

COMPUTER GRAFICA TRIDIMENSIONALE ASSONOMETRIA E PROIEZIONI ORTOGONALI

Già da qualche numero abbiamo cominciato a parlare di Computer Grafica tridimensionale. Abbiamo trattato il problema della prospettiva, realizzando dei programmi con i quali abbiamo visualizzato su carta, tramite plotter, o sul monitor alta risoluzione dell'Apple II, oggetti tridimensionali fissi od in movimento.

In questo numero allargheremo il discorso dalla prospettiva agli altri metodi di rappresentazione bidimensionale di oggetti tridimensionali, l'assonometria e la proiezione ortogonale, e vedremo come i tre metodi non differiscono molto tra di loro.

Anzi, per evidenziare la loro comune logica, utilizzeremo nei nostri programmi gli stessi dati base che abbiamo visto nei programmi di prospettiva.

Negli scorsi numeri abbiamo esaminato programmi il cui scopo era quello di avere una o più viste prospettiche di un dato oggetto. Poiché in questo numero continueremo il discorso allargandolo alle assonometrie e alle proiezioni ortogonali, è necessario fare un breve riassunto delle puntate precedenti.

In figura 1 è riportato il listato di un programma denominato PRO/ZERO, già pubblicato nell'articolo del numero 8 di MICROCOMPUTER, ma alleggerito dei REM e con DATA differenti.

Tale programma realizza sul monitor Apple II una vista prospettica di alcuni solidi geometrici posizionati in un certo sistema di riferimento. In figura 2 abbiamo riportato l'output, sul plotter WATANABE 4671, dei nostri solidi.

Dobbiamo soffermarci sul sistema di codificazione dei dati da visualizzare. La fi-

gura solida è scomposta in varie spezzate. I singoli punti della spezzata sono individuati dalle proprie coordinate cartesiane nello spazio e da un flag F%(I), che indica solo la condizione di inizio e continuazione spezzata a seconda che sia posto uguale a 1 o a 0.

Se un punto da visualizzare appartiene a più spezzate, occorrerà immetterlo più volte.

Quello che ci proponiamo ora di fare è di realizzare programmi di Assonometria e di Proiezione ortogonale utilizzando lo stesso sistema di codificazione dei dati da visualizzare.

Questo ci permetterà, ad esempio, di realizzare programmi di Grafica Tridimensionale che utilizzino gli stessi dati, indipendentemente dal sistema di visualizzazione scelto.

L'Assonometria

L'assonometria, come detto nel numero 8 è un sistema di rappresentazione tridimensionale abbastanza semplice da realizzare (con matita e carta) in quanto fa conservare ai singoli segmenti da visualizzare le esatte coordinate lungo gli assi XYZ.

Nella figura 3 è riportato il metodo grafico da utilizzare per realizzare una assonometria e che noi abbiamo direttamente riportato nel programma, per posizionare il punto PS (XS, YS), originato dal P(XYZ) nello spazio, sul foglio di carta. Le formule da usare, ricavabili con semplici passaggi trigonometrici, sono:

$$XS = X * \cos(P1) - Z * \cos(P2)$$

$$YS = X * \sin(P1) + Z * \sin(P2) - Y$$

Gli angoli P1 e P2 possono essere variati a volontà. Le assonometrie classiche sono però quelle riportate in figura 4.

È evidente che il sistema assonometrico, potendo essere variati a volontà gli angoli caratteristici, non può dare una vista realistica dell'oggetto. Ciononostante, data la facilità d'uso il sistema è molto diffuso.

Il programma realizzato (listato in figura 5 e output in figura 6) ripete fedelmente il sistema grafico sopra descritto.

I punti che formano le varie spezzate sono ND, memorizzati nei DATA (da riga 460 in poi). Anche gli assi di riferimento sono posti nei DATA. Per comodità, gli

```

10 HOME :ND = 63: DIM XS(ND),YS(ND),ZS(ND),FX(ND),FY(ND)
20 A = 3:V2 = 6000:D1 = 4000:D2 = 8000
30 FOR I = 1 TO ND: READ X(I),Y(I),Z(I),F(I): NEXT I
40 C = COS(A):S = SIN(A):X9 = 279:Y9 = 191:X8 = 139:Y8 = 95:5
50 XM = -9999:NM = 9999:VM = -9999:VN = 9999:FOR I = 1 TO ND
60 X1 = X(I) * C + Z(I) * S:V1 = Y(I) - V2:Z1 = -X(I) * S + Z(I) * C
70 XS(I) = (D1 * X1) / (D2 - Z1):YS(I) = -(D1 * V1) / (D2 - Z1)
80 IF XS(I) < XM THEN XM = XS(I)
82 IF XS(I) > XM THEN XM = XS(I)
84 IF YS(I) < VM THEN VM = YS(I)
86 IF YS(I) > VM THEN VM = YS(I)
90 NEXT I:DX = XM - XN:DV = VM - VN
100 SX = X9 / DX:SY = Y9 / DV:SC = SX:IF SX > SY THEN SC = SY
110 HORZ :HCOLOR = 3:FOR I = 1 TO ND
120 X = (XS(I) - XM - DX / 2) * SC + X8:Y = (YS(I) - VM - DV / 2) * SC + Y8
130 IF NOT F(I) THEN HPLLOT X,Y: NEXT I
140 HPLLOT TO X,Y: NEXT I:END
990 DATA 0,0,0,0,400,0,0,1,0,0,0,0,0,400,0,1,0,0,0,0,0,600,1
1000 REM PARALLELEPIPEDO
1010 DATA 20,0,180,0,140,0,20,1,300,0,200,1,260,0,360,1,20,0,100,1
1020 DATA 20,60,190,0,140,60,20,1,300,60,200,1,260,60,360,1,20,60,100,1
1030 DATA 20,0,180,0,20,60,180,1,140,0,20,0,140,60,20,1,300,0,200,0
1040 DATA 200,60,220,1,260,0,360,0,260,60,360,1
1050 REM TRONCO-PIRAMIDE
1060 DATA 300,60,220,0,270,60,246,1,210,60,246,1,180,60,220,1,210,60,194,1
1070 DATA 270,60,194,1,300,60,220,1,340,240,220,0,290,240,263,1,190,240,263,1
1080 DATA 140,240,220,1,190,240,177,1,290,240,177,1,340,240,220,1,300,60,220,0
1090 DATA 340,240,220,1,270,60,246,0,290,240,263,1,210,60,246,0,190,240,263,1
1100 DATA 190,60,220,0,140,240,220,1,210,60,194,0,190,240,177,1,270,60,194,0
1110 DATA 290,240,177,1
1120 REM PIRAMIDE CORICATA
1130 DATA 220,60,140,0,160,60,60,1,150,150,60,1,210,150,140,1,220,60,140,1
1140 DATA 220,60,140,0,30,60,220,1,160,60,60,0,30,60,220,1,150,150,60,0
1150 DATA 30,60,220,1,210,150,140,0,30,60,220,1

```

Figura 1 - Listato del programma PRO/ZERO. Per comprendere nel dettaglio come funziona il programma, vi rimandiamo alla lettura del numero 8 di MC, pagina 50.

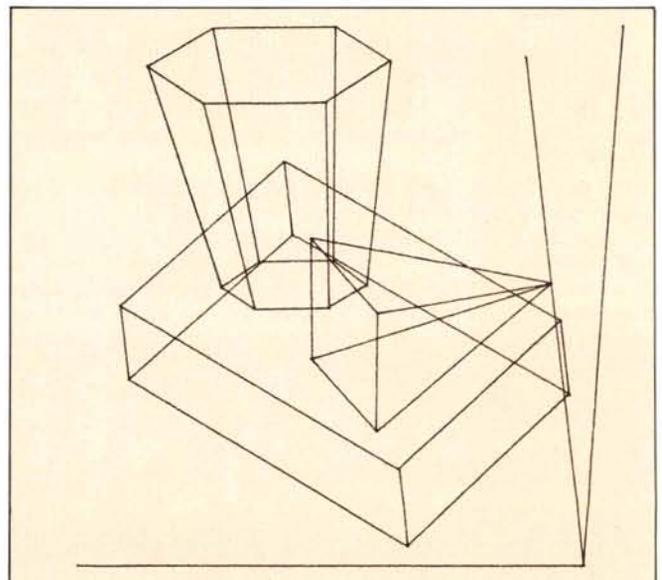


Figura 2 - Output su monitor del programma PRO/ZERO. Gli assi di riferimento possono essere trattati come tutti gli altri DATA da visualizzare. Sono semplicemente tre segmenti individuati da sei punti.

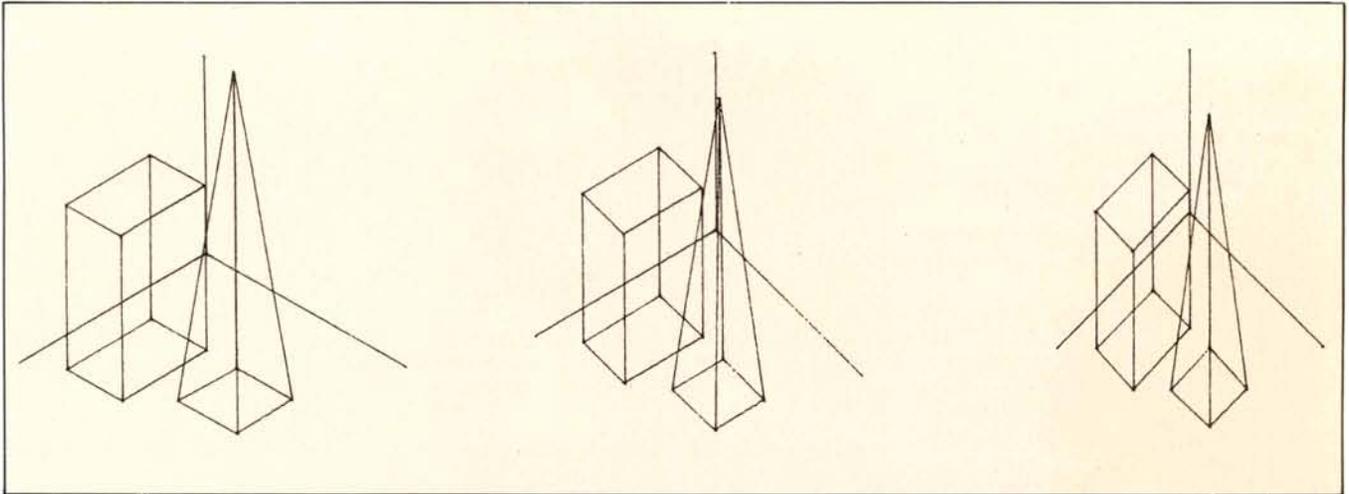


Figura 6 - Output del programma Assonometria. Sono state realizzate tre uscite ponendo in riga 140 gli angoli $P1, P2$ successivamente uguali a 120, 120 poi 135, 135 infine 120, 135.

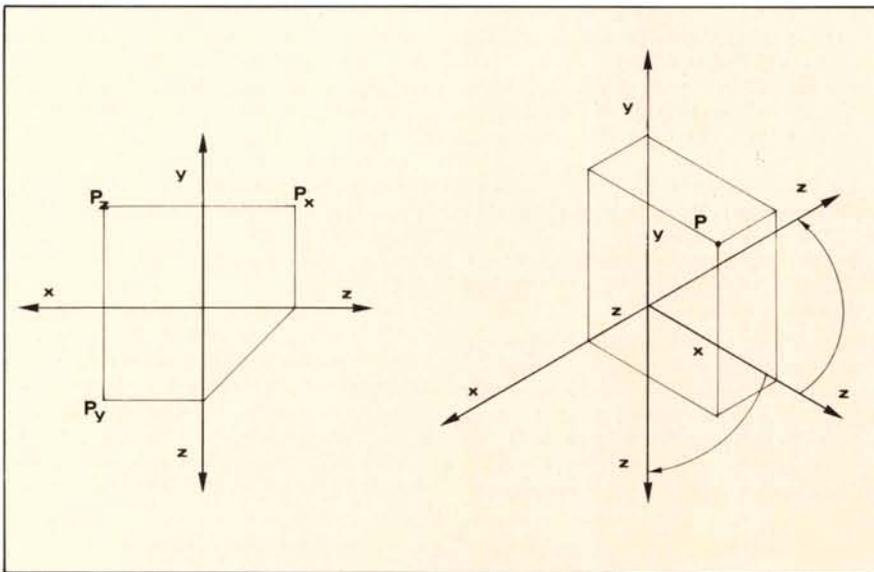


Figura 7 - Costruzione grafica della proiezione ortogonale. Il singolo punto P nello spazio genera le tre proiezioni P_x, P_y, P_z sui tre piani PYZ, PXZ, PXY nello spazio. Per riportarci al piano si ipotizza di ribaltare i piani YZ e XZ ambedue sul piano XY .

ce, realizza l'output delle tre proiezioni in sequenza sul monitor Apple II, senza metterle in relazione tra di loro.

Il secondo è invece una implementazione dei programmi di prospettiva e realizza un unico output con le tre viste poste nei tre quadranti classici e completi dei segmenti che uniscono ciascun punto delle tre viste.

Ambedue i programmi utilizzano la stessa codifica dei dati di ingresso, che quindi possono essere immediatamente intercambiati con tutti i programmi pubblicati, facendo dei merge.

Il programma PRO/ORT/1

Il primo programma di proiezioni ortogonali è molto elementare, presenta in sequenza sul monitor HGR dell'Apple II le tre viste degli oggetti riportati nei DATA (vedi il listato in figura 8).

Per avere ad esempio la vista sul piano XZ basterà caricare i dati relativi nei vetto-

ri $A\%(I), B\%(I)$, che rappresentano, in questo caso, le coordinate dello schermo. Su tali dati vengono eseguite, una volta per ciascuna vista, le operazioni di scaling e di centratura per l'output.

Il limite di questo programma è la indipendenza dei vari output, che non permette di riconoscere la corrispondenza dei punti sulle tre viste, e che addirittura eseguendo tre volte, una per ogni vista, lo scaling, visualizza con scale differenti.

A questi "inconvenienti" pone rimedio il programma PRO/ORT/2 che ha una routine di scaling unica, generale, ovvero sceglie un fattore di scala unico per le tre viste e le posiziona centrando, ciascuna nel suo riquadro.

Il programma PRO/ORT/2

Se ricordate la procedura che abbiamo utilizzato per realizzare la prospettiva, vedi articolo sul numero 8 di MC ed in particolare la figura 5 di pagina 50, vi sarà facile comprendere come allontanando l'osser-

```

100 HOME : PRINT "LETTURA DATI " : ND = 59
110 DIM A%(ND), B%(ND), X%(ND), Y%(ND), Z%(ND), F%(ND)
120 FOR I = 1 TO ND: READ X%(I), Y%(I), Z%(I), F%(I): NEXT I
130 FOR K = 1 TO 3: ON K GOTO 140, 160, 180
140 FOR I = 1 TO ND: A%(I) = X%(I): B%(I) = - Y%(I): NEXT
150 HOME : VTAB (22): PRINT "VISTA SUL PIANO X,Y " : GOTO 200
160 FOR I = 1 TO ND: A%(I) = X%(I): B%(I) = Z%(I): NEXT
170 HOME : VTAB (22): PRINT "VISTA SUL PIANO X,Z " : GOTO 200
180 FOR I = 1 TO ND: A%(I) = Z%(I): B%(I) = - Y%(I): NEXT
190 HOME : VTAB (22): PRINT "VISTA SUL PIANO Y,Z "
200 AM = - 9999: AN = 9999: BM = - 9999: BN = 9999
210 FOR I = 1 TO ND
220 IF A%(I) > AM THEN AM = A%(I)
230 IF A%(I) < AN THEN AN = A%(I)
240 IF B%(I) > BM THEN BM = B%(I)
250 IF B%(I) < BN THEN BN = B%(I)
260 NEXT I
270 DA = AM - AN: DB = BM - BN
280 SA = 279 / DA: SB = 159 / DB
290 SC = SA: IF SA > SB THEN SC = SB
300 REM DISEGNO
310 HGR : HCOLOR= 3
320 FOR I = 1 TO ND
330 A1 = (A%(I) - AN - DA / 2) * SC + 140
340 B1 = (B%(I) - BN - DB / 2) * SC + 80
350 IF F%(I) = 0 THEN HPLT TO A1, B1
360 HPLT TO A1, B1
370 NEXT I, K: END

```

Figura 8 - Listato del programma PRO/ORT/1.

Il programma ha un output su monitor Apple II, fornisce le tre viste ortogonali in sequenza, utilizzando per ciascuna di esse interamente lo schermo.

vatore dall'oggetto, e lasciando invece prossimo all'oggetto lo schermo, la traccia dell'oggetto sullo schermo si avvicina sempre più alla sua dimensione reale. Potremo dire che al limite per D2 che tende all'infinito la traccia sullo schermo tende alla dimensione reale dell'oggetto.

Quindi per realizzare le tre proiezioni ortogonali nella nostra figura abbiamo pensato di utilizzare per tre volte la stessa procedura che esegue la prospettiva, da tre punti differenti e prefissati.

I tre punti di vista sono posti sugli assi cartesiani a distanza infinita dall'origine $P_x(\infty, 0, 0)$, $P_y(0, \infty, 0)$, $P_z(0, 0, \infty)$.

In pratica eseguiamo per tre volte il programma di prospettiva, ponendo per ogni punto di vista D1, distanza oggetto/ schermo, pari a 100, D2, distanza oggetto/ osservatore, pari a 99999 e realizzando lo spostamento da un asse all'altro ruotando il punto di osservazione di 90 gradi dapprima rispetto all'asse Y e poi rispetto all'asse X.

Esaminando direttamente il programma, vedi listato in figura 9, vediamo come i valori D1, D2 sono comuni alle tre viste, come anche i valori angolari A1(I), A2(I), e X9, Y9 che rappresentano il formato output.

Invece specifici per ogni vista sono i valori angolari A1(I), A2(I), che rappresentano le rotazioni rispetto a Y e X, e i due valori XC(I), YC(I), che rappresentano il

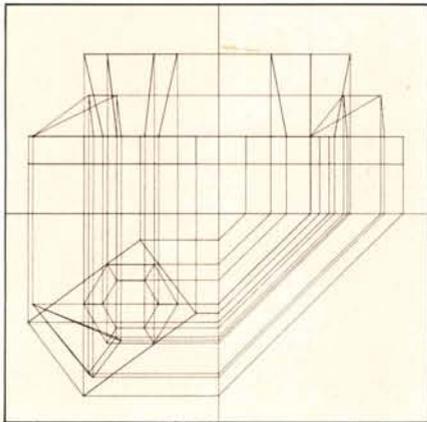


Figura 10 - Output del programma PRO/ORT/2. I dati rappresentano un tronco di piramide, un parallelepipedo e una piramide coricata e sono immagazzinati nei DATA. Per visualizzare un altro soggetto basta inserire le informazioni relative alle spezzate nei DATA e ND, numeri dei dati, nella riga 120 del programma.

centro del disegno nel proprio riquadro.

Quindi, dopo aver caricato i dati iniziali comuni alle tre viste, e quelli specifici di ogni singola vista (righe 100-170), vengono eseguiti nel loop L da 1 a 3 tutti i calcoli per ottenere i valori XS(I,L), YS(I,L) di tutti i punti della figura nelle tre viste (righe 180-330). Viene poi sempre per ciascuna vista eseguita la ricerca del fattore di scala SC(L) (righe 340-370).

Finito il loop L (riga 380), viene scelto il più piccolo valore SC(L) come fattore di scala generale. È chiaro infatti che le tre proiezioni devono essere tutte riprodotte

```

100 REM INIZIALIZZAZIONI
110 PL = 0: G4 = CHR$(4): HOME: PRINT "LETTURA DATI "
120 X9 = 71: Y9 = 71: ND = 39: D1 = 100: D2 = 99999: SF = 10
130 DIM XS(ND,3), YS(ND,3), XL(ND,3), YL(ND,3)
140 DIM XS(ND), YS(ND), ZS(ND), FX(ND)
150 FOR I = 1 TO 3: READ XL(I), YL(I): NEXT I
160 FOR I = 1 TO 3: READ XS(I), YS(I): NEXT I
170 FOR I = 1 TO ND: READ XS(I), YS(I), ZS(I), FX(I): NEXT I
180 REM CALCOLO DELLE L VISTE
190 FOR L = 1 TO 3: PRINT "PASSO "L
200 PL = XL(L): C1 = COS(A1): S1 = SIN(A1)
210 PL = YL(L): C2 = COS(A2): S2 = SIN(A2)
220 XN(L) = -29999*XN(L) = 29999*YR(L) = -29999*YR(L) = 29999
230 FOR I = 1 TO ND
240 XI = XS(I) * C1 + ZS(I) * S1
250 YI = YS(I) * C2 + ZS(I) * S2
260 ZI = -XS(I) * S1 + YS(I) * S2 + ZS(I) * (C1 - C2)
270 XSC(I,L) = (XI + YI) / (C2 - ZI)
280 YSC(I,L) = -(XI + YI) / (C2 - ZI)
290 IF XSC(I,L) < XN(L) THEN XN(L) = XSC(I,L)
300 IF XSC(I,L) > XN(L) THEN XN(L) = XSC(I,L)
310 IF YSC(I,L) < YN(L) THEN YN(L) = YSC(I,L)
320 IF YSC(I,L) > YN(L) THEN YN(L) = YSC(I,L)
330 NEXT I
340 REM CALCOLO FATTI SCALA E TRASLAZIONE
350 DX(L) = XN(L) - XN(L): DY(L) = YN(L) - YN(L)
360 SX(L) = X9 / DX(L): SY(L) = Y9 / DY(L)
370 SC(L) = SX(L): IF SY(L) > SY(L) THEN SC(L) = SY(L)
380 NEXT L
390 REM RICERCA FATTORE SCALA GENERALE
400 SC = SC(1)
410 IF SC(2) < SC THEN SC = SC(2)
420 IF SC(3) < SC THEN SC = SC(3)
430 REM SUB CALCOLO PUNTI DISEGNO
440 FOR L = 1 TO 3
450 FOR I = 1 TO ND
460 XI(I,L) = (XN(L) - XN(L) + (X1 - X1) / 2) * SC + XC(L)
470 YI(I,L) = (YN(L) - YN(L) + (Y1 - Y1) / 2) * SC + YC(L)
480 YTAB(I,5) = PRINT I + ND - (I - 1) * ND - 1: NEXT I
490 NEXT L
500 REM VISUALIZZAZIONE
510 IF NOT PL THEN HORZ: PLOTURE 3
520 REM SUBROUTINE
530 X1 = X1: Y1 = 0: GOSUB 680: X2 = 100: Y2 = 100: GOSUB 720
540 X2 = 0: Y2 = 0: GOSUB 720: X2 = 100: Y2 = 100: GOSUB 720
550 X2 = 0: Y2 = 90: GOSUB 680: X2 = 100: Y2 = 100: GOSUB 720
560 X2 = 90: Y2 = 0: GOSUB 680: X2 = 100: Y2 = 100: GOSUB 720
570 REM SINGOLE FIGURE
580 FOR I = 1 TO 3: FOR I = 1 TO ND: X2 = XI(I,L): Y2 = YI(I,L)
590 IF NOT FX(I) THEN GOSUB 680: GOTO 610
600 GOSUB 720
610 NEXT I:L
620 REM TRATTI DI COLLEGAMENTO
630 FOR I = 1 TO ND
640 X2 = XI(I,1): Y2 = YI(I,1): GOSUB 680: X2 = XI(I,2): Y2 = YI(I,2): GOSUB 720
650 X2 = XI(I,2): Y2 = YI(I,2): GOSUB 720: X2 = XI(I,3): Y2 = YI(I,3): GOSUB 720
660 X2 = XI(I,3): Y2 = YI(I,3): GOSUB 720: X2 = XI(I,1): Y2 = YI(I,1): GOSUB 720
670 NEXT I: END
680 REM FUNZIONE PLOTTER MOVE
690 IF NOT PL THEN PLOT 10: Y2: RETURN
700 X3 = INT (X2 + SP) * VS = 2400 + INT (Y2 + SP)
710 PRINT D;"PRM1": PRINT "R";X3;"Y";Y2: PRINT D;"PRM0": RETURN
720 REM FUNZIONE PLOTTER DPAH
730 IF NOT PL THEN PLOT 10: Y2: Y2: RETURN
740 X3 = INT (X2 + SP) * VS = 2400 + INT (Y2 + SP)
750 PRINT D;"PRM1": PRINT "D";X3;"Y";Y2: PRINT D;"PRM0": RETURN
    
```

Figura 9 - Listato del programma PRO/ORT/2. Il programma esegue prima i calcoli relativi alle tre viste, poi formatta i dati per poterli visualizzare, infine li visualizza.

nella stessa scala (righe 390-420).

Ciò fatto vengono calcolati tutti i punti di ciascuna proiezione nel riferimento generale (righe 430-490), cioè calcolati considerando il fattore di scala generale e introducendo il dato centro del disegno XC, YC per ogni vista (righe 440-490).

Dalla riga 500 alla 670 c'è la visualizzazione, che al solito può avvenire anche sul plotter WATANABE, ponendo PL = 1 in riga 100.

La visualizzazione si compone di tre fasi:

- la squadratura, righe 520-560, in cui

vengono predisposti i quattro riquadri in cui si esegue il disegno;

- il disegno delle singole figure, che avviene grazie al loop L da 1 a 3, con ciascuna figura nel proprio riquadro (righe 570-610);

- il disegno dei tratti di collegamento tra i punti corrispondenti delle tre viste. Viene in pratica eseguita una spezzata chiusa che unisce i tre punti corrispondenti nelle tre viste e attraverso obliquamente il quadrante vuoto (righe 620-670).

Nelle subroutine di righe 680-710 e righe 720-750 ci sono le solite istruzioni per il disegno vero e proprio. Le istruzioni per il PLOTTER sono state più volte illustrate nei numeri precedenti della rivista.

Esaminando gli output prodotti (figure 10 e 11) si vede come anche la visualizzazione di un oggetto semplice comporti il tracciamento di un gran numero di righe, che spesso si sovrappongono, specialmente quando l'oggetto presenta facce parallele ai piani di riferimento.

Quindi visualizzando un oggetto complesso sul monitor Apple II si può avere anche un disegno poco comprensibile, anche perché dividiamo il monitor in quattro quadranti, pur avendo come noto il monitor Apple II una buona definizione.

Realizzando l'output con il plotter e quindi avendo una maggiore definizione e avendo la possibilità di usare più penne di colore differente si ottengono risultati accettabili.

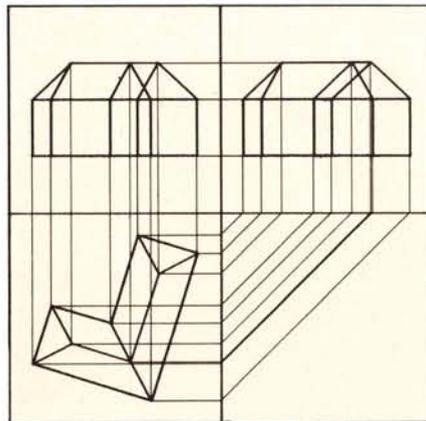


Figura 11 - Output del programma PRO/ORT/2. Sono moltissime le linee che si sovrappongono, specie se le facce degli oggetti da visualizzare sono parallele ai piani di riferimento.