



a cura di Maurizio Mauri

# L'interfaccia stampante

di Sergio e Dario Neddi

L'interfaccia verso la stampante è implementata con comuni circuiti TTL; la porta parallela in uscita verso la stampante è allocata all'indirizzo di I/O &H91 mentre all'indirizzo (sempre di I/O) &H90 si trovano i bit di controllo: lo STROBE (uscita) ed il BUSY (ingresso).

L'interfaccia è compatibile Centronics, ma usa un connettore a piedinatura ridotta rispetto allo standard e non prevede l'eventuale uso di molti segnali di controllo.

Per interfacciare una stampante MSX, è necessario munirsi del cavo apposito o nel peggiore dei casi farselo o farselo fare.

In tal caso può essere utile il confronto tra il connettore MSX e quello standard Centronics che riportiamo nella pagina accanto.

Chi volesse saperne di più sullo standard Centronics può rileggersi l'articolo di Tommaso Pantuso sul numero 45 di MC, tenendo conto che il pin 14 del connettore Centronics in alcune stampanti è Signal Ground come il pin 16, e in altre (come Mannesmann Tally) è Auto Feed TX. Il numero 16 è sempre Signal Ground.

## Note per chi lavora in L/M

Per i programmatori in Assembly il

BIOS mette a disposizione 2 routine, ecco gli ENTRY POINT:

### &H00A5

Invia alla stampante il carattere contenuto nel registro A (dello Z80).

### &H00A8

Legge il segnale di BUSY della stampante.

La prima routine attende che la stampante sia pronta a ricevere un nuovo carattere (tramite la routine che legge lo stato del BUSY) ed in caso affermativo lo invia in uscita, altrimenti si blocca in un loop di attesa dal quale si può uscire con CTRL-STOP.

In caso di interruzione della stampa con CTRL-STOP la routine ritorna il flag di CARRY settato in modo da avvertire del fatto il programma chiamante.

L'esecuzione di questa routine non altera i registri dello Z80, salvo quello di stato.

La seconda routine si limita a leggere il segnale di BUSY della stampante, ritornando il flag di zero resettato nel caso che la stampante sia pronta e settato in caso contrario.

Il rilevamento della pressione dei tasti CTRL-STOP nella routine di stampa viene eseguito da un richiamo alla routine di indirizzo &H00B7.

Parliamo degli ENTRY POINT del-

le routine descritte: a questi indirizzi (come di consueto per gli ENTRY POINT del BIOS) si trovano delle istruzioni di jump alle routine vere e proprie nella ROM del BIOS. Questi indirizzi non sono standard ma possono variare da computer a computer. Ad ogni modo nella maggior parte dei casi dovrebbero corrispondere come nello Spectravideo 728, a questi valori: per la routine &H00A5 il jump è a &H085D, per la routine &H00A8 il jump è a &H0884 e per la routine &H00B7 il jump è a &H046F.

Tenete comunque presente che questi indirizzi NON sono standard e pertanto non indirizzate queste routine direttamente, ma solo tramite chiamate agli ENTRY POINT menzionati sopra.

Ad ogni modo può essere interessante dare un'occhiata al disassemblato delle due routine che hanno a che fare con la stampante.

Bisogna fare una precisazione a proposito dei «rimandi in RAM» menzionati nel listato: si tratta semplicemente di chiamate in una zona alta della RAM. A questi indirizzi di default sono memorizzati semplicemente dei RET (codice operativo &HC9), quindi normalmente non servono a nulla, infatti il microprocessore salta lì e ritorna indietro.

Se però, invece di lasciare il RET di default si memorizza un jump ad una nostra routine, ecco che possiamo modificare la funzione originale, per esempio possiamo gestire una stampante che non sia Centronics. Chi scrive ha utilizzato per molti mesi una stampante Commodore, adattata a funzionare con l'MSX grazie ad una routine L/M allocata nella parte alta della RAM. Questa routine prevede un jump sistemato all'indirizzo &HFFB6 e si occupa anche del problema dell'apertura del file verso la stampante (una stampante Commodore deve sempre venire «chiamata» prima di stampare), dell'invio seriale dei dati e problemini connessi, tutto tramite la normale porta stampante dell'MSX.

In questo modo la stampante ha funzionato egregiamente sia in Basic con LPRINT ed LLIST che con programmi L/M (MS TEXT, versione cassetta).

Come vedremo di questi «rimandi in RAM» ce ne sono una marea, e lavorando sopra si possono ottenere tante belle cose.

Alcuni sono utilizzati dal Basic per il dialogo con il drive (od altre periferiche, come la RS-232), ed è proprio attraverso di essi, inizializzati all'accensione dalla cartuccia dell'interfaccia, che vengono «lanciate» molte funzioni del drive stesso.

La locazione &HF415 che compare nel listato contiene l'attuale posizione della testina di stampa e viene incrementata tutte le volte che viene stampato un carattere con l'istruzione Basic LPRINT; la funzione LPOS(0) è perfettamente equivalente a PEEK(&HF415).

Più interessante, anche se sinceramente non sappiamo quanto utile in Basic, è la locazione &HF416. Facendo POKE &HF416,1: PRINT «CIAO» la stringa «CIAO» viene in-

viata alla stampante anziché al video. Questa locazione è in effetti un flag che viene utilizzato dal Basic per dirigere l'output verso la stampante.

La routine di output generico, presente nel BIOS ed utilizzata dal Basic ha il suo ENTRY POINT all'indirizzo &H0018. Questa importante routine

#### Confronto tra le piedinature Centronics ed MSX

Pin	Centronics Segnale	MSX Segnale
1	Data strobe	Data strobe
2	Data 1	Data 1
3	Data 2	Data 2
4	Data 3	Data 3
5	Data 4	Data 4
6	Data 5	Data 5
7	Data 6	Data 6
8	Data 7	Data 7
9	Data 8	Data 8
10	Acknowledge	.....
11	Busy	Busy
12	Printer enable	.....
13	Printer select	.....
14	Auto feed TX	Signal ground
15	.....	.....
16	Signal ground	.....
17	Chassis ground	.....
18	+5 Volt	.....
19	Twisted pair gnd	.....
20	Twisted pair gnd	.....
21	Twisted pair gnd	.....
22	Twisted pair gnd	.....
23	Twisted pair gnd	.....
24	Twisted pair gnd	.....
25	Twisted pair gnd	.....
26	Twisted pair gnd	.....
27	Twisted pair gnd	.....
28	Twisted pair gnd	.....
29	Twisted pair gnd	.....
30	Twisted pair gnd	.....
31	Input prime	.....
32	Fault	.....
33	Signal ground	.....
34	.....	.....
35	.....	.....
36	.....	.....

ha come dispositivo di default il video, ma inizializzando opportunamente alcuni parametri permette l'invio di dati ad altre periferiche.

#### Esempi di stampa di un carattere:

```
LD A,41H ; Lettera «A»
CALL 00A5H ; STAMPA
LD A,0DH ; RETURN: provoca la
           ; STAMPA
CALL 00A5H ; Svuotando il BUFFER
LD A,0AH ; LINE FEED: Va alla
CALL 00A5H ; Linea di STAMPA
           ; successiva
RET ; Torna al programma
           ; chiamante
```

Notare che è sempre opportuno chiudere un'operazione di stampa con l'invio dei caratteri CR ed LF, come abbiamo visto in questo esempio che stampa semplicemente una lettera «A».

```
LD A,1
LD (0F416H),A ; Setta Flag di stampa
LD A,41H ; Lettera «A»
RST 0018H ; Cioè CALL 0018H
           ; abbreviato
LD A,0DH ; CR
RST 0018H
LD A,0AH ; LF
RST 0018H
XOR A ; Azzera il registro A
LD (0F416H),A ; Azzera Flag di stampa
RET ; Ritorno
```

Questa routine fa la medesima cosa della precedente, però ha una particolarità interessante: ponendo a 0 il flag di stampa all'inizio della routine (con LD A,0 oppure XOR A) od omettendo questa inizializzazione, l'output avviene sul video.

#### Disassemblato

```
085D CALL 0FFB6H ; Rimando in RAM
0860 PUSH AF ; Salva il carattere da stampare
0861 CALL 046FH ; Cioè CALL 00B7H: test dei tasti
           ; CTRL-STOP
0864 JR C, 0878H ; Jump se premuti CTRL-STOP
0866 CALL 0884H ; Cioè CALL 00A8H: test segnale
           ; BUSY
0869 JR Z, 0861H ; LOOP di attesa
086B POP AF ; La stampante è pronta, recupera il
           ; carattere
086C PUSH AF ; Risalva il carattere
086D OUT (91H),A ; Invia il carattere
086F XOR A ; Azzera il reg. A
0870 OUT (90H),A ; Strobe = 0
0872 DEC A ? ; OFFH nel reg. A
0873 OUT (90H),A ; STROBE = 1
0875 POP AF ; Recupera AF
0876 AND A ; Serve a resettare il Flag di CARRY
0877 RET ; Ritorno al programma chiamante
```

```
0878 XOR A ; Qui salta se vengono pigiati
           ; CTRL-STOP
0879 LD (0F415H),A ; Azzera la locazione &HF415
087C LD A, 0DH ; Pone un CR in A
087E CALL 086CH ; Invia alla stampante per scaricarne il
           ; Buffer
0881 POP AF ; Recupera AF
0882 SCF ; Setta il CARRY per indicare
           ; l'interruzione
0883 RET ; Ritorna al programma chiamante
```

#### Routine di test del segnale di BUSY

```
0884 CALL 0FFBBH ; Rimando in RAM
0887 IN A, (90H) ; INPUT
0889 RRCA ; Pone il BIT di
088A RRCA ; BUSY nel CARRY
088B CCF ; Lo complementa
088C SBC A,A ; Sottrae
088D RET ; Ritorna al programma principale
```

L'utilità pratica consiste nel poter inviare l'output al video oppure alla stampante a seconda dello stato di un flag.

Anche sul video è necessario fare CR ed LF per posizionarsi all'inizio della linea successiva, altrimenti la

stampa avviene a partire dall'inizio della medesima linea se viene omissa il line feed.

Diciamo questo per i lettori abituati con i computer e le stampanti Commodore, che con CR sottintendono l'LF.

Con questo anticipiamo l'argomento dei prossimi articoli che sarà proprio il VDP.

Vedremo di svelare i segreti dei vari screen dell'MSX 1, ripromettendoci in seguito di penetrare pure in quelli dell'MSX 2.

## Un programma di Hard Copy

Tra le numerose istruzioni del Basic MSX ne manca una che permette di fare l'hard-copy dello schermo: sopperire a questa mancanza è abbastanza semplice per gli screen di testo, ma decisamente meno facile per gli screen grafici soprattutto se si vuol tener conto dei colori. Il programma che segue permette di fare la copia dello screen 2 anche se l'effetto che si ottiene qualche volta non è ottimale. Interessante è anche l'uso come comando esteso, cioè richiamabile da Basic con l'istruzione CALL.

### Descrizione del programma

La comprensione del listato richiede la conoscenza di alcune nozioni trattate nella rubrica MISTER MSX: ci limiteremo a darne una descrizione generale, rimandando per i particolari alla rubrica citata. La prima parte del programma (che deve essere mandato in esecuzione con BLOAD«no-

me»,R) ricerca la memoria RAM in pagina 1 (fra 4000H e 7FFFH) nei 16 possibili slot secondari, aggiorna la tabella degli attributi di slot (SLTATR) ed infine trasferisce la routine che gestisce il comando esteso in questa parte della memoria, avvertendo con un messaggio che la ricerca ha avuto esito positivo. Se il vostro computer non ha a disposizione 64K di RAM ovviamente non potete far uso di comandi estesi in RAM. Per utilizzare il nuovo comando dovete costruirvi una immagine in alta risoluzione e richiamare la routine con CALL LCOPY. Se non vi trovate in screen 2 otter-

rete ovviamente cose senza senso. In ogni momento potete fermare la stampa premeendo «CTRL STOP». La routine controlla che il nome del comando chiamato sia quello giusto «LCOPY», legge quindi i 6K dalla VRAM e li elabora prima di mandarli alla stampante. La necessità di elaborazione è dovuta al fatto che 1 byte rappresenta sullo schermo 8 punti disposti orizzontalmente, mentre 1 byte inviato alla stampante controlla 8 punti in verticale. Per salvare bisogna digitare:

```
BSAVE"LCOPY",&HC000,&HC165.
```



### LCOPY

```
C000: 3E 80 F5 F3 26 40 CD 24 00 21 FD 5F 7E 2F 77 BE >...&@.#.!.../w.
C010: 28 25 F1 3C FE 90 38 EA CD 9A C0 52 41 4D 20 69 (%...B...RAM :
C020: 6E 73 75 66 66 69 63 69 65 6E 74 65 0D 0A 00 3A nsufficiente...:
C030: C1 FC 26 40 C3 24 00 F1 47 17 17 17 17 E6 30 B0 ..&@.#..G.....0.
C040: E6 3C 3C 06 00 4F 21 C9 FC 09 36 20 21 A7 C0 11 <<<.D!...6 !...
C050: 00 40 01 C7 01 ED B0 CD 6C 00 3A AE F3 CB 2F D6 .@...!...:..7.
C060: 0F 67 2E 0A CD C6 00 CD 9A C0 49 6C 20 63 6F 6D .g.....Il com
C070: 61 6E 64 6F 20 65 73 74 65 73 6F 20 65 60 20 69 ando esteso e' 1
C080: 6E 73 74 61 6C 6C 61 74 6F 0D 0A 00 06 04 21 00 nstallato.....!
C090: 00 2B 7C B5 20 FB 10 F6 18 95 E3 7E 23 47 28 05 .+!.....*#.(.
C0A0: CD A2 00 18 F6 E3 C9 41 42 00 00 10 40 00 00 00 .....AB...@...
C0B0: 00 00 00 00 00 00 00 E5 21 B9 FD 11 22 40 1A BE .....@...
C0C0: 20 0D A7 23 13 20 F7 18 09 4C 43 4F 50 59 00 E1 ..#. ...LCOPY..
C0D0: 37 C9 21 B3 40 CD 98 40 21 00 00 3E 18 32 AB 40 7.'@...@!...>.2.@
C0E0: 22 AD 40 11 C7 40 01 00 01 CD 59 00 3E 20 32 AC ".@...@!...>.2.@
C0F0: 40 21 C7 40 E5 11 BF 40 0E 08 06 08 E1 E5 CB 06 @!..@...@...@...
C100: 1F 23 10 FA 12 13 0D 20 F1 D1 E5 3A AC 40 3D 32 .#@...@...@...@=2
C110: AC 40 20 E4 E1 EB 36 0D 23 36 0A 21 B7 40 CD 98 .@ ...@.#@...!..@..
C120: 40 2A AD 40 24 3A AB 40 3D 20 B2 21 AF 40 CD 98 @#@#@...@...!..@..
C130: 40 E1 7E B7 28 07 FE 3A 28 03 23 1B F5 A7 C9 4E @..(..(..#...N
C140: 23 46 23 7E CD A5 00 38 06 0B 79 B0 20 F4 C9 E1 #F#"...B...y...
C150: 1B DF 00 00 00 02 06 1B 41 02 00 1B 42 0B 01 .....A...B...
C160: 1B 53 30 32 35 36 00 00 00 00 00 00 00 00 00 .S0256.....
```

```
1 ; L C O P Y Copia in hi-res dello screen 2
2 ;
3 ;
4 ; Maurizio Mauri
5 ; dicembre 1985
6 ;
7 ; Si richiama da basic con: CALL LCOPY
8 ;
9 ; Versione per stampante MSI
10 ;
11 ;.i80
12 ;f2
13 ;.print: # MSI LCOPY #
14 ;end:
15 ;
16 ;
17 ;
18 ; Equates
19 ;
20 ; idrav equ 005h ; VRAM-VRAM
21 ; istmot equ 004h ; invia alla stampante il contenuto dell'accumulatore
22 ; cput equ 004h ; stampa sul video un carattere
23 ; enasit equ 002h ; abilita uno slot
24 ; posit equ 004h ; posiziona il cursore
25 ; iniat equ 004h ; inizializza lo schermo in modo testo
26 ; lin40 equ 014ah ; numero di colonne in modo testo
27 ; sltatr equ 01c7h ; contiene gli attributi di ogni pagina
28 ; procm equ 01d9h ; nome del comando esteso
29 ; cr equ 17 ; carriage return
30 ; lf equ 10 ; line feed
31 ;
32 ;
33 ;.aseq
34 ;.ore 100h
35 ;.d9f0 01ah ; basic header
36 ;.d9f0 01ah ; first ; last
37 ;.d9f0 01ah ; last0-l'last-init ; end
38 ;.d9f0 01ah ; first ; start
39 ;.d9f0 01ah ; origine
```

```
38 ; Ricerca la memoria RAM in pagina 1
39 ; first:
40 ; id a,80h ; slot 0-0
41 ; ntbl: push af
42 ; di
43 ; id h,40h ; abilita la pag. 1
44 ; call enasit
45 ; id h1,5f0h ; una locazione a caso di pag. 1
46 ; id a,(h1)
47 ; cpl
48 ; id (h1),a ; modifica il valore
49 ; cp (h1) ; e' uguale?
50 ; jr z,ramok
51 ; pop af
52 ; inc a ; prossimo slot
53 ; cp 90h
54 ; if c,nsitit
55 ; RAM non trovata
56 ; call displ
57 ; defb "RAM insufficiente",cr,lf,0
58 ;
59 ;
60 ;
61 ;
62 ;
63 ;
64 ;
65 ;
66 ;
67 ;
68 ;
69 ;
70 ;
71 ;
72 ;
73 ;
74 ;
75 ;
76 ;
77 ;
78 ;
79 ;
80 ;
81 ;
82 ;
83 ;
84 ;
85 ;
86 ;
87 ;
88 ;
89 ;
90 ;
91 ;
92 ;
93 ;
94 ;
95 ;
96 ;
97 ;
98 ;
99 ;
100 ;
101 ;
102 ;
103 ;
104 ;
105 ;
106 ;
107 ;
108 ;
109 ;
110 ;
111 ;
112 ;
113 ;
114 ;
115 ;
116 ;
117 ;
118 ;
119 ;
120 ;
121 ;
122 ;
123 ;
124 ;
125 ;
126 ;
127 ;
128 ;
129 ;
130 ;
131 ;
132 ;
133 ;
134 ;
135 ;
136 ;
137 ;
138 ;
139 ;
140 ;
141 ;
142 ;
143 ;
144 ;
145 ;
146 ;
147 ;
148 ;
149 ;
150 ;
151 ;
152 ;
153 ;
154 ;
155 ;
156 ;
157 ;
158 ;
159 ;
160 ;
161 ;
162 ;
163 ;
164 ;
165 ;
166 ;
167 ;
168 ;
169 ;
170 ;
171 ;
172 ;
173 ;
174 ;
175 ;
176 ;
177 ;
178 ;
179 ;
180 ;
181 ;
182 ;
183 ;
184 ;
185 ;
186 ;
187 ;
188 ;
189 ;
190 ;
191 ;
192 ;
193 ;
194 ;
195 ;
196 ;
197 ;
198 ;
199 ;
200 ;
201 ;
202 ;
203 ;
204 ;
205 ;
206 ;
207 ;
208 ;
209 ;
210 ;
211 ;
212 ;
213 ;
214 ;
215 ;
216 ;
217 ;
218 ;
219 ;
220 ;
221 ;
222 ;
223 ;
224 ;
225 ;
226 ;
227 ;
228 ;
229 ;
230 ;
231 ;
232 ;
233 ;
234 ;
235 ;
236 ;
237 ;
238 ;
239 ;
240 ;
241 ;
242 ;
243 ;
244 ;
245 ;
246 ;
247 ;
248 ;
249 ;
250 ;
251 ;
252 ;
253 ;
254 ;
255 ;
256 ;
257 ;
258 ;
259 ;
260 ;
261 ;
262 ;
263 ;
264 ;
265 ;
266 ;
267 ;
268 ;
269 ;
270 ;
271 ;
272 ;
273 ;
274 ;
275 ;
276 ;
277 ;
278 ;
279 ;
280 ;
281 ;
282 ;
283 ;
284 ;
285 ;
286 ;
287 ;
288 ;
289 ;
290 ;
291 ;
292 ;
293 ;
294 ;
295 ;
296 ;
297 ;
298 ;
299 ;
300 ;
301 ;
302 ;
303 ;
304 ;
305 ;
306 ;
307 ;
308 ;
309 ;
310 ;
311 ;
312 ;
313 ;
314 ;
315 ;
316 ;
317 ;
318 ;
319 ;
320 ;
321 ;
322 ;
323 ;
324 ;
325 ;
326 ;
327 ;
328 ;
329 ;
330 ;
331 ;
332 ;
333 ;
334 ;
335 ;
336 ;
337 ;
338 ;
339 ;
340 ;
341 ;
342 ;
343 ;
344 ;
345 ;
346 ;
347 ;
348 ;
349 ;
350 ;
351 ;
352 ;
353 ;
354 ;
355 ;
356 ;
357 ;
358 ;
359 ;
360 ;
361 ;
362 ;
363 ;
364 ;
365 ;
366 ;
367 ;
368 ;
369 ;
370 ;
371 ;
372 ;
373 ;
374 ;
375 ;
376 ;
377 ;
378 ;
379 ;
380 ;
381 ;
382 ;
383 ;
384 ;
385 ;
386 ;
387 ;
388 ;
389 ;
390 ;
391 ;
392 ;
393 ;
394 ;
395 ;
396 ;
397 ;
398 ;
399 ;
400 ;
401 ;
402 ;
403 ;
404 ;
405 ;
406 ;
407 ;
408 ;
409 ;
410 ;
411 ;
412 ;
413 ;
414 ;
415 ;
416 ;
417 ;
418 ;
419 ;
420 ;
421 ;
422 ;
423 ;
424 ;
425 ;
426 ;
427 ;
428 ;
429 ;
430 ;
431 ;
432 ;
433 ;
434 ;
435 ;
436 ;
437 ;
438 ;
439 ;
440 ;
441 ;
442 ;
443 ;
444 ;
445 ;
446 ;
447 ;
448 ;
449 ;
450 ;
451 ;
452 ;
453 ;
454 ;
455 ;
456 ;
457 ;
458 ;
459 ;
460 ;
461 ;
462 ;
463 ;
464 ;
465 ;
466 ;
467 ;
468 ;
469 ;
470 ;
471 ;
472 ;
473 ;
474 ;
475 ;
476 ;
477 ;
478 ;
479 ;
480 ;
481 ;
482 ;
483 ;
484 ;
485 ;
486 ;
487 ;
488 ;
489 ;
490 ;
491 ;
492 ;
493 ;
494 ;
495 ;
496 ;
497 ;
498 ;
499 ;
500 ;
501 ;
502 ;
503 ;
504 ;
505 ;
506 ;
507 ;
508 ;
509 ;
510 ;
511 ;
512 ;
513 ;
514 ;
515 ;
516 ;
517 ;
518 ;
519 ;
520 ;
521 ;
522 ;
523 ;
524 ;
525 ;
526 ;
527 ;
528 ;
529 ;
530 ;
531 ;
532 ;
533 ;
534 ;
535 ;
536 ;
537 ;
538 ;
539 ;
540 ;
541 ;
542 ;
543 ;
544 ;
545 ;
546 ;
547 ;
548 ;
549 ;
550 ;
551 ;
552 ;
553 ;
554 ;
555 ;
556 ;
557 ;
558 ;
559 ;
560 ;
561 ;
562 ;
563 ;
564 ;
565 ;
566 ;
567 ;
568 ;
569 ;
570 ;
571 ;
572 ;
573 ;
574 ;
575 ;
576 ;
577 ;
578 ;
579 ;
580 ;
581 ;
582 ;
583 ;
584 ;
585 ;
586 ;
587 ;
588 ;
589 ;
590 ;
591 ;
592 ;
593 ;
594 ;
595 ;
596 ;
597 ;
598 ;
599 ;
600 ;
601 ;
602 ;
603 ;
604 ;
605 ;
606 ;
607 ;
608 ;
609 ;
610 ;
611 ;
612 ;
613 ;
614 ;
615 ;
616 ;
617 ;
618 ;
619 ;
620 ;
621 ;
622 ;
623 ;
624 ;
625 ;
626 ;
627 ;
628 ;
629 ;
630 ;
631 ;
632 ;
633 ;
634 ;
635 ;
636 ;
637 ;
638 ;
639 ;
640 ;
641 ;
642 ;
643 ;
644 ;
645 ;
646 ;
647 ;
648 ;
649 ;
650 ;
651 ;
652 ;
653 ;
654 ;
655 ;
656 ;
657 ;
658 ;
659 ;
660 ;
661 ;
662 ;
663 ;
664 ;
665 ;
666 ;
667 ;
668 ;
669 ;
670 ;
671 ;
672 ;
673 ;
674 ;
675 ;
676 ;
677 ;
678 ;
679 ;
680 ;
681 ;
682 ;
683 ;
684 ;
685 ;
686 ;
687 ;
688 ;
689 ;
690 ;
691 ;
692 ;
693 ;
694 ;
695 ;
696 ;
697 ;
698 ;
699 ;
700 ;
701 ;
702 ;
703 ;
704 ;
705 ;
706 ;
707 ;
708 ;
709 ;
710 ;
711 ;
712 ;
713 ;
714 ;
715 ;
716 ;
717 ;
718 ;
719 ;
720 ;
721 ;
722 ;
723 ;
724 ;
725 ;
726 ;
727 ;
728 ;
729 ;
730 ;
731 ;
732 ;
733 ;
734 ;
735 ;
736 ;
737 ;
738 ;
739 ;
740 ;
741 ;
742 ;
743 ;
744 ;
745 ;
746 ;
747 ;
748 ;
749 ;
750 ;
751 ;
752 ;
753 ;
754 ;
755 ;
756 ;
757 ;
758 ;
759 ;
760 ;
761 ;
762 ;
763 ;
764 ;
765 ;
766 ;
767 ;
768 ;
769 ;
770 ;
771 ;
772 ;
773 ;
774 ;
775 ;
776 ;
777 ;
778 ;
779 ;
780 ;
781 ;
782 ;
783 ;
784 ;
785 ;
786 ;
787 ;
788 ;
789 ;
790 ;
791 ;
792 ;
793 ;
794 ;
795 ;
796 ;
797 ;
798 ;
799 ;
800 ;
801 ;
802 ;
803 ;
804 ;
805 ;
806 ;
807 ;
808 ;
809 ;
810 ;
811 ;
812 ;
813 ;
814 ;
815 ;
816 ;
817 ;
818 ;
819 ;
820 ;
821 ;
822 ;
823 ;
824 ;
825 ;
826 ;
827 ;
828 ;
829 ;
830 ;
831 ;
832 ;
833 ;
834 ;
835 ;
836 ;
837 ;
838 ;
839 ;
840 ;
841 ;
842 ;
843 ;
844 ;
845 ;
846 ;
847 ;
848 ;
849 ;
850 ;
851 ;
852 ;
853 ;
854 ;
855 ;
856 ;
857 ;
858 ;
859 ;
860 ;
861 ;
862 ;
863 ;
864 ;
865 ;
866 ;
867 ;
868 ;
869 ;
870 ;
871 ;
872 ;
873 ;
874 ;
875 ;
876 ;
877 ;
878 ;
879 ;
880 ;
881 ;
882 ;
883 ;
884 ;
885 ;
886 ;
887 ;
888 ;
889 ;
890 ;
891 ;
892 ;
893 ;
894 ;
895 ;
896 ;
897 ;
898 ;
899 ;
900 ;
901 ;
902 ;
903 ;
904 ;
905 ;
906 ;
907 ;
908 ;
909 ;
910 ;
911 ;
912 ;
913 ;
914 ;
915 ;
916 ;
917 ;
918 ;
919 ;
920 ;
921 ;
922 ;
923 ;
924 ;
925 ;
926 ;
927 ;
928 ;
929 ;
930 ;
931 ;
932 ;
933 ;
934 ;
935 ;
936 ;
937 ;
938 ;
939 ;
940 ;
941 ;
942 ;
943 ;
944 ;
945 ;
946 ;
947 ;
948 ;
949 ;
950 ;
951 ;
952 ;
953 ;
954 ;
955 ;
956 ;
957 ;
958 ;
959 ;
960 ;
961 ;
962 ;
963 ;
964 ;
965 ;
966 ;
967 ;
968 ;
969 ;
970 ;
971 ;
972 ;
973 ;
974 ;
975 ;
976 ;
977 ;
978 ;
979 ;
980 ;
981 ;
982 ;
983 ;
984 ;
985 ;
986 ;
987 ;
988 ;
989 ;
990 ;
991 ;
992 ;
993 ;
994 ;
995 ;
996 ;
997 ;
998 ;
999 ;
1000 ;
```



```

75 C03F 80          or      b
76 C040 E8 3C     and     3ch
77 C042 3C        inc     a
78 C043 06 00     ld      b,0
79 C045 4F        ld      c,a
80 C046 21 FCCY   ld      hl,statr
81 C049 09        add     hl,bc
82 C04A 38 20     ld      (hl),20h ; attributo di un comando esteso
83 C04C 21 C0A7   ld      hl,last0 ; inizio del blocco da trasferire
84 C04F 11 4900   ld      de,4900h ; destinazione
85 C052 01 01C7   ld      bc,last-inst ; lunghezza del blocco
86 C055 E8 80     ld      bc,80
87 C057 CD 006C   call   instat ; screen 0
88 C05A 24 F3AE   ld      a,(lin40) ; centra il messaggio
89 C05D CD 2F     sra     a
90 C05F 06 0F     sub     15
91 C061 67        ld      h,a
92 C063 2C 04     ld      l,10
93 C064 CD 00C8   call   posit ; locate
94 C067 CD C09A   call   displ
95 C06A 49 6C 20 63 ; "il comando esteso e' installato",cr,lf,0
96 C06E 6F 6B 61 6E
97 C072 64 6F 20 65
98 C076 73 74 65 73
99 C07A 6F 20 65 60
100 C07E 20 69 6E 71
101 C082 74 61 6C 6C
102 C086 61 74 6F 60
103 C08A 04 00
104 C08C 06 04     ld      b,4
105 C08E 21 0000   delay: ld      hl,0 ; pausa
106 C091 28        delayt dec hl
107 C092 7C        ld      a,h
108 C093 85        or      l
109 C094 20 F8     jr      nz,delav0
110 C096 10 F6     djnz   delav
111 C098 18 95     jr      retrdr ; ritorna al Basic
112
113 ; Routine di stampa sullo schermo
114 C09A E3        displ: ex     (spl),hl
115 C098 7E        displ: ld     a,(hl)
116 C09C 23        inc     hl
117 C09D 47        and     a
118 C09E 28 05     jr      t,devit
119 C0A0 CD 00A2   call   chput
120 C0A3 18 F6     jr      displ
121 C0A5 E3        devit: ex     (spl),hl
122 C0A6 C9        ret
123 C0A7
124
125 ;
126
127 ; Inizio del comando esteso
128 ; phase 04000h
129 4000
130 4000 41 42     orb     "08" ; simulazione di una cartidge in ROM
131 4002 0000 4010 defw 0,stat0,0,0
132 4006 0000 0000
133 400A
134
135 ; Controllo del nome
136 stat0: push hl ; punta il primo carattere dopo il nome
137 4011 21 FDB9   ld      hl,procha ; nome del comando chiamato
138 4014 11 4022   ld      de,stat0a ; confronta con "LCOPY"
139 4017 14        nomlop: ld     a,(de)
140 4019 9E        cp      (hl)
141 401B 20 90     jr      nz,invnoa ; nome non valido
142 401B 47        and     a
143 401C 23        inc     hl
144 401D 17        inc     de
145 401E 20 F7     jr      nz,nomlop
146 4020 18 8F     jr      lcopy
147
148 4022 4C 43 4F 50 ; stat0a defb "LCOPY",0
149 4026 59 00
150 4028 E1        invnoa: pop hl ; lascia invariato il puntatore
151 4029 37        scf
152 402A C9        ret ; segnala l'errore
153
154 ; Main program
155 ;
156 ; Inizializzazioni
157 ;
158 ;
159 402B
160 4029 21 4082   ld      hl,interl ; fissa l'interlinea
161 402E CD 4098   call   print
162
163 ;
164 4031 21 0000   ld      hl,0 ; inizio della P.O.T.
165 4034 3E 18     ld      a,24 ; 24 linee
166
167 ; start:
168 4036 1D (savlin),a ; salva il byte nel buffer
169 4037 1D (svrvin),hl ; salva il byte nel buffer
170 403C 11 40C7   ld      de,buf
171 4042 CD 9059   call   idirav ; scarica 256 bytes
172
173 ;
174 4045 3E 20     ld      a,32
175 4047 32 404C   ld      (savcol),a ; colonne
176 404A 21 40C7   ld      hl,buf
177 404E 11 40BF   push   hl
178
179 4051
180 4051 9E 08     invert: ld     c,8 ; bytes da invertire
181 4053
182 4053 06 08     inv: ld      b,8 ; bits
183 4055 E1        pop     hl ; carica l'ind. del byte
184 4056 E5        push   hl ; salva
185
186 ;
187 4057
188 4057 CD 06     rlc     (hl) ; deposita un bit nel carry
189 4059 1F        rra

```

```

189 405A 23        inc     hl
190 405B 10 FA     djnz   inloop
191
192 405D 12        ld      (de),a ; salva il byte nel buffer
193 405E 1A        inc     de
194 405F 08        dec     c
195 4060 20 F1     jr      nz,inv ; inverti 8 bytes
196
197 4062 01        pop     de ; aggiusta i puntatori
198 4063 E5        push   hl
199
200 4064 34 40AC   ; completa tutta la riga
201 4067 2D        ld      a,(svrcol)
202 4068 32 40AC   dec     a
203 406B 20 E4     jr      nz,invert
204
205 ;
206 406B E1        ; stampa i 256 bytes
207 406E EB        pop     hl
208 406F 36 00     ex     de,hl
209 4071 23        ld      (hl),cr
210 4072 38 0A     inc     hl
211 4074 21 4087   ld      hl,ndata
212 4077 CD 409E   call   print
213
214 407A 24 40A0   ; altra riga
215 407D 24        ld      hl,(svrvin)
216 407E 24        inc     hl
217
218 407E 34 40A8   ;
219 4081 2D        ld      a,(savlin)
220 4082 20 82     dec     a
221 4084 21 40AF   jr      nz,start ; ripristina l'interlinea
222 4087 CD 409E   ld      hl,txtint
223 call   print
224
225 ; ritorno al basic senza errore
226 ; posizione il puntatore alla fine dell'istruzione Basic
227 408A
228 408A E1        pop     hl
229 408B 7E        exloop: ld     a,(hl)
230 408C 87        or      a ; e' zero?
231 408D 28 07     jr      z,exit0
232 408F FE 34     cp      "i"
233 4091 28 03     jr      z,exit0
234 4092 23        inc     hl
235 4094 18 F5     jr      exloop
236 4096 47        exit0: and    a ; resetta il carry
237 4097 C9        ret
238
239 ;
240
241 ; Subroutine di stampa
242
243 4098
244 4098 00        print:
245 4098 4E        ld      c,(hl) ; in BC il numero dei caratteri
246 4099 23        inc     hl ; da stampare
247 409A 46        ld      b,(hl)
248 409B 23        sloop: inc hl
249 409C 7E        ld      a,(hl)
250 409D CD 00A5   call   lptout
251 409E 28 06     jr      c,return ; interrotto?
252 40A0 08        dec     bc
253 40A1 79        ld      a,c
254 40A2 80        or      b
255 40A3 80        jr      nz,sloop
256 40A4 80        return: pop hl
257 40A5 18 8F     jr      exit
258
259 ;
260
261 ; Buffers
262
263 40A8 00        savlin: defb 0
264 40A9 00        svrvin: defb 0
265 40AA 02        cora: defb 2
266 40AB 02        trint: defb 2
267 40AC 18 41     defb 27,65
268 40AD 02        interl: defb 2
269 40AE 18 42     defb 27,66
270 40AF 0108     ndata: defb 24
271
272 40B0 18 53 30 32 ; defb 27,"SO256" ; modo grafico
273 40B1 35 36
274 40B2 08        buf1: defb 8
275 40B3 25        buf2: defb 256
276
277 ;
278
279 ; last: .dephase
280 end

```

## Macros:

```

$define:
40C7 BUFF      40BF BUFF1    00A2 CHPUT
00A0 CR        C09E DELAY   C091 DELAY0
00A5 DEIT      C09A DISPL  C098 DISPL1
0024 ENASLT    409A EXIT   4096 EXIT0
408B ELOOP     C090 FIRST  4000 INIT
00A3 INITIT    4057 INLOOP  4083 INTERL
4053 INV       4051 INVERT  4028 INWINM
41C7 LAST      C0A7 LAST0   4028 LCOPY
0059 LDIRMV    000A LF     F0AE LIN40
00A5 LPTOUT    4017 NDMLP   C0A2 NXTSLT
4087 PDATA     407B PLOOP   00C6 POSIT
4098 PRINT     F0B9 PROCNM  C037 RANDM
C02F RETADR    40A8 RETURN  40A4 SAVCOL
40A8 SAVLIN    FCC9 SLTATR  4036 START
4010 STATM     4022 STNMN   40A0 SVRVIN
40A7 TXTINT

```

No Fatal errors!

MC