Bit-Plane, Bitmap e Color Palette

di Giuseppe Cardinale Ciccotti

La necessità di visualizzare immagini ad un numero sempre crescente di colori, è risolta progettando architetture e dispositivi di elevata efficienza; di pari passo si sviluppano tecniche e organizzazioni per rispondere alle esigenze degli utilizzatori dei sistemi grafici. Questa montante complessità si riflette in specifiche di elevate bande passanti per i componenti elettronici utilizzati, al limite delle possibilità tecnologiche degli stessi; si devono pertanto trovare delle soluzioni alternative per gestire in modo efficiente le grosse richieste di memoria necessarie ad immagini ad alta risoluzione ed elevato numero di colori. Una strada è quella di parallelizzare l'organizzazione del dispositivo grafico in modo da soddisfare la richiesta di un throughput sempre maggiore

Color Palette

La più comune organizzazione di un dispositivo grafico è senz'altro il frame buffer, una zona di memoria i cui bit sono in corrispondenza biunivoca con i pixel del monitor. È noto che per visualizzare un determinato colore dobbiamo fornire al monitor stesso tre diverse tensioni per i cannoni elettronici del rosso, del verde e del blu. Si deve quindi organizzare il frame buffer in modo da allocare le informazioni relative ai colori, tali informazioni digitali saranno poi convertite in tensioni analogiche per guidare i cannoni elettronici. La cosa più semplice da fare è certamente prevedere di assegnare, per esempio, un byte a ciascun pixel: in tal modo possiamo individuare 256 colori diversi. Del tutto autonomamente potremo scegliere la corrispondenza tra i valori 0-255 del byte e i colori che desideriamo ottenere sul monitor. A valle del frame buffer saranno presenti tre convertitori digitali analogici (DAC), uno per cannone che convertono il byte in tensione analogica. Il problema che sorge è quello intuitivo di come distribuire i bit del byte ai DAC: infatti otto non è multiplo di tre e perciò uno dei DAC non sarà a tre bit ma a due. Ciò implica che il cannone guidato da quel DAC fornirà soltanto quattro intensità diverse rispetto agli altri due che avranno otto possibilità. Come intuitivamente ci si rende conto, tale organizzazione condiziona in modo arbitrario, in fase di progettazione, lo spettro dei colori a disposizione dell'utente, quindi è necessario trovare una maniera alternativa per l'allocazione dei bit destinati al colore.

Una prima risposta consiste nel considerare un numero di bit multiplo di 3 per gli indici dei colori, tuttavia si preferisce evitare allocazioni che non riempiano i byte o per lo meno i nibble per evitare problemi di accesso alla memoria. Infatti allocando 9 bit per pixel, vale a dire un byte più un bit, ci troviamo a scegliere fra le seguenti alternative: il sacrificio dei restanti sette bit del secondo byte oppure impaccando tutti i bit in byte consecutivi, la complicazione di maggiori tempi di accesso ai dati relativi ai pixel, è necessario infatti eseguire due estrazioni di bit, tra l'altro variabile secondo la posizione del pixel. Questa organizzazione è perciò adottata soltanto quando si assegnano tre byte per pixel, disponendo in uscita di 16777216 colori diversi. Una delle alternative consiste nel prevedere una tavola di conversione che riceve in ingresso l'indice del colore e restituisca in uscita un numero di bit multiplo di 3 da mandare ai DAC. Questi dispositivi vengono chiamati «Color Palette»; sono costituiti da un numero di registri pari ai colori disponibili nel frame buffer, ognuno dei quali è lungo un numero di bit multiplo di tre maggiore del numero di bit assegnati a ciascun pixel.

Durante la scansione del frame buffer per formare l'immagine video, i valori contenuti nel frame buffer sono mandati alle linee di indirizzo delle Color Palette che sono abilitate in lettura, le linee di uscita sono connesse agli ingressi dei DAC. Visto che il numero di bit della Color Palette è maggiore di quelli del frame buffer, variando i valori del contenuto dei registri si può disporre di più colori di quanti il solo frame buffer ne possa fornire. Saranno infatti visualizzabili 2^f colori contemporaneamente da 2^c possibili (f è pari al numero di bit per pixel del frame buffer e c'è il numero di bit di ciascun registro della Color Palette). Tali dispositivi allora risolvono anche il problema di poter disporre di un grande numero di colori con frame buffer non eccessivamente grandi; questo sistema sembra funzionare abbastanza bene nelle applicazioni più comuni perché spesso si ha necessità di po-

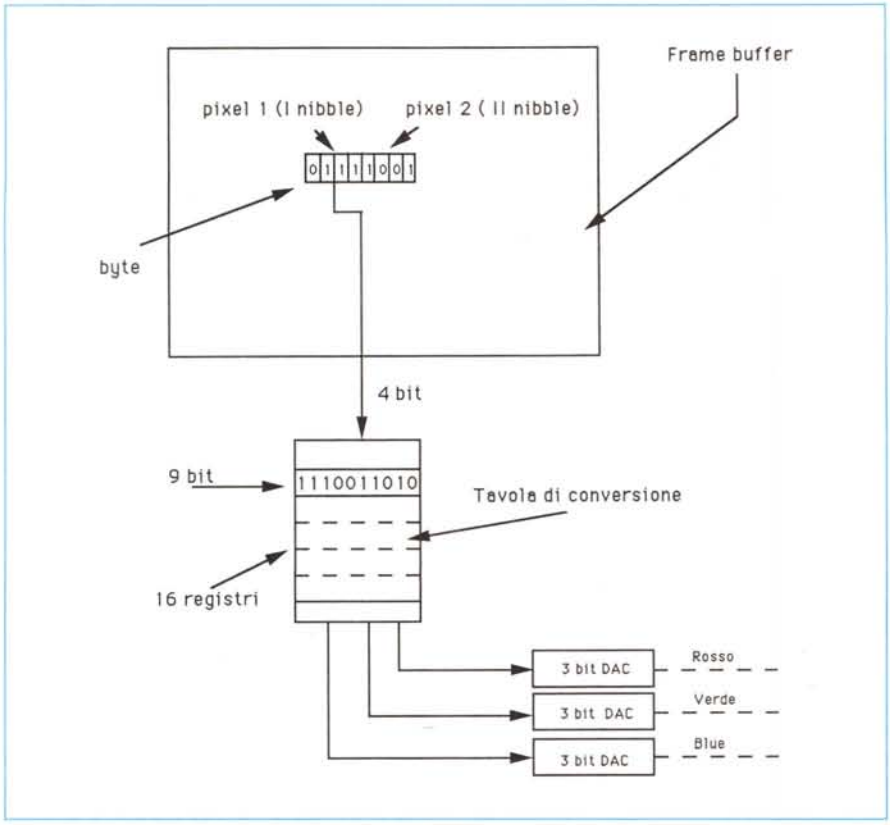

Figura 1 - Frame buffer a «byte impaccato». Ogni colore occupa un certo numero di bit e un byte contiene informazioni relative a più pixel. A valle è posta una tabella di conversione che permette di ottenere una scelta di colori maggiore di quella possibile con il solo frame buffer.

chi ma precisi colori. In figura 1 è mostrata un'architettura schematica di frame buffer con Color Palette e DAC. I dispositivi in commercio nella maggior parte dei casi integrano anche i convertitori che devono essere necessariamente veloci perché i tempi disponibili per la conversione sono molto ristretti e dipendono dal numero di pixel e dalla frequenza di refresh verticale; per fissare le idee: con risoluzione 1280×1024 e 60 Hz di refresh si hanno disponibili per ciascun pixel 12 ns circa pari ad una frequenza di 79 MHz se invece la risoluzione è di 2048×2048 la frequenza sale a 250 MHz circa. Questo è il motivo per il quale molto spesso le Color Palette sono realizzate in tecnologia ECL.

L'organizzazione bit-plane

Un'altra possibilità rispetto all'utilizzo di Color Palette consiste nel disporre il frame buffer a «piani di bit» o come più comunemente si dice a «bit-plane». Secondo questa architettura, i bit relativi ad uno stesso pixel, non vengono impaccati nello stesso byte ma sono separati in locazioni diverse in modo che in un byte siano allocati 8 bit relativi a 8 pixel consecutivi, riferitevi alla figura 2. Come si vede i bit di ciascun pixel sono allocati in zone diverse della memoria, si può pensare al frame buffer a bit-plane come ad una struttura tridimensionale, una pila di frame buffer monodimensionali ognuno dei quali fornisce un bit ai DAC a valle. Naturalmente i piani di bit saranno allocati in memoria ad indirizzi base diversi ma fra i bit omologhi si manterrà lo stesso ordinamento. In questo modo non si hanno i problemi visti precedentemente poiché si possono disporre un numero qualsiasi di bit per pixel, prevedendo altrettanti bit-plane.

Tuttavia i tempi necessari per una qualsiasi operazione sono molto elevati in quanto per ogni pixel bisogna accedere in genere ad un numero di byte pari al numero dei bit-plane. Può allora accadere se non si struttura l'architettura in modo opportuno, che il tempo complessivo necessario all'operazione stessa sia comparabile con il tempo di ritraccia del quadro. Si ottiene quindi l'apparizione di colori fittizzi mentre sono stati modificati alcuni bit-plane e non è ancora avvenuto l'accesso agli altri. Adottando allora uno schema a bit-plane è necessario ricorrere ad un accesso multiplo della memoria. L'architettura a bit-plane sarà quindi in genere costituita da più canali di accesso alla memoria, in tal modo si possono parallelizzare gli accessi ai piani di bit del frame buffer, e portare a termine le operazioni su essi simultaneamente. In figura 3 potete osservare una schematizzazione della struttura interna del processore grafico AMD 95C60 che gestisce quattro bit-plane contemporaneamente, per controllarne di più è previsto che vengano aggiunti altri processori.

La scansione del frame buffer necessaria al rinfresco del monitor soffre dello stesso problema, è possibile che non riesca a fornire tutte le informazioni necessarie ad accendere il pixel nel tempo richiesto a causa dei molteplici accessi; tuttavia, in questo caso il fatto che i pixel debbano essere rinfrescati in ordine, ci permette di sfruttare l'accesso a byte o a word che viene fatto per leggere la memoria. Accedendo in un solo ciclo di lettura, a 8 o 16 bit, bastano un numero di cicli pari al numero di bit-plane per poter rinfrescare 8 o 16 pixel. Sarà naturalmente necessario un dispositivo che serializzi le informazioni dei pixel. Questa tecnica può essere sfruttata naturalmente anche in scrittura perciò risulta molto semplice riempire zone rettangolari del frame buffer. Tuttavia se i limiti della zona non sono allineati al byte del frame buffer sono necessarie numerose istruzioni aggiuntive per costruire i byte di bordo.

L'operazione «bitblt»

Tale problema di velocità è ancora più grave quando si voglia spostare un'area rettangolare da una zona all'altra del frame buffer. Bisognerà copiare bit a bit l'area sorgente in quella destinazione e poi cancellare l'area sorgente.
Se le due aree non sono allineate al byte o alla word, né tantomeno hanno lo stesso offset rispetto ai byte dove

sono posizionati i limiti dell'area, ai tempi di lettura e scrittura della memoria, si aggiungeranno i tempi di estrazione dei bit della sorgente, dell'inserimento dei bit della destinazione e ancora di un inserimento per la sorgente nell'operazione di cancellazione.

Questa operazione di trasferimento detta «bit block transfer» e indicata col termine abbreviato «bitblt», ha ormai assunto un'importanza rilevante nelle caratteristiche dei dispositivi grafici attuali, basta pensare che i sistemi di interfaccia a finestre richiedono principalmente operazioni di questo tipo. Le case costruttrici di dispositivi elettronici hanno fornito risposte diverse ma sempre molto efficaci all'esigenza da parte dell'utente di disporre di un'operazione semplice e veloce di bitblt. Generalmente viene integrato nel controllore grafico un «barrel shifter» in figura 4, che è in sostanza uno shift register capace di spostare il contenuto di un byte o word a piacere in un solo colpo di clock. Naturalmente è presente una logica di controllo che permette di calcolare il numero di shift da effettuare e gli eventuali riporti. L'utente perciò vede l'operazione di bitblt come trasparente, fornendo soltanto le coordinate della sorgente, della destinazione e le dimensioni dell'area da spostare.

Alcune case come la National prevedono addirittura un barrel-shifter per bit-plane, mentre l'approccio per esempio di Motorola prevede che l'operazione di bitblt sia eseguita con istruzioni di estrazione ed inserimento fornite dal potente set del 68020 e 68030. C'è comunque da notare che la Motorola è l'unica fra le maggiori case produttrici di microprocessori a non avere in catalogo un controllore grafico appositamente progettato. Nel campo dei personal computer, l'Amiga ha per primo integrato nel suo chip custom un dispositivo che esegue assai rapidamente il bit block transfer e viene perciò pittorescamente chiamato «Blitter»; nel PC e compatibili, invece non è previsto un dispositivo dedicato a tale scopo e perciò è il processore centrale che si incarica di qualsiasi operazione sul frame buffer, tra l'altro poiché la VGA è vista dal processore come un set di registri, un'operazione di bitblt è assai più lenta che in un sistema con la CGA che è mappata direttamente in memoria.

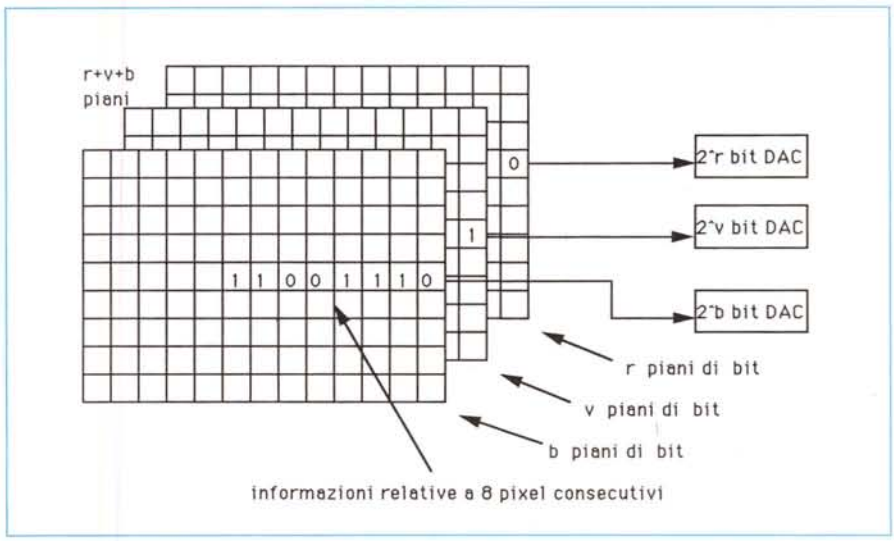

Figura 2 - Organizzazione a bit-plane. Ogni piano di bit contribuisce ad un solo bit per il valore finale del pixel.

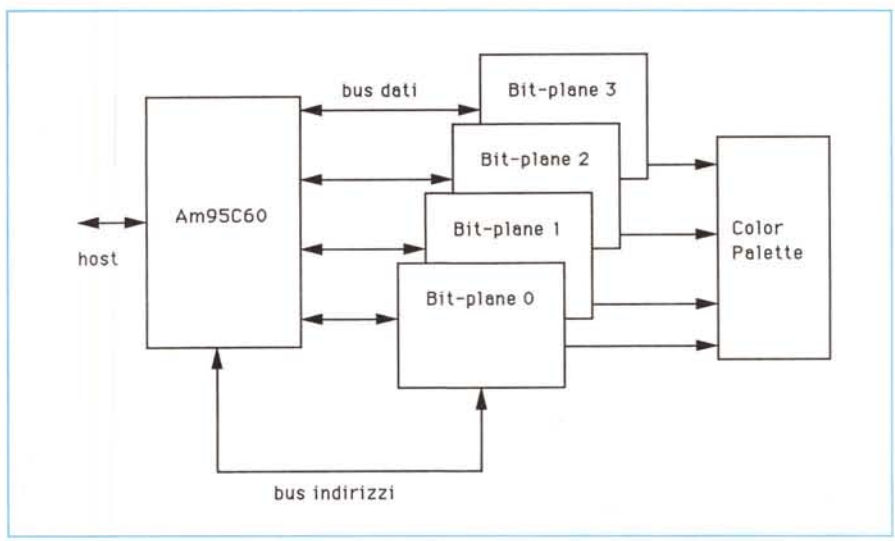

Figura 3 - Schema tipico di applicazione dell'AM 95C60. Come si può vedere l'accesso ai quattro bit-plane disponibili avviene in parallelo su quattro bus separati.

Architettura per le interfacce a finestre

Da quando la Xerox ha inventato il sistema WIMP, Window Icon Mouse Pointer, le interfacce grafiche a finestre si sono moltiplicate e non esiste sistema oggi che non le preveda.

Un tale sistema può essere implementato senza grosse complicazioni in software, come per esempio avviene per MS-Windows oppure tramite dispositivi che facilitino la gestione delle finestre stesse.

Un approccio totalmente software naturalmente non potrà essere molto veloce specie se il frame buffer ha numerosi bit-plane: la quantità di dati da spostare è veramente elevata.

Abbiamo già detto come i barrel shifter possono efficacemente effettuare lo spostamento di zone rettangolari, è per-

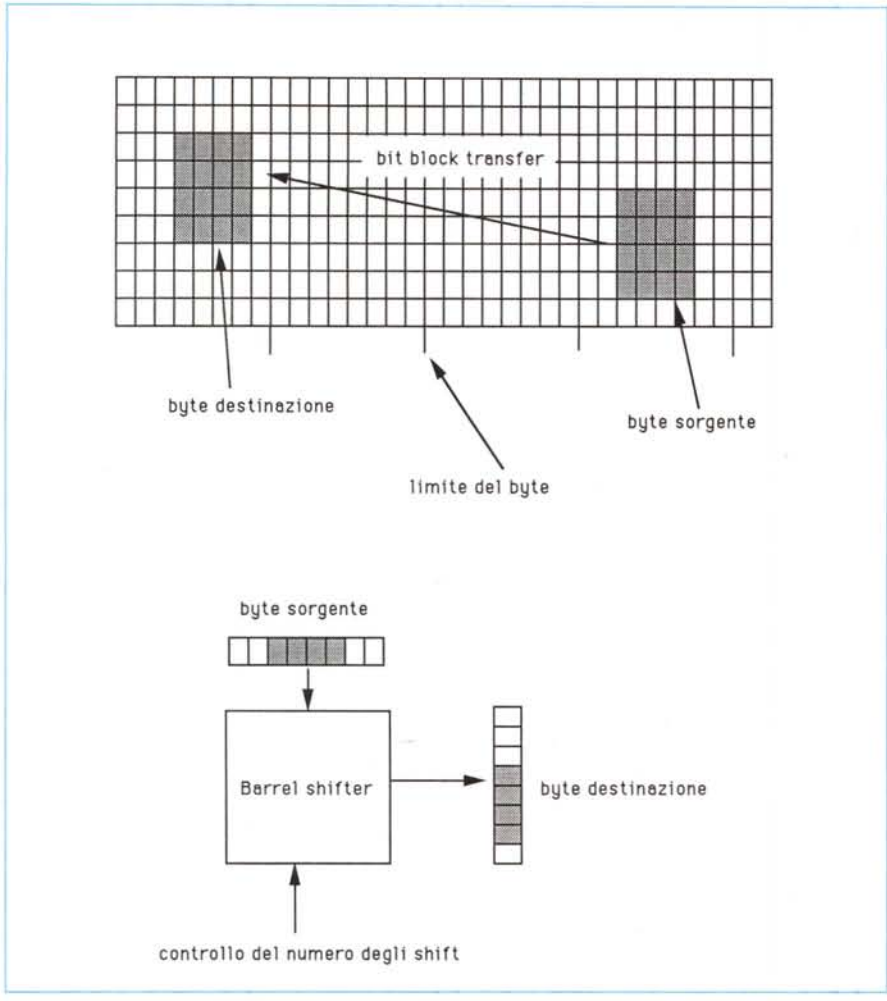

Figura 4 - Operazione bit block transfer e dispositivo barrel shifter che la esegue direttamente in hardware.

ciò naturale che la maggior parte dei costruttori basi le istruzioni per la gestione delle finestre sulle capacità di tale componente.

Sono però possibili altri approcci come quello proposto ad Intel che per l'82786 ha previsto di gestire le finestre mediante una serie di registri. Le aree delle finestre sono fisicamente allocate in zone diverse del frame buffer, rispetto all'immagine che risulta sul monitor; nei registri viene invece mantenuta la posizione che ogni finestra deve avere sul monitor e la sua priorità per risolvere eventuali coperture.

Lo schermo è poi suddiviso in strisce e per ogni striscia sono mantenute le informazioni delle finestre che appartengono a quella striscia, queste sono l'indirizzo di memoria della parte di finestra che cade nella striscia considerata e il numero di pixel orizzontali da considerare. Il controllore video, durante il rinfresco del video non fa altro che accedere a questa lista e prelevare i dati dal frame buffer agli indirizzi specificati, ricavando così i dati per i pixel. In questo modo è molto semplice e veloce trasferire una finestra, poiché non c'è necessità di trasferire fisicamente tutto il blocco di informazioni ad esso associato ma basta spostare un puntatore. Inoltre specificando per riga il numero di pixel orizzontali è possibile ottenere finestre dal bordo irregolare.

Questo approccio consente naturalmente le operazioni di bitblt che pertanto non sono supportate da barrel shifter. Naturalmente è necessario per ragioni di tempo, che il numero di finestre per riga sia limitato e precisamente non si possono avere nell'82786 più di 16 finestre per striscia. In questo caso quindi, il frame buffer non è l'immagine del video ma soltanto una zona di memoria dedicata alle informazioni grafiche a cui ha accesso il controllore video.

La bitmap

Lo stesso concetto di finestra si può applicare a tutto lo schermo, considerando appunto questo come una finestra, e predisponendo un frame buffer di dimensioni maggiori dello schermo stesso. In questo modo è possibile, per esempio, effettuare scroll globali dello schermo in maniera velocissima in quanto è sufficiente comunicare al controllore video l'indirizzo dell'area da cui prelevare i dati necessari alla visualizzazione.

In letteratura si indica questo modo di gestire il frame buffer come «superbitmap», dato che normalmente la memoria del frame buffer è detta «bitmap» ad indicare che è proprio una «mappa» di bit dell'immagine video.

Come si vede nei sistemi evoluti si ha la necessità di prevedere dei controlli video, i dispositivi che fisicamente si occupano di scandire la memoria del frame-buffer per generare l'immagine video, programmabili in modo da ottenere elevata flessibilità.

Attualmente sono presenti in commercio dei controllori video molto evoluti come l'Inmos GV300, i cui parametri di visualizzazione sono dinamicamente programmabili: è possibile variare risoluzione fra una riga e l'altra e addirittura all'interno di una stessa riga in modo da poter ottenere un sistema a finestre molto flessibile.

Conclusioni

Questa discussione sulle varie architetture del frame buffer e sulle finestre ci prepara a capire un problema squisitamente tecnologico: la grande massa di dati che devono essere processati in tempi molto brevi necessita di architetture che consentano accessi multipli alle risorse condivise, la memoria, in modo da superare il collo di bottiglia delle collisioni di accesso.

Il prossimo articolo sarà dedicato alle Video-RAM recenti dispositivi che aiutano a risolvere questo problema.

MS

I nostri PROGRAMMI per l'edilizia

PriMus è il più potente, versatile, economico, particolarmente facile ed altamente professionale programma per il COMPUTO METRICO e la CONTABILITA DEI LAVORI. Il pacchetto comprende: computo metrico; elenco prezzi unitari; libretto misure; registro di contabilità; sommario del registro di contabilità; stato d'avanzamento lavori; certificato di pagamento; situazione contabile; quadro comparativo per perizia di variante; stima dei lavori; richiesta offerta; liste settimanali degli operai, mezzi d'opera e provviste; modulistica (consegna lavori, inizio lavori, fine lavori, ripresa lavori, ultimazione lavori, verbale nuovi prezzi, certificato regolare esecuzione, verbale di pesa, etc.); vidimazione dei registri in bianco. Dimensionamenti: MILLE tariffari con 5 mila voci di tariffa per ognuno; DIECIMILA numeri d'ordine con 500 misurazioni (circa 160 mila pagine di libretto misure); MILLE miliardi di importo. Ogni computo può essere suddiviso in 100 sottocomputi (opere) ed il tutto in ulteriori 100 categorie di lavoro. Nessun limite sul numero dei lavori gestibili e sulle perizie di variante per ogni lavoro. **PriMus-A** è il programma per l'ANALISI DEI PREZZI nei lavori sia pubblici che privati nonché per la gestione e redazione di ELENCHI PREZZI in generale. Potenti funzioni di input e di Word Processor permettono di gestire un archivio di voci elementari, uno di voci composte ed uno di analisi. Ogni voce composta può essere formata da voci elementari e da altre voci composte. Ogni analisi può essere formata da voci elementari, voci composte e da altre analisi. Effettuata la scelta delle voci finite per un lavoro, verranno stampate automaticamente tutte le analisi che compongono tali voci, nonché tutti i prezzi composti ed elementari compresi in dette analisi. Oltre alla gestione ed al calcolo delle analisi il programma può gestire anche le sole descrizioni delle voci con unità di misura, prezzi unitari e numeri di tariffa per generare automaticamente nuovi elenchi prezzi da utilizzare con PriMus. La gestione delle descrizioni è affidata ad un WP e quindi non ci sono limitazioni significative sulla lunghezza e tipo di descrizione delle voci. Tante utilità permettono la copia delle descrizioni fra le varie voci, il ricalcolo con percentuali di aumento o diminuzione nonché la possibilità di stabilire il prezzo finito di una analisi per risalire, in funzione delle quantità e dei prezzi esistenti, ai prezzi che dovrebbero avere i singoli elementi che la compongono al fine di ottenere il detto prezzo finito. **PriMus-C** non è un semplice programma di video scrittura, ma è un potente strumento per la redazione di sei capitolati speciali d'appalto previsti per legge nella redazione di progetti dei lavori sia pubblici che privati. Compresi nel programma sono già interamente archiviati i seguenti Capitolati Speciali Tipo: Lavori Edili; - Lavori Stradali; - Impianti Elettrici; - Impianti Idrico-Sanitari; - Impianti termici; - Acquedotti e Fognature. **CanTus** è il nuovissimo programma per la contabilità industriale particolarmente dedicato alle imprese edili (stradali, di impiantistica, etc.) e quindi alla risoluzione dei problemi inerenti la CONTABILITA CANTIERI. Con CanTus si potranno gestire uno o più cantieri (con la possibilità di suddivisione in sottocantieri e lavorazioni), uno o più magazzini ed altri punti di carico (fornitori, dipendenti, macchine, spese generali, attrezzature, etc.) da cui attingere le varie risorse. L'aggiornamento viene fatto in base al rapportino di cantiere, o qualsiasi altro documento, imputando su ogni cantiere i materiali utilizzati, le spese di mano d'opera, le spese di macchine e noleggi, l'incidenza del costo delle attrezzature, etc. che ogni giorno concorrono a formare il costo del cantiere. Il movimento dei materiali può essere fatto da fornitore a cantiere e/o magazzino, da magazzino a cantiere, da cantiere a cantiere o da magazzino a magazzino con aggiornamento in tempo reale delle giacenze sia dei magazzini che dei cantieri. Con CanTus, quindi, sarà facile conoscere in ogni istante il costo dei vari cantieri, le giacenze di materiali, l'impiego di mano d'opera o di macchine e quant'altro necessario per una moderna e razionale gestione dell'Impresa.

FIERE:

TECNORAMA UFFICIO 91
BARI 14/18 febbraio 1991
pad. 11 - stands 172-173

ROMA UFFICIO 91
ROMA 24/27 marzo 1991
pad. 10 - stand 59

ACCA® SOFTWARE

ACCA s.r.l. - Via M.lo Cianciulli, 41 * 83048 MONTELLA (AV)
telefoni 0827/69504-601235- 089/953581