

```

2290 002533* 001000 0000257 33720 XMA A 0000257
2291 002534* 001000 0000062 33740 STA DURES* FOLLOW CRUNCHING
2292 002533* 000000 001544*
2293 002536* 000000 002531*
2294 002537* 001000 0000010 33760 MVI C,5 I/COUNT OF CHARS AT LEAST 5
2295 002540* 000000 0000005
2296 002541* 000000 0000021 33780 LXI D,BUF I/SETUP DESTINATION POINTER
2297 002542* 001000 001431*
2298 002543* 000000 002535*
2299 002544* 001000 0000176 33800 KLOOP: MOV A,M I/GET CHARACTER FROM BUF
2300 002545* 001000 0000376 33820 CPI " " I/IS IT A SPACE WE WANT TO SAVE
2301 002546* 000000 0000040
2302 002547* 001000 0000312 33840 JZ STUFFH I/YES, STUFF IN DESTINATION LINE.
2303 002550* 000000 002607*
2304 002551* 000000 002542*
2305 002552* 001000 0000107 33860 MOV B,A I/GET A CHARACTER FROM THE LINE
2306 33880 I/SETUP B WITH A QUOTE IF IT IS A STRING
2307 002553* 001000 0000376 33900 CPI 34 I/QUOTE SIGN?
2308 002554* 000000 0000042
2309 002555* 001000 0000312 33920 JZ STRNG I/YES, GO TO SPECIAL STRING HANDLING
2310 002556* 000000 002727*
2311 002557* 000000 002550*
2312 002560* 001000 002607* 33940 ORA A I/END OF LINE?
2313 002561* 001000 0000312 33960 JZ CRDNE I/YES, DONE CRUNCHING
2314 002562* 000000 002753*
2315 002563* 000000 002556*
2316
2317 002564* 001000 000072 33980 IFN STRING,<
2318 002565* 000000 001544* 34000 LDA DURES I/IN DATA STATEMENT AND NO CRUNCH?
2319 002566* 000000 002562*
2320 002567* 001000 0000267 34020 ORA A
2321 002570* 001000 0000107 34040 MOV B,A I/INITIALIZE RESERVED WORD COUNT
2322 002571* 001000 0000176 34060 MOV A,M I/GET THE CHARACTER AGAIN
2323 002572* 001000 0000302 34080 JNZ STUFFH I/IF NO CRUNCHING JUST STORE
2324 002573* 000000 002607*
2325 002574* 000000 002565*
2326
2327 34100
2328 002575* 001000 0000376 34120 IFN LENGTH,<
2329 002576* 000000 0000077 34140 CPI "?" I/NOT FOR 4K VERSION
2330 002577* 001000 0000076 34160 MVI A,PRINTK I/A UNMARK?
2331 002600* 000000 0000231
2332 002601* 001000 0000312 34180 JZ STUFFH I/THEN USE A "PRINT" TOKEN
2333 002602* 000000 002607*
2334 002603* 000000 002573*
2335 002604* 001000 0000176 34200 MOV A,M
2336 002605* 001000 0000376 34220 CPI "0" I/SKIP NUMERICS
2337 002606* 000000 0000060
2338 002607* 001000 0000332 34240 JC MUSTCH I/SINCE CRUNCHING IS SLOW
2339 002610* 000000 002617*
2340 002611* 000000 002602*
2341 002612* 001000 0000376 34260 CPI 60 I/"0" ALSO PUT IN QUICKLY
2342 002613* 000000 0000074

```

```

2343 002614* 001000 0000332 34280 JC STUFFH
2344 002615* 000000 002607*
2345 002616* 000000 002610*
2346 002617*
2347 002617* 001000 0000325 34300 MUSTCH: >
2348 34320 PUSH D I/SAVE STORE POINTER
2349 34340 IFE STRING,<
2350 34360 MVI B,0 I/INIT RESERVED WORD COUNT
2351 002620* 001000 0000021 34380 LXI D,RESLST-1 I/INIT TO RESERVED WORD LIST
2352 002621* 000000 0000171*
2353 002622* 000000 002615*
2354 002623* 001000 0000345 34400 PUSH H I/SAVE IN BUF POINTER
2355 002624* 001000 0000076 34420 XWD "01000","076 I/"MVI" AROUND CHRGET
2356 002625* 001000 0000327 34440 NEXTRES: CHRGET I/GET CHAR FROM LINE
2357 002626* 001000 0000025 34460 INX D I/BUMP DEPOSIT POINTER
2358 002627* 001000 0000032 34480 RESER: LDAX D I/GET A BYTE FROM RESERVED WORD LIST
2359 002630* 001000 0000346 34500 ANI 127 I/GET RID OF SIGN BIT
2360 002631* 000000 0000177
2361 002632* 001000 0000312 34520 JZ TABEND I/END OF RESERVED WORD TABLE
2362 002633* 000000 002655*
2363 002634* 000000 002621*
2364 002635* 001000 0000276 34540 CMP M I/NO CHARS THE SAME?
2365 002636* 001000 0000302 34560 JNZ NTHIS I/AND, DIFFERENT
2366 002637* 000000 002736*
2367 002640* 000000 002635*
2368 002641* 001000 0000032 34580 LDAX D I/GET RESERVED WORD BYTE
2369 002642* 001000 0000267 34600 ORA A I/SET CONDITION CODES
2370 002643* 001000 0000362 34620 JP NEXTRES I/IF SIGN SET, RESERVED WORD FOUND
2371 002644* 000000 002625*
2372 002645* 000000 002637*
2373 002646* 001000 0000361 34640 FOUND: POP PSH I/TAKE OFF GARBAGE ORIG POINTER
2374 002647* 001000 0000170 34660 MOV A,B I/GET RESERVED WORD #
2375 002648* 001000 0000366 34680 ORI 128 I/SET MSB TO FLAG AS RESERVED WORD
2376 002649* 000000 0000000
2377 002652* 001000 0000362 34700 XWD "01000","0362 I/"JMP" AROUND THE POP H AND MOV A,M
2378 002653* 001000 0000341 34720 TABEND: POP H I/GET BACK ORIG POINTER
2379 002654* 001000 0000176 34740 MOV A,M I/GET BACK ORIG CHAR
2380 002655* 001000 0000521 34760 POP D I/GET STUFF POINTER BACK
2381 002656* 001000 0000353 34800 IFE LENGTH=<,<
2382 002657* 001000 0000376 34820 XCHG I/[M,L]=STUFF POINTER
2383 002660* 000000 0000220 34840 CPI ELSETK I/HAVE TO PUT A HIDDEN
2384 34860
2385 002661* 001000 0000066 34880 MVI H,"" I/COLON IN FRONT OF "ELSE"
2386 002662* 000000 0000072 I/STORE IT
2387 002663* 001000 0000314 34900 CZ INXHT# I/ADVANCE POINTER ON "ELSE"
2388 002664* 000000 0000000
2389 002665* 000000 002644*
2390 34920
2391
2392 002666* 001000 0000353 34940 STUFFH: XCHG> I/[O,E]=STUFF POINTER
2393 002667* 001000 0000443 34960 INX H I/ENTRY TO BUMP [M,L]
2394 002670* 001000 0000022 34980 STAX D I/SAVE CHARACTER IN CRUNCHED LINE
2395 002671* 001000 0000023 34960 INX D I/BUMP SAVE POINTER

```

```

2396 002672' 001000 000014 35000 INR C #BUMP CRUNCHED CHARACTER COUNT
2397 35020 IFN STRING,< SUI "!" #SEE IF IT IS A COLON
2398 35040 SUI "!" #SEE IF IT IS A COLON
2399 002673' 001000 000026 35000 JZ COLIS #IF SO ALLOW CRUNCHING AGAIN
2400 002674' 000000 000072 35000 JZ COLIS #IF SO ALLOW CRUNCHING AGAIN
2401 002675' 000000 000012 35000 JZ COLIS #IF SO ALLOW CRUNCHING AGAIN
2402 002676' 000000 002705' 35000 JZ COLIS #IF SO ALLOW CRUNCHING AGAIN
2403 002677' 000000 002664' 35000 CPI DATATK="!" #SEE IF IT IS A DATA TOKEN
2404 002701' 000000 000011 35100 JNZ NODATT #SEE IF IT IS A DATA TOKEN
2405 002702' 001000 000030 35100 JNZ NODATT #SEE IF IT IS A DATA TOKEN
2406 002703' 000000 002710' 35120 COLIS: STA DORES #SETUP FLAG
2407 002704' 000000 002676' 35140 NODATT: SUI REMTK="!" #WAS IT A REM STATEMENT
2408 002705' 001000 000062 35100 JNZ KLOOP #KEEP LOOPING
2409 002706' 000000 001544' 35100 IFE STRING,<SUI REMTK> #WAS IT A REM STATEMENT
2410 002707' 000000 002703' 35100 JNZ KLOOP #KEEP LOOPING
2411 002710' 001000 000026 35140 NODATT: SUI REMTK="!" #WAS IT A REM STATEMENT
2412 002711' 000000 000124 35100 IFE STRING,<SUI REMTK> #WAS IT A REM STATEMENT
2413 35100 JNZ KLOOP #KEEP LOOPING
2414 002712' 001000 000030 35100 JNZ KLOOP #KEEP LOOPING
2415 002713' 000000 002544' 35100 JNZ KLOOP #KEEP LOOPING
2416 002714' 000000 002706' 35100 JNZ KLOOP #KEEP LOOPING
2417 002715' 001000 000017 35200 MOV B,A #BUMP DOESN'T STOP ON "!", ONLY ON A ZERO
2418 002716' 001000 000176 35220 STR1: MOV A,M #SET CONDITION CODES
2419 002717' 001000 000267 35240 OWA A #SET CONDITION CODES
2420 002720' 001000 000312 35260 JZ CROONE #IF END OF LINE THEN DONE
2421 002721' 000000 002753' 35260 JZ CROONE #IF END OF LINE THEN DONE
2422 002722' 000000 002713' 35260 JZ CROONE #IF END OF LINE THEN DONE
2423 002723' 001000 000070 35260 CMP B #END OF GOBBLE
2424 002724' 001000 000312 35300 JZ STUFFH #IF YES, DONE WITH STRING
2425 002725' 000000 002667' 35300 JZ STUFFH #IF YES, DONE WITH STRING
2426 002726' 000000 002721' 35300 JZ STUFFH #IF YES, DONE WITH STRING
2427 002727' 001000 000043 35320 STRNG: INX H #INCREMENT TEXT POINTER
2428 002730' 001000 000022 35340 STAX D #STORE CHAR
2429 002731' 001000 000014 35360 INR C #BUMP COUNT
2430 002732' 001000 000023 35380 INX O #AND POINTER
2431 002733' 001000 000503 35400 JMP STR1 #KEEP LOOPING
2432 002734' 000000 002716' 35400 JMP STR1 #KEEP LOOPING
2433 002735' 000000 002725' 35400 JMP STR1 #KEEP LOOPING
2434 35400 JMP STR1 #KEEP LOOPING
2435 002736' 001000 000341 35440 NTHIS: POP H #RESTORE TEXT STRING
2436 002737' 001000 000345 35440 NTHIS: POP H #RESTORE TEXT STRING
2437 002740' 001000 000004 35480 INR B #INCREMENT RESERVED WORD #
2438 002741' 001000 000353 35500 XCHG #RESLST POINTER INTO (H,L)
2439 002742' 001000 000666 35520 NTHIS: OWA M #TEST BITS IN RESERVED WORD LIST
2440 002743' 001000 000043 35540 INX H #TEST BITS IN RESERVED WORD LIST
2441 002744' 001000 000362 35560 JP NTHIS1 #SKIP MORE
2442 002745' 000000 002742' 35560 JP NTHIS1 #SKIP MORE
2443 002746' 000000 002734' 35560 JP NTHIS1 #SKIP MORE
2444 002747' 001000 000353 35580 XCHG #RESLST POINTER INTO (D,E)
2445 35580 XCHG #RESLST POINTER INTO (D,E)
2446 002750' 001000 000363 35620 JMP RESER #TEXT POINTER INTO (H,L)
2447 002751' 000000 002627' 35620 JMP RESER #DONE, MOVE TO NEXT RESERVED WORD
2448 002752' 000000 002745' 35620 JMP RESER #DONE, MOVE TO NEXT RESERVED WORD

```

```

2449 002753' 001000 000041 35640 CROONE: LXI H,BUFMIN #LEAVE WITH (H,L) POINTER TO START OF LINE
2450 002754' 000000 001430' 35640 CROONE: LXI H,BUFMIN #LEAVE WITH (H,L) POINTER TO START OF LINE
2451 002755' 000000 002751' 35640 CROONE: LXI H,BUFMIN #LEAVE WITH (H,L) POINTER TO START OF LINE
2452 002756' 001000 000022 35660 STAX D #NEED THREE 0'S ON THE END
2453 002757' 001000 000023 35680 INX D #ONE FOR END-OF-LINE
2454 002760' 001000 000022 35700 STAX D #ONE 2 FOR A ZERO LINK
2455 002761' 001000 000023 35720 INX D #SINCE IF THIS IS A DIRECT STATEMENT
2456 002762' 001000 000022 35740 STAX D #ITS END MUST LOOK LIKE THE END OF A PROGRAM
2457 002763' 000000 000311 35760 RET #END OF CRUNCHING
2458 35780 RET #END OF CRUNCHING
2459 35800 RET #END OF CRUNCHING
2460 35820 RET #END OF CRUNCHING
2461 35840 RET #END OF CRUNCHING
2462 35860 RET #END OF CRUNCHING
2463 35880 RET #END OF CRUNCHING
2464 35900 RET #END OF CRUNCHING
2465 35920 RET #END OF CRUNCHING
2466 35940 RET #END OF CRUNCHING
2467 002764' 001000 000005 35960 LINLIN: OCA B #BACK ARROW SO DECREMENT COUNT
2468 002765' 001000 000053 35980 OCA H #BACK UP POINTER
2469 36000 IFN REALIO,< OUTCHR> #BACK UP POINTER
2470 002766' 001000 000337 36020 OUTCHR> #BACK UP POINTER
2471 002767' 001000 000302 36040 JNZ INLIN #NOT TOO MANY SO CONTINUE
2472 002770' 000000 000035' 36040 JNZ INLIN #NOT TOO MANY SO CONTINUE
2473 002771' 000000 002754' 36060 INLIN: IFN REALIO,< OUTCHR> #NOT TOO MANY SO CONTINUE
2474 002772' 001000 000337 36080 INLIN: IFN REALIO,< OUTCHR> #NOT TOO MANY SO CONTINUE
2475 002772' 001000 000337 36100 INLIN: IFN REALIO,< OUTCHR> #NOT TOO MANY SO CONTINUE
2476 002773' 001000 000315 36120 CALL CRDO #PRINT THE #, OR A SECOND # IF THERE
2477 002774' 000000 004523' 36120 CALL CRDO #WERE TOO MANY
2478 002775' 000000 002770' 36120 CALL CRDO #TYPE A CRLF
2479 002776' 000000 000041 36140 INLIN: LXI H,BUF #TYPE A CRLF
2480 002777' 000000 001431' 36140 INLIN: LXI H,BUF #TYPE A CRLF
2481 003000' 000000 002774' 36160 MVI B,1 #CHARACTER COUNT
2482 003001' 001000 000006 36180 INLIN: CALL INCHR #GET A CHARACTER
2483 003002' 000000 000001 36180 INLIN: CALL INCHR #GET A CHARACTER
2484 003003' 001000 000315 36180 INLIN: CALL INCHR #GET A CHARACTER
2485 003004' 000000 003126' 36180 INLIN: CALL INCHR #GET A CHARACTER
2486 003005' 000000 002777' 36180 INLIN: CALL INCHR #GET A CHARACTER
2487 003006' 001000 000376 36182 IFN LENGTH,< CPI 7 #IS IT BOB ALBRECHT RINGING THE BELL
2488 003007' 000000 000007 36184 CPI 7 #IS IT BOB ALBRECHT RINGING THE BELL
2489 003010' 000000 000312 36186 JZ GOODCH> #FOR SCHOOL KIDS?
2490 003011' 000000 003044' 36186 JZ GOODCH> #FOR SCHOOL KIDS?
2491 003012' 000000 003004' 36200 CPI 13 #IS IT A CARRIAGE RETURN?
2492 003013' 001000 000376 36220 JZ FININL #IF SO FINISH UP
2493 003014' 000000 000015 36240 CPI 32 #CHECK FOR FUNNY CHARACTERS
2494 003015' 000000 000312 36260 JZ FININL #IF SO FINISH UP
2495 003016' 000000 004516' 36260 JZ FININL #IF SO FINISH UP
2496 003017' 000000 003011' 36260 JZ FININL #IF SO FINISH UP
2497 003020' 001000 000376 36260 JZ FININL #IF SO FINISH UP
2498 003021' 000000 000044' 36260 JZ FININL #IF SO FINISH UP
2499 003022' 001000 000332 36260 JZ FININL #IF SO FINISH UP

```

```

2502 003023* 000000 003003*
2503 003024* 000000 003016*
2504 003025* 001000 000376 36260 CPI 125
2505 003026* 000000 000175
2506 003027* 001000 000522 36300 JNC INLINC FBIG ONES BAD TOO
2507 003030* 000000 003003*
2508 003031* 000000 003023*
2509 003032* 001000 000376 36320 CPI "0" FLINE DELETE?
2510 003033* 000000 000100
2511 003034* 001000 000312 36340 JZ INLINC
2512 003035* 000000 002772*
2513 003036* 000000 003036*
2514 003037* 001000 000376 36360 CPI "A" ICHARACTER DELETE?
2515 003040* 000000 000137
2516 003041* 001000 000312 36380 JZ LINLIN
2517 003042* 000000 002764*
2518 003043* 000000 003035*
2519 003044* 001000 000117 36400 GOODCH: MOV C,A
2520 003045* 001000 000170 36420 MOV A,B
2521 003046* 001000 000376 36440 CPI BUFLIN
2522 003047* 000000 000110
2523 003050* 001000 000076 36460 MVI A,7 IGET A BELL IN CASE LINE TOO LONG,
2524 003051* 000000 000007
2525 003052* 001000 000322 36480 JNC OUTBEL FLINE TOO LONG, RING BELL,
2526 003053* 000000 003061*
2527 003054* 000000 003042*
2528 003055* 000000 000171 36500 MOV A,C
2529 003056* 001000 000161 36520 MOV M,C ISTORE THIS CHARACTER
2530 003057* 001000 000043 36540 INX H
2531 003060* 001000 000004 36560 INR B
2532 003061*
2533 OUTBEL:
2534 IFN REALIO,<
2535 003062* 001000 000303 36600 OUTCH: INLINC
2536 003063* 000000 003003*
2537 003064* 000000 003053*
2538 003065*
2539 OUTCON:
2540 003065* 001000 000302 36660 IFN CONTR,<
2541 003066* 000000 010024* 36680 JNZ PPSWRT INO, DO OUTPUT
2542 003067* 000000 003063*
2543 36720 IFN REALIO,<
2544 36740 IFN LPTSW,<
2545 LDA LPTPOS
2546 OKA A PRTFLG ISEE IF WE WANT TO TALK TO LPT
2547 JZ TTYCHK ITEST BITS
2548 POP PSH IIF ZERO THEN NOT
2549 PUSH PSH IGET CHARACTER WE WANT TO PRINT
2550 CPI 15 IIS IT CARRIAGE RETURN?
2551 CZ PRINTW IFORCE OUT A LINE
2552 JC PPSWRT IIF FUNNY CONTROL CHARACTER (LF) DO NOTHING
2553 LDA LPTPOS IWHERE ARE WE?
2554 CPI LPTLEN IAT THE END?

```

```

2555 36960 CNC PRINTW IYES, START OVER
2556 36980 INR A
2557 37000 STA LPTPOS
2558 37020 LPTWAT: IN 2
2559 37040 ANI 2
2560 37060 JZ LPTWAT
2561 37080 POP PSH
2562 37100 OUT 3 ISEND OUT CHAR
2563 37120 RET IRETURN
2564 37140 PATLPT: BLOCK 20
2565 FINLPT: XRA A IRESET PRINT FLAG SO OUTPUT
2566 37160 STA PRTFLG IGOS TO THE TERMINAL
2567 37180 LDA LPTPOS ISEE IF ANY LEFTOVERS MUST BE
2568 37200 OKA A IFORCED OUT
2569 37220 RZ IBY LOOKING AT LPTPOS
2570 37240 PRINTW: IN 2 IMAKE SURE LAST PRINT
2571 37260 ANI 2 IFINISHED BY TESTING DONE
2572 37280 JZ PRINTW IBIT
2573 I SEE IF BUFFER MUST BE EMPTIED
2574 37300 LDA LPTPOS
2575 37320 OKA A
2576 37340 JNZ PRINTW ICHARACTERS IN THE BUFFER?
2577 37360 MVI A,1 IIF SO DON'T CLEAR THE BUFFER
2578 37380 OUT 2 IOTHERWISE BUFFER MUST BE EMPTIED
2579 37400 OUT 2 ICLEAR THE BUFFER
2580 37420 PRINTW: MVI A,1 ITO PRINT A BLANK LINE
2581 37440 OUT 2 ITELL LPT TO PRINT
2582 37460 OCA A ISTATUS REG
2583 37480 STA LPTPOS I[A]#0
2584 37500 RET IRESET LINE PRINT POSITION
2585 003070*
2586 37520 TTYCHK:
2587 IFN STRING,<
2588 POP PSH IGET THE CHARACTER
2589 PUSH PSH IAND SAVE IT AGAIN
2590 CPI 32 IIS THIS A MEANINGFUL CHARACTER?
2591 JC TRYOUT IIF IT'S A NON-PRINTING CHARACTER
2592 37600
2593 37680 IFN LENGTH:CONTRW:LPTSW,<
2594 37700 LDA TTYPOS ISEE IF PRINT HEAD IS AT THE END OF THE LINE
2595 37720
2596 003077* 001000 000072 37740 CPI LINLEN IMODIFIED BY "TERMINAL WIDTH?" QUESTION IN INIT
2597 003100* 000000 000047*
2598 003101* 000000 003075*
2599 003102* 001000 000376 37760 LINPT:=1,1
2600 003103* 000000 000110 37780 CZ CRDO ITYPE CRLF AND SET TTYPOS AND (A)#0 IF SO
2601 003104* 001000 000314
2602 003105* 000000 000523*
2603 003106* 000000 003100*
2604 003107* 001000 000074 37800 INR A
2605 37820 IINCREMENT TTYPOS SINCE WE'RE
I GOING TO PRINT A CHARACTER.

```

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF MACRO 47(113) 03:12 10-SEP-75 PAGE 8-14
F3 MAC 6-SEP-64 03:11 ERROR HANDLER, READY, COMPACTIFICATION, NEW, CLEAR, MAIN

```

2608 003110' 001000 000062 57840 STA TTYPOS STORE NEW PRINT HEAD POSITION
2609 003111' 000000 000047' 37860 TRYOUT:
2610 003112' 000000 003105' 37880 IFN REALIO,<
2611 003113' 37900 NOPRIN: IN 0
2612 003113' 001000 000333 37920 CNLCA1=*,*1
2613 003114' 000000 000000 37940 ANI 0000E JGET STATUS
2614 003115' 001000 000340 37960 JNZ NOPRIN> JKEEP LOOPING
2615 003116' 000000 000000 37980 PUP PSW JGET CHARACTER BACK
2616 003117' 001000 000302 38000 OUT TIOCHN JSEND OUT THE CHAR
2617 003120' 000000 003113' 38020 CNLCB1=*,*1 JCONSOLE COMMAND CHANGE LOC
2618 003121' 000000 003111' 38040 RET JRETURN FROM OUTCHR
2619 003122' 001000 000361 38060 INCHM1 IFN REALIO,<
2620 003123' 001000 000323 38100 TRYIN: IN 0 JGET STATUS
2621 003124' 000000 000001 38120 CNLCA2=*,*1 JCONSOLE COMMAND CHANGE LOC
2622 003125' 001000 000311 38140 ANI 1000E JTEST BIT
2623 003126' 000000 000000 38160 JNZ TRYIN> JGO BACK & DO IT AGAIN
2624 003127' 001000 000346 38180 IN TIOCHN JGET A CHAR
2625 003128' 000000 000001 38200 CNLCB2=*,*1 JCONSOLE COMMAND CHANGE LOC
2626 003129' 001000 000346 38220 ANI 127 JGET RID OF PARITY BIT
2627 003130' 000000 000000 38240 IFN CONTRM,<
2628 003131' 001000 000376 38260 CP1 CINTM JIS IT SUPPRESS OUTPUT?
2629 003132' 000000 000000 38280 MNZ JGET LINE NUMBER INTO [D,E]
2630 003133' 001000 000000 38300 LDA CNTHFL JSAVE IN I/O FLAG
2631 003134' 000000 000000 38320 CMA STA JGET LINE GREATER THAN [D,E]
2632 003135' 001000 000000 38340 STA CNTHFL> JCOMPLEMENT ITS STATE
2633 003136' 000000 000000 38360 PAGE JSAVE BACK
2634 003137' 001000 000000 38380 IFN LENGTH=2,<
2635 003138' 000000 000000 38400 LFTSH,<
2636 003139' 001000 000000 38420 LIST: MVI A,1 JGET NON ZERO VALUE
2637 003140' 000000 000000 38440 CALL LINGET JSAVE IN I/O FLAG
2638 003141' 001000 000000 38460 MNZ JGET LINE NUMBER INTO [D,E]
2639 003142' 000000 000000 38480 CALL FNDLIN JSAVE IN I/O FLAG
2640 003143' 001000 000000 38500 PUSH B JSAVE START POINTER
2641 003144' 000000 000000 38520 CALL FNDLIN JGET POINTER TO LINE
2642 003145' 001000 000000 38540 PUSH B JSAVE START POINTER
2643 003146' 000000 000000 38560 MOV A,B JGET LINE GREATER THAN [D,E]
2644 003147' 001000 000000 38580 ORA C JSAVE START POINTER
2645 003148' 000000 000000 38600 JZ READY JGET LINE GREATER THAN [D,E]
2646 003149' 001000 000000 38620 IFN LISTEN,< JCHECK FOR CONTROL-C
2647 003150' 000000 000000 38640 CALL JSCNTC JPUT BACK ON
2648 003151' 001000 000000 38660 CALL CKOD JGO CNL TO START OUT
2649 003152' 000000 000000 38680 XTHL JSAVE LINE #
2650 003153' 001000 000000 38700 CALL LINPRT JGET LINE # INTO [H,L]
2651 003154' 000000 000000 38720 MOV A,M JAND WE WANT [H,L] ON THE STACK
2652 003155' 001000 000000 38740 CALL LINPRT JPRINT AS INT WITOUT LEADING SPACE
2653 003156' 000000 000000 38760 MOV A,M JRESTORE POINTER TO START OF TEXT
2654 003157' 001000 000000 38780 CALL LINPRT JALWAYS A SPACE AFTER THE LINE #
2655 003158' 000000 000000 38800 MOV A,M JGET A CHARACTER FROM LINE.
2656 003159' 001000 000000 38820 CALL LINPRT JIS IT A RESERVED WORD
2657 003160' 000000 000000 38840 MOV A,M JINCREMENT POINTER INTO TEXT
2658 003161' 001000 000000 38860 CALL LINPRT JZERO, END OF LINE, GET NEXT LINE
2659 003162' 000000 000000 38880 MOV A,M JREGULAR CHAR, JUST PRINT IT
2660 003163' 001000 000000 38900 CALL LINPRT JGET RID OF SIGN BIT AND ADD ONE
2661 003164' 000000 000000 38920 MOV A,M JGET RESERVED WORD # IN C
2662 003165' 001000 000000 38940 CALL LINPRT JSAVE CURRENT POSIT
2663 003166' 000000 000000 38960 MOV A,M JGET RESLST POINTER,
2664 003167' 001000 000000 38980 CALL LINPRT JSAVE
2665 003168' 000000 000000 39000 MOV A,M JGET CHARACTER FROM RESLST
2666 003169' 001000 000000 39020 CALL LINPRT JBUMP RESLST POINTER
2667 003170' 000000 000000 39040 MOV A,M JTEST BITS
2668 003171' 001000 000000 39060 CALL LINPRT JNOT AT END OF RESERVED WORD YET
2669 003172' 000000 000000 39080 MOV A,M JDECREMENT CHAR
2670 003173' 001000 000000 39100 CALL LINPRT JFROM START POINTER HERE
2671 003174' 000000 000000 39120 MOV A,M JNOT AT END OF RESLST YET,
2672 003175' 001000 000000 39140 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2673 003176' 000000 000000 39160 MOV A,M JSET CONDITION CODES
2674 003177' 001000 000000 39180 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2675 003178' 000000 000000 39200 MOV A,M JSET CONDITION CODES
2676 003179' 001000 000000 39220 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2677 003180' 000000 000000 39240 MOV A,M JSET CONDITION CODES
2678 003181' 001000 000000 39260 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2679 003182' 000000 000000 39280 MOV A,M JSET CONDITION CODES
2680 003183' 001000 000000 39300 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2681 003184' 000000 000000 39320 MOV A,M JSET CONDITION CODES
2682 003185' 001000 000000 39340 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2683 003186' 000000 000000 39360 MOV A,M JSET CONDITION CODES
2684 003187' 001000 000000 39380 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2685 003188' 000000 000000 39400 MOV A,M JSET CONDITION CODES
2686 003189' 001000 000000 39420 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2687 003190' 000000 000000 39440 MOV A,M JSET CONDITION CODES
2688 003191' 001000 000000 39460 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2689 003192' 000000 000000 39480 MOV A,M JSET CONDITION CODES
2690 003193' 001000 000000 39500 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2691 003194' 000000 000000 39520 MOV A,M JSET CONDITION CODES
2692 003195' 001000 000000 39540 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2693 003196' 000000 000000 39560 MOV A,M JSET CONDITION CODES
2694 003197' 001000 000000 39580 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2695 003198' 000000 000000 39600 MOV A,M JSET CONDITION CODES
2696 003199' 001000 000000 39620 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2697 003200' 000000 000000 39640 MOV A,M JSET CONDITION CODES
2698 003201' 001000 000000 39660 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2699 003202' 000000 000000 39680 MOV A,M JSET CONDITION CODES
2700 003203' 001000 000000 39700 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2701 003204' 000000 000000 39720 MOV A,M JSET CONDITION CODES
2702 003205' 001000 000000 39740 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2703 003206' 000000 000000 39760 MOV A,M JSET CONDITION CODES
2704 003207' 001000 000000 39780 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2705 003208' 000000 000000 39800 MOV A,M JSET CONDITION CODES
2706 003209' 001000 000000 39820 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2707 003210' 000000 000000 39840 MOV A,M JSET CONDITION CODES

```

BASIC MCS 8080 GATES/ALLEN/DAVIDOFF MACRO 47(113) 03:12 10-SEP-75 PAGE 9
F3 MAC 6-SEP-64 03:11 THE "LIST" COMMAND

```

2655 58420 SUBTTL THE "LIST" COMMAND
2656 58440 IFN LENGTH=2,<
2657 58460 IFN LFTSH,<
2658 58480 LIST: MVI A,1 JGET NON ZERO VALUE
2659 58500 CALL LINGET JSAVE IN I/O FLAG
2660 58520 MNZ JGET LINE NUMBER INTO [D,E]
2661 58540 CALL FNDLIN JSAVE IN I/O FLAG
2662 58560 PUSH B JSAVE START POINTER
2663 58580 CALL FNDLIN JGET POINTER TO LINE
2664 58600 PUSH B JSAVE START POINTER
2665 58620 MOV A,B JGET LINE GREATER THAN [D,E]
2666 58640 ORA C JSAVE START POINTER
2667 58660 JZ READY JGET LINE GREATER THAN [D,E]
2668 58680 IFN LISTEN,< JCHECK FOR CONTROL-C
2669 58700 CALL JSCNTC JPUT BACK ON
2670 58720 CALL CKOD JGO CNL TO START OUT
2671 58740 XTHL JSAVE LINE #
2672 58760 CALL LINPRT JGET LINE # INTO [H,L]
2673 58780 MOV A,M JAND WE WANT [H,L] ON THE STACK
2674 58800 CALL LINPRT JPRINT AS INT WITOUT LEADING SPACE
2675 58820 MOV A,M JRESTORE POINTER TO START OF TEXT
2676 58840 CALL LINPRT JALWAYS A SPACE AFTER THE LINE #
2677 58860 MOV A,M JGET A CHARACTER FROM LINE.
2678 58880 CALL LINPRT JIS IT A RESERVED WORD
2679 58900 MOV A,M JINCREMENT POINTER INTO TEXT
2680 58920 CALL LINPRT JZERO, END OF LINE, GET NEXT LINE
2681 58940 MOV A,M JREGULAR CHAR, JUST PRINT IT
2682 58960 CALL LINPRT JGET RID OF SIGN BIT AND ADD ONE
2683 58980 MOV A,M JGET RESERVED WORD # IN C
2684 59000 CALL LINPRT JSAVE CURRENT POSIT
2685 59020 MOV A,M JGET RESLST POINTER,
2686 59040 CALL LINPRT JSAVE
2687 59060 MOV A,M JGET CHARACTER FROM RESLST
2688 59080 CALL LINPRT JBUMP RESLST POINTER
2689 59100 MOV A,M JTEST BITS
2690 59120 CALL LINPRT JNOT AT END OF RESERVED WORD YET
2691 59140 MOV A,M JDECREMENT CHAR
2692 59160 CALL LINPRT JFROM START POINTER HERE
2693 59180 MOV A,M JNOT AT END OF RESLST YET,
2694 59200 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2695 59220 MOV A,M JSET CONDITION CODES
2696 59240 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2697 59260 MOV A,M JSET CONDITION CODES
2698 59280 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2699 59300 MOV A,M JSET CONDITION CODES
2700 59320 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2701 59340 MOV A,M JSET CONDITION CODES
2702 59360 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2703 59380 MOV A,M JSET CONDITION CODES
2704 59400 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2705 59420 MOV A,M JSET CONDITION CODES
2706 59440 CALL LINPRT JGET A CHARACTER FROM RESERVED WORD
2707 59460 MOV A,M JSET CONDITION CODES

```