

HP41CX Program Listing

Title:	Logic Gates
Program Label:	LBL "GATE" / 297 bytes
Version:	1.0
Date:	15/08/1989

Description

This program determines the output level of a gate no matter how many inputs it has and what their states are. The six gate types supported are: NOR, NAND, OR, AND, EXOR, EXNOR. Helpful if you are gathering data working your way around an IC or debugging a circuit.

Core Logic

The program uses a nice way of displaying a scrolling menu and just hitting a key to divert the execution to another subroutine within **LBL A**. A presaved data file containing the gate names (in the above order) is recalled from extended memory and stored in RAM whereupon each is displayed in turn while the machine waits for a keypress. When one is detected, the program diverts to requesting the number of inputs for the gate. The user is then prompted for the status of the inputs by entering the letter **H** or **L** for High and Low.

A check is made that the correct number of inputs have been entered.

By calculating the equivalent numeric value of these two letters and multiplying by the number of inputs it is possible to determine if all the inputs are all high, low, or in some intermediate state.

By testing the status of flags set by the logic simulators **LBL 11-16** the program establishes the final output state of the gate.

OR	
ALL 0	0
ALL 1	1
MIXED	1

NOR	
ALL 0	1
ALL 1	0
MIXED	0

AND	
ALL 0	0
ALL 1	1
MIXED	0

NAND	
ALL 0	1
ALL 1	0
MIXED	1

The six gate types supported by the program.

EXOR	
ALL 0	0
ALL 1	0
MIXED	1

EXNOR	
ALL 0	1
ALL 1	1
MIXED	0

Execution

When the program is called, the user is presented with a sequential list of gates in rotation, when the appropriate gate is displayed, touch Enter. The program will then prompt for the number of inputs:

TOT INPUTS? Respond by entering the number and touching **R/S**

The program then prompts for the status of the inputs:

STATUS? Touch the **H** and **L** keys as appropriate. Touch **R/S**

If the number of inputs is not the same as that entered earlier the program will briefly show:

INPUT ERR and return the user to the **STATUS?** display.

The status of the gate will be shown in following manner: **NAND: LOW**

To try a different set of inputs to this gate, touch **R/S** and the **STATUS?** prompt will be displayed. Set flag 16 if you need to follow the display on a monitor or printer.

Extended Memory File

The names of the six gates are saved in an extended memory file so the data can be shared with other programs and to reduce the size of the program in main memory.

The **GADA** file serves this purpose for the **GATE** program.

Create an extended memory file of 6 registers called **GADA** by entering **6** in the X reg. and **GADA** in the Alpha reg. and executing **CRFLD**.

Enter the six gate names, **NOR, NAND, OR, AND, EXOR, EXNOR** into six consecutive main memory registers such **020 - 025** by entering them in turn into the Alpha register and executing **ASTO** for each with the appropriate register number.

You then have the following data in main memory:

Register	Data
020	NOR
021	NAND
022	OR
023	AND
024	EXOR
025	EXNOR

This block of registers is then copied to the registers of the **GADA** file by first setting the pointer to the beginning to the file:

Enter **0** in the X reg. and **GADA** in the Alpha reg. and execute **SEEKPTA**.

Copy the main memory registers by entering the range **20.025** in the X reg. and executing **SAVERX**.

This file is then available to the **GATE** program and to any others until you purge (delete) it from extended memory.

Program listing follows:



Line	Instruction	Comments
1	LBL "GATE"	
2	"15/08/89 V1"	
3	"GADA"	
4	CLX	
5	SEEKPTA	
6	11.016	
7	GETRX	Recalls gate names from GADA in registers 11 to 16
8	LBL A	Start of menu display
9	11.01601	Range of registers to scan with a counter as the last digit
10	STO 10	
11	LBL 20	Sub loop
12	CLA	
13	ARCL IND 10	Recall a gate name
14	AVIEW	
15	1	
16	GETKEYX	Capture a keypress
17	X<>Y	
18	X≠0?	
19	↓ GTO 00	Escape out of loop if key is pressed
20	ISG 10	
21	↑ GTO 20	
22	↑ GTO A	Re-display menu from start
23	LBL 00	
24	"TOT INPUTS?"	Enter no. of inputs
25	PROMPT	
26	STO 09	
27	LBL B	
28	CF 00	
29	CF 05	
30	"STATUS?"	Enter string of H & Ls corresponding to the state of the inputs
31	AON	
32	STOP	
33	append: " OK"	
34	AVIEW	
35	AOFF	
36	ALENG	Test of inputs entered
37	3	
38	-	
39	RCL 09	
40	X≠Y?	

Line	Instruction	Comments
41	↓ GTO 31	
42	ENTER^	Load stack with No. of inputs
43	ENTER^	
44	ENTER^	
45	1000.01	Calculate the values that would represent all high and all low states.
46	+	
47	1000	
48	÷	
49	STO 06	
50	CLX	
51	72	
52	*	
53	STO 07	
54	CLX	
55	76	
56	*	
57	STO 08	
58	CLST	
59	LBL 21	Read H & L string character by character and subtract from totals
60	ATOX	established above.
61	+	
62	ISG 06	Is either register brought to zero by subtracting characters?
63	↑ GTO 21	
64	ST- 07	
65	ST- 08	
66	RCL 07	
67	X=0?	
68	↓ GTO 01	
69	RCL 08	
70	X=0?	
71	↓ GTO 02	Call logic simulator.
72	SF 01	
73	LBL 01	
74	SF 00	
75	LBL 02	
76	← XEQ IND 10	
77	LBL 22	Compile final display on the basis of flags set or cleared by logic
78	CF 00	simulators.
79	CLA	
80	ARCL IND 10	

Line	Instruction	Comments
81	append: ": "	
82	FS? 05	
83	"HIGH"	
84	FC? 05	
85	"LOW"	
86	PROMPT	Display result
87	↑ GTO B	Loop back for another run
88	LBL 30	
89	FS?C 01	
90	↑ GTO 22	
91	RTN	
92	LBL 11	NOR simulation
93	↑ XEQ 30	
94	LBL 32	
95	CF 05	
96	FC?C 00	
97	SF 05	
98	RTN	
99	LBL12	NAND simulation
100	SF 05	
101	↑ XEQ 30	
102	↑ XEQ 32	
103	RTN	
104	LBL 13	OR simulation
105	SF 05	
106	↑ XEQ 30	
107	LBL 33	
108	SF 05	
109	FC?C 00	
110	CF 05	
111	RTN	
112	LBL 14	AND simulation
113	↑ XEQ 30	
114	↑ XEQ 33	
115	RTN	
116	LBL 15	EXOR simulation
117	SF 05	
118	↑ XEQ 30	
119	CF 05	
120	RTN	

Line	Instruction	Comments
121	LBL 16	EXNOR simulation
122	↑ XEQ 30	
123	SF 05	
124	RTN	
125	LBL 31	No. of inputs in H & L string do not match stated no. of inputs at start.
126	BEEP	(STO 09)
127	"INPUT ERR"	
128	AVIEW	
129	PSE	
130	CLD	
131	↑ GTO B	Loop back for another run.
132	END	

For more HP41CX programs, visit <<http://www.inventors-emporium.co.uk>>

This program listing is provided "as is" for personal and experimental uses only.
 Liability cannot be accepted for any use made of this information or for any consequences to your equipment, business or life in general. If you don't agree with these terms, please don't use the program.