TECHNICAL REPORT No. 8

A GUIDE TO THE USE OF HYCOMP1

STUART C. SHAPIRO
DAVID A. GRACE

DECEMBER 10, 1973

A GUIDE TO THE USE OF HYCOMP1

Stuart C. Shapiro
David A. Grace

HYCOMP1 is a program designed to give the novice programmer a feel for the way a computer executes programs at the machine language level. It simulates on the ADDS Consul 880, attached via the KRONOS interactive time sharing system to the CDC6600, the HYCOMP computer discussed in Terry Walker's introductory computer text.[1]

HYCOMP1 is a diminished version of HYCOMP in that:

i) memory consists of 100 words, as opposed to Walker's 1000

ii) only machine language, not assembly language, is understood.

A word consists of a sign and five digits. Alphanumeric and special characters are stored using two digits to code each character, two characters per word, in the rightmost four digits. Table 1 give the HYCOMP1 character codes.

When the contents of a word is interpreted as an instruction, the sign and the third digit are ignored, the first two digits are the operation code and the last two digits are the operand (usually an address).

±	Ø	P		s	s

1. Walker, Terry M., Introduction to Computer Science: An Interdisciplinary Approach, Allyn and Bacon, Inc., Boston, 1972.

The acceptible op. codes, organized by function are given in Table 2.

Interacting with HYCOMP1

Once logged into TELEX on the ADDS Consul 880, the following COMMANDS should be given.

```
BATCH,40000
GET,HYCOMP1/UN=7438
SNOBOLC,I=HYCOMP1,L=0
```

HYCOMP1 will then clear the screen and fill it with the 100 words of central memory (CM), the overflow, underflow and end-of-file flags (OF, UF and EOF), the instruction counter (IC), control word (CU) and accumulator (REG). The screen will then appear as in figure 1.

At this point any of the commands listed below may be entered.

LOAD XX - XX is a two digit number which designates the initial location into which the instructions and data are to be loaded. If XX is omitted, 00 is assumed. Note that XX is put into IC, the instruction counter. Also, OF, UF and EOF are initialized to 0 and the input line is positioned at the first "card" of data.

RUN XX - XX is a two-digit number which designates where the first instruction to be executed is. This initiates the program in CM. If XX is omitted, the program starts executing at instruction IC. If the program enters an infinite loop, it will be necessary to perform the following "dead start" procedure:

i) press the BREAK key

ii) type STOP

iii) issue the following commands:

REWIND,HYCOMP1.

SNOBOLC,I=HYCOMP1,L=0.

CY - one machine cycle is executed, consisting of the following steps:

i) The contents of the word whose address is in the IC is copied into the CU.

ii) IC is incremented by 1.

iii) The instruction in CU is executed.

Note that IC must have the value of the proper location. (No XX option is available.) This can be accomplished (if IC is incorrect) by issuing an IC instruction with the proper XX, then the CY command.

IC XX - XX is a two digit number. IC is set to XX.

UF X - X is 0 or 1. UF is set to X.

OF X - X is 0 or 1. OF is set to X.

EOF X - X is 0 or 1. EOF is set to X.

REG N - N is an optionally signed integer with no more than five digits. REG is set to N.

QUIT - this terminates HYCOMP1. Control is returned to TELEX Command Mode.

DATA - permits entering of a new set of data, one "card" at a time. Data must be no more than 80 characters (width of screen). After each "card" (80 or less characters), hit new line. All numeric data must have a sign in column 1.

Since HYCOMP1 input instructions read from a new card, only 1 datum is permitted per "card". To signal end of data, enter # in column 1. HYCOMP1 permits only 50 "cards", including the end of file.

DATA ADD - permits adding of data to end of existing data queue. The end of file (#) is automatically erased, but must be re-entered to signal end of data, as above.

HELP - this command prints out a list of all the above commands and a brief description of each.

To return after HELP or after an error message is issued, type a period and press the NEWLINE key.

It should be noted that as long as the "?" appears (signifying "ready"), any command may follow any other. For example, although it would be pointless, one could issue

LOAD 10 , LOAD 37 , LOAD , LOAD 94.

The result would be the same as issuing LOAD 94.

Care should be exercised in issuing a command (or entering an instruction or datum) only after "?" appears. Upon completion of entering the instruction (or command or datum), hit <new line> once. Response time varies with the system load, so be patient!

```
00:         01:         02:         03:         04:         05:         06:
07:         08:         09:         10:         11:         12:         13:
14:         15:         16:         17:         18:         19:         20:
21:         22:         23:         24:         25:         26:         27:
28:         29:         30:         31:         32:         33:         34:
35:         36:         37:         38:         39:         40:         41:
42:         43:         44:         45:         46:         47:         48:
49:         50:         51:         52:         53:         54:         55:
56:         57:         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:          OF=     UF=     IC=         CU=          REG=
                        EOF=
TO SEE A LIST OF AVAILABLE COMMANDS, TYPE HELP
TO PROCEED AFTER AN ERROR MESSAGE APPEARS, HIT  NEW LINE
TO LEAVE A MODE, JUST TYPE NEW COMMAND.   ? MEANS READY
INPUT APPEARS ON LINE 22, OUTPUT ON LINE 23
?_
```

Fig. 1. CRT screen after initialization

TABLE 1. HYCOMP1 Character Codes

ø	00	/	08	F	16	N	24	V	32	3	43
.	01	,	09	G	17	O	25	W	33	4	44
(	02	=	10	H	18	P	26	X	34	5	45
+	03	A	11	I	19	Q	27	Y	35	6	46
$	04	B	12	J	20	R	28	Z	36	7	47
*	05	C	13	K	21	S	29	0	40	8	48
)	06	D	14	L	22	T	30	1	41	9	49
-	07	E	15	M	23	U	31	2	42		

TABLE 2. HYCOMP1 Instruction Set

Data Transfer

01	REG←(ss)
02	ss ←(REG)

I/O (Numeric)

11	ss←input line
13	output line←(ss)

I/O (Character)

12	[ss,ss+(REG)-1] ←input line
14	output line←[ss, (ss+(REG)-1)]

Integer Arithmetic

21	REG←(REG)+(ss)
22	REG←(REG)-(ss)
23	REG←(REG)×(ss)
24	REG←(REG)/(ss)

Control

00	Stop
30	IC←ss
31	IC←ss if (REG)=0
32	IC←ss if (REG)<0
33	IC←ss if (REG)>0
34	IC←ss if (REG)≤0
35	IC←ss if (REG)≥0
36	IC←ss if (EOF)=1
37	IC←ss if (OF)=1
38	IC←ss if (UF)=1

Floating Point Arithmetic

41	REG←(REG)+(ss)
42	REG←(REG)-(ss)
43	REG←(REG)×(ss)
44	REG←(REG)/(ss)