

How To Use IC-Tester Program:

I. PROGRAMS

There are 2 version of IC-Tester right now. Original version program; CHIP.EXE (QuickBasic Ver 4.5 compiled), this is made by Paul Stenning (1993). Secondly, ICTESTER.EXE (TurboPascal Ver 7.0 compiled), made by me. The original version used serial communication (COM1 or COM2) port hardware, while mine used LPT printer parallel port (LPT1 ~ LPT3) hardware auto detect. But my program also can run on both versions of hardwares. I wonder if you still could find Paul's home page, because at my last visited, this site already disappeared. This site is <http://www.paulandmark.u-net.com/electron/computer/chiptester/chiptester.htm>.

This program used to test any simple IC function/gate like; and, or, nand, nor, not, buffer, flip-flop, counter, etc. Any logic likes input or output type. But cannot test analog type like; pll, op-amp, time delay, etc. Tri-state logic will be interpret as hi logic. Any IC that has a boolean truth table usually can be implemented to get its library. The perform only for a single IC and cannot do in circuit test.

II. MAIN MENU

After you run this program, the screen will display the menu. The option can be selected by choosing the numbers or hi-lite characters, or by using direction key and return key. This menu is;



Press the number or hi-lite key or move the cursor and press the enter key to choose one of these options;

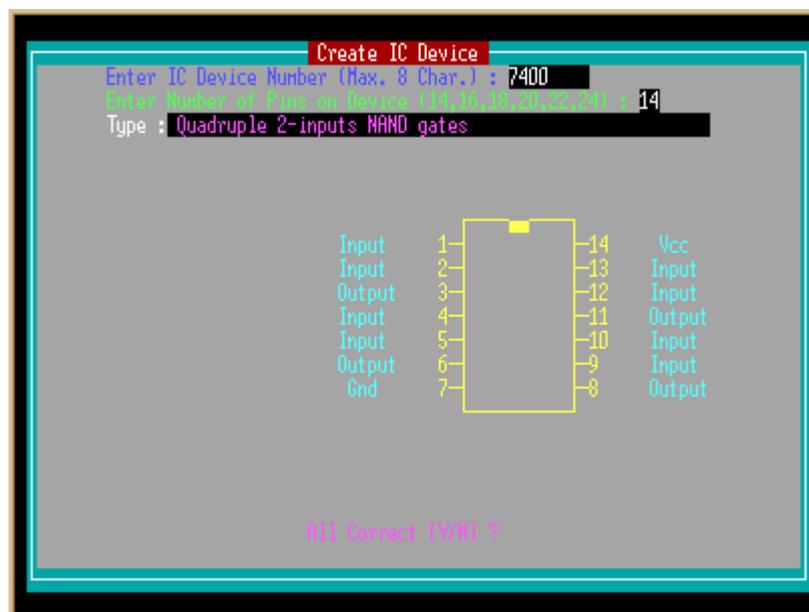
1. Create New IC Library
This must be run for the first time or when the IC libraries not create yet. You must make it by yourself, or using a few type accompanying with if it exists.
2. Load Existing IC Library
Load IC library from disk to IC data buffer.
3. Edit Existing IC Library
Edit IC library which already in the IC data buffer.
4. Debugging Test IC Device
This option must be run for the first time after you create the IC library, this IC must be test step by step to be sure the sequence is reliable.
5. Run Test IC Device
For every use option to test the IC and the perform only for one cycle.
6. Continuous Test IC Device
To test IC until failure for some that almost damage, more than one cycle.
7. Change Hardware Port
Setting the hardware device, which ports the hardware is connected.

III.1. CREATE NEW IC LIBRARY

This option only used for the first time when the IC library not exists or creates yet. After run this option, you will be asked for the IC library name. It is better to use the generic type for the name because the file name only 8 characters long (still used old DOS naming system directory), eg; 4000, 4001 ... and so on, for the CMOS family, 7400, 7401 ... and so on, for the TTL family. No need to add an extension, because the program will add automatically (the library extension is *.CHP and will be put to \LIBRARY subdirectory). At this prompt you can see the IC library file that already create or if you don't want to write the same name which already exist or if you doubt that the IC library already create or not, put a wildcard like *.CHP and press enter. The entire existing IC library file that match to this criteria will be displayed.



Then you will be asked for the pin quantity. IC that could be checked only which has pin quantity of 14-pins, 16-pins, 18-pins (seldom), 20-pins, 22-pins (seldom) and 24-pins. After that you will be asked for each of this pin whether it is an Input, Output, Clock, Vcc, Ground or Unused (not connected) pin. Select by inputting the number consecutively.



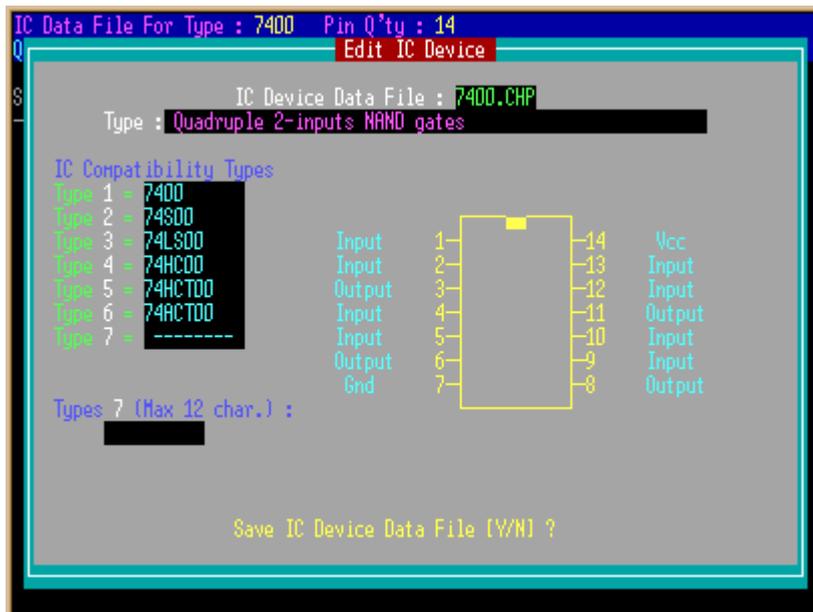
The different between +clock and -clock are; for +clock, the signal is a rising edge and standby lo while for -clock and the signal is a falling edge and standby hi. The input is any of logic gates input, counter input, flip-flop input, decoder, encoder, preset, reset, clear, set, etc. The output is any of logic gates output, counter output, flip-flop output, decoder, encoder, tri-state, etc. There are 7 sets of power connection. These sets are taken for standardize only and for the simplicity of the circuit. 6 sets are the common

configuration of power connection, ie; for 14-pins IC, the power takes pin 7 (ground) and pin 14 (Vcc), for 16-pins IC, the power takes pin 8 (ground) and pin 16 (Vcc), for 18-pins IC, the power takes pin 9 (ground) and pin 18 (Vcc), for 20-pins IC, the power takes pin 10 (ground) and pin 20 (Vcc), for 22-pins IC, the power takes pin 11 (ground) and pin 22 (Vcc), and for 24-pins IC, the power takes pin 12 (ground) and pin 24 (Vcc), while the last one for user define power connection. This last option must use a jumper wire and used for checking non-standard power configuration, eg; like a CMOS 4049, 16-pins IC, the power takes pin 8 (ground) and pin 1 (Vcc). The power configuration setting for this conditional used a jumper from pin 1 of the socket (ground) and pin 24 of the socket (Vcc). When the inputted pins completed, you go to the editor screen.

IC Data File For Type : 7400 Pin Q'ty : 14		
Quadruple 2-inputs NAND gates		
STEP	INPUTS	OUTPUTS
	1 2 4 5 9 10 12 13	3 6 8 11
1	0 0 0 0 0 0 0 0	1 1 1 1
2	0 1 0 0 0 0 0 0	1 1 1 1
3	1 0 0 0 0 0 0 0	1 1 1 1
4	1 1 0 0 0 0 0 0	0 1 1 1
5	0 0 0 1 0 0 0 0	1 1 1 1
6	0 0 1 0 0 0 0 0	1 1 1 1
7	0 0 1 1 0 0 0 0	1 0 1 1
8	0 0 0 0 0 1 0 0	1 1 1 1
9	0 0 0 0 1 0 0 0	1 1 1 1
10	0 0 0 0 1 1 0 0	1 1 0 1
11	0 0 0 0 0 0 0 1	1 1 1 1
12	0 0 0 0 0 0 1 0	1 1 1 1
13	0 0 0 0 0 0 1 1	1 1 1 0
14	1 1 1 1 1 1 1 1	0 0 0 0
15	1 0 1 1 1 1 1 1	1 0 0 0
16	0 1 1 1 1 1 1 1	1 0 0 0
17	0 0 1 1 1 1 1 1	1 0 0 0
18	1 1 1 0 1 1 1 1	0 1 0 0
19	1 1 0 1 1 1 1 1	0 1 0 0

←→/Tab>next entry,<Space>/<Enter>-set entry,<Ins>-insert line,<Esc>-cancel,

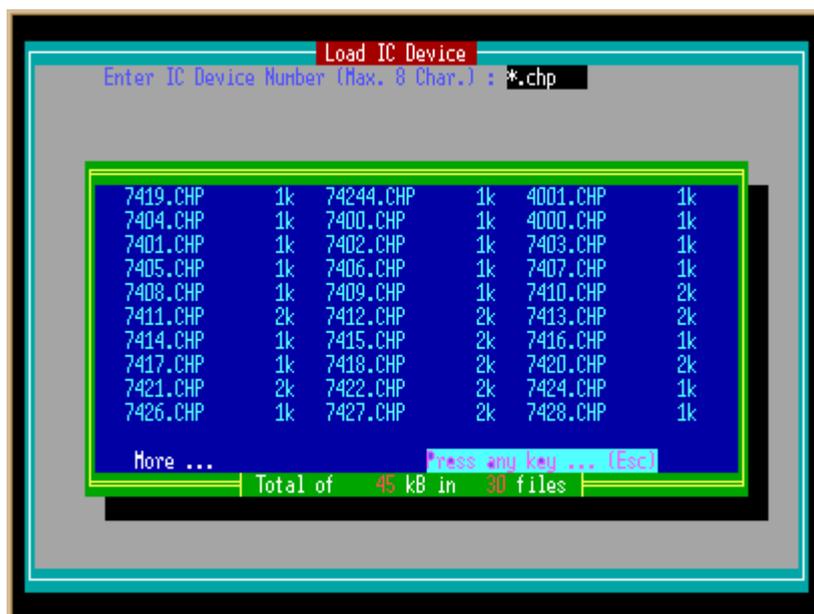
This screen shows the step for logic state sequence check. Step can be up to 100 count. These are an implementation of boolean truth table from the data book, like: 7400 TTL IC, this is a quadruple NAND gates, so if one of the input is lo, or both inputs are lo, the output is hi and if both inputs are hi, the output will be lo. You are freely to put the data combination as you want, as long as the boolean logics are true for each gate. Please refer to the sample of a few IC libraries that already created by Paul or by me. After you think the steps are enough, press the <Q> key to quit. The screen will display IC compatibility names and prompt you to input these names. There are 7 field preserve for these name. If you only inputted less than 7 fields, just press the <Esc> key to bypass the rest prompt. Then you will be asking for confirmation to save the file.



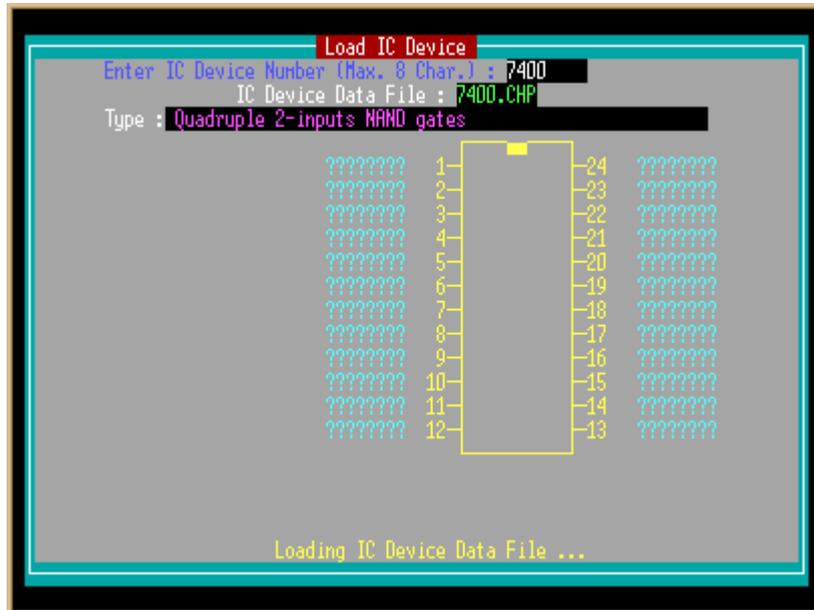
The new IC library file will be created, just so simple, isn't it?

III.2. LOAD EXISTING IC LIBRARY

This option used for every times you want to check another type of IC and different from an existing IC library which reside in the buffer memory. No needs to load the same type of IC library with already exist in buffer memory, just run the test sequence. When you choose this option, you will be asked for IC library name. No need to add an extension, because the program will add automatically (the library extension is *.CHP and will be get from \LIBRARY subdirectory).



At this prompt you can see the IC library file that already create or if you don't remember the IC library file name which you want to check, put a wildcard like *.CHP and press enter. The entire existing IC library file that match to this criteria will be displayed. After the file name inputted, you would be back to the main menu.



III.3. EDIT EXISTING IC LIBRARY

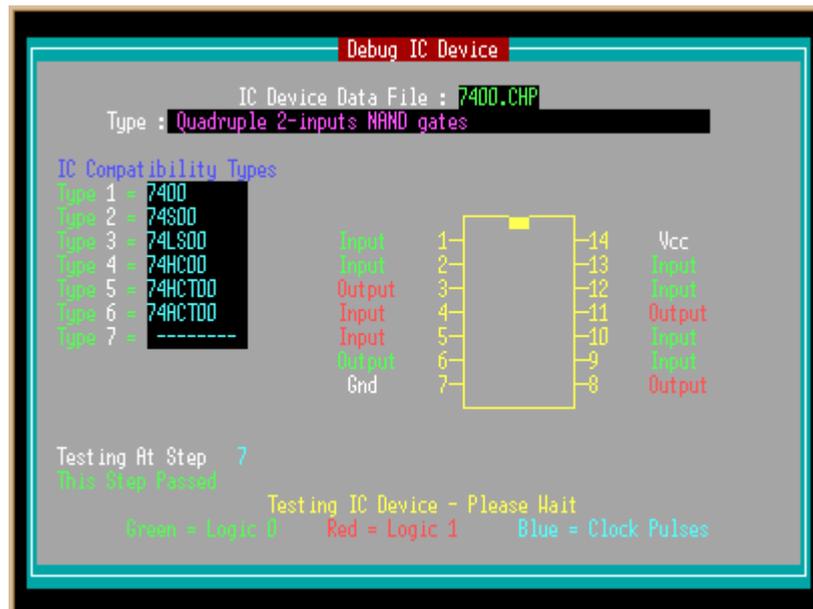
This option used to modify the existing IC library that already resided in the buffer memory. When you choose this option, you will be brought to the editor screen. From here you can modify or delete any step that do not run well. The rest of the process is the same as when you create a new one.

IC Data File For Type : 7400 Pin Q'ty : 14													
Quadruple 2-inputs NAND gates													
STEP	INPUTS								OUTPUTS				
	1	2	4	5	9	10	12	13	3	6	8	11	
1	0	0	0	0	0	0	0	0	1	1	1	1	
2	0	1	0	0	0	0	0	0	1	1	1	1	
3	1	0	0	0	0	0	0	0	1	1	1	1	
4	1	1	0	0	0	0	0	0	0	1	1	1	
5	0	0	0	1	0	0	0	0	1	1	1	1	
6	0	0	1	0	0	0	0	0	1	1	1	1	
7	0	0	1	1	0	0	0	0	1	0	1	1	
8	0	0	0	0	0	1	0	0	1	1	1	1	
9	0	0	0	0	1	0	0	0	1	1	1	1	
10	0	0	0	0	1	1	0	0	1	1	0	1	
11	0	0	0	0	0	0	0	1	1	1	1	1	
12	0	0	0	0	0	0	1	0	1	1	1	1	
13	0	0	0	0	0	0	1	1	1	1	1	0	
14	1	1	1	1	1	1	1	1	0	0	0	0	
15	1	0	1	1	1	1	1	1	1	0	0	0	
16	0	1	1	1	1	1	1	1	1	0	0	0	
17	0	0	1	1	1	1	1	1	1	0	0	0	
18	1	1	1	0	1	1	1	1	0	1	0	0	
19	1	1	0	1	1	1	1	1	0	1	0	0	

<D>/<P>-next/pre line,-delete line,<←>/<Shft><Tab>/<Bk><Space>-pre entry,

III.4. DEBUGGING TEST IC DEVICE

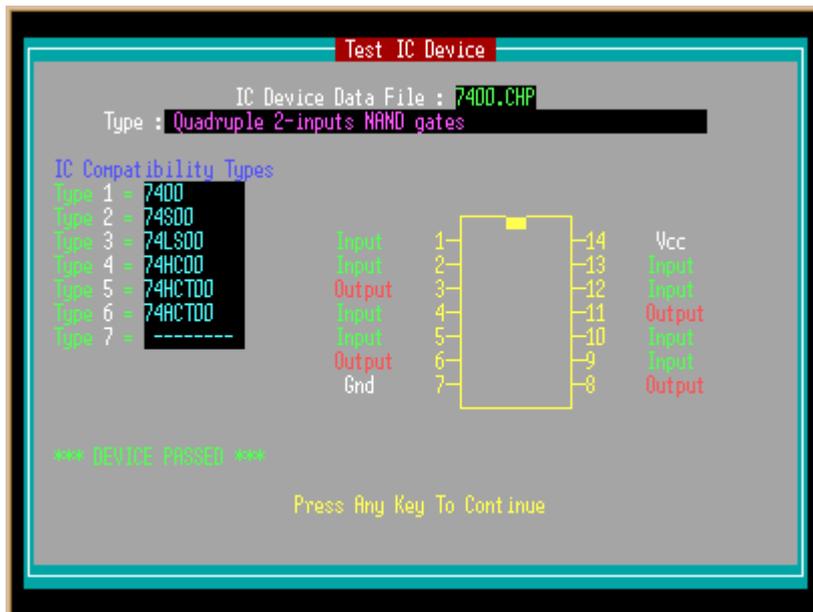
This option used to test the existing IC library that already resided in the buffer memory. Usually you go to this sub menu then change the state to edit mode and then run debugging again, a few times until you get the right step sequence. When run this option, you need a hardware connected to the PC, otherwise a message will occur, and you cannot continue the rest of it. Put a new IC that is the same type as the IC library type. If everything is okay, the sequence will perform.



When the wrong step detected, it will display the message 'This Step Failed' with the red color, if the step is right, the message 'This Step Passed' will display with green color. After the last step, it will display the conclusion for that IC whether the IC is a good one or a bad one.

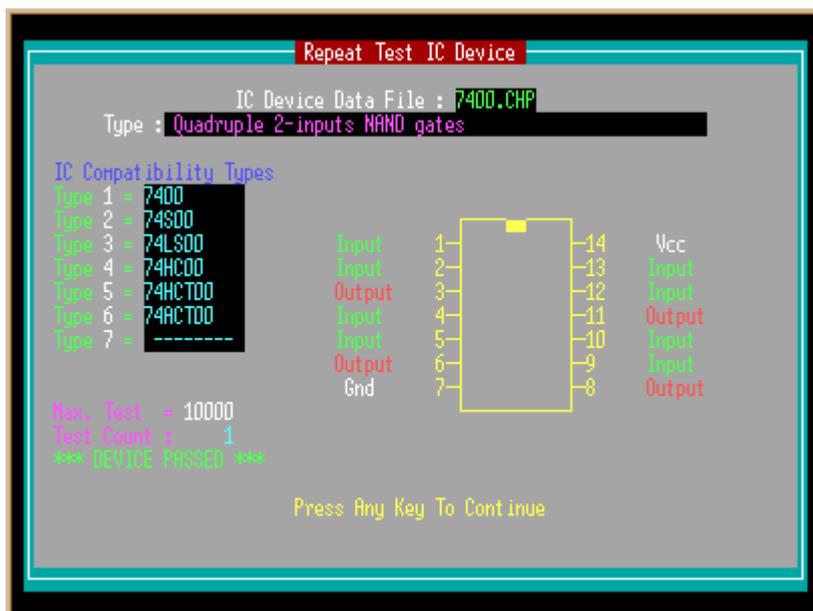
III.5. RUN TEST IC DEVICE

This option is the same as with sub menu debugging test IC device. The different only for this option is the sequence perform faster, and the performance of each step does not displayed, only the result. The rests are the same. This option also needs a hardware connected to the PC.



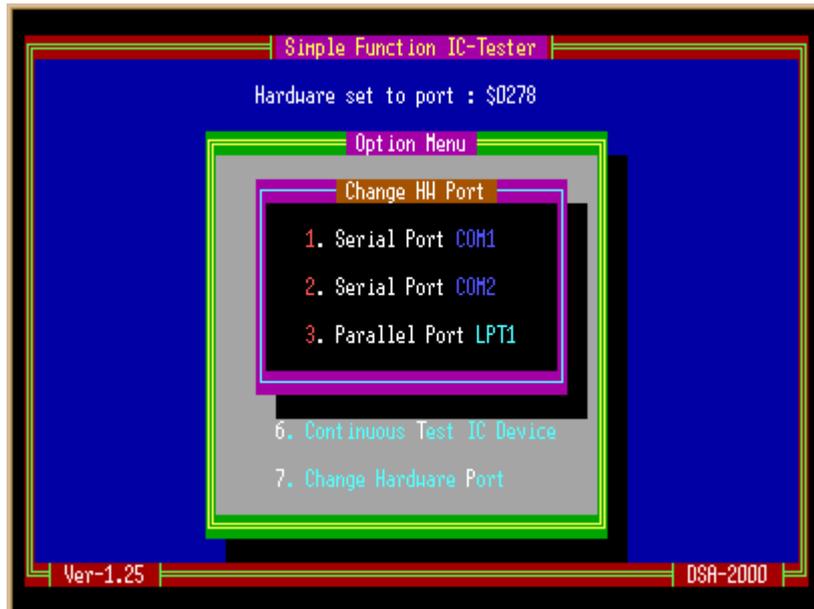
III.6. CONTINUOUS TEST IC DEVICE

This option is the same as with sub menu run test IC device. The different only the sequence test will be repeated for 10000 times. These need when the IC does not show the broken characteristic immediately. The process must repeatedly until the bad step occur, and the process will be stopped at the repeated step. If the last step was reached, the IC is a good one. The loop count unadjustable, so if you just want to check the loop for only 5 times or maybe 10 times, just press the <Esc> key after the repeated step passed away. This option also needs a hardware connected to the PC.



III.7. CHANGE HARDWARE PORT

This option used to select the port where the device was connected. Any port can be select are COM1 and COM2 for serial communication ports, and any LPT1 through LPT3 for LPT printer parallel ports. COM port used for hardware designed by Paul Stenning and LPT port used for hardware designed by me. For LPT port, it is automatically scanned which one where the hardware connected. When the hardware not detect, a message will be appeared, and the default port will be set again.



----- End Of File – ICTester Help File for Ver 1.25 – April 5, 2001 – DSA -----