

# IBM 1401 Simulator Usage

Robert M Supnik

31-Mar-2015

Copyright (c) 1993-2015, Robert M Supnik

COPYRIGHT NOTICE and LICENSE are at the end of this document.

## Contents

- Simulator Files
- IBM 1401 Features
- CPU
  - Memory
  - Registers
- 1402 Card Reader/Punch (CDR, CDP, STKR)
  - Card Reader
  - Card Punch
  - Stacker
  - Error handling
- 1403 Line Printer (LPT)
  - VFU
  - Registers
  - Error handling
- 1407 Inquiry Terminal (INQ)
  - Registers
- 1311 Disk Pack (DP)
  - Registers
  - Error handling
- 729 Magnetic Tape (MT)
  - Registers
  - Error handling
- Symbolic Display and Input
- Character Sets
- Old Conversions
- COPYRIGHT NOTICE and LICENSE

This memorandum documents the IBM 1401 simulator.

## Simulator Files

- sim/
  - scp.h
  - sim\_console.h
  - sim\_defs.h
  - sim\_fio.h
  - sim\_rev.h
  - sim\_sock.h
  - sim\_tape.h
  - sim\_timer.h
  - sim\_tmxr.h
  - scp.c
  - sim\_console.c
  - sim\_fio.c
  - sim\_sock.c
  - sim\_tape.c
  - sim\_timer.c
  - sim\_tmxr.c
- sim/i1401/
  - i1401\_defs.h
  - i1401\_dat.h
  - i1401\_cpu.c
  - i1401\_cd.c
  - i1401\_iq.c
  - i1401\_lp.c
  - i1401\_dp.c
  - i1401\_mt.c
  - i1401\_sys.c

## IBM 1401 Features

The IBM 1401 simulator is configured as follows:

device names	simulates
CPU	IBM 1401 CPU with 16K of memory
CDR,CDP	IBM 1402 card reader/punch
LPT	IBM 1403 line printer
INQ	IBM 1407 inquiry terminal
DP	IBM 1311 disk pack with five drives
MT	IBM 729 7-track magnetic tape controller with six drives

The IBM 1401 simulator implements many unique stop conditions. On almost any kind of error the simulator stops:

- Unimplemented opcode
- Reference to non-existent memory
- Reference to non-existent device
- No word mark under opcode
- Invalid A address
- Invalid B address
- Invalid instruction length
- Invalid modifier character
- Invalid branch address
- Invalid magtape unit number
- Invalid magtape record length
- Write to locked magtape drive
- Skip to unpunched carriage control tape channel
- Card reader hopper empty
- Address register wrap-around
- I/O check with I/O stop switch set
- Invalid disk drive
- Invalid disk sector address
- Invalid disk sector count
- Invalid disk address compare

The LOAD command is used to load a line printer carriage-control tape.

The DUMP command is not implemented.

## CPU

The CPU options include a number of special features and the size of main memory. Note that the Modify Address special feature is always included when memory size is greater than 4K.

SET CPU XSA	enable advanced programming special feature
SET CPU NOXSA	disable advanced programming
SET CPU HLE	enable high/low/equal special feature
SET CPU NOHLE	disable high/low/equal
SET CPU BBE	enable branch on bit equal special feature
SET CPU NOBBE	disable branch on bit equal
SET CPU MR	enable move record special feature
SET CPU NOMR	disable move record
SET CPU EPE	enable extended print edit special feature
SET CPU NOEPE	disable extended print edit
SET CPU MDV	enable multiply/divide special feature
SET CPU NOMDV	disable multiply/divide
SET CPU 4K	set memory size = 4K
SET CPU 8K	set memory size = 8K
SET CPU 12K	set memory size = 12K
SET CPU 16K	set memory size = 16K

If memory size is being reduced, and the memory being truncated contains non-zero data, the simulator asks for confirmation. Data in the truncated portion of memory is lost. Initially, memory size is 16K, and all special features are enabled.

## Memory

Memory is implemented as 7 bit BCD characters, as follows:

```
6      5      4      3      2      1      0
```

```
word  B bit A bit 8      4      2      1
mark  <- zone ->  <----- digit ----->
```

In BCD, the decimal digits 0-9 are (octal) values 012, 001, 002, 003, 004, 005, 006, 007, 010, 011, respectively. Signs are encoded in the zone bits, with 00, 01, and 11 being positive, and 10 being negative.

## Registers

CPU registers include the visible state of the processor. The 1401 has no interrupt system.

name	size	comments
IS	14	instruction storage address register (PC)
AS	14	A storage address register
BS	14	B storage address register
ASERR	1	AS invalid flag
BSERR	1	BS invalid flag
SSA	1	sense switch A
SSB	1	sense switch B
SSC	1	sense switch C
SSD	1	sense switch D
SSE	1	sense switch E
SSF	1	sense switch F
SSG	1	sense switch G
EQU	1	equal compare indicator
UNEQ	1	unequal compare indicator
HIGH	1	high compare indicator
LOW	1	low compare indicator
OVF	1	overflow indicator
IOCHK	1	I/O check switch
PRCHK	1	process check switch
ISQ[0:63]	14	IS prior to last branch;most recent IS change first
WRU	8	interrupt character

The CPU can maintain a history of the most recently executed instructions. This is controlled by the SET CPU HISTORY and SHOW CPU HISTORY commands:

SET CPU HISTORY	clear history buffer
SET CPU HISTORY=0	disable history
SET CPU HISTORY=n	enable history, length = n
SHOW CPU HISTORY	print CPU history
SHOW CPU HISTORY=n	print first n entries of CPU history

The maximum length for the history is 65536 entries.

The original character encoding used by the 1401 simulator was revised to be compatible with the coding used by Paul Pierce's 709X and 1401 simulators. The user can select between the original (old) and compatible (new) encodings, as follows:

SET CPU OLDCONVERSIONS	use original character encoding
SET CPU NEWCONVERSIONS	use compatible character encoding

NEWCONVERSIONS is the default.

## 1402 Card Reader/Punch (CDR, CDP, STKR)

The IBM 1402 card/reader punch is simulated as three independent devices: the card reader (CDR), the card punch (CDP), and the reader and punch stackers (STKR). STRK units 0, 1, 2, and 4 correspond to the reader normal stacker, reader stacker 1, shared stacker 2/8, and punch stacker 4, respectively.

Card punch and stacker units support both the business (1403 print chain A) and Fortran (1403 H chain) character sets:

SET CDP BUSINESS	business character set
SET CDP FORTRAN	Fortran character set

The business character set is the default.

### Card Reader

The card reader supports the BOOT command. BOOT CDR reads a card image into locations 1-80, sets a word mark under location 1, clears storage, and then transfers control to location 1.

The card reader normally reads data from disk files, while the punch and stackers write data to disk files. Text cards are simulated as ASCII text lines with terminating newlines; column binary cards are simulated as ASCII text lines with adjacent characters supplying half of the 12b column code. For each unit, the POS register specifies the number of the next data item to be read or written. Thus, by changing POS, the user can backspace or advance these devices.

As a convenience for testing, the simulator supports entering cards from the console keyboard. To enable and disable default input from the keyboard:

```
SET CDR DEFAULT          default to keyboard if no file attached
SET CDP NODEFAULT        always take input from an attached file
```

The card reader signals its readiness to read a card by typing:

ENTER CARD

Input is terminated by a carriage return. Backspace (rubout) will erase the previously entered character and echo backslash (\). ^E will interrupt the wait loop and return control to the simulator control package. While taking input from the keyboard, the card reader's "last card" logic is disabled.

The card reader registers are:

name	size	comments
LAST	1	last card indicator
ERR	1	error indicator
S1	1	stacker 1 select flag
S2	1	stacker 2 select flag
POS	32	position in reader input file
TIME	24	delay window for stacker select
BUF[0:79]	8	reader buffer

## Card Punch

The card punch registers are:

name	size	comments
ERR	1	error indicator
S4	1	stacker 4 select flag
S8	1	stacker 8 select flag
POS	32	position in punch output file

## Stacker

The stacker registers are:

name	size	comments
POS0	32	position, normal reader stack
POS1	32	position, reader stacker 1
POS2	32	position, shared stacker 2/8
POS4	32	position, punch stacker 4

## Error handling

Error handling is as follows:

device	error	processed as
reader	end of file	if SSA set, set LAST indicator on next read, report error and stop
reader	not attached	report error and stop
punch	OS I/O error	print error message. if IOCHK set, report error and stop. otherwise, set ERR indicator.
stacker	not attached	ignored
	OS I/O error	print error message. if IOCHK set, report error and stop.

## 1403 Line Printer (LPT)

The IBM 1403 line printer (LPT) writes its data, converted to ASCII, to a disk file. The POS register specifies the number of the next data item to be read or written. Thus, by changing POS, the user can backspace or advance the printer.

The line printer implements both 48- and 64-character print chains:

```
SET LPT 64                64-character print chain
SET LPT                   48 48-character print chain
```

The line printer also implements both the business (1403 print chain A) and Fortran (1403 H chain) character sets:

```
SET LPT BUSINESS          business print character set
SET LPT FORTRAN           Fortran character set
```

The default is the 64 character print chain with the business set.

The line printer can be set to output to the console window if no file is attached:

```
SET LPT DEFAULT           default output to console if not attached
SET LPT NODEFAULT         always output to a file.
```

## VFU

In addition, the line printer can be programmed with a carriage control tape. The LOAD command loads a new carriage control tape:

```
LOAD <file>               load carriage control tape file
```

The format of a carriage control tape consists of multiple lines. Each line contains an optional repeat count, enclosed in parentheses, optionally followed by a series of column numbers separated by commas. Column numbers must be between

1 and 12; a column number of zero denotes top of form. The following are all legal carriage control specifications:

<blank line>	no punch
(5)	5 lines with no punches
1,5,7,8	columns 1, 5, 7, 8 punched
(10)2	10 lines with column 2 punched
1,0 column	1 punched; top of form

The default form is 66 lines long, with column 1 and the top of form mark on line 1, and the rest blank.

## Registers

The line printer registers are:

name	size	comments
LINES	8	number of newlines after next print
LFLAG	1	carriage control flag (1 = skip, 0 = space)
CCTP	8	carriage control tape pointer
CCTL	8	carriage control tape length (read only)
ERR	1	error indicator
POS	32	position in the output file
CCT[0:131]	32	carriage control tape array

## Error handling

Error handling is as follows:

error	processed as
not attached	report error and stop
OS I/O error	print error message. If IOCHK set, report error and stop. Otherwise, set ERR indicator.

## 1407 Inquiry Terminal (INQ)

The IBM 1407 inquiry terminal (INQ) is a half-duplex console. It polls the console keyboard periodically for inquiry requests.

The inquiry terminal supports both the business (1403 print chain A) and Fortran (1403 H chain) character sets for output:

SET INQ BUSINESS	business character set
SET INQ FORTRAN	Fortran character set



The business character set is the default.

## Registers

The inquiry terminal registers are:

name	size	comments
INQC	7	inquiry request character (initially ESC)
INR	1	inquiry request indicator
INC	1	inquiry cleared indicator
TIME	24	polling interval

When the 1401 CPU requests input from the keyboard, the message [Enter] is printed out, followed by a new line. The CPU hangs waiting for input until either the return/enter key is pressed, or the inquiry request character is typed in. The latter cancels the type-in and sets INC.

The inquiry terminal has no errors.

## 1311 Disk Pack (DP)

The disk pack controller supports 5 drives, numbered 0 through 4. Disk pack options include the ability to enable address writing (formatting).

```
SET DPn ADDROFF          set unit n address enable off
SET DPn ADDRON           set unit n address enable on
```

Units can also be set ENABLED or DISABLED.

Unlike most simulated disks, the 1311 includes explicit representation for sector addresses. This is to support non-standard formats, such as the inclusion of the drive number in the sector address. As a result,

1311 sectors are 106 characters long: 6 address characters and 100 data characters. If the 1311 has not been formatted, the addresses are blanks and are synthesized, if needed, based on the sector number.

The 1311 also supports two modes of operation: move mode and load mode. In move mode, word marks are ignored on writes and left untouched on reads, and sectors hold 100 characters. In load mode, word marks are included on writes and stored on reads, and sectors hold 90 characters. No attempt is made to deal with sectors written in load mode and read in move mode, or vice versa; on a real 1401, this causes a fatal parity error.

## Registers

The disk pack controller implements these registers:

name	size	comments
ACC	1	access error indicator
PWC	1	parity or write check error indicator
WLR	1	wrong length record error indicator
UNA	1	unequal address compare error indicator
DSK	1	any disk error indicator
BSY	1	disk access busy indicator
LASTF	3	most recent function
TIME	24	seek time

The 1311 has a primitive overlapped seek capability. If TIME is set non-zero, the 1311 will report itself busy for the specified amount of time following a seek. This allows programs to utilize the seek time for processing.

### Error handling

Error handling is as follows:

error	processed as
not attached	set DSK indicator. if IOCHK set, report error and stop.

1311 data files are buffered in memory; therefore, end of file and OS I/O errors cannot occur.

## 729 Magnetic Tape (MT)

The magnetic tape controller supports six drives, numbered 1 through 6. Magnetic tape options include the ability to make units write enabled or write locked.

```
SET MTn LOCKED           set unit n write locked
SET MTn WRITEENABLED     set unit n write enabled
```

Magnetic tape units can be set to a specific reel capacity in MB, or to unlimited capacity:

```
SET MTn CAPAC=m          set unit n capacity to m MB (0 = unlimited)
SHOW MTn CAPAC            show unit n capacity in MB
```

Units can also be set ENABLED or DISABLED.

The magnetic tape simulator supports the BOOT command. BOOT MT reads the first record off tape, starting at location 1, and then branches to it. BOOT -N MT reads the next record off tape, without rewinding, starting at location 1, and then branches to it.

## Registers

The magnetic tape controller implements these registers:

name	size	comments
END	1	end of file indicator
ERR	1	error indicator
PAR	1	parity error indicator
POS1..POS6	32	position, drives 1 to 6

## Error handling

Error handling is as follows:

error	processed as
not attached	report error and stop
end of file	set error indicator
OS I/O	error print error message, set error indicator. if IOCHK set, report error and stop.

## Symbolic Display and Input

The IBM 1401 simulator implements symbolic display and input. Display is controlled by command line switches:

```
-c    display as single character (BCD for CPU and MT, ASCII for others)
-s    display as wordmark terminated BCD string (CPU only)
-m    display instruction mnemonics (CPU only)
-d    display 50 characters per line, with word marks denoted
      by "1" on the line below
```

In a CPU character display, word marks are denoted by ‘.

Input parsing is controlled by the first character typed in or by command line switches:

```
' or ", -c or -s    characters (BCD for CPU and MT, ASCII for others)
alphabetic          instruction mnemonic
numeric             octal number
```

Instruction input is free format, with spaces separating fields. There are six instruction formats: 1, 2, 4, 5, 7, and 8 characters:

```
1 character          opcode
```

```

2 character      opcode 'modifier
4 character      opcode address
5 character      opcode address 'modifier
7 character      opcode address address
8 character      opcode address address 'modifier

```

Addresses are always decimal, except for special I/O addresses in the A field, which may be specified as %xy, where x denotes the device and y the unit number.

For the CPU, string input may encompass multiple characters. A word mark is denoted by ‘ and must precede the character to be marked. All other devices can only accept single character input, without word marks.

## Character Sets

The IBM 1401 uses a 6b character code called BCD (binary coded decimal). Some of the characters have no equivalent in ASCII and require different representations:

BCD code	ASCII representation	IBM 1401 character	print chains
00	space		
01	1		
02	2		
03	3		
04	4		
05	5		
06	6		
07	7		
10	8		
11	9		
12	0		
13	# or =		= in H chain
14	@ or ’		’ in H chain
15	:		blank in A, H chains
16	>		blank in A, H chains
17	{	tape mark	blank in A, H chains
20	^	alternate blank	blank in A, H chains
21	/		
22	S		
23	T		
24	U		
25	V		
26	W		
27	X		
30	Y		

BCD code	ASCII representation	IBM 1401 character	print chains
31	Z		
32		record mark	
33	,		
34	% or (		( in H chain
35	~	word mark	blank in A, H chains
36	\		blank in A, H chains
37	"		blank in A, H chains
40	-		
41	J		
42	K		
43	L		
44	M		
45	N		
46	O		
47	P		
50	Q		
51	R		
52	!		
53	\$		
54	*		
55	]		blank in A, H chains
56	;		blank in A, H chains
57	—	delta	blank in A, H chains
60	&		
61	A		
62	B		
63	C		
64	D		
65	E		
66	F		
67	G		
70	H		
71	I		
72	?		
73	.		
74	)	lozenge	
75	[		blank in A, H chains
76	<		blank in A, H chains
77	}	group mark	blank in A, H chains

## Old Conversions

Starting with V3.5-1, the 1401 simulator was changed to use the same character set as the SimH 7094 (and other 7094 simulators). This involved the following

changes:

code	V3.5-0 or earlier	V3.5-1 or later
13	#	# or = on input
14	@	@ or ' on input
17	(	{
32	,	
34	%	% or ( on input
35	=	~
37	+	"
60	&	& or + on input
77	"	}

In addition, the word mark indicator was changed from ~ to '.

The 1401 simulator can be set to operate with either set of conventions:

```
SET CPU OLDCONVERSIONS
```

```
SET CPU NEWCONVERSIONS
```

The default is NEWCONVERSIONS.

## COPYRIGHT NOTICE and LICENSE

The following copyright notice applies to the SIMH source, binary, and documentation:

Original code published in 1993-2015, written by Robert M Supnik

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL ROBERT M SUPNIK BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the names of the authors shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from each author.