



Consiglio Nazionale delle Ricerche

ISTITUTO DI
ELABORAZIONE DELLA
INFORMAZIONE

Pisa

*HDRC25 2.01: Un sistema di Hard Disk
Recording-Editing di segnali audio per Windows.
Realizzazione con Visual Basic 1.0.*

A. Belfiore, G. Bertini, V. Mazzacca, D. Montebovi

Nota Interna B4-24

Luglio 1993

***HDRC25 2.01: Un sistema di Hard Disk
Recording-Editing di segnali audio per Windows.
Realizzazione con Visual Basic 1.0.***

A. Belfiore, G. Bertini, V. Mazzacca, D. Montebovi

Nota Interna B4-24

Luglio 1993

INDICE

Presentazione.....	2
Introduzione.....	3
1. Analisi delle funzionalità di un sistema di Hard Disk Recording	7
1.1 Hard Disk Recording e Riascolto	7
1.2 Visual Editing.....	8
1.3 PlayList.....	8
1.4 Trattamento di segnali audio	8
2. Supporto di memorizzazione: hard disk	9
3. Supporti hardware scelti per il sistema	10
4. Flusso dei dati.....	13
5. Strumenti di programmazione.....	15
5.1 Visual Basic: un nuovo approccio alla programmazione	15
5.2 Disegno dell'interfaccia utente	16
5.3 Inizializzazione delle proprietà	16
5.4 Sviluppo del programma	17
6. Presentazione del programma HDRC25 2.01	18
6.1 Modalità d'uso.....	19
Conclusioni.....	23
Bibliografia	24

HDRC25 2.01: Un sistema di Hard Disk Recording-Editing di segnali audio per Windows. Realizzazione con Visual Basic 1.0.

A.Belfiore*, G.Bertini**, V. Mazzacca***, D. Montebovi***

*Conservatorio Musicale L. Cherubini, Firenze

**IEI-CNR, Pisa

***Collaboratore esterno

Presentazione

In questa nota si descrive il progetto e la realizzazione prototipale di un sistema di acquisizione-memorizzazione-editing di segnali in banda audio basato su elaboratore personale della classe PC IBM e compatibili; nel seguito indicheremo il sistema con la sigla HDR (Hard Disk Recording), comprendendo l'insieme di tutte le sue funzioni, come normalmente usato in letteratura.

Nelle applicazioni per il trattamento numerico di segnali è in forte aumento l'uso di sistemi digitali basati su PC con livelli di prestazione differenziati. Ciò si verifica anche per il settore dei segnali audio ad es. nella strumentazione per la produzione di musica elettronica e la postproduzione audio-video (colonne sonore di film, lavori televisivi, spot pubblicitari, ecc.), nei sistemi multimediali, ecc..

Presso l'IEI/CNUCE-CNR ed in collaborazione con ditte esterne è in corso da alcuni anni un'attività per lo sviluppo di stazioni di lavoro modulari, basate su PC e strumenti HW e SW specializzati per il DSP, orientate al trattamento numerico di segnali in banda audio (segnali di rumore, vocali, musicali, ecc.). In tale ambito è stata sviluppata la scheda LeonardC25 basata su un microprocessore TMS320C25 della Texas Instr.. L'obiettivo del presente progetto è quello di sviluppare un sistema di HDR integrandolo nell'ambiente di lavoro suddetto per aumentarne le potenzialità.

Nella nota, dopo un panoramica fatta su alcuni sistemi di HDR rappresentativi, vengono individuate le operazioni tipiche e le problematiche relative alla gestione di file che contengono la codifica di segnali audio. Utilizzando le schede DSP a nostra disposizione, le LeonardC25 ed il linguaggio Visual Basic-1.0 per la realizzazione delle interfacce utente, è stata sviluppata una versione del sistema (HDRC25-2.01), contraddistinto da un insieme di funzionalità ridotto, ma sufficiente per effettuare delle valide esperienze di uso.

Nota: una prima versione HDREC è stata realizzata sotto MS-DOS dalla Soc. Leonardo (L.M. Del Duca, D. Montebovi).

Si ringrazia M. Marani e L. Leodori per il contributo dato alla messa a punto del sistema.

Introduzione

La registrazione, la riproduzione e l'editing di segnali in banda acustica é una classica operazione che nei vari ambiti applicativi ha usufruito delle innovazioni che la tecnologia nel suo evolversi ha riversato nella strumentazione messa a disposizione: basta ricordare i miglioramenti delle prestazioni dei registratori a nastro magnetico con le tecniche di riduzione del rumore (vari tipi di Dolby, ecc.), l'introduzione del CD (Compact Disk) e ultimamente del DCC (Digital Compact Cassette) e del MD (Mini Disc).

Le funzioni tipiche eseguite con modalit  analogica sono quelle di: controllo della dinamica, sovrapposizione e missaggi di tracce, operazioni di taglia e cucii manuali o meccanizzate, ecc. effettuate con molta difficolt , dispendio di tempo ed energie, e sempre in modo impreciso.

Per eseguire le suddette operazioni ormai le tecniche analogiche, pur essendo arrivate a fornire buone prestazioni sono gradualmente abbandonate e vengono largamente sostituite con tecniche digitali. Si ritiene qui utile fare una breve panoramica sulla situazione attuale dei sistemi per il trattamento dei segnali musicali, rimanando alla bibliografia specifica per quanto riguarda gli aspetti storiografici dei rapporti fra musica e tecnologia [TARABELLA-92].

I sistemi digitali impiegati nel trattamento di segnali offrono prestazioni sempre pi  sofisticate e sostituiscono i sistemi analogici in un numero sempre crescente di applicazioni; oltre che nei settori tradizionalmente trattati (telecomunicazioni, militari, controllo di processi, ecc.) ultimamente le tecniche digitali si vanno diffondendo nel settore dell'audio e del video professionale, consentendo anche lo sviluppo di sistemi multimediali nei quali l'elaboratore elettronico pu  controllare e integrare fra loro una molteplicit  di sorgenti di informazione: testi, grafica, immagini fisse e in movimento, audio, voce e musica sintetizzati e naturali.

Al riguardo   bene sottolineare le forti implicazioni che tale settore ha dal punto di vista produttivo e del volume di mercato che induce. Si pensi a tutta l'attivit  legata allo sviluppo di sistemi di trattamento automatico della messaggistica e della postproduzione, di colonne sonore (film, lavori televisivi, spot pubblicitari, ecc.) Un altro settore di interesse rilevante che fa un notevole uso di programmi di questo tipo   quello della masterizzazione di CD.

I vantaggi maggiori offerti dall'impiego di sistemi di hard-disk recording digitali sono, oltre alla maggiore precisione e stabilit  di operazioni tipicamente eseguite con i sistemi analogici, l'introduzione di nuove possibilit  non offerte dalle vecchie tecniche, quali ad es.:

- compressione-espansione di pezzi di brani, possibilit  di collegare pezzi presi da brani diversi, potendo curare in maniera diversa il tipo di collegamento tra pezzi distinti (*crossfade*).

- interventi sulla dinamica del brano, editing al microlivello, cio  direttamente sulla forma d'onda che rappresenta il segnale in modo da eliminare i difetti e migliorare la qualit  del risultato finale.

- facilitazioni per l'esecuzione di analisi sul segnale, per indagini di tipo statistico (FFT, valore di picco, valor medio ecc.).

La registrazione digitale di segnali audio fino agli anni '80 veniva effettuata soltanto tramite sofisticati e costosi hardware dedicati; sistemi di registrazione digitale erano disponibili a costi elevati (decine di milioni, a volte centinaia). In

seguito, grazie al continuo perfezionamento delle tecnologie di acquisizione ed elaborazione del segnale, all'abbattimento dei costi di produzione dei materiali ed alla sempre maggiore capacità degli elaboratori di manipolare grandi quantità di dati numerici quali sono, appunto, i dati digitalizzati, è stato possibile realizzare dei sistemi di registrazione digitale basati su personal computer. Si è assistito ad un forte aumento delle prestazioni e delle potenzialità di calcolo di personal computer, con la conseguente diffusione in molti ambiti di utilizzo.

Grazie all'avvento della tecnologia VLSI si sono avuti via via a disposizione vari tipi di microprocessori, alcuni per uso generale, altri invece orientati ad applicazioni più specializzate, sempre più veloci, potenti e questi ultimi con un set di istruzioni particolarmente orientato all'implementazione di algoritmi tipici dell'elaborazione digitale dei segnali (microprocessori per *DSP*, *Digital Signal Processing*).

Si è verificata, così, la tendenza a realizzare sistemi a costo contenuto, per l'elaborazione numerica in tempo reale di segnali, costituiti da calcolatori delle classe PC MS-DOS, Macintosh, ecc. equipaggiati con moduli di calcolo progettati appositamente, basati su microprocessori per DSP con funzioni di coprocessori programmabili.

Uno dei primi prodotti in linea con questa tendenza, il Sound Tools, è stato realizzato alla fine degli anni '80 dalla DigiDesign (costo 3.5 milioni circa). Basato su computer Macintosh 2, Sound Tools permetteva di registrare, modificare e riprodurre due canali audio con qualità da CD. Si è assistito quindi ad un proliferare di registratori/editori su hard-disk basati su computer: Turtle Beach 56K, sistema basato su PC IBM 286 e 386 compatibili, Spectral Synthesis Digital Studio, Digital Audio Card System, fino ad arrivare ai più recenti Deck/AudioMedia, Sound Tools 2 (versione a quattro canali) e il multi traccia Pro Tools, prodotti sempre dalla DigiDesign. Sono stati sviluppati inoltre anche sistemi autosufficienti come il Roland DM-80, disponibile nelle versioni a quattro e otto tracce, eventualmente controllabile attraverso un computer Macintosh, nel qual caso può arrivare a trentadue tracce, l'Akai DD1000, lo Yamaha CBX-D5, ecc.

Oggi sul mercato c'è quindi un discreto assortimento di sistemi di registrazione digitale; un sistema di media potenza (non professionale) a due tracce costa attualmente intorno ai due milioni e mezzo di lire, senza contare il computer e l'hard-disk, quest'ultimo in particolare riveste un ruolo delicato viste le richieste di memoria (circa 10Mbyte per un minuto di suono stereo hifi a 44100Hz) e di tempo di accesso (<28msec).

Al di sotto di questa fascia si trovano anche dei programmi di HDR di corredo alle molte schede audio di tipo consumer aventi limitate possibilità operative e quindi di scarso interesse per usi professionali

Il sistema oggetto del nostro progetto e realizzato in forma prototipale utilizza una scheda che pur essendo di basso costo (0.6 MLt.) è dotata di un microprocessore per DSP, il *TMS320C25* della Texas Instruments [BERTINI-92]. La scheda, denominata LeonardC25, è stata prodotta nell'ambito di una collaborazione tra il Centro Tecnologico Culturale Leonardo di Massa e l'IEI di Pisa e destinata ad impieghi generali che richiedano l'elaborazione numerica in tempo reale di segnali in campo industriale, biomedico e audio. Inoltre all'IEI con lo scopo di avere potenze di calcolo maggiori ed in modo scalabile è stato progettato e realizzato un

SW che consente di far operare in parallelo fino ad otto schede LeonardC25 [BERTINI-93a]; si possono assemblare così delle vere e proprie stazioni di lavoro (*MULTIC25*) particolarmente orientate alla sintesi di segnali musicali con elevato grado di polifonicità e politimbricità [BERTINI-93b].

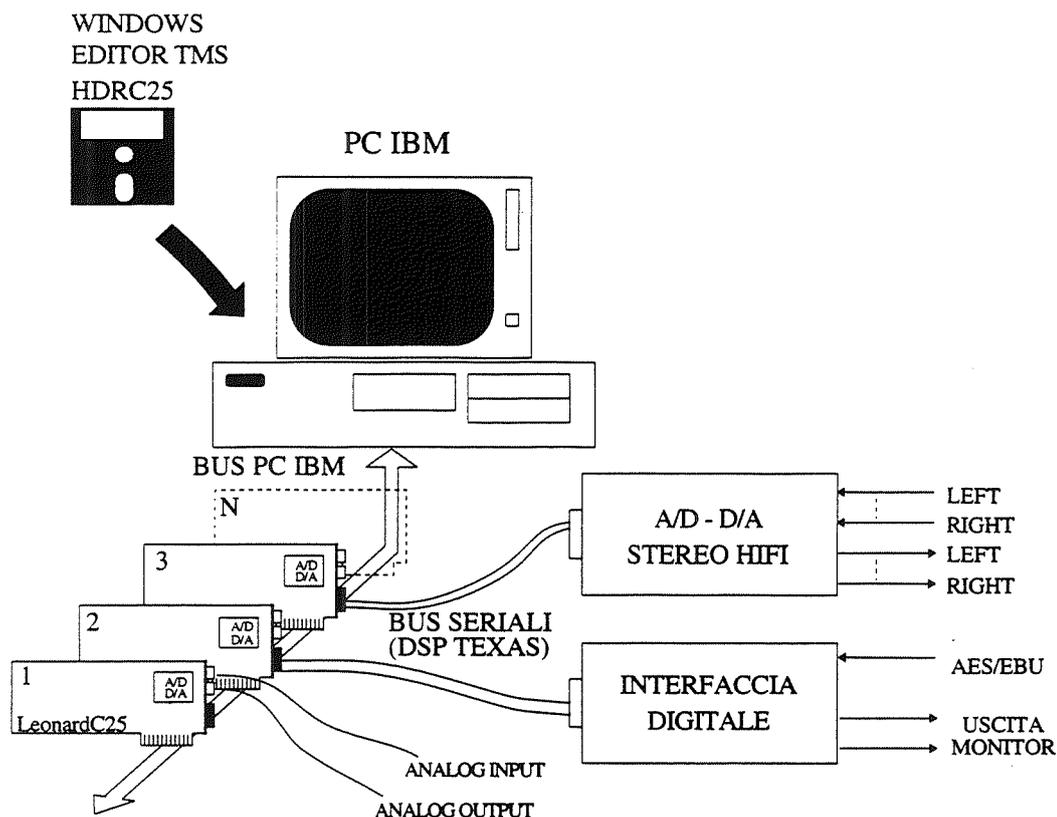


Fig. 1 Stazione di lavoro musicale MUSTC25.

Sempre per tale settore di applicazione sono stati realizzati o sono in via di completamento una serie di *tools* (editore grafico per algoritmi di sintesi, software di integrazione con dispositivi MIDI, libreria di algoritmi di sintesi, ecc.) in grado di rendere possibile l'uso ad alto livello delle potenzialità della stazione di lavoro stessa (fig. 1). Si cerca cioè di creare un ambiente di lavoro che fornisca una vasta gamma di strumenti fruibili in modo amichevole da musicisti e generici utenti non interessati al lavoro di programmazione a basso livello. L'obiettivo del presente progetto è perciò quello di aggiungere una funzionalità di Hard Disk Recording alla stazione di lavoro *MULTIC25*.

I suddetti *tools* come filosofia sono esportabili e utilizzabili con supporti hardware con più alti livelli di prestazioni (vedi DSP con floating point).

Non è stato naturalmente trascurato l'obiettivo di rendere il più professionale possibile, nei limiti consentiti dalle potenzialità dell'hardware sottostante, questo

sistema di HDR, tenendo presente le esigenze dell'utente finale, che sono poi in ultima analisi quelle che portano alla richiesta di nuove funzionalità di cui corredare un programma di hard-disk recording.

Curando particolarmente l'interfaccia utente e l'ottimizzazione delle prestazioni rispetto alle caratteristiche della scheda DSP a disposizione, vengono create tutte le funzioni tipiche di tools simili (registrazione, riascolto, ecc.), riscontrabili in sistemi professionali, con l'aggiunta di funzioni tipiche del DSP. Inoltre l'obiettivo comprende anche la verifica della possibilità di integrare il tool nel software della stazione di lavoro multischeda .

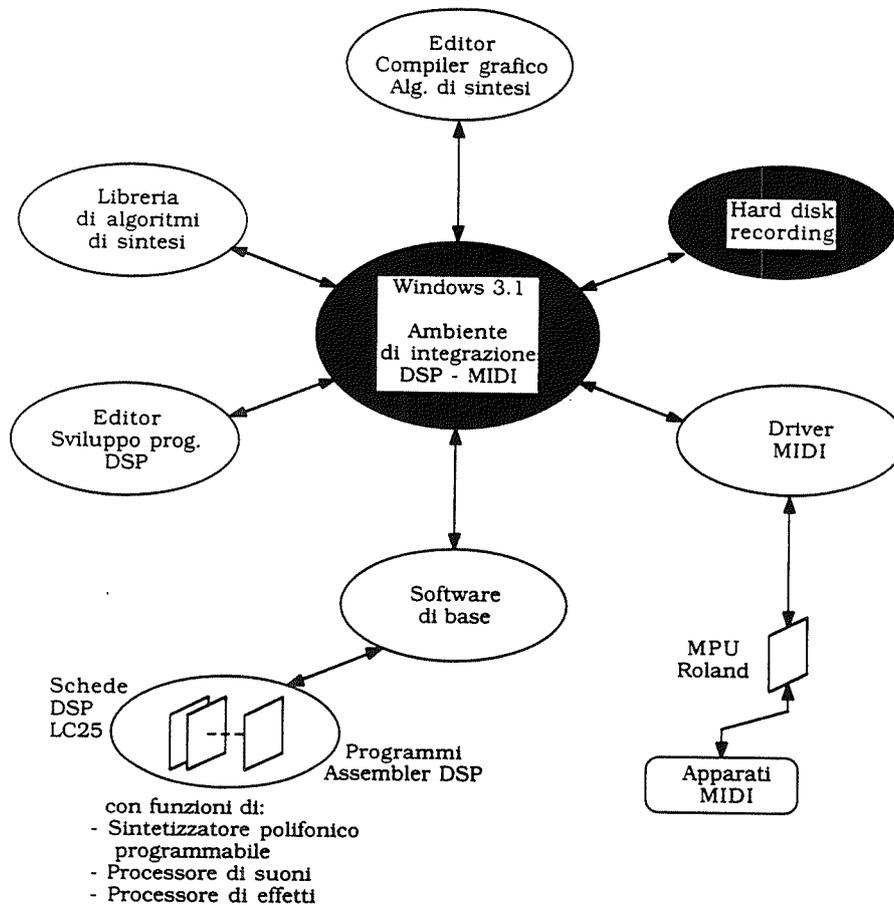


Fig. 2 Rappresentazione grafica dei tools della stazione di lavoro MULTIC25.

Nel seguito vengono analizzati i concetti principali e le problematiche relative all'acquisizione, al trattamento e al recupero di segnali audio.

Successivamente viene fornita una descrizione delle caratteristiche funzionali e architetturali del sistema proposto, sia per le risorse HW che SW. Per prima si esamina la scheda LeonardC25 adottata come principale componente hardware del sistema, in particolare per gli aspetti che riguardano il suo utilizzo, poi gli strumenti utilizzati nello sviluppo del SW residente sull'elaboratore ospite, come la programmazione in ambiente Microsoft Windows e Visual Basic 1.0. Segue la descrizione delle funzionalità del programma di HDR.

1. Analisi delle funzionalità di un sistema di Hard Disk Recording

Lo scopo di questo paragrafo è quello di esaminare brevemente le principali funzioni di un sistema di hard disk recording ed editing di segnali audio.

Un sistema di HDR è un programma che, sfruttando particolari risorse hardware (PC, dispositivi di conversione, schede acceleratrici, interfacce digitali, campionatori, ecc.), deve fornire un insieme di strumenti per il trattamento del suono, deve cioè permettere di acquisire suoni con un dispositivo di campionamento, editarli, salvarli sull'hard disk, per poi recuperarli, eventualmente elaborarli e riprodurli; inoltre l'utilizzo di microprocessori per il DSP (*Digital Signal Processing*) permette di fare operazioni di elaborazione sul segnale in tempo reale, creazione di effetti, filtraggi, eco, riverbero, ecc..

In pratica una volta che il segnale audio è stato acquisito sotto forma di campioni digitali ed immagazzinato in file, su tali file si deve poter intervenire con tutte le funzioni tipiche dell'editing, con in più la possibilità di disporre anche di una rappresentazione grafica del segnale contenuto nel file sulla quale poter intervenire, il tutto sfruttando al massimo le tecniche di elaborazione digitale del segnale. Per quanto riguarda l'editing dei segnali, oltre alle funzioni standard dell'editing distruttivo (cut, paste, copy, ecc.), ci sembrano di grande interesse le funzioni per la realizzazione di editing non distruttivo, che cioè non alterano il segnale originale in nessun modo.

Un sistema di hard disk recording ed editing di segnali audio può essere diviso in quattro sezioni base:

- a) *Hard Disk Recording e Riproduzione;*
- b) *Visual Editing;*
- c) *Creazione ed Editing di PlayList;*
- d) *Trattamento di segnali audio.*

1.1 Hard Disk Recording e Riascolto

In questa sezione sono comprese tutte quelle funzioni tipiche di un registratore digitale (e analogico), un segnale audio può essere registrato in tempo reale direttamente sull'hard disk del PC e riascoltato; il riascolto (*play*) può partire da un punto qualsiasi del segnale, ci si può posizionare in qualsiasi punto del segnale (senza i noiosi tempi d'attesa per il riavvolgimento del nastro), si può eseguire il play di una porzione particolare del segnale, agendo direttamente sulla rappresentazione grafica del segnale stesso, ecc..Un'altra possibile funzione è la registrazione su più tracce e ovviamente il riascolto di più tracce, caratteristica questa dei registratori multitraccia, utilizzati soprattutto a livello professionale negli studi di registrazione.

Il sistema di hard disk recording può quindi essere visto come un dispositivo di registrazione digitale, il cui tempo totale di registrazione è limitato solo dalla dimensione dell'hard disk con cui si lavora, si tenga presente al riguardo che un minuto di suono stereo con caratteristiche hi-fi richiede circa 10 Mbyte di spazio su disco.

Va sottolineato infine che l'implementazione delle funzioni tipiche di questa sezione richiede maggiormente l'uso di moduli hardware particolari: schede acceleratrici, interfacce analogico/digitale e digitale/analogico, ecc..

1.2 Visual Editing

In questa sezione possono essere incluse tutte le funzioni standard nel dominio del tempo e della frequenza come cut, copy, paste, fade, crossfade, mix, merge, creazione di loop, equalizzazione grafica e parametrica, visualizzazione della FFT a tre dimensioni, analisi di tipo statistico (valore di picco, valor medio, ecc.), modifica della lunghezza di un suono campionato (espansione, compressione) senza cambiare il valore di picco e viceversa.

Il vantaggio di usare il visual editing è quello di riuscire ad eseguire su PC queste funzioni più velocemente e facilmente rispetto all'editing classico di file.

Va sottolineata l'importanza di disporre di due tipi di editing: *distruttivo e non distruttivo*. L'editing distruttivo, che modifica direttamente il segnale editato, è il modo di editare un suono se lo si vuole inviare ad un campionatore, ma se vogliamo registrare il suono direttamente su hard disk è utile disporre anche di strumenti di editing non distruttivo, per evitare di alterare il segnale originale acquisito e per rendere le operazioni di editing stesse più veloci; un caso particolare di editing non distruttivo, che analizzeremo a parte nel successivo paragrafo, è la Playlist.

1.3 Playlist

La playlist è sicuramente una delle funzioni più importanti e innovative realizzabili con un sistema di hard disk recording, sfruttando i mezzi messi a disposizione dall'elaborazione digitale dei segnali, ed è il migliore esempio di editing non distruttivo.

La creazione e l'esecuzione di una playlist permette la riproduzione in tempo reale di regioni di segnali audio diverse, precedentemente create, nell'ordine imposto dalla playlist e tutto questo senza alterare in nessun modo i segnali originali, dai quali sono state scelte le regioni che formano la playlist.

Tra le altre cose va sottolineato che si può scegliere il tipo di collegamento (*crossfade*) fra una regione e la successiva, nonché la durata di tale collegamento; anche sul tipo di collegamento si può variare: dalla semplice giustapposizione, al crossfade lineare, dal crossfade esponenziale, al sinusoidale, ecc..

Questo è sicuramente un'opzione molto utile negli studi di registrazione, dove rende possibile la modifica dell'ordine d'esecuzione di parti di brani musicali, senza modificare in alcun modo il materiale sonoro di partenza, e di aggiustare il collegamento fra le varie parti finché non si trova una soluzione soddisfacente, che a quel punto può eventualmente essere salvata su hard disk.

1.4 Trattamento di segnali audio

Le funzioni di questa sezione permettono di agire sui parametri del segnale acquisito modificando, da un punto di vista qualitativo, la forma d'onda del segnale stesso: le modifiche consistono ad esempio nella creazione di effetti particolari, creazioni di echi, riverberi, cambi di frequenza di campionamento, ecc., realizzate implementando delle operazioni di filtraggio digitale.

L'esecuzione di algoritmi di filtraggio digitale, su file audio relativi a registrazioni di lunga durata, con le potenzialità di PC anche evoluti porta a tempi di calcolo esagerati: per cui risulta utile, a volte indispensabile, l'uso di schede acceleratrici dotate di microprocessori per il DSP.

2. Supporto di memorizzazione: *hard disk*

I sistemi per il trattamento dei segnali digitali con caratteristiche stereo hi-fi lavorano su quantità di dati enormi, basti pensare che un minuto di suono stereo con caratteristiche di hi-fi richiede circa 10 MByte di spazio in memoria, che è il risultato della memorizzazione di 44100 campioni al secondo (frequenza di campionamento dei CD) per 60 (secondi al minuto) per 2 (numero di byte per campione) per 2 (numero di canali); si capisce che se anche si dovesse rinunciare alle caratteristiche di stereofonia ed hi-fi comunque, a meno di un degrado eccessivo del segnale digitalizzato, il numero di byte richiesti per la memorizzazione di segnali audio digitalizzati si manterrebbe elevato. Al riguardo sono correntemente impiegate tecniche di compressione dati (ADPCM, MPEG audio, ecc.) che però degradano il segnale e pertanto non possono essere usate con fattori di compressione elevati (oltre il 2-3). L'approfondimento di queste tecniche, di grande interesse e delle quali si tiene conto in altre applicazioni, non rientra negli scopi del nostro lavoro in quanto non ci siamo preoccupati della valutazione di questi aspetti. Per questo motivo fra tutti i componenti di cui si può aver bisogno, uno dei più critici è certamente l'*hard disk*. Si tratta, come noto, di dispositivi che permettono di immagazzinare grandi quantità di dati e successivamente recuperarli in tempi molto ridotti. Le prestazioni di un *hard disk* sono caratterizzate da alcuni parametri tipici:

- a) *capacità (>500MByte);*
- b) *tempo di accesso medio (average seek time, <15msec.);*
- c) *tempo di latenza medio (average latency time, <15msec.);*
- d) *velocità di trasferimento (transfer rate, 1.5MByte/sec.);*

ci sono anche altri parametri per una valutazione più specialistica delle caratteristiche di un *hard disk*, cosa che però va oltre i nostri scopi e per la quale si rimanda alla letteratura specialistica. Collegata al problema della velocità di trasferimento è la presenza di un'interfaccia tra l'*hard disk* e il computer che permette appunto il dialogo tra questi due dispositivi. Il mondo dei PC ha proposto in questi ultimi anni tre tipi principali di interfacce tra computer e *hard disk*: *IDE*, *ESDI* e *SCSI*; la prima gode una notevole diffusione nel mondo degli IBM compatibili, la seconda è prerogativa pressoché esclusiva delle macchine IBM originali, la terza inizialmente ha conosciuto grande diffusione in ambiente Macintosh per poi espandersi anche fuori da questo ambiente. La differenza più rilevante tra *IDE*, *ESDI* e *SCSI* è che le prime due sono *interfacce punto-punto*, permettono cioè solo il colloquio bidirezionale tra due dispositivi (computer e disco), mentre la terza permette l'indirizzabilità di diversi dispositivi ad essa connessi. Proprio per questo motivo e per l'elevata velocità di trasferimento (soprattutto nelle evoluzioni *SCSI2* e *SCSI3*) l'interfaccia *SCSI* sta riscuotendo un grande successo, in particolare nel campo dell'*hard disk recording*.

Per concludere vale la pena sottolineare l'importanza che hanno le modalità di organizzazione dei file su disco, quindi in pratica l'importanza del *file system*. Una memorizzazione contigua dei file su disco non può che apportare benefici nelle operazioni di recupero dei dati tipiche delle applicazioni legate all'*hard disk recording*, non a caso molto spesso sui manuali di installazione ed uso di programmi di HDR si consiglia periodicamente una riorganizzazione e ricompattamento del disco.

3. Supporti hardware scelti per il sistema

In base all'analisi fatta nei paragrafi precedenti si è individuato un sottinsieme di funzioni per i nostri scopi con l'obiettivo di realizzare un sistema di hard disk recording a basso costo per uso sperimentale, che utilizza come hardware minimo un PC-IBM compatibile e una scheda per il DSP. In modo semplificato i supporti per il nostro sistema sono riportati in figura 3.

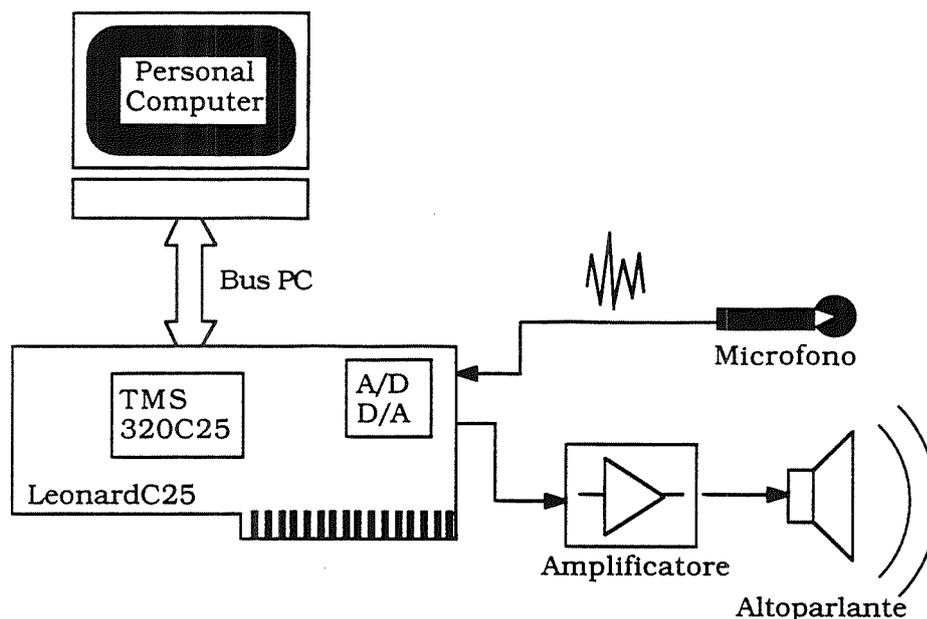


Fig. 3 Architettura del sistema HDRC25.

Per quanto riguarda la scelta della scheda è ricaduta su una LeonardC25, le parti principali e le caratteristiche essenziali della quale sono descritte di seguito.

Un **microprocessore DSP TMS320C25** con frequenza di clock a 40MHz, a 16 bit, in grado di eseguire un ampio set di istruzioni tipiche del DSP, fra cui citiamo la moltiplicazione/somma di due operandi (multiply/accumulate) che viene eseguita in 100 nsec.. Per mantenere le caratteristiche di compattezza richieste è stato impiegato questo microprocessore come nucleo di calcolo, scelta giustificata da una serie di motivi, fra cui il buon rapporto prestazioni/costo e il fatto che tale dispositivo è tuttora uno standard industriale, supportato da una serie di strumenti di sviluppo (sia software che hardware) continuamente aggiornati. Inoltre, il software sviluppato per il TMS320C25 può essere impiegato eventualmente anche da alcuni processori più veloci della generazione successiva (TMS320C50), di recente apparizione sul mercato.

Un **banco di memoria RAM** da 28 kwords e un banco di memoria EPROM da 4 kwords, allocati nello spazio di memoria programma del microprocessore.

Un'interfaccia **parallela di comunicazione bidirezionale** con l'elaboratore ospite. La volontà di mantenere la compatibilità con i PC di tipo XT, ha avuto come conseguenza il fatto che per l'interfacciamento è stato usato il bus dati a 8 bit del PC. Poichè il bus dati del TMS320C25 è a 16 bit le operazioni di comunicazione sono eseguite in due tempi. L'interfaccia è inoltre in grado di inviare richieste di interrupt al controllore di interruzioni dell'elaboratore ospite.

Un'interfaccia verso il mondo dei segnali analogici costituita da un convertitore integrato A/D-D/A e da una sezione di condizionamento per i segnali di ingresso. La scheda è provvista di tre prese RCA per i due ingressi e per l'uscita analogica.

Una porta seriale esterna: il modulo può comunicare con dispositivi esterni tramite una comunicazione seriale full-duplex (bidirezionale) con velocità massima di 5 Mbit/sec, che permette l'implementazione del protocollo seriale della Texas Instruments offerto dal microprocessore. Il collegamento è realizzato con un connettore Cannon a 15 poli.

Un'apposita rete (Control & Switch) che effettua la commutazione della porta seriale del TMS sul convertitore interno o verso la seriale esterna.

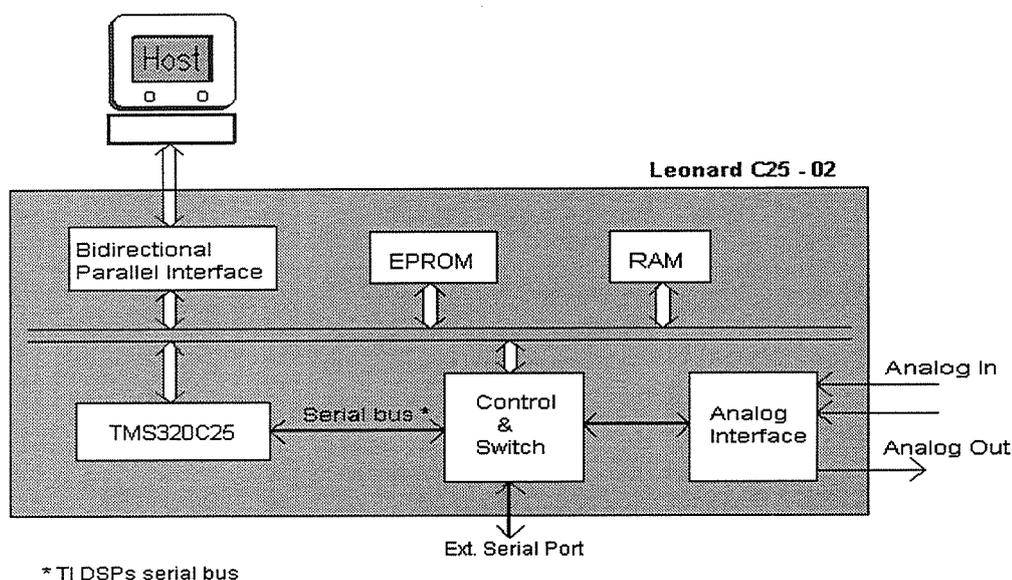


Fig. 4 Architettura del modulo LeonardC25.

La scelta del modulo LeonardC25 è legata al fatto che tale scheda, realizzata presso i laboratori dell'IEI, ha un buon rapporto costo/prestazioni rispetto a quelle disponibili sul mercato. A titolo di informazione schede di produzione estera, basate anch'esse sul TMS320C25 costano almeno il doppio e schede audio di tipo *consumer*, che costano circa la metà della LeonardC25, come ad esempio la Sound Blaster, pur con buone caratteristiche audio, hanno però una carenza fondamentale: sono prive di microprocessore, non sono quindi programmabili in modo generale. Infatti il chip VLSI, di cui è dotata la Sound Blaster, è di tipo dedicato (*application specific*), può eseguire una sola tecnica di sintesi di suoni, la FM, e pochi altri effetti.

Va sottolineato che comunque il software da noi realizzato per l'HDR non è strettamente legato alla scheda acceleratrice utilizzata e più in generale al particolare hardware utilizzato.

Altro hardware richiesto è: un PC IBM compatibile, una sorgente di segnali audio, un amplificatore e un altoparlante.

Il PC utilizzato è un IBM compatibile modello AT con processore INTEL 80486 a 50 MHz, con 4 MByte di RAM, con un hard disk da 209 MByte di capacità, sul PC è installato Windows 3.1.

Per quanto riguarda l'hard disk è caratterizzato da un tempo medio d'accesso di 16 msec, tempo medio di seek di 18.2 msec, velocità di trasferimento di 314.4 KByte/sec, questi dati sono stati ottenuti col programma *Norton*. Appare evidente da questi dati che le caratteristiche dell'hard disk non rientrano nei valori tipici richiesti per programmi di hard disk recording di più alti livelli di prestazioni, ma per quelli che sono i nostri scopi vanno bene anche tali valori. Naturalmente un hard disk di elevate prestazioni (ad es. SCSI) porterebbe ovvi benefici alle prestazioni, ma comporterebbe un aumento dei costi del sistema e quindi tale soluzione può essere considerata per lo sviluppo di sistemi professionali.

Come fonte di segnali audio è stato utilizzato un DAT (Digital Audio Tape) DA-R100 della CASIO, l'uso del DAT non è ovviamente indispensabile e può essere sufficiente anche un microfono con adatta sensibilità, purché compatibile elettricamente con gli ingressi della scheda.

C'è bisogno anche di un amplificatore e di un altoparlante per poter riascoltare i risultati del trattamento e delle modifiche dei segnali acquisiti, anche in questo caso non ci sono particolari richieste.

Altro hardware aggiuntivo collegabile al sistema minimo potrebbe essere un'interfaccia SCSI, che permetterebbe di collegare al PC più dispositivi (in particolare hard disk di elevate prestazioni), e un dispositivo di conversione di più elevate prestazioni, collegabile all'ingresso seriale della scheda e il cui uso è tra l'altro previsto dal progetto realizzato.

Al riguardo è disponibile presso i laboratori dell'IEI un modulo di conversione A/D D/A stereo hi-fi che sfrutta la *sigma delta modulation*; tale modulo è stato sviluppato con una collaborazione fra l'IEI di Pisa, la Leonardo di Massa e la SEED di Massa, e sfrutta come componenti base per la conversione A/D due convertitori della Motorola (AD776AQ) e per la conversione D/A un convertitore della Philips (SAA7323GP), che lavorano entrambi con precisione su 16 bit e 20 kHz di banda passante.

4. Flusso dei dati

Il programma realizzato permette l'acquisizione in tempo reale di segnali analogici di banda compresa tra 0 e circa 10 KHz, consentendo la memorizzazione dei campioni, sempre in tempo reale, sull'hard disk del PC.

I limiti imposti sulla frequenza dei segnali acquisiti sono legati al tipo di interfaccia e al meccanismo di comunicazione fra la scheda e il PC, nonché all'uso di hard disk di più alte prestazioni (SCSI).

A tale proposito c'è da dire che sono stati utilizzati i driver di comunicazione forniti dai progettisti del software di base della scheda e che in teoria i tempi di trasferimento potrebbero essere ottimizzati sfruttando le ultime versioni di Windows previste per ambienti multimediali

Comunque l'IEI e la SEED stanno sviluppando una versione di scheda che oltre a contenere nuovi dispositivi per la conversione, adotterà meccanismi di comunicazione col PC tali da permettere di superare questi inconvenienti in modo da arrivare ai 44.1 KHz stereo (uso di Dual Port, DMA, ecc.).

Diamo ora una breve descrizione dei vari passi fatti dal segnale dal suo ingresso analogico nella scheda alla memorizzazione sottoforma di campioni a 14 bit sull'hard disk del PC.

Il segnale analogico, in ingresso alla scheda, viene sottoposto al processo di filtraggio e conversione ad opera del TLC, quindi i campioni prodotti vengono inviati al TMS, che a sua volta li memorizza in un buffer circolare nella memoria programma disponibile sulla scheda, in attesa di poterli inviare al PC. Sul PC la situazione è simile: sulla RAM viene implementato un buffer in cui vengono memorizzati i campioni ricevuti dalla scheda prima di memorizzarli permanentemente sul disco. Il processo appena descritto può essere schematizzato come segue:

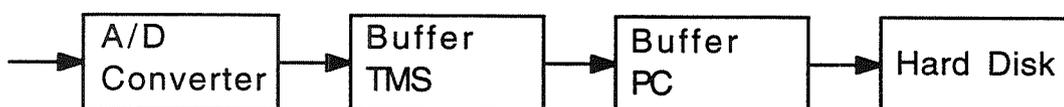


Fig. 5 Flusso dei dati.

La memorizzazione temporanea dei campioni nei buffer è necessaria per cercare di rendere meno stretta la forma di *rendez-vous* che si crea nella comunicazione tra scheda e PC e per evitare perdite di dati, derivanti dal fatto che il processo di campionamento avviene in maniera continuativa nel tempo, mentre il processo di trasferimento dei campioni su hard disk, come è noto, non può avvenire secondo la stessa modalità.

Nel processo di riproduzione il segnale digitale compie passi analoghi in senso inverso; anche in questo caso si utilizza il meccanismo di bufferizzazione dei campioni nel percorso dall'hard disk all'uscita analogica della scheda.

Nel caso si decida di utilizzare il convertitore hi-fi esterno alla scheda, caratterizzato da frequenze di campionamento superiori a quella consentita dal programma, sono necessari degli opportuni algoritmi di decimazione del segnale in arrivo dall'interfaccia digitale seriale della scheda con fattore di riduzione della

frequenza opportuno. Questa operazione si realizza con l'uso di filtri digitali in tempo reale sul TMS [SHAFER-73]. Viceversa quando si vuole riascoltare i segnali occorre eseguire il sovracampionamento effettuato sempre con operazioni di interpolazione, come noto implementabili con filtri digitali [SHAFER-73]. Nonostante l'inconveniente della riduzione della frequenza di campionamento del segnale memorizzato nel file audio, può essere utile sfruttare questi dispositivi di conversione hi-fi perché sono migliori del TLC per quanto riguarda il rapporto segnale/rumore ($>90\text{dB}$) e per la maggiore dinamica (16 bit).

5. Strumenti di programmazione

Fra gli obiettivi prefissati nella realizzazione del sistema uno dei principali è quello di realizzare un ambiente di lavoro che permetta all'utente del sistema di poter controllare ed interagire con l'applicazione nel modo più agevole possibile. L'interfaccia con l'utente, realizzata sull'elaboratore ospite, deve essere quindi confortevole e non richiedere tempi e sforzi di apprendimento elevati, in modo che l'utente stesso possa concentrare la sua attenzione sull'utilizzo del sistema.

In generale prima di intraprendere un lavoro di programmazione è importante valutare attentamente gli strumenti disponibili ed operare le giuste scelte.

La scelta che è stata fatta per questo lavoro è quella di realizzare il programma per la gestione del sistema su PC come un'applicazione per il sistema operativo *Microsoft Windows*.

L'ambiente Windows, come è noto, presenta delle caratteristiche che lo rendono preferibile da parte di molti utenti di PC rispetto al classico ambiente DOS (per il quale era stata sviluppata una prima versione del sistema HDRC25): l'interfaccia grafica con finestre, menu, icone, ecc.; il funzionamento regolare e prevedibile per tutte le applicazioni; il multitasking; la possibilità di utilizzare tutta la memoria RAM del computer; funzionamento in modalità protetta.

La preferenza verso Windows risulta ancora più accentuata dall'uso di modelli di PC con CPU veloci, come le 386 e 486, e hard disk con tempi di accesso ridottissimi e grosse capacità di memoria.

Per questo motivo, molte case costruttrici stanno realizzando versioni per Windows di molti loro programmi, mentre sviluppano le nuove applicazioni direttamente per questo ambiente.

Una volta deciso quindi di sviluppare il programma per Windows, la scelta da fare riguarda quali strumenti di programmazione da utilizzare.

Il nostro progetto è stato realizzato con il programma *Visual Basic 1.0*, che è esso stesso un'applicazione Windows, cambia completamente l'approccio tradizionale: permette, anche ad utenti non esperti, di creare delle applicazioni perfettamente e immediatamente funzionanti. Tutto ciò è reso possibile dai due fattori che danno il nome al programma: la costruzione *visiva* di un'applicazione, disegnando direttamente le finestre, i loro componenti e tutto ciò che riguarda l'interfaccia tramite un *editor grafico*, e l'utilizzo del linguaggio *Basic*.

Al riguardo va subito sottolineato come il Basic in questione, pur mantenendo la semplicità d'uso del Basic tradizionale, ha tutte le caratteristiche di un linguaggio strutturato: dichiarazione ed uso di procedure, funzioni, tipi di dato, istruzioni composte, ecc..

5.1 Visual Basic: un nuovo approccio alla programmazione

Le tradizionali applicazioni DOS si pongono esse stesse al centro del funzionamento; lo sviluppatore dell'applicazione si concentra prima di tutto sull'applicazione stessa, sul contenuto, sulla dinamica, e considera l'utente solo come un fornitore di dati, praticamente alla stregua di un file.

L'approccio di Windows, e quindi di Visual Basic, è diametralmente opposto. Il centro dell'applicazione è rappresentato dall'utente, o più precisamente dall'interfaccia utente.

La conseguenza immediata di questa inversione di posizioni è data dal fatto che l'utente può scegliere in qualunque momento cosa fare, ad esempio facendo click sulla parte di schermo che interessa. Ovviamente tale cambiamento modifica profondamente il modo di concepire e realizzare le applicazioni.

Una delle difficoltà nello sviluppo di applicazioni utilizzando linguaggi come Turbo Pascal o C è data dal fatto che il metodo di sviluppo è ancora tradizionale: lo sviluppatore deve scrivere un programma per disegnare le finestre.

Visual Basic nello sviluppo di un'applicazione segue invece i seguenti passi:

- a) disegno dell'interfaccia utente, cioè delle finestre e dei loro componenti, tramite uno strumento di disegno interattivo;
- b) inizializzazione delle proprietà che costituiscono attributi o caratteristiche di ogni elemento dell'interfaccia;
- c) sviluppo del programma.

Il programma viene quindi sviluppato dopo aver disegnato l'interfaccia, chiaramente è possibile tornare al passo precedente in qualsiasi momento.

Questa linea di sviluppo fa sì che, prima di scrivere anche una sola riga di programma, sarà possibile far funzionare l'applicazione, ovviamente solo per ciò che riguarda la visualizzazione dell'interfaccia utente.

5.2 Disegno dell'interfaccia utente

Le applicazioni Visual Basic sono costituite da finestre, come la maggior parte delle applicazioni Windows. La finestra principale dell'applicazione, le finestre secondarie o finestre di dialogo (modali e non modali) vengono chiamate *form*. Su tali form vengono posizionati i *control*, elementi dell'interfaccia utente: campi di inserimento testo, bottoni, liste, ecc..

L'ambiente Visual Basic comprende un insieme di strumenti che permettono di disegnare i control, di posizionarli, di modificarne le dimensioni e così via.

Tutto ciò viene effettuato in maniera intuitiva e visiva, il risultato di un'azione sarà immediatamente visibile sullo schermo. Lo sviluppatore dell'applicazione vede quindi in qualunque momento come essa apparirà all'utente finale durante il suo utilizzo.

5.3 Inizializzazione delle proprietà

Ogni oggetto creato, form o control che sia, dispone di proprietà che gli appartengono; alcune proprietà sono comuni a quasi tutti i control (testo, colore, ecc.), mentre altre sono tipiche di un determinato tipo di control.

Le proprietà di un oggetto hanno ad ogni istante un valore che caratterizza l'oggetto stesso. Tali valori vengono inizializzati nel momento in cui l'oggetto viene creato (ogni proprietà ha inizialmente un valore di default) e possono essere modificati in due modi: agendo direttamente sulla rappresentazione dell'oggetto e sulle relative proprietà, in fase di costruzione (disegno) dell'interfaccia, oppure tramite opportune istruzioni che modificano il valore di una proprietà, in fase di esecuzione. Va però sottolineato che alcune proprietà possono essere modificate e/o lette solo in uno dei due modi. Pertanto, dopo aver creato una form o un control, il progettista può modificarne il valore delle proprietà al fine di rendere l'oggetto in questione rispondente ai requisiti richiesti.

5.4 Sviluppo del programma

I due passi precedenti permettono di disegnare l'interfaccia dell'applicazione, che a questo punto può funzionare, ma non è in grado di fare molto, se non permettere di vedere come l'interfaccia si presenta all'utente; è necessario sviluppare il codice per far corrispondere all'azione sugli elementi dell'interfaccia un'elaborazione specifica.

La struttura generale di un programma in Visual Basic è molto diversa da quella di un'applicazione classica. Il programma viene associato ad ogni oggetto dell'applicazione, form o control, in particolare agli eventi ai quali può rispondere un oggetto.

La programmazione delle applicazioni in Visual Basic avviene *per eventi*, in contrapposizione alla programmazione *lineare* tradizionale.

La struttura generale di un programma tradizionale può essere vista come segue:

- punto di entrata nel programma;
- punto d'uscita;
- fra l'entrata e l'uscita una sequenza di istruzioni con eventuali salti;
- possibilità di effettuare chiamate a procedure o funzioni.

La *programmazione per eventi* differisce dallo schema appena descritto, non esiste più una sequenza di esecuzione che si svolge dall'inizio alla fine del programma, un'applicazione Visual Basic è costituita da un insieme di procedure indipendenti fra loro.

Una procedura, scritta in Basic, viene associata ad un oggetto, cioè ad uno degli elementi della form: la form stessa, un bottone, una lista, un campo di inserimento testo, ecc. La procedura viene richiamata da Visual Basic nel momento in cui ha luogo un evento per l'oggetto corrispondente.

Se nella procedura, preposta a gestire un tipo di evento per un determinato oggetto, non è stato scritto alcun programma, quando l'evento ha luogo non accade nulla.

Per sviluppare il programma di un'applicazione è opportuno determinare gli eventi ai quali si desidera reagire e gli oggetti per i quali l'evento ha luogo, ciò determina le procedure che compongono il programma. Si capisce quindi l'importanza del ruolo svolto dall'utente che si trova all'origine della maggior parte degli eventi (click o doppio click su un control, trasporto di un control, posizionamento su un control, cambiamento del contenuto di un campo di immissione testo, ecc.).

6. Presentazione del programma HDRC25 2.01

Il programma permette l'acquisizione in tempo reale di segnali analogici di banda compresa tra 0 e circa 10 kHz, consentendo la memorizzazione dei campioni, sempre in tempo reale, su Hard Disk sotto forma di file. Successivamente è possibile intervenire sul file di campioni per eseguire operazioni tipiche di programmi di questa classe, cioè operazioni di editing del segnale (Taglia, Copia, Incolla ecc.).

Si è tralasciato, in previsione di sviluppi futuri, l'implementazione di funzionalità per la creazione e l'esecuzione di playlist, per il trattamento e la sovrapposizione di segnali audio.

All'avviamento del programma viene presentata all'utente la finestra principale, visibile in figura 6.

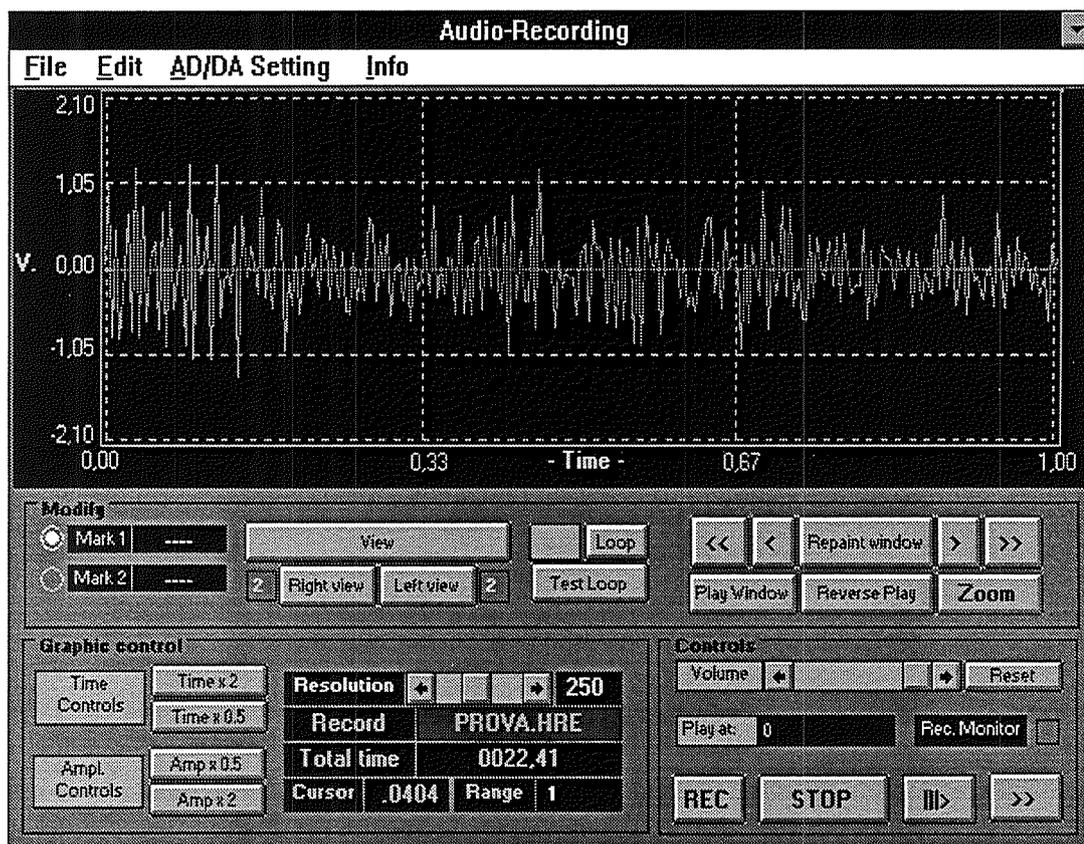


Fig. 6 Finestra principale.

E' possibile notare che i controlli disponibili a questo livello sono stati raggruppati secondo le funzionalità espletate.

Abbiamo dunque i seguenti gruppi di controlli:

- controlli per la gestione della visualizzazione (*Graphic Control*);
- controlli per la gestione della selezione (*Modify*);
- controlli per la gestione delle operazioni di acquisizione e riproduzione (*Controls*).

Nel menù si trovano le voci relative a:

- gestione dei file (Menu File);
- gestione delle operazioni di editing (Menu Edit);
- gestione dei controlli per i dispositivi di acquisizione (Menu AD/DA Setting);
- comando per la finestra informazioni (Menu Info).

Nella parte superiore della finestra è visibile infine una finestra destinata alla visualizzazione del segnale acquisito.

6.1 Modalità d'uso.

All'avviamento il programma si presenta con alcuni controlli disabilitati (menu AD/DA Setting) e inoltre la finestra dei campioni non contiene la rappresentazione grafica di alcun file. Per poter lavorare con il programma è necessario impostare un file di lavoro. Tale file può essere uno dei file creati durante le precedenti sessioni di lavoro oppure un file creato durante la sessione corrente.

La creazione di un file avviene tramite il comando File>Select_File. Viene visualizzata la finestra FILES (figura 7). Con il mouse si sceglie la directory che dovrà contenere il nuovo file. Scegliendo il bottone New, viene visualizzata una finestra di dialogo che richiede l'immissione del nome del nuovo file di lavoro. Il nome del file può essere lungo al massimo 8 caratteri. Il programma assegna automaticamente l'estensione ".HRE" al file, che diventa il file di lavoro corrente.

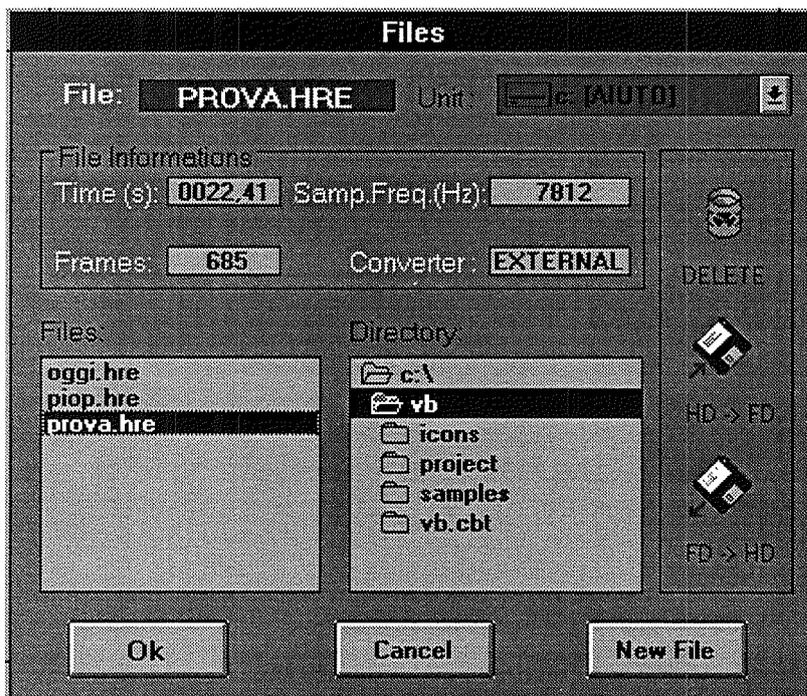


Fig. 7 Finestra per l'impostazione del file di lavoro.

A questo punto la finestra di dialogo viene chiusa e si rientra nella finestra principale del programma.

Il nome del file viene visualizzato nella casella "Record". Nella casella "Total time" è leggibile la durata della registrazione che all'inizio è, ovviamente, nulla, mentre nella casella "Record" troviamo il nome.

Una volta impostato il file di lavoro, diventano attivi sia il check-box "Rec Monitor", presente nel gruppo "Controls", sia la voce di menu relativa all'impostazione degli apparati di conversione "AD/DA Setting".

Un nuovo file è composto da un Header da 512 bytes, (128 doublewords), seguito dai campioni.

L'Header del file contiene varie informazioni, tra le quali ad esempio, quelle relative alla sua lunghezza, alla frequenza di campionamento con cui sono stati acquisiti i campioni, e al tipo di dispositivo di conversione adottato (la scelta è tra quello presente a bordo della scheda LeonardC25 e uno esterno).

I valori di default contenuti nell'Header sono:

Convertitore	Guadagno del TLC	Guadagno porta analogica	Filtro Passa basso	Filtro SIN(x)/x	Freq. di conversione
Interno	3 Volt Full Scale	2 X	Off	Off	18519 kHz
Esterno	-	-	-	-	8000 kHz

Attivando il check-box "Rec-Monitor" è possibile ascoltare il segnale analogico presente in ingresso alla scheda LeonardC25. E' possibile regolare il guadagno dello stadio di ingresso della scheda agendo sul menu "AD/DA Setting" viene presentata la finestra seguente:

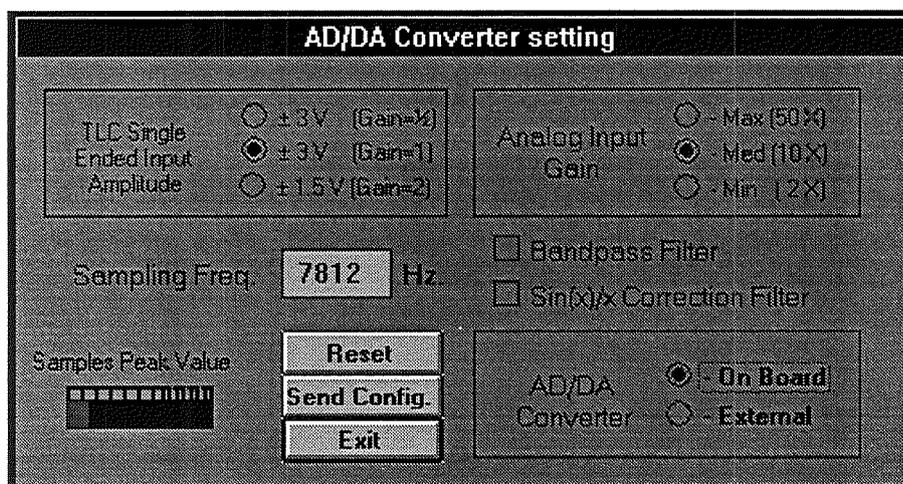


Fig. 8 Finestra per l'impostazione dei parametri del convertitore.

Nel caso si stia lavorando con il convertitore TLC32044 (cioè quello montato sulla scheda), sono disponibili i controlli per intervenire sia sul guadagno interno del dispositivo, sia sul guadagno a monte, cioè proprio all'ingresso dei segnali analogici nella scheda. Si notano inoltre due controlli di tipo check-box: il primo per l'inserimento o l'esclusione del filtro passa basso; il secondo per l'inserimento o l'esclusione del filtro di correzione del ripple fuori banda Sin(x)/x.

Una volta modificate le impostazioni visualizzate è possibile testare la configurazione agendo sul bottone "Send Config.". Si ha in tal modo immediato riscontro auditivo delle modifiche apportate alla configurazione. All'interno della finestra è presente anche un controllo visivo per monitorare i picchi del segnale. Ogni 100 campioni tale controllo visualizza il loro valore massimo.

Il bottone di "Reset" riporta i controlli ai valori di default.

Il bottone "Exit" imposta il convertitore e provoca la chiusura della finestra.

I controlli descritti fino a questo punto influenzano direttamente il convertitore montato sulla scheda LeonardC25.

Agendo sul gruppo di bottoni radio "AD/DA Converter" è possibile selezionare un dispositivo di conversione esterno, connesso tramite la porta seriale presente sulla scheda LeonardC25. In caso di connessione mancante o difettosa il programma può presentare mal funzionamenti, in quanto esso è basato sui segnali che transitano attraverso la porta seriale.

Dopo aver impostato il convertitore seriale presente sulla scheda ed aver predisposto i controlli di guadagno ai corretti valori, il bottone "Exit" permette di chiudere la finestra "AD/DA Converter Setting" e di tornare alla finestra principale.

L'attivazione del check-box "Rec Monitor" permette, nel caso di segnali analogici in banda audio, di controllare il segnale presente all'ingresso del convertitore. Si può dunque scegliere con un certo grado di precisione l'istante in cui iniziare ad acquisire il segnale.

Ciò avviene tramite il bottone "REC" presente nel gruppo "Controls". La fase di acquisizione è confermata dall'avanzamento del valore presente nella casella "Play at".

La pressione del bottone "STOP" pone fine al processo: nella casella "Total time" è leggibile la durata della registrazione. Alla fine di una acquisizione, nella finestra per la visualizzazione del segnale normalmente il segnale acquisito non appare. Per visualizzarlo si può procedere in vari modi:

- a) agire sul controllo "Reset", presente nel gruppo "Controls";
- b) spostarsi nella finestra con i bottoni di direzione adiacenti il bottone "Repaint window";
- c) impostare un valore nella casella "Play at" e premere invio (oppure fare click su "Refresh"): la visualizzazione del segnale avrà inizio dall'istante temporale impostato.

Nel primo caso la visualizzazione parte dall'istante zero; nel caso b) ci si muove all'interno della finestra di intervalli pari al valore della casella "Range" se si preme la doppia freccia, altrimenti pari a ("Range")/2.

Per modificare la visualizzazione si può agire sui controlli che modificano l'ampiezza del segnale rappresentato ("Amp x 2" e "Amp x 0.5"), sui controlli relativi al tempo ("Time x 2" e "Time x 0.5") e con il bottone di "Zoom".

I primi modificano la finestra temporale visualizzata e aggiornano conseguentemente la casella "Range". Se ad esempio in "Range" è presente il valore "1" ciò vuol dire che nella finestra si ha la visualizzazione di 1 secondo di suono. Facendo click su "Time x 0.5" si otterrà il grafico di 0.5 secondi di suono e così via. Il segnale verrà graffiato a partire dall'istante in "Play at". Facendo click su "Zoom" è possibile ingrandire una porzione del segnale visualizzato: trascinando il mouse sul grafico si apre un rettangolo con il quale si può scegliere la porzione di grafico di interesse.

La risoluzione del grafico può essere impostata con la barra di scorrimento relativa a "Resolution" (gruppo "Graphic Controls"), e va da un minimo di 50 ad un massimo di 500 punti.

I bottoni "Play Windows" e "Reverse play" presenti nel gruppo "Modify" permettono la riproduzione di quanto visualizzato nella finestra, rispettivamente in maniera diretta o inversa.

La selezione di una porzione di brano è resa possibile dai pulsanti radio etichettati con "Mark1" e "Mark2" presenti nel gruppo "Modify". Selezionando ad esempio "Mark1" e facendo click sul grafico, appare una barra verticale nella posizione del cursore. Tale posizione viene riportata anche nel controllo relativo a "Mark1". La barra rappresenta il limite sinistro della selezione. Analogamente, agendo su "Mark2" è possibile impostare il limite destro. A questo punto dunque il brano selezionato è quello compreso tra le due barre verticali. Oltre alle classiche operazioni di editing (Taglia, Copia, Incolla, ecc..) presenti nel menu "Edit", ed effettuabili sulla parte di brano selezionata, nel gruppo "Modify" esistono dei controlli per la gestione di loops sulla selezione. In particolare, impostando un valore intero maggiore di 2 nella casella adiacente il bottone "Loop" e agendo sul bottone "Test Loop" si ottiene l'esecuzione del brano con loop.

Il brano ha inizio a partire da X secondi prima della barra di selezione sinistra, dove X è il valore presente nella casella adiacente il bottone "Left view", prosegue ciclando tra le barre di selezione per il numero di cicli impostato, e poi prosegue per Y secondi, dove Y è il valore impostato nella casella adiacente il bottone "Right view".

Tra i bottoni "Test loop" e "Loop" vi è una differenza: il primo provoca solo l'ascolto dell'effetto di loop mentre il secondo aggiorna il file di lavoro in maniera permanente.

Il bottone "View" allinea la finestra di visualizzazione ai margini impostati dai marks. Ad esempio se la finestra va da 0 sec. a 3 sec. e i marks sono impostati a 1 secondo e a 2 secondi, facendo click su "View" la nuova finestra visualizzerà da 1 sec. a 2 sec.

Facendo click con il mouse sulla barra di selezione corrispondente al mark attivo, la barra di selezione scompare ed il valore nella casella corrispondente viene sostituito da "----".

Il bottone di riproduzione si trova a destra del bottone di "STOP". Facendovi click si dà il via alla riproduzione del brano acquisito, a partire dall'istante visualizzato in "Play at". Con la barra di scorrimento relativa al volume si controlla il volume di riproduzione.

L'opzione per il convertitore esterno nella finestra "AD/DA converter setting" permette di lavorare con dispositivi di conversione di caratteristiche differenti dal TLC32044 presente sulla scheda LeonardC25. Il programma è stato strutturato per essere impiegato con il modulo di conversione AD/DA HI-FI Stereo. Per quanto riguarda questa modalità di impiego non c'è molto da aggiungere a quanto già descritto nel precedente paragrafo. Il programma si utilizza in maniera analogica. La differenza sta nel fatto che in questo caso l'acquisizione viene fatta attraverso il modulo HI-FI con frequenza di campionamento pari a circa 39kHz. La riproduzione avviene invece tramite il convertitore presente sulla scheda, la cui frequenza di campionamento è impostata a circa 7 kHz.

Conclusioni

Il sistema di HDR realizzato, pur essendo caratterizzato da un'insieme di funzionalità ridotto, fornisce uno strumento sufficiente per effettuare delle valide esperienze di uso nel campo dell'elaborazione digitale di segnali audio.

Dato il basso costo dell'hardware richiesto il programma HDRC25 rappresenta di fatto un sistema facilmente acquisibile per un primo approccio all'elaborazione digitale dei segnali in banda audio su personal computer, si ritiene utile quindi il suo utilizzo per scopi didattici in laboratori di istituti di elettronica che trattano l'introduzione alle tecniche digitali, laboratori di scuole di musica elettronica (Conservatori, ecc.), utenti consumer, ecc..

Va infine sottolineata l'importanza del lavoro svolto, per la realizzazione di questo progetto, in termini di esperienza maturata da utilizzare per ulteriori sviluppi del sistema stesso.

Bibliografia

- [BELLADONNA-92] A. Belladonna, M. Graziani, "*Musica Elettronica, AudioMedia: un sistema di hard disk recording per Apple Macintosh*", AUDIOREVIEW, n. 113, febbraio 1992.
- [BERTINI-91] G. Bertini, L. Tarabella, G. Bacchiocchi, M. Balestrieri, "*Tecniche di gestione delle risorse in sistemi multi-DSP per la sintesi e l'elaborazione in tempo reale di segnali audio*", Nota interna IEI/CNR, agosto 1991.
- [BERTINI-92] G. Bertini, D. Fabbri, "*Progetto e funzionamento del modulo di acquisizione ed elaborazione di segnali in tempo reale LeonardC25-ver.02*", Nota interna IEI/CNR, B4-57 novembre 1992.
- [BERTINI-93a] G. Bertini, D. Fabbri, "*MULTIC25, un sistema multi-DSP con schede LeonardC25*", Nota interna IEI/CNR, B4-14 maggio 1993.
- [BERTINI-93b] G. Bertini, D. Fabbri, L. Tarabella, "*MuStC25 una stazione di lavoro musicale con il sistema MULTIC25. Descrizione e manuale operativo*", Nota interna IEI/CNR, B4-20 giugno 1993.
- [BERTINI-93c] G. Bertini, D. Fabbri, M. Marani, L. Tarabella, "*MUSTC25, stazione di lavoro musicale con schede LeonardC25*", Atti del X Colloquio di Informatica Musicale, pag. 307-310, Milano 2-4 dicembre 1993.
- [CELIN-93] G. Celin, E. Tanteri, "*Microsoft Sound System, hard + soft = microsoft*", Multimedia PC, n. 18, febbraio 1993.
- [CHIANURA-93] P. Chianura, "*Mixer/registratore a sei tracce a cassette VESTAX MR-66S*", SM, n.158, ottobre 1993.
- [CURIEL-93] G. Curiel, "*Hard disk e dintorni*", SM, n. 158, ottobre 1993.
- [DANIELI-93] G. Danieli, "*A ogni scheda il suo software*", Elettronica Oggi, n. 164, luglio/agosto 1993.
- [DIGIDESIGN-90] DigiDesign, "*Sound Designer 2, User's guide*", 1990.
- [DORIA-93] A. Doria, "*Suoni d'autore*", SM, n.157, settembre 1993.
- [FRANTZ-93] G. Frantz, "*Programmare con Visual Basic 2*", APOGEO, 1993.
- [GIANNANGELI-93a] G. Giannangeli, "*Sistema di hard disk recording Korg SoundLink, registrando con il software*", SM, n. 154, maggio 1993.
- [GIANNANGELI-93b] G. Giannangeli, "*Registratore DAT portatile: Sony TCD-D10PRO-2*", SM, n. 157, settembre 1993.

- [MARANS-92] M. Marans, "*Registrazione digitale, l'imbarazzo della scelta: gioie e dolori delle nuove tecnologie*", parte 1, SM-Elettronica, 1992.
- [MOORER-90] J. A. Moorer, "*Hard-disk recording and editing of digital audio*", AES, presented at the 89th convention, 1990 september 21-25 Los Angeles.
- [PANIZZA-93a] M. Panizza, "*Creative Labs, Sound Blaster 16 ASP, hardware per IBM compatibili*", SM, n. 157, settembre 1993.
- [PANIZZA-93b] M. Panizza, "*Sistema di hard disk recording per IBM compatibili: DigiDesign Session 8*", SM, n. 158, ottobre 1993.
- [PEROTTI-93] G. Perotti, "*Sistema hw/sw di hard disk recording: Motu Digital Performer e Digital Waveboard*", SM Computer e Musica, n. 154, maggio 1993.
- [PETRONI-93] F. Petroni, "*Microsoft Windows Sound System*", MCmicrocomputer, n. 126, febbraio 1993.
- [PICCOLI-93] M. Piccoli, "*Dove va l'home recording*", SM, n. 158, ottobre 1993.
- [SHAFER-73] R. W. Shafer, L. R. Rabiner, "A Digital Signal Processing Approach to Interpolation", IEEE, pag. 692-702, giugno 1973.
- [TARABELLA-92] L. Tarabella, G. Bertini, A. Caioli, A. Guerra, "*Informatica e Musica*", Jackson Libri, 1992.
- [TMS-87] "*Second Generation TMS320 User's Guide*", Texas Instruments, 1987.