INTERNATIONAL FIRST PRIZE

1997 International Microprocessor Competition

This article supplies a condensed description of a stand-alone IC tester for SSI (small-scale integration) logic ICs (with up to 24 pins) from the well-known 74xx and 40xx series. The elementary building blocks that make up the design are an 80C535 microcontroller, a large EPROM, an LCD display, a small keyboard and an RS232 interface. The latter allows the tester to receive new IC test vectors produced by a test vector compiler which runs on a PC. Another DOS program allows new test vectors to be tested using the same RS232 interface, so you don't have to re-program the system EPROM or dig up an EPROM emulator.

software available on cd-rom
µP–µC hard & software '97–'98

By Laurent Lamesch (Luxembourg)

IC tester

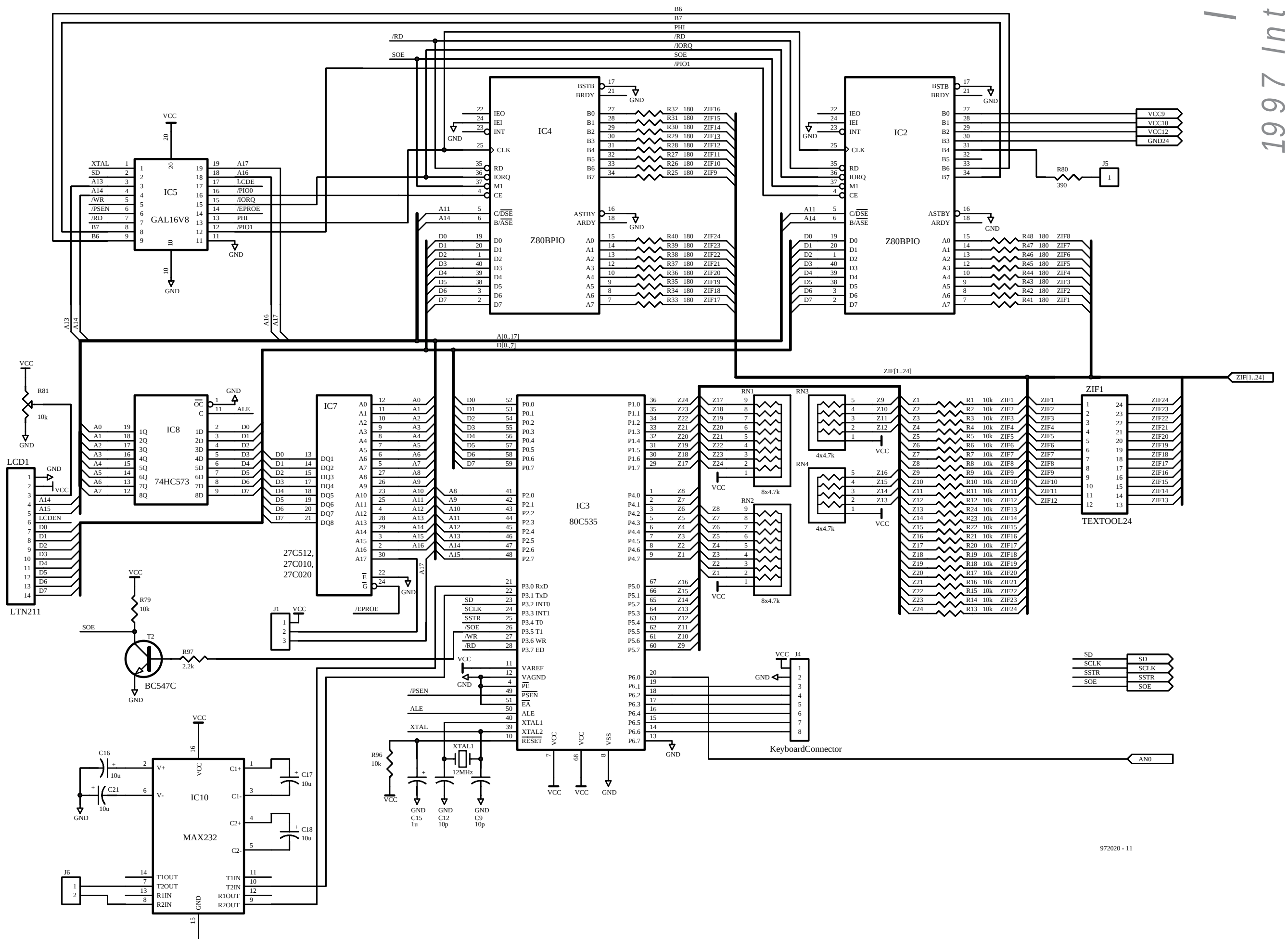


Figure 1. Circuit diagram of the digital control circuit and the RS232 interface.

All input pins of the device under test (DUT) inserted in the Textool zero-insertion force socket, ZIF1, may be pulled to the logic high (H) or logic low (L) level by means of the outputs on the Z80PIOs and current limiting resistors. The same PIOs also enable the logic states of the DUT outputs to be checked, while ports 1, 4 and 5 of the 80C535 are used to detect which DUT pins represent a high impedance. Furthermore, the supply voltage pins of the DUT may be con-

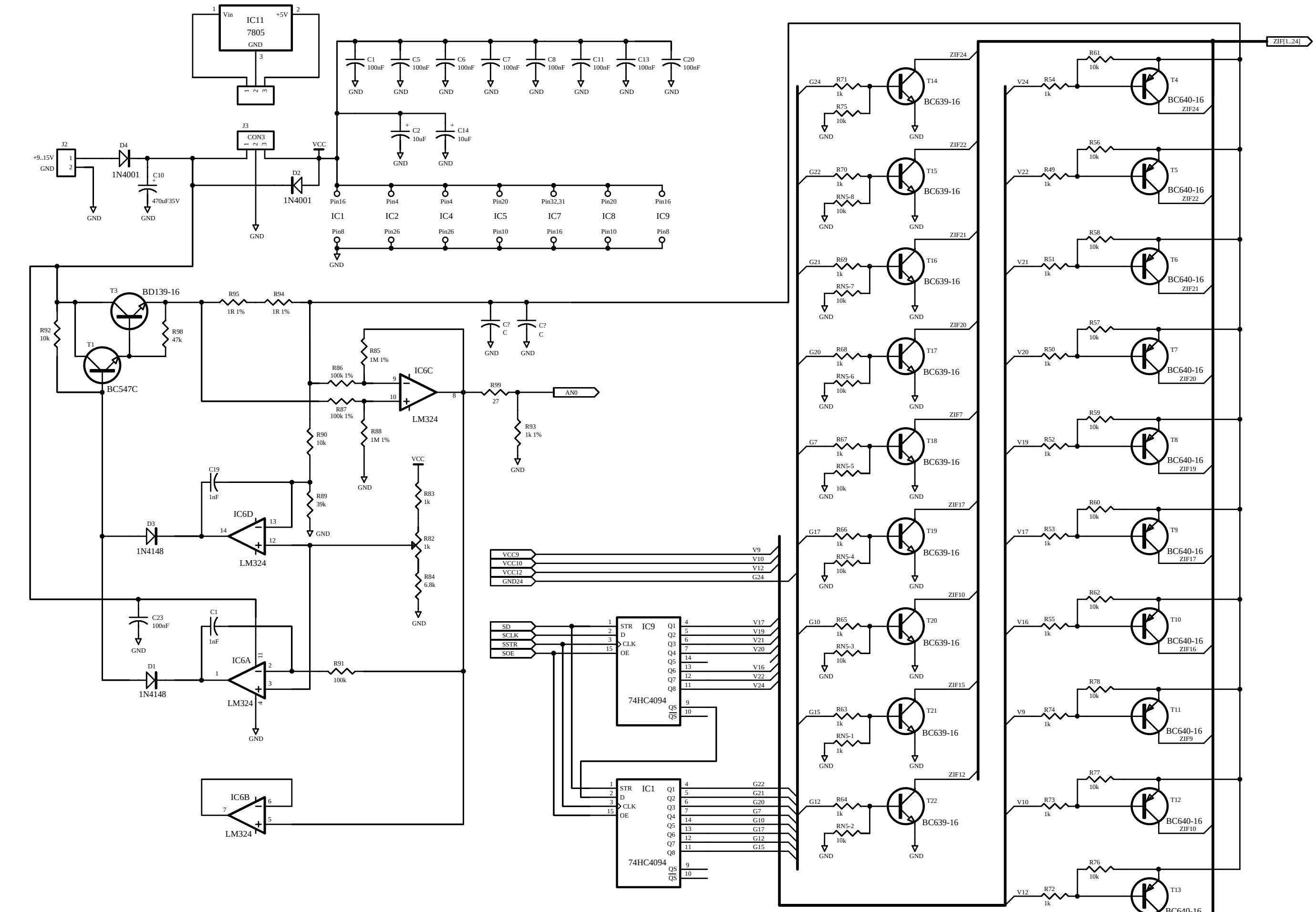


Figure 2. Circuit diagram showing the DUT interface and the power supplies.

nected to GND or a current-limited voltage source via switching transistors.

This voltage source (built around IC6) supplies 5.2 V, and its output current is limited to about 0.2 A. The actual output current is converted into a proportional voltage for measurement by the controller.

Depending on the size of the test vector files, the control program of the IC tester is contained in a 27C512, 27C010 or 27C020 EPROM. If multiple banks are present inside the EPROM, then the selection between the 64-kByte chunks is accomplished by outputs B6 and B7 of IC2. The 80C535 also controls an LC display and a keyboard with 6 keys. A GAL, IC5, looks after the address decoding, and also generates the PHI signal for the Z80PIOs.

The reason for using the Z80PIO to control and monitor the DUT inputs and outputs is that this chip is the only widely available 16-bit parallel port IC of which all port line directions are individually controllable, while the output drivers for all port lines consist of push-pull circuits.

Jumper J1 selects between a 28-pin and a 32-pin EPROM in position IC7. When a 28-pin EPROM is used, J1-1 is connected to J1-2. When a 32-pin EPROM is used, connect J1-2 to J1-3.

A 9-way sub-D socket is connected to J6. This enables you to connect the IC tester to the RS232 serial port on your PC. Ground for the serial interface is taken from J2. The pin connections of the sub-D socket are as follows:

9-way sub-D	J6	J2
2	1	
3	2	
5		2

Adjustment

The only adjustment in the circuit is the DUT supply output voltage. This is set to 5.2 V ±0.05 V using preset R82.

Operation

The tester is operated using six keys labelled Enter, Escape, dn (scroll down), up, dn2 (fast scroll down), and up2 (fast scroll up). The up and dn keys have an auto-repeat function which causes the repeat rate to be automatically increased as the key is held depressed.

LED D5 lights to indicate that the IC under test is being powered, and should not be removed from the ZIF socket.

Pressing the escape key takes you to the main menu. There, the following functions may be selected:

1. **Test IC:** the user picks an IC from an IC library, and the DUT is checked for correct operation. The test may be repeated. If indicated by the test vectors, the current consumption of the IC under test is measured and displayed.
2. **Identify:** this allows you to identify the type number of an unknown IC. If the GND and Vcc pins are unknown, only those test vectors are used that have the GND and Vcc pins at the same positions. The GND/Vcc pin entry is optional. Next, you can select the libraries that have to be scanned.
3. **Retest IC:** once an IC has been tested or identified, it may be tested again without having to pick it from the libraries.
4. **Trace:** all test vectors and the response of the DUT to these vectors appear in succession on the LC display.
5. **Options:** here, you can define global options.
6. **Info:** information on version and copyright.
7. **Self Check:** the IC tester hardware

may be checked using this function and a voltmeter.

8. **Remote Mode:** connect a PC to the RS232 interface and debug test vectors using the DOS program TVCHK.EXE.

The up/dn keys are used to scroll one item up or down. The up2/dn2 keys do the same, but then five items at a time. The ent(er) key is used to confirm a selection. Esc, finally, jumps to the main menu.

Test vector compiler and debugger

ICVC.EXE is the test vector compiler which generates a Test Vector Binary File from the Test Vector Source File. The Binary File has to be appended to microcontroller program, ICT.BIN, before an EPROM may be programmed for the IC tester. File appending is achieved with the simple 'copy /b ...' command.

ICTVC.EXE is launched by typing ICTVC scrfile.TVC at the DOS prompt. So, for instance, ICTVC VECT.TVC. It generates the following files:
TVC.OUT: Test Vector Binary File
ERR.OUT: Error report
LIST.OUT: List file containing information on the source file, the binary file, a copy of the source file with line numbers, plus, for each line, the bytes that were generated from this line.
TMP.OUT: temporary file, used by ICTVC.EXE.
If an error occurs during the compilation phase, the relevant error reports are written to ERR.OUT. Error reports are not displayed on the PC monitor.

TVCHK is a shell program which calls ICTVC.EXE, and enables the generated test vectors to be debugged. This program has to be launched using the COM port to which the IC tester is connected, or the address and interrupt line of this COM port, as an appended switch. So, for instance, TVCHK 2.

The batch file **EDT.BAT** is used to launch an external text editor (word processor). If another editor than EDIT.COM is used, this batch file has to be modified accordingly. The batch file launches the editor and conveys the first batch file parameters. The Test Vector Source File is relatively large, so your external editor may not be able to load it entirely. If that happens, you should create a file which only contains the new test vectors (if new test vectors are being debugged). Once the new vectors have been debugged, they may be appended to VECT.TVC without using the TVCHK program. the option 'Compile tv source code' is then used to compile the complete file.

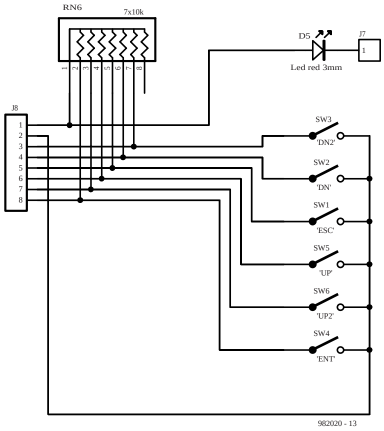


Figure 3. The keyboard circuit.

A file called **SMALL.TVC** is supplied for practicing. Select option 2 (Edit & compile tv source file) in the TVCHK program and then enter SMALL.TVC instead of VECT.TVC.

Building this project

As already mentioned, this article presents a condensed description of a

Index of IC test vectors in VECT.TVC

Library: 74xxx	74:35*	74:133*	74:191
74:00	74:37	74:136*	74:192
74:01*	74:38	74:137	74:193
74:02	74:39*	74:138	74:194
74:03	74:40	74:139	74:237*
74:04	74:42	74:140*	74:238*
74:05	74:45	74:147	74:239
74:06	74:46*	74:145*	74:240
Parent: 74:05	74:47	74:148	74:241
74:07	74:48*	74:150	74:242*
74:08	74:49*	74:151	74:243
74:09	74:51 St,S*	74:153*	74:244
74:10	74:51 LS,L	74:154	74:245
74:11	74:54*	74:155	74:247*
74:12*	74:55*	74:156*	74:248*
74:13	74:73	74:157	74:249*
74:14	74:74	74:158*	74:250*
74:15*	74:75	74:159*	74:251
74:16	74:76	74:160	74:253
74:17	74:83	74:161	74:257
74:18*	74:86 -C,-L	74:162	74:258*
74:19*	74:86 C,L*	74:163	74:259
74:20	74:90	74:164	74:260*
74:21	74:92	74:165	74:266
74:22*	74:93	74:166	74:273
74:24*	74:95A,B	74:168*	74:280
74:25	74:100	74:169	74:283
74:26	74:107	74:170	74:290*
74:27	74:109	74:173	74:293
74:28*	74:119	74:174	74:299
74:30	74:125	74:175	74:323
74:31*	74:126*	74:180*	74:347*
74:32	74:128*	74:184*	74:348*
74:33	74:131*	74:185*	74:352*
74:34*	74:132	74:190	74:353*

74:365	74:595	74:811*	4019	4082*
74:366*	74:596*	74:1000*	4020	4093
74:367	74:620*	74:1002*	4021	4094
74:368*	74:621*	74:1003*	4022*	4099*
74:373	74:622*	74:1004*	4023	40014*
74:374	74:623*	74:1008*	4024	40102
74:375	74:638*	74:1010*	4025	40103
74:377*	74:639*	74:1020*	4027	40105
74:386*	74:640	74:1032*	4028	40106
74:390	74:641*	74:1034*	4029	40160
74:393	74:642*	74:1035*	4030	40161
74:412	74:643*	74:1036*	4040	40162
74:425*	74:644*	74:1240*	4041*	40163
74:426*	74:645	74:1244*	4042	40174
74:445*	74:646	74:1245*	4043	40175
74:447*	74:647*	74:1640*	4044*	40192
74:465*	74:648*	74:1645*	4049**	40193
74:466*	74:649*	74:2240*	4050**	40194
74:467*	74:668*	74:2241*	4051	4502
74:468*	74:669*	74:2244*	4052	4508
74:518*	74:670	74:2540*	4053	4510
74:519*	74:682	74:2541*	4056	4511
74:520*	74:683*	74:7245*	4060	4512
74:521	74:684	74:7266*	4066	4514
74:522*	74:685*	74:7540*	4067	4515*
74:533*	74:688	74:7541*	4068 -RCA*	4516*
74:534*	74:689*	**Library: 40xxx**	4068 RCA*	4518
74:540*	74:699	4001	4069	4520
74:541	74:746**	4002	4070	4522
74:563*	74:747**	4009*	4071	4526
74:564*	74:756*	4010*	4072	4529
74:573	74:757*	4011	4073	4539
74:574	74:758	4012	4075	4543*
74:576*	74:759*	4013	4076	4555
74:580*	74:760*	4014*	4077	4556*
74:590	74:762*	4015	4078 -RCA*	4584
74:591*	74:763*	4016	4078 RCA*	4724
74:592	74:810*	4017	4081	

* *Test vector for this IC not yet verified with a correctly operating IC.*
** *This IC has not been fully tested using the IC test vectors.*
: *Any TTL family identifier, except if the type number has a suffix.*
-X *Test vectors not valid for X family device of this IC (e.g. 74:86 -C means not for 74C86).*
X *Test vectors apply only to X family device of this IC (e.g. 74:86 C means for 74C86 only).*

fairly large project which consists of hardware, software and extensive documentation. Unfortunately, the overall size of the project as submitted by the author is such that it is not possible to reproduce it in full in this supplement. After all, we want to show you some of the other prize-winning entries, too!

The IC tester project will be described as a project for home construction in a future issue of *Elektor Electronics*, complete with a printed circuit board, a ready-programmed EPROM and tested project software on disk.

Meanwhile, the design (as we received it) will be included on a Contest Compilation CD-ROM which we hope to publish by the end of January 1998.

(982020-1)

Development tools used

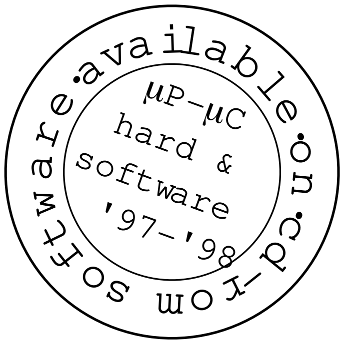
IC Tester firmware:	Metalink 8051 Cross Assembler V1.2h Signetics HEX2BIN.EXE
PC programs:	Borland Turbo C++ V3.0
Schematic layout[1]:	Protel Advanced Schematic V3.5
PCB layout[2]:	Easytrax V2.06
GAL assembler:	c't Galio

[1] cosmetic changes by Elektor
[2] layouts not reproduced in this article

FIRST NATIONAL PRIZE UK

1997 International Microprocessor Competition

PIC on the Rocks is a system for PIC users that allows them to do in-circuit debugging of PIC programs. It uses a simple serial connection with a host PC running Windows 95 or NT. The PIC program the user wants to check also needs to embed a small piece of telemetry code. This code enables the PIC side of the telemetry link. The project software includes source code for both the PIC and the host PC. It also comprises a manual in Word format and some examples.



By Ben de Waal (USA)

PIC on the Rocks

The PIC micro controller range from Microchip has become one of the leading micro controllers used by engineers and hobbyists to solve simple and medium complexity embedded problems. Personally I have used PICs in pagers, Virtual Reality peripherals, home security systems and even hamster treadmill measurements.

One of the major advantages of PICs is that they are relatively inexpensive. The basic development tools are also fairly cheap and this makes it easy for the hobbyist (*Elektor Electronics* reader) to develop software for it. The major drawback is that PIC debugging tools and In Circuit Emulators (ICE) are very expensive. This creates the common scenario that average Joe Hobbyist has the development tools but also has to burn and erase endless number of times to get the solution 'just right'. Most of the time he tries to figure out 'what goes on' inside the PIC since he cannot afford or justify the cost of an ICE.

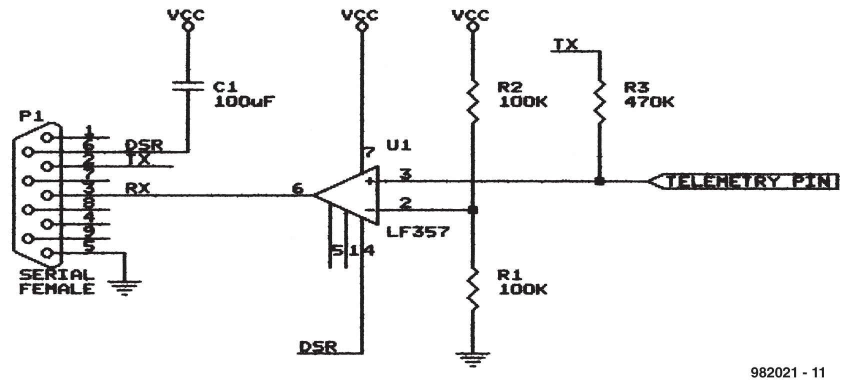


Figure 1. The opamp solution. Note: depending on the serial cable, you may have to swap pins 2 and 3 as well as pins 4 & 6. DSR should be an output.

This project addresses the PIC debugging issue by providing an almost non-intrusive solution for PIC telemetry (remote debugging). It allows you to pause PIC execution, look at the internal registers (files), edit them and continue till the next 'break point', etc.

IC tester

Part 1: circuit description and construction

This article describes a stand-alone IC tester for logic ICs (with up to 24 pins) from the well-known 74xx (TTL) and 40xx (CMOS) series. The elementary building blocks that make up the design are an 80C535 microcontroller, a large EPROM, an LCD display, a small keyboard and an RS232 interface. In this first instalment, our main subjects will be hardware-related.

This design was awarded the International First prize in the Elektor Electronics 1997 Design Competition. A condensed description of the project appeared in the January 1998 Supplement on prize-winning contest entries.

Design by L. Lamesch

In the January 1998 issue we promised a full-blown and tested version of the IC Tester, complete with PCB layouts and a ready-programmed GAL and EPROM. Many of you will have eagerly looked forward to the present article, which, we hope, lives up to expectation.

The IC tester is a fairly complex project, and an interesting combination of mixed-mode (analogue/digital) electronics on the one hand, and software (microprocessor as well as PC-hosted), on the other. The IC tester is capable of working in stand-alone mode, i.e., any connection to a PC is entirely optional. The standard version of the tester is capable of testing all ICs listed in **Table 1**. A number of software utilities are available which allow advanced users to add ICs to the library as listed in Table 1, and this subject will be covered in detail in part 2 of this article. We do, however, reckon that the 'default' library will be sufficient in the majority of cases — extending it with 'new' devices is specialist work!

This tester is a great tool for anyone who's ever had to do go/non-go testing on a vast amount of integrated circuits from the '74' TTL and '4000' CMOS series. These ICs are still extensively used by hobbyists because (1) they can often be salvaged from surplus equipment or bought 'ten a penny' or at least at knock-down prices, and (2) their datasheets are widely available, so you know how to use them in a design of your own. As a matter of course, the tester is also perfectly suitable for use in an electronics repair shop, where it will be necessary from time to time to check suspect devices. Lastly, the IC tester may also be used to identify unknown ICs, i.e., those with obscure print or a label someone painstakingly ground off…

How it works

The circuit diagram of the IC tester is shown in **Figure 1**. It may be divided into a digital control (microcontroller) section, a precision power supply, an RS232 interface, a zero-insertion-force (ZIF) socket (which receives the device under test, DUT), a keyboard section, an LCD and a power supply.

Around the DUT

Let's start with a look at the electronics around the ZIF socket, position IC12. All input pins of the DUT inserted in the zero-insertion force socket may be pulled to the logic high (H) or logic low (L) level using current limiting resistors and appropriate control levels at the outputs of two Z80PIO I/O blocks, IC2 and IC4. The output states of the DUT may be interrogated via the same PIOs for subsequent evaluation by the CPU (IC3). The power supply pins of DUTs with 14, 16, 18, 20, 22 and 24 pins may be connected to ground or a current-limited supply voltage via BC639/BC640 switching transistors. These are controlled by outputs on counter cascade IC1-IC9. While the PIOs enable the exact logic states of the DUT outputs to be checked (i.e., 0 or 1), ports P1, P4 and P5 of the 80C535 are used to detect which DUT pins represent a high impedance (high-Z).

DUT power supply

The voltage source used to power the DUT is built around quad opamp IC6. It supplies an accurately regulated voltage of 5.2 V, and its output current is limited to about 0.2 A. The output current is converted into a proportional voltage for measurement by the 80C535 CPU via its AN0 (analogue) input. The current-sense resistors are R94 and R95. The output voltage is 5.2 V rather than 5.0 V (the typical supply voltage of all TTL ICs) in order to compensate the collector-emitter drop of the BC640 transistors when they are switched on.

Logic circuitry and firmware EPROM

The control program of the IC tester and the test vectors for the ICs that may be tested are contained in a single 27C512 EPROM, which may be obtained ready-programmed from the Publishers under order code **986507-1**. Larger (32-pin) EPROMs like the 27C020 or 27C021 may also be used in this design. For the 27C021 EPROM, jumper JP1 has to be set to the A17 position. More about this in next month's concluding instalment. The selection between the 64-kByte banks in the EPROM is accomplished by outputs B6 and B7 of IC2. These lines are applied to GAL IC5, which contains logic to control the A16 and A17 address lines of EPROM IC7. Basically, when a 28-pin EPROM is used, JP1 is set to the bevelled edge position ('A'). When a 32-pin EPROM is used, the A17 signal is required, so JP1 is set to the other position.

The 80C535 CPU runs at a clock speed of 12 MHz. The CPU does not contain firmware code, and fetches all of its instructions and data from the system EPROM. None the less, it directly controls the LC display and a keyboard with 6 keys. The GAL, IC5, looks after the address decoding, and also generates the PHI signal for the Z80PIOs, as well as other essential control signals in the circuit. The GAL, like the EPROM, is supplied ready-programmed by the Publishers, the order code is **986506-1**.

The reason for using the Z80PIO to control and monitor the DUT inputs and outputs is that this chip is the only widely available 16-bit parallel port IC of which all port line directions are individually controllable, while the output drivers for all port lines consist of push-pull circuits.

User I/O

The system interacts with the user via a small keyboard (circuit diagram in Figure 2), an LED, D6, and an LCD (liquid crystal display). The LCD is a general-purpose type with 2⇔16 characters, optionally with back-lighting. Its contrast is adjustable with preset P1. The LED lights to inform the user that the DUT is being powered and should not be removed from the ZIF socket.

An RS232 serial interface, traditionally designed around the MAX232, enables the IC tester to (optionally)

Table 1. Index of ICs that may be tested (default EPROM contents)

Library: 74xxx	74:38	74:139	74:239	74:386*	74:596*	74:758	74:7541*	4050**	40161
74:00	74:39*	74:140*	74:240	74:390	74:620*	74:759*		4051	40162
74:01*	74:40	74:147	74:241	74:393	74:621*	74:760*	**Library: 40xxx**	4052	40163
74:02	74:42	74:145*	74:242*	74:412	74:622*	74:762*	4001	4053	40174
74:03	74:45	74:148	74:243	74:425*	74:623*	74:763*	4002	4056	40175
74:04	74:46*	74:150	74:244	74:426*	74:638*	74:810*	4009*	4060	40192
74:05	74:47	74:151	74:245	74:445*	74:639*	74:811*	4010*	4066	40193
74:06	74:48*	74:153*	74:247*	74:447*	74:640	74:1000*	4011	4067	40194
Parent: 74:05	74:49*	74:154	74:248*	74:465*	74:641*	74:1002*	4012	4068 -RCA*	4502
74:07	74:51 St,S*	74:155	74:249*	74:466*	74:642*	74:1003*	4013	4068 RCA*	4508
74:08	74:51 LS,L	74:156*	74:250*	74:467*	74:643*	74:1004*	4014*	4069	4510
74:09	74:54*	74:157	74:251	74:468*	74:644*	74:1008*	4015	4070	4511
74:10	74:55*	74:158*	74:253	74:518*	74:645	74:1010*	4016	4071	4512
74:11	74:73	74:159*	74:257	74:519*	74:646	74:1020*	4017	4072	4514
74:12*	74:74	74:160	74:258*	74:520*	74:647*	74:1032*	4019	4073	4515*
74:13	74:75	74:161	74:259	74:521	74:648*	74:1034*	4020	4075	4516*
74:14	74:76	74:162	74:260*	74:522*	74:649*	74:1035*	4021	4076	4518
74:15*	74:83	74:163	74:266	74:533*	74:668*	74:1036*	4022*	4077	4520
74:16	74:86 -C,-L	74:164	74:273	74:534*	74:669*	74:1240*	4023	4078 -RCA*	4522
74:17	74:86 C,L*	74:165	74:280	74:540*	74:670	74:1244*	4024	4078 RCA*	4526
74:18*	74:90	74:166	74:283	74:541	74:682	74:1245*	4025	4081	4529
74:19*	74:92	74:168*	74:290*	74:563*	74:683*	74:1640*	4027	4082*	4539
74:20	74:93	74:169	74:293	74:564*	74:684	74:1645*	4028	4093	4543*
74:21	74:95A,B	74:170	74:299	74:573	74:685*	74:2240*	4029	4094	4555
74:22*	74:100	74:173	74:323	74:574	74:688	74:2241*	4030	4099*	4556*
74:24*	74:107	74:174	74:347*	74:576*	74:689*	74:2244*	4040	40014*	4584
74:25	74:109	74:175	74:348*	74:580*	74:699	74:2540*	4041*	40102	4724
74:26	74:119	74:180*	74:352*	74:590	74:746**	74:2541*	4042	40103	
74:27	74:125	74:184*	74:353*	74:591*	74:747**	74:7245*	4043	40105	
74:28*	74:126*	74:185*	74:365	74:592	74:756*	74:7266*	4044*	40106	
74:30	74:128*	74:190	74:366*	74:595	74:757*	74:7540*	4049**	40160	
74:31*	74:131*	74:191	74:367						
74:32	74:132	74:192	74:368*						
74:33	74:133*	74:193	74:373						
74:34*	74:136*	74:194	74:374						
74:35*	74:137	74:237*	74:375						
74:37	74:138	74:238*	74:377*						

*	Test vector for this IC not yet verified with a correctly operating IC.
**	This IC has not been fully tested using the IC test vectors.
:	Any TTL family identifier, except if the type number has a suffix.
-X	Test vectors not valid for X family device of this IC (e.g. 74:86 -C means not for 74C86).
X	Test vectors apply only to X family device of this IC (e.g. 74:86 C means for 74C86 only).

1

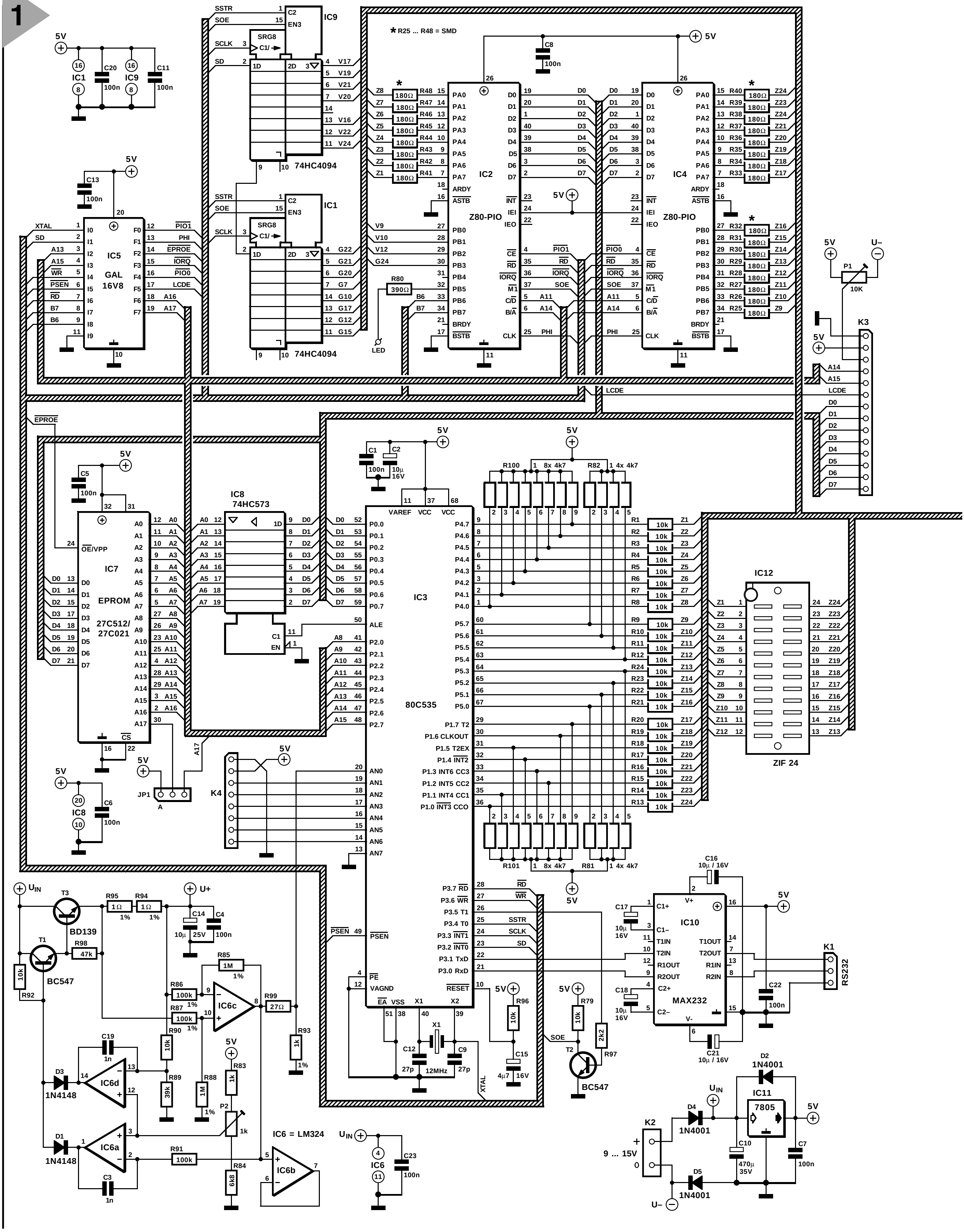
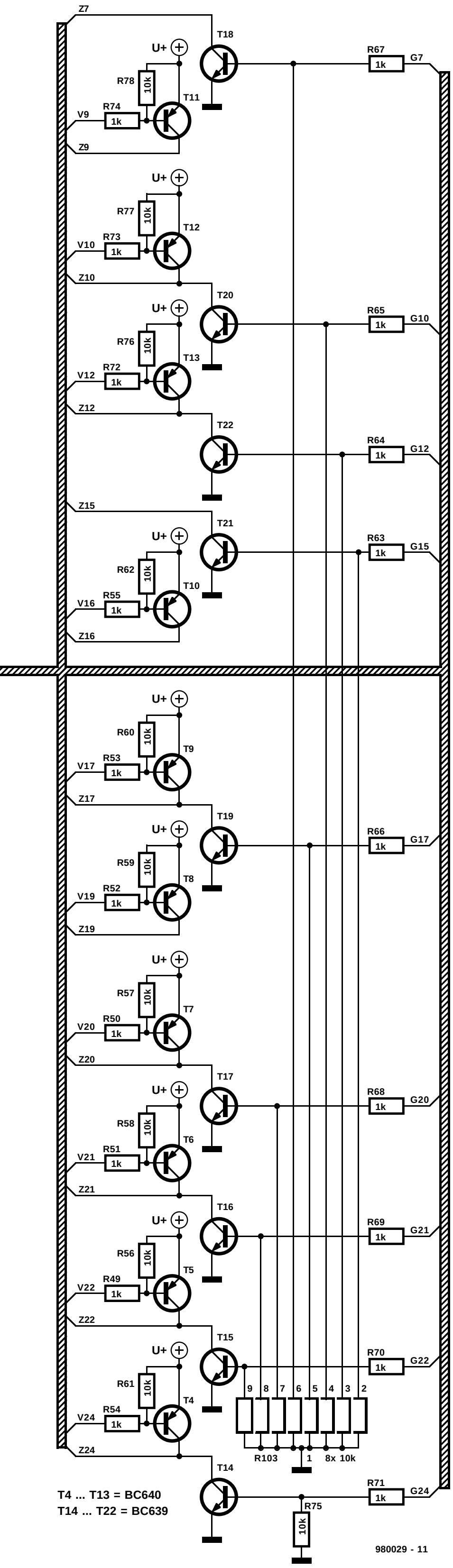


Figure 1. Circuit diagram of the IC tester, an interesting mixed-mode design based in the powerful 80C535 microcontroller from Siemens.

communicate with a PC running specially written programs. Details on these programs will be given in next month's final instalment. Briefly, the PC may be used to write your own IC test vectors and debug them without removing the system EPROM. The link between the 9-way sub-D socket and the MAX232 on the board is accomplished via pinheader K1 as



illustrated in **Figure 2**. Note that no active handshaking is used; the IC tester only uses RxD (received data), GND (ground) and TxD (transmitted data).

Power supply
The on-board 5-volt power supply for the IC tester is based on a single 7805 3-pin voltage regulator (IC11). A small negative voltage is created mainly for older LCDs by a stealer diode (D5) between the negative line of the unregulated input voltage and the circuit ground. This negative voltage provides a proper range for the LCD contrast setting, P1.

The circuit may be powered by an inexpensive mains adaptor with an output voltage between 9 VDC and 15 VDC. Battery powering is also possible. The unregulated voltage is also applied to the DUT supply discussed above. The current consumption of the circuit will be of the order of 150 mA.

Construction

As you can see from the photographs on this article and the artwork in **Figure 3**, the printed circuit board designed for the IC tester is densely populated. The board being double-sided and through-plated, production with hobbyists tools will be almost impossible, hence our recommendation to buy it ready-made through our Readers Services or one of the kit suppliers advertising in this magazine. Remember, successful construction almost entirely depends on *accuracy and soldering skills*.

Start by separating the keyboard from the main board. Put the keyboard section aside for later.

Populating the main board may take quite some time as there are relatively many components to sort and solder in place.

Resistors, diodes, capacitors, crystal
Start by fitting the SMD (surface-mount device) resistors, R25-R48. This should be done with a low-power (8-watt) soldering iron and lots of care and precision. Use an ohm-meter to check your work on each and every SMD resistor. All other resistors are mounted upright to save space on the board. Make sure you know the value of each and every resistor and capacitor before mounting it. If necessary, use your DMM and the component overlay printed on the board (and shown in Figure 3) to be absolutely sure. Also, the SIL resistor arrays on the board must be fitted the right way around, so make sure you know where the 'common' terminal goes. Likewise, observe the orientation of the electrolytic capacitors and diodes. Capacitors C1 (100 nF miniature ceramic) and C2 (10μF, 16V, also miniature) are mounted **at the solder side of the board**. The two presets are fitted and set to mid-travel.

Semiconductors
Fitting the transistors and the voltage regulator should not present problems,

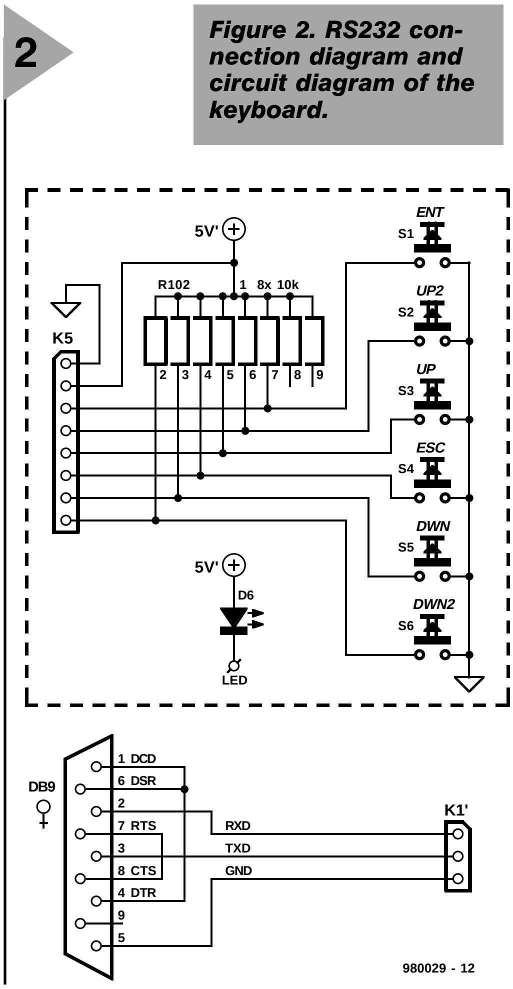


Figure 2. RS232 connection diagram and circuit diagram of the keyboard.

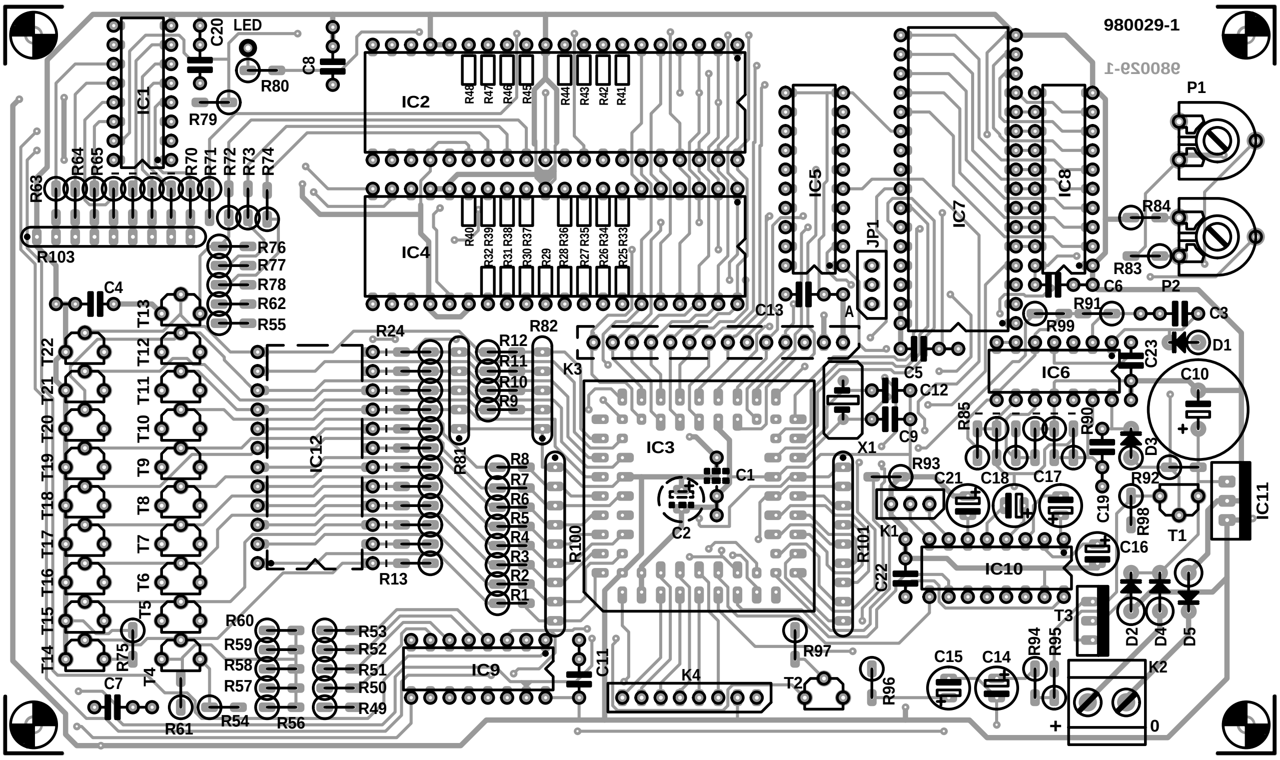
although you have to be careful not to mix up the BC639s and the BC640s. All ICs are mounted in sockets. With the possibility of future extensions in mind, it is recommended to fit a 32-pin IC socket in position IC7. However, as 32-pin wide-DIL are few and far between, you may have to make one yourself by truncating a 40-pin socket. If you use the ready-programmed (28-pin) 27C512 EPROM supplied through the Readers Services, its pin 14 should go in socket pin 16. In other words, the EPROM is then inserted with its corner pins 14/15 close to the edge of the board.

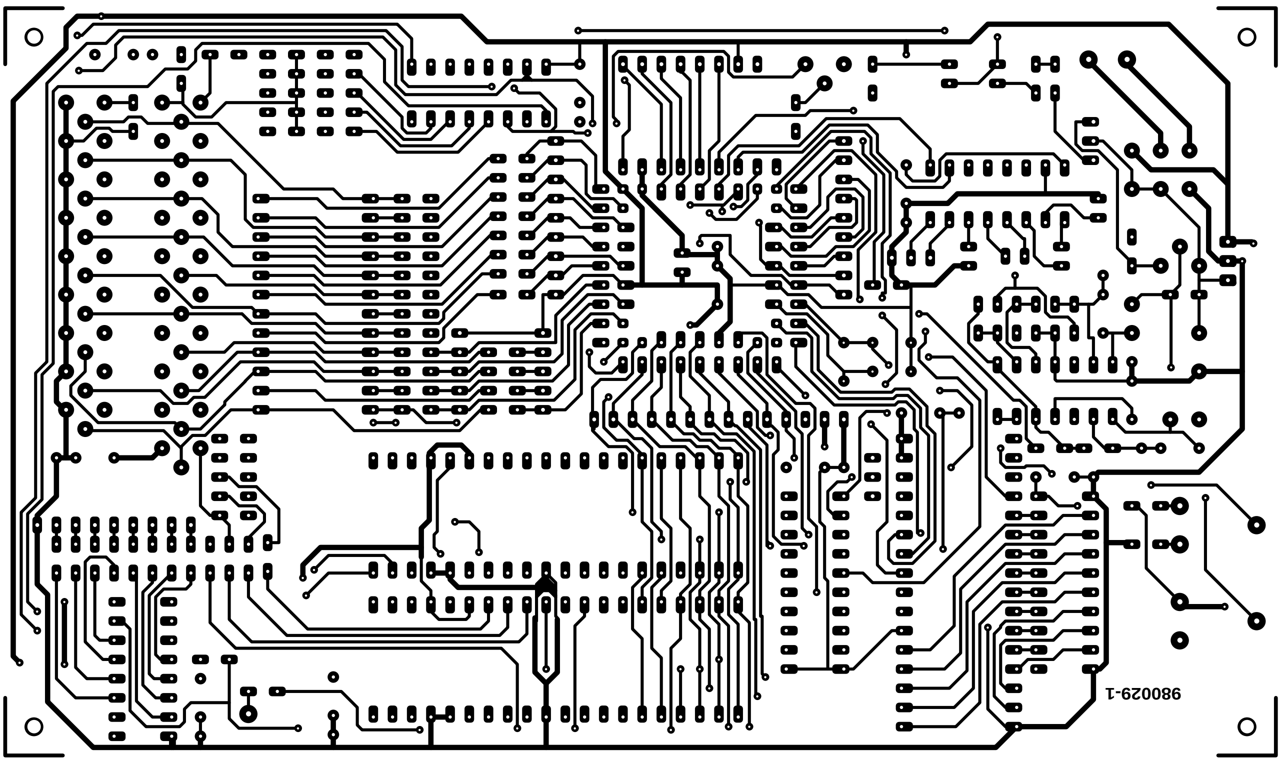
The CPU socket has a bevelled edge which is also indicated on the overlay to assist in positioning.

The socket in position IC12 is not intended for an integrated circuit and it is **fitted at the solder side of the board**. You can (carefully) solder the socket pins at the component side of the board.

Do not insert the ICs in their sockets yet.

Connectors and ZIF socket
Simple 3-way pinheaders are used in positions K1 and JP1. The former receives a mating socket, the second, a jumper. There are two longer pinheaders, K4 and K3. The latter is





COMPONENTS LIST

Resistors:
R1-R24,R56-R62,R75-R79,R90,R92,R96 = 10kΩ
R25-R48 = 180Ω SMD
R49-R55,R63-R74,R83 = 1kΩ
R80 = 390Ω
R81,R82 = SIL resistor array 4 x 4kΩ7
R84 = 6kΩ8
R85,R88 = 1MΩ 1%
R86,R87 = 100kΩ 1%
R89 = 39kΩ
R91 = 100kΩ
R93 = 1kΩ 1%
R94,R95 = 1Ω 1%
R97 = 2kΩ2
R98 = 47kΩ
R99 = 27Ω
R100,R101 = SIL resistor array 8 x 4kΩ7
R102 = SIL resistor array 8 x 10kΩ
R103 = SIL resistor array 8 x 10kΩ
P1 = 10kΩ preset
P2 = 1kΩ preset

Capacitors:
C1,C4-C8,C11,C13,C20,C22,C23 = 100nF
C2,C16,C17,C18,C21 = 10µF 16V radial
C3,C19 = 1nF
C9,C12 = 27pF
C10 = 470µF 35V radial
C14 = 10µF 25V radial
C15 = 4µF7 16V radial

Semiconductors:
D1,D3 = 1N4148
D2,D4,D5 = 1N4001
D6 = LED
T1,T2 = BC547
T3 = BD139
T4-T13 = BC640

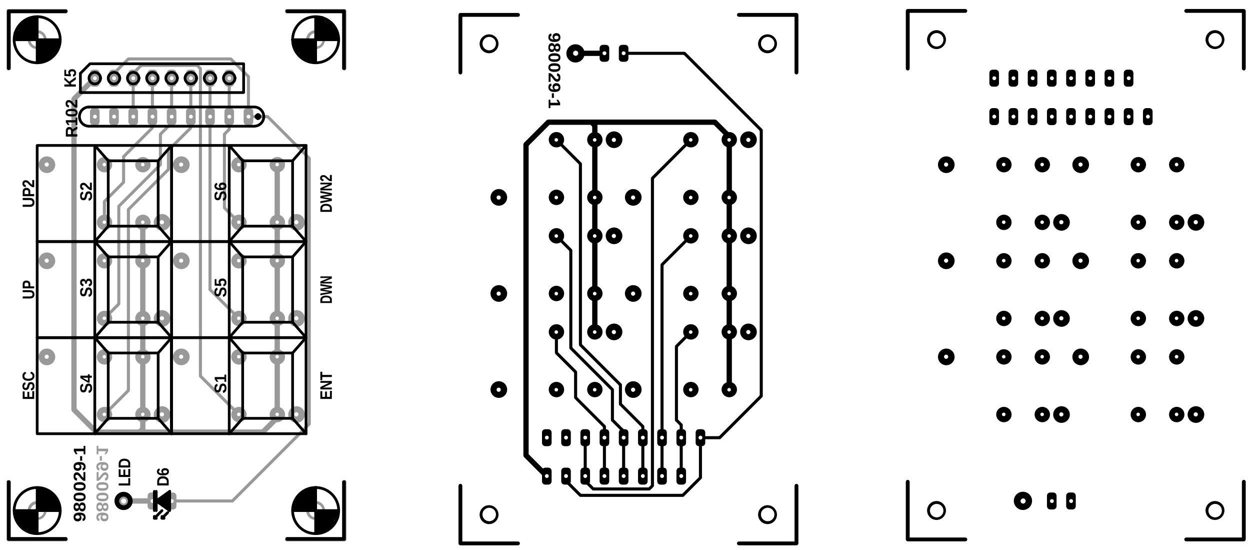
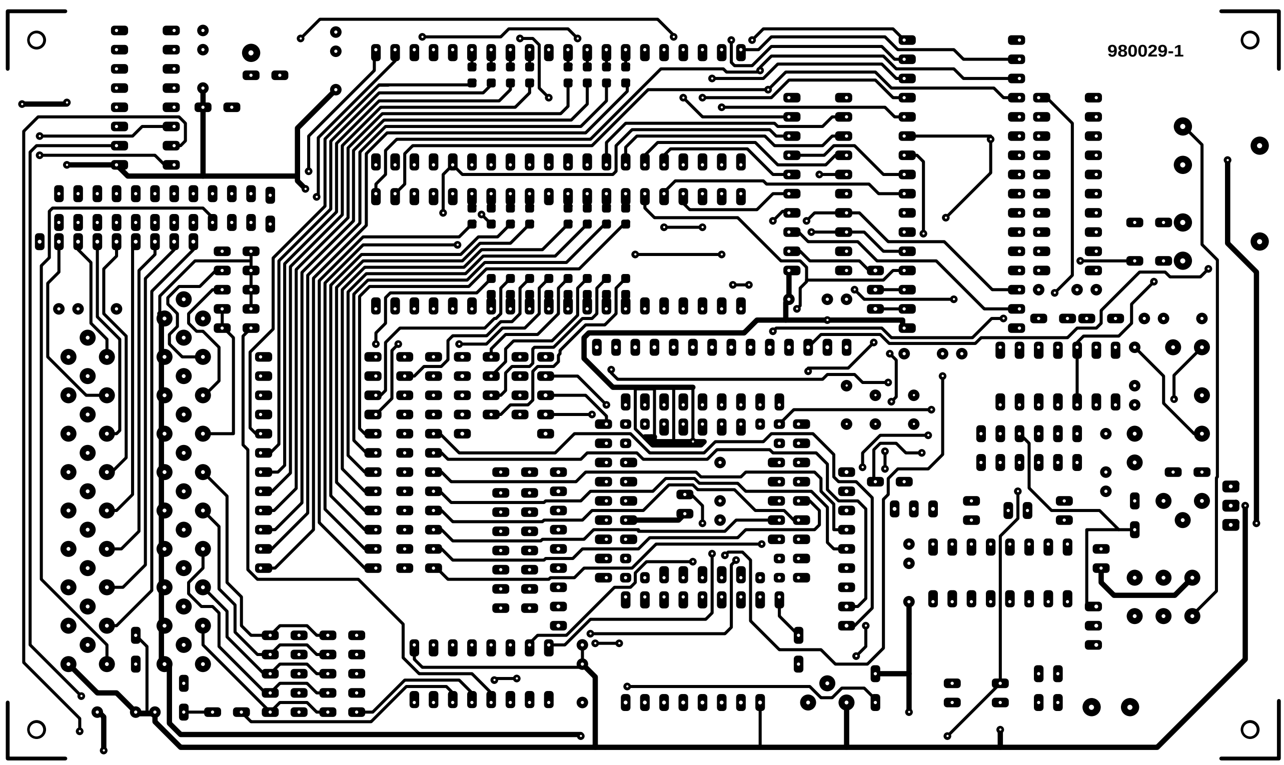


Figure 3. Copper track layouts and component overlays (actual size) of the double-sided, through-plated printed circuit board.

Items available from the Publishers:
PCB, disk, GAL and EPROM; set, order code 980029-C.
PCB only, order code 980029-1.
GAL 16V8 only, order code 986506-1.
EPROM 27C512 only, order code 986507-1.
Disk only, order code 986014-1.



T14-T22 = BC639
IC1,IC9 = 74HC4094
IC2,IC4 = Z80PIO
IC3 = SAB80C535-N
IC5 = GAL 16V8 (order code 986506-1)
IC6 = LM324
IC7 = EPROM 27C512 (order code 986507-1) (but use 32-pin socket!)
IC8 = 74HC573
IC10 = MAX232
IC11 = 7805
IC12 = 24-pin ZIF-socket (wide slots; Aries, Farnell)

Miscellaneous:
X1 = 12MHz quartz crystal
K1 = 3-pin SIL header
K2 = 2-way PCB terminal block (pitch 5mm)
K3 = 14-pin SIL header
K4,K5 = 8-pin SIL header
S1-S6 = Digitast push button (ITT Schadow), 4 black caps, 1 white (Ent), 1 red (Esc)
JP1 = 3-pin SIL header + jumper
LCD module, 2x16 characters
9-way sub-D socket (female)

mounted **at the solder side of the board** (and soldered at the component side). Its long pins receive a mating socket whose pins are soldered to the 14 copper spots on the LCD module. Doing so allows the LCD to be given a slant angle for optimum viewing.

Insert a second 24-pin socket into the one you soldered at the solder side in position IC12, and then insert the ZIF socket to make a nice stack.

K2 is a two-way PCB mount terminal block which receives the mains adaptor output voltage.

Keyboard
This is a simpler and smaller board which should not present any difficulty. Although a connector (K5) is indicated on the overlay, the 8-way flatcable between the keyboard PCB and the main board may be soldered directly to the spots at the solder side. One separate wire is used to control the LED, D6. It goes to a solder pin marked 'LED' on the main board.

That concludes the construction of the PCBs. Now's a good time to review your work so far. Any blatant errors?

Testing

With the ICs still waiting to be inserted in the respective sockets, connect-up the input voltage and run a quick check on the presence of the 5-volt supply voltage at the relevant pins of all IC sockets. Switch off and insert the LM324 (IC6). Connect the DMM to ground and the top wire of R61, switch on again and adjust P2 for a reading of 5.2 V.

Switch off and carefully insert all ICs. Note their orientations on the board! Switch on again. The LCD should read

IC Tester
1:Test

Okay so far? Congratulations!

Adjustments

The DUT supply output voltage, U_+, has to be set to 5.2 V ±0.05 V using preset P2. Next, P1 is adjusted for optimum contrast of the texts that appear on the LCD.

Operation

The tester is operated using six keys labelled Ent (enter), Esc (escape), dn (scroll down), up, dn2 (fast scroll down), and up2 (fast scroll up). The up and dn keys have an auto-repeat function which causes the repeat rate to be automatically increased as the key is held depressed. LED D6 lights to indicate that the IC under test is being powered, and should not be removed from the ZIF socket. **All ICs to be tested should be aligned towards pins 12/13 of the ZIF socket. Their 'top' notch is at the side of the ZIF socket lever.**

Pressing the Esc (escape) key takes you to the main menu. There, the following functions may be selected:
1. Test IC: the user picks an IC from an IC library, and the DUT is checked for correct operation. The test may be repeated. If indicated by the test vectors, the current consumption of the IC under test is measured and displayed.
2. Identify: this allows you to identify the type number of an unknown IC. If the GND and Vcc pins are unknown, only those test vectors are used that have the GND and Vcc pins at the same positions. The GND/Vcc pin entry is optional. Next, you can select the libraries that have to be scanned.
3. Retest IC: once an IC has been tested or identified, it may be tested again without having to pick it from the libraries.
4. Trace: all test vectors and the response of the DUT to these vectors appear in succession on the LC display.
5. Options: here, you can define global options.
6. Info: information on version and copyright.
7. Self Check: the IC tester hardware may be checked using this function and a voltmeter.
8. Remote Mode: connect a PC to the RS232 interface and debug test vectors using the DOS program TVCHK.EXE.

The up/dn keys are used to scroll one item up or down. The up2/dn2 keys do the same, but then five items at a time. The Ent key is used to confirm a selection. Esc, finally, jumps to the main menu.

Next month

Next month's second and final instalment will discuss the structure of the various menus which appear on the LCD, as well as the ins and outs of test vector creation, downloading, debugging and EPROM programming.

(980029-1)

4

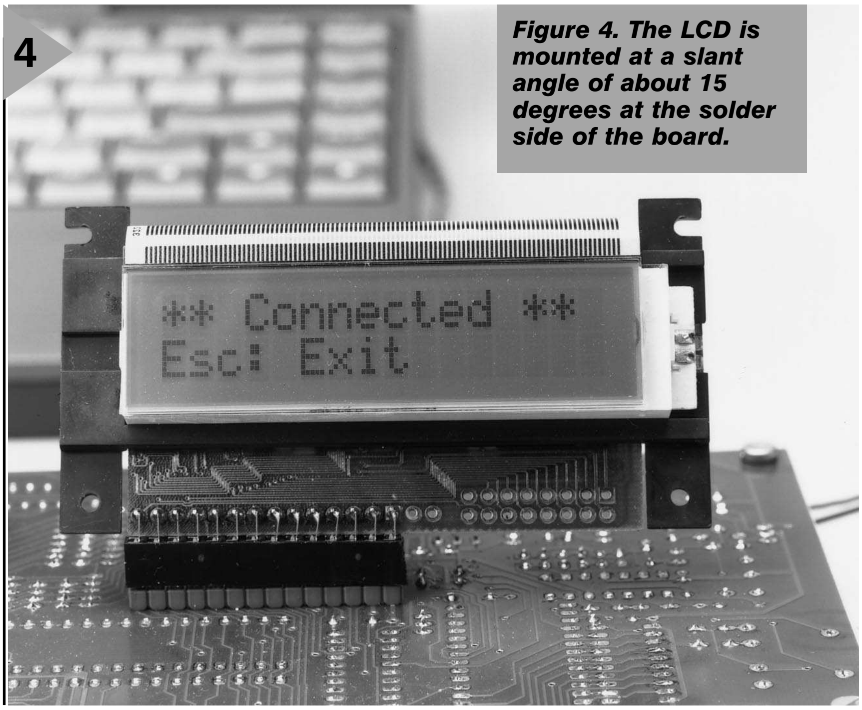


Figure 4. The LCD is mounted at a slant angle of about 15 degrees at the solder side of the board.

5

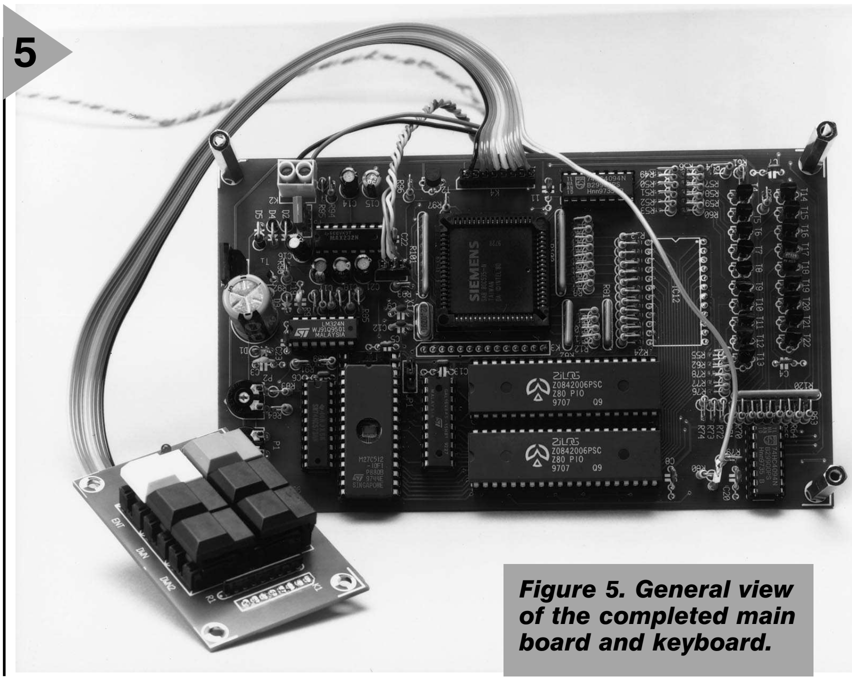


Figure 5. General view of the completed main board and keyboard.

IC tester

Part 2: menu structure and test-vector building

Last month we gave you the low-down on the hardware aspects of this powerful and versatile IC tester. In this second and concluding instalment, we turn to programming matters.

To be able to apply the information presented in this instalment, you have to have a working IC tester available, together with all the files found on the diskette with order number 986014-1 or CD-ROM 986001-1 (see further on). The tester is very likely to function properly if the message

IC Tester
1:Test

appears on the LCD when you switch the power on. However, this project features a number of clever hardware checking options. Although you may not need to do any hardware checking at all, the relevant routines are briefly mentioned further on. For now, it is assumed that the tester works properly.

Menu structure

As already mentioned in last month's instalment, you are looking at a test instrument which offers three communication channels with the real world: a keyboard, an LCD and a serial computer interface. Throughout the operation of the IC tester, menus are used to interact with you, the user. Selecting from the plethora of functions offered by the instrument is basically very simple and easy to learn once you know

1

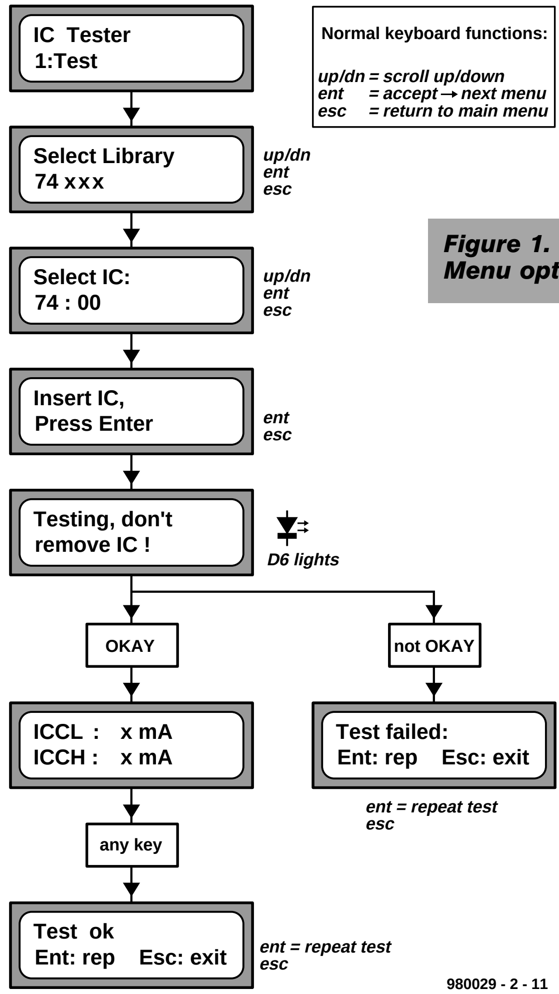


Figure 1. Structure of Menu option 1: Test.

the functions of the (few) keys on the keyboard. Fortunately, the use of these keys is consistent in most menus. To help you stay oriented, the structure of each of the major sub-menus offered by the tester is shown in **Figures 1 through 8**. Where a key has a non-standard function, this is indicated separately.

Many of you will be perfectly happy to use the '1:Test' menu most of the time, or the '3:Retest' menu if you are looking at a pile of identical ICs which should be subjected to a go/non-go test.

Test vector file

From here on, we are addressing the more advanced users among you who require a deeper knowledge of the way the instrument uses built-in software to test logic integrated circuits. For a good understanding of the hardware/software interaction, it is necessary to analyse the structure of the file Test Vector File which is permanently stored in the system EPROM, together with the program executed by the microcontroller.

A copy of the test vector file which is compiled, converted to binary and then stored in the 'default' system EPROM (order code 986507-1) may be found on the project diskette, order code 986014-1, as well as on the '*µP-µC Hard & Software 97-98*' CD-ROM, order

Design by L. Lamesch

code 986001-1 (look in the subdirectory /INT/BIN). The file is called VECT.TVC, and contains all information the system needs to check the massive amount of ICs listed in Table 1 in last month's instalment. For now, let's load 'VECT.TVC' into a word processor, and examine the syntax used. What do all these keywords mean?

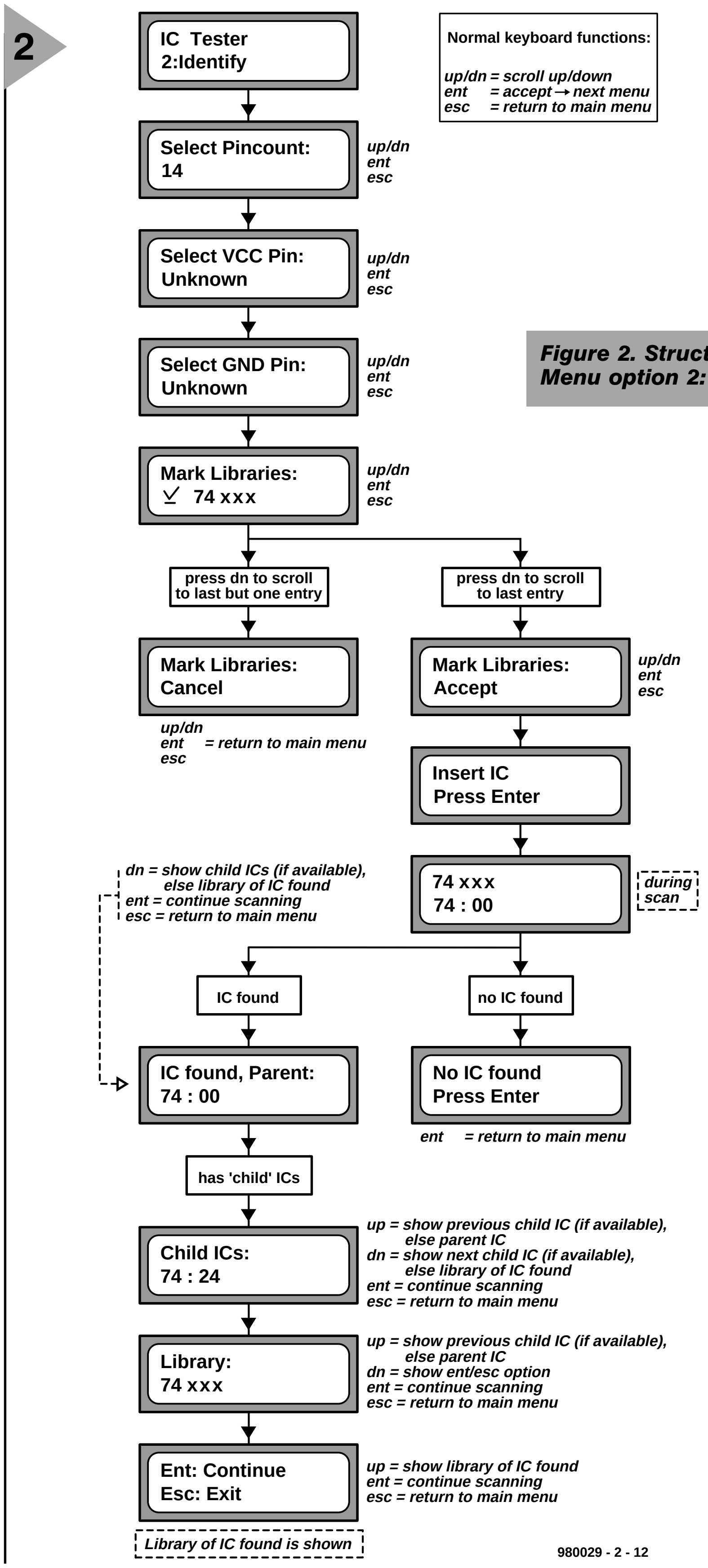


Figure 2. Structure of Menu option 2: Identify.

;
Comment delimiter. All characters following the semicolon are treated as comment, and ignored by the system.

LIBRARY lib_name
Defines the start of an IC library. This should be the first keyword in the .TVC file (except comment, of course). All IC types which follow this keyword belong to the library with the name 'lib_name'. The name of the library may consist of up to 15 characters. There is no limit to the number of libraries in a .TVC file.

NAME ic_name
Defines the start of IC data, and launches the test vector set for the IC with the name 'ic_name' (max. length is 15 characters). There is no limit to the number of ICs in a library. The end of the test vector set is marked by the next occurrence of 'NAME', 'LIBRARY', or the end of the .TVC file.

CHILD parent_ic_name
If an IC employs the same test vectors as one already defined, it is sufficient to identify it as a 'child' of a 'parent'. For example, the 74:132 has the same function and pinout as the 74:00, except that the gates are of the Schmitt trigger type (which is not recognised by the tester). A 74:132 is defined as follows:

```
NAME     74:132   ;define new IC
CHILD    74:00    ;declare as 74:00
                   offspring
```

A parent may have up to 100 children.

PINS pin_count
Unless a certain IC is a 'child', its test vectors have to be defined. In that case, the first keyword to use is always 'PINS' which defines the number of pins on the IC.

PINORDER pin_order
Links the individual IC pins to the columns that supply the test vectors. 'PINORDER' may only follow 'PINS', and the pins are identified using their pin numbers. Individual pin numbers should be separated by a space character. All IC pins have to be identified in 'pin_order', including pins which are not tested.

PINDEF pin_definition
This defines the function of each individual pin. The following functions are available:
O output
I input
G ground pin
V Vcc (+supply) pin
The individual pin functions may be separated by spaces (not obligatory, though), and all pins must be included in 'pin_definition'. Pins which should not be tested are defined as outputs ('O'), and not tested in the 'VECT' line ('X'). 'PINDEF' should precede the first 'VECT'. After a 'VECT' line, the pin functions may be redefined using 'PINDEF'.

3

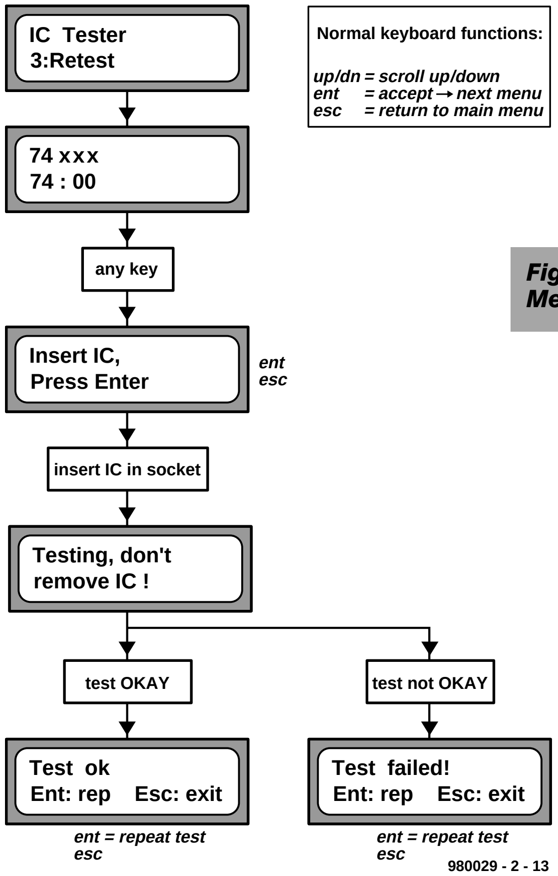


Figure 3. Structure of Menu option 3: Retest IC.

VECT **test_vector**
A test vector may consist of the following elements:
1 output pin: check if pin is at 1;
input pin: apply 1 to pin.
0 output pin: check if pin is at 0;
input pin: apply 0 to pin.
z test if pin is at high impedance.
x do not test this pin.

The individual elements may be separated by spaces (not obligatory). If a pin is defined as 'GND', it should have a 0 at the relevant position in the test vector. Likewise, 'VCC' should be matched with a 1.

REPEAT **count,** ENDR
This keyword allows a loop to be implemented, containing test vectors (VECT) and pin function definitions (PINDEF). The loop is repeated 'count' times. Each loop has to be terminated with an 'ENDR' keyword. Nested loops are not allowed.

PULL **on_off**
This keyword tells the IC tester hardware to connect the IC outputs to pull-down resistors during the test (on_off = 1) or not (on_off = 0). When on_off is at 0, the outputs are continuously loaded by a pull-up resistor, and testing for a high-Z state is not possible.

ICCL
When this instruction is encountered, the regulated power supply measures the IC supply current. After the test, the supply current is indicated as 'ICCL'. 'ICCL' may only be used once for any one IC.

ICCH
When this instruction is encountered, the regulated power supply measures the IC supply current. After the test, the supply current is indicated as 'ICCH'. 'ICCH' may only be used once for any one IC.

The following points should be noted when writing your own test vectors for ICs not included in the default library.

◗ All IC inputs have to be made logic 0 and logic 1 at least once, in a manner that ensures that this change can be detected on at least one IC output.
◗ All IC outputs have to go 0 and 1 at least once during the test, and also 'Z' (high-impedance or tri-state) when an output can assume this state.
◗ With ICs having a sequential internal circuit, IC inputs driving the clock input of a register flip-flop may not change state in unison with inputs of these flip-flops, if this transition equals the active edge of the clock input. For example, the clock input of a 74:74 may not change from 0 to 1 when the level at the data input changes at the same time. This proviso also applies to clock and enable inputs of synchronous counters.
◗ If an enable input of a latching flip-flop toggles, the level at the data input of the flip-flop is not allowed to toggle at the same time. This applies to rising as well as falling pulse edges.

To close off this section, **Figure 9** shows

4

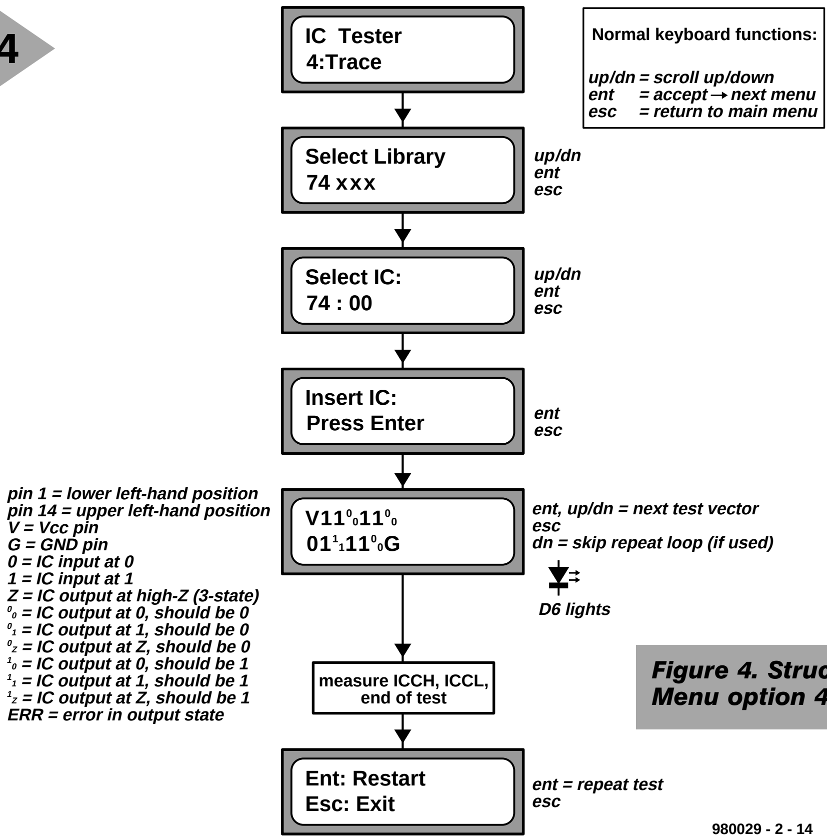


Figure 4. Structure of Menu option 4: Trace.

an example of a set of test vectors written for the type 4040 CMOS 12-stage ripple-carry binary counter. Although not all compiler keywords are used, the example is still useful to unravel the structure of the vectors in relation to the internal operation of the IC. A pinout diagram is included for your convenience. Note how the 'repeat' instruction is used to toggle the logic level at the clock pulse input (cp). Depending on the number of clock pulses applied in this way, a particular IC output (Q_x) should go high. The hardware monitors this output, and its level is compared with that stated by the test vector. For example, a logic 1 should occur on output Q8 of the 4040 after 128 clock pulses. If this logic one is not measured after the 'endr' instruction, in other words, if the response of the IC under test does not match the test vector after 'endr', the IC is identified as faulty, and may be binned! Comment is also used in this test vector script: the fourth line contains the logic labels of the IC pins. Note that some manufacturers of the 4040 start with output label Q1 rather than Q0 as indicated by the comment. This does not affect the operation of the IC, however.

More IC test vector scripts and internal diagrams are given on this month's *Datasheets*.

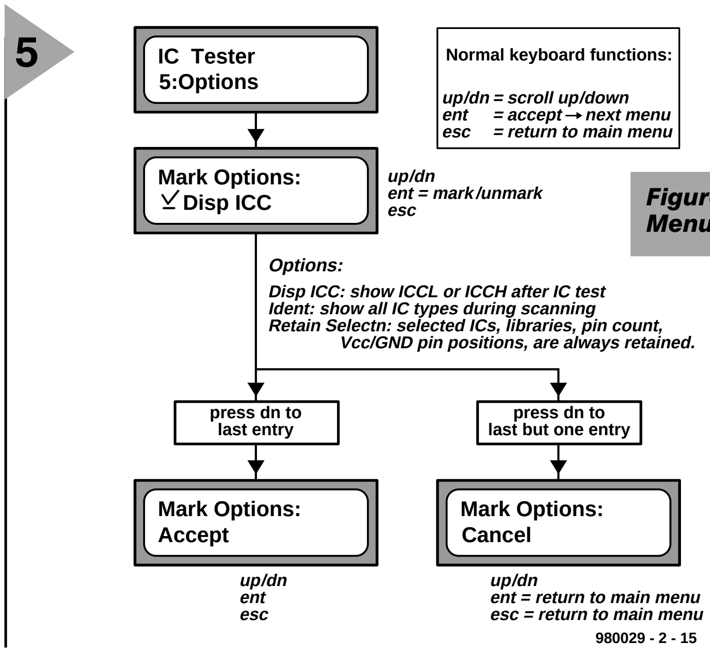


Figure 5. Structure of Menu option 5: Options.

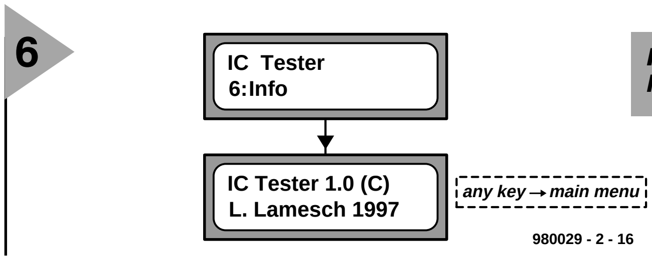


Figure 6. Structure of Menu option 6: Info.

TEST VECTOR COMPILING AND DEBUGGING

The purpose of analysing the master test-vector file as we just did is to enable you to write your own test scripts for ICs not included in the default library (see Table 1 in part 1). In principle, you only need a datasheet of the IC to reason how it *should* work. Eventually, you may want to add the new test vectors to the ones already available in 'vect.tvc', and burn the lot into a new system EPROM. The programs and general procedures to do so will be described below. Remember that all of the information presented below may be totally academic to you if you are satisfied with the collection of ICs in the default library.

Do not launch the programs directly from the floppy disk. First run `CHECK 1` from the DOS prompt as indicated on the floppy to make sure the data is intact and virus-free. Next, copy all files on the floppy to a suitably named subdirectory on the hard disk.

ICTVC.EXE

This is the **Test Vector Compiler**. Its function is to turn a test vector source file (like VECT.TVC) into a test vector binary file. Next, the latter file has to be appended to the microcontroller program, ICT.BIN, to create a large binary file that can be burned into an EPROM. File appending, by the way, is achieved with the aid of the DOS command `copy ict.bin+vect.out /b eprom.bin`.

The compiler is invoked by typing

`ICTVC [source file.TVC]`

For example,

`ICTVC VECT.TVC`.

It generates the following files:

TVC.OUT: test vector file (binary);
ERR.OUT: error report;
LIST.OUT: list-file containing information on the source file, binary file, a copy of the source file with line num-

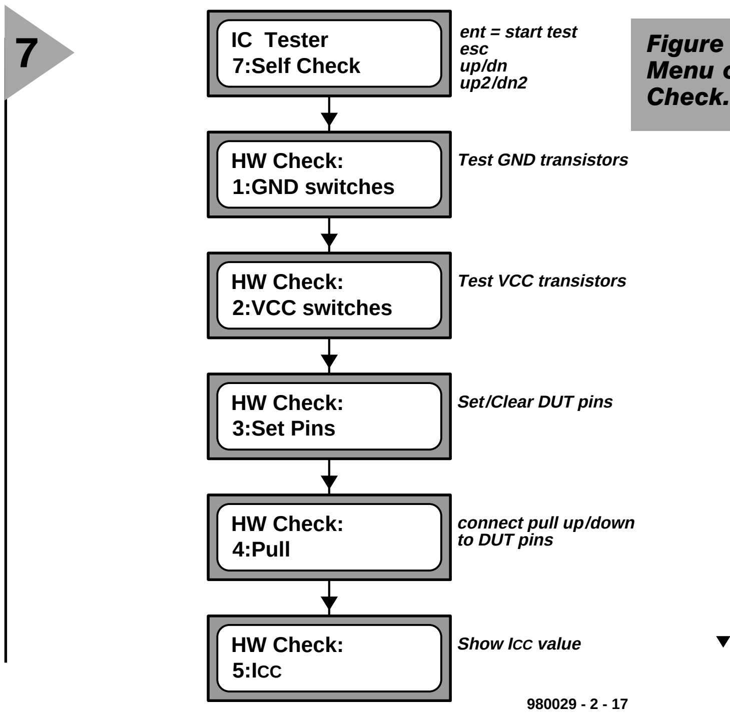


Figure 7. Structure of Menu option 7: Self Check.

8

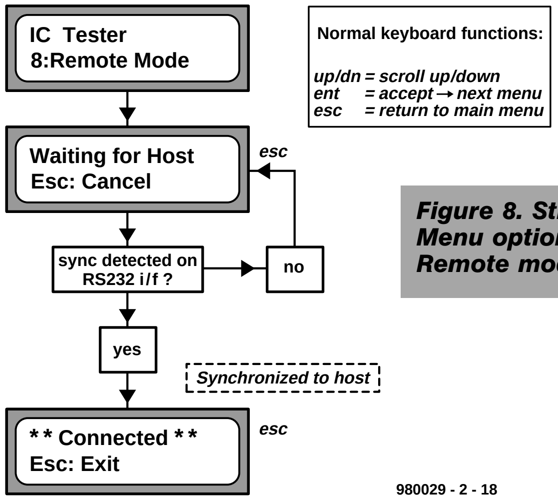


Figure 8. Structure of Menu option 8: Remote mode.

9

```
name        4040
; 12-bit CMOS ripple carry counter
pins        16
pinorder  10  11  1   15  14  12  13  4   2   3   5   6   7   9   8   16
;         /cp mr  q11 q10 q9  q8  q7  q6  q5  q4  q3  q2  q1  q0  gnd vcc
pindef    I   I   O   O   O   O   O   O   O   O   O   O   O   O   G   V
vect      0   1   0   0   0   0   0   0   0   0   0   0   0   0   0   1
vect      0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   1
vect      1   0   0   0   0   0   0   0   0   0   0   0   0   0   0   1
vect      0   0   0   0   0   0   0   0   0   0   0   0   0   1   0   1
vect      1   0   0   0   0   0   0   0   0   0   0   0   0   1   0   1
vect      0   0   0   0   0   0   0   0   0   0   0   0   1   0   0   1
vect      1   0   0   0   0   0   0   0   0   0   0   0   1   0   0   1
vect      0   0   0   0   0   0   0   0   0   0   0   0   1   1   0   1
vect      1   0   0   0   0   0   0   0   0   0   0   0   1   1   0   1
vect      0   0   0   0   0   0   0   0   0   0   0   1   0   0   0   1
repeat 4
vect      0   0   0   0   0   0   0   0   0   0   0   X   X   X   0   1
vect      1   0   0   0   0   0   0   0   0   0   0   X   X   X   0   1
endr
vect      0   0   0   0   0   0   0   0   0   0   1   0   0   0   0   1
repeat 8
vect      0   0   0   0   0   0   0   0   0   0   X   X   X   X   0   1
vect      1   0   0   0   0   0   0   0   0   0   X   X   X   X   0   1
endr
vect      0   0   0   0   0   0   0   0   0   1   0   0   0   0   0   1
repeat 16
vect      0   0   0   0   0   0   0   0   0   X   X   X   X   X   0   1
vect      1   0   0   0   0   0   0   0   0   X   X   X   X   X   0   1
endr
vect      0   0   0   0   0   0   0   0   1   0   0   0   0   0   0   1
repeat 32
vect      0   0   0   0   0   0   0   0   X   X   X   X   X   X   0   1
vect      1   0   0   0   0   0   0   0   X   X   X   X   X   X   0   1
endr
vect      0   0   0   0   0   0   0   1   0   0   0   0   0   0   0   1
repeat 64
vect      0   0   0   0   0   0   0   X   X   X   X   X   X   X   0   1
vect      1   0   0   0   0   0   0   X   X   X   X   X   X   X   0   1
endr
vect      0   0   0   0   0   0   1   0   0   0   0   0   0   0   0   1
repeat 128
vect      0   0   0   0   0   0   X   X   X   X   X   X   X   X   0   1
vect      1   0   0   0   0   0   X   X   X   X   X   X   X   X   0   1
endr
vect      0   0   0   0   0   1   0   0   0   0   0   0   0   0   0   1
repeat 256
vect      0   0   0   0   0   X   X   X   X   X   X   X   X   X   0   1
vect      1   0   0   0   0   X   X   X   X   X   X   X   X   X   0   1
endr
vect      0   0   0   0   1   0   0   0   0   0   0   0   0   0   0   1
repeat 512
vect      0   0   0   0   X   X   X   X   X   X   X   X   X   X   0   1
vect      1   0   0   0   X   X   X   X   X   X   X   X   X   X   0   1
endr
vect      0   0   0   1   0   0   0   0   0   0   0   0   0   0   0   1
repeat 1024
vect      0   0   0   X   X   X   X   X   X   X   X   X   X   X   0   1
vect      1   0   0   X   X   X   X   X   X   X   X   X   X   X   0   1
endr
vect      0   0   1   0   0   0   0   0   0   0   0   0   0   0   0   1
vect      0   1   0   0   0   0   0   0   0   0   0   0   0   0   0   1
vect      0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   1
iccl      ; measure IC current
icch      ; measure IC current
```

bers added as well as the bytes generated from each line;
TMP.OUT: temporary file used by ICTVC.EXE.

Any errors occurring during the compilation process are recorded in ERR.OUT only. They do not appear on the PC screen.

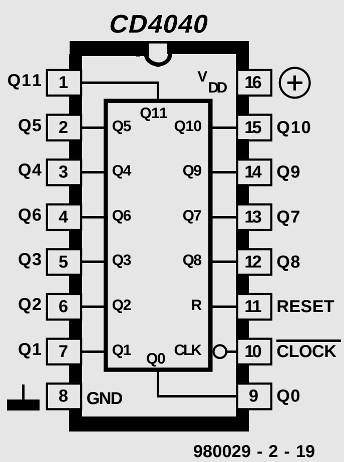
TVCHK.EXE

This is a kind of shell program that launches ICTVC.EXE, and enables test vector scripts to be debugged. For this program to operate you have to connect the IC Tester to your PC via the 3-wire serial link. TVCHK should be launched with an appended parameter which is either the COM port number (1-4), or the COM port address (in hex) followed by the associated interrupt line (1-7). Example: *TVCHK 2*. Obviously, we are talking of the COM port to which the IC Tester is connected! A screendump illustrating some of the options offered by TVCHK is shown in **Figure 9**.

EDT.BAT

This extremely small batch file is used to launch the word processor you will be using to load, modify and save test vector files. If you do not want to employ EDIT.COM, change EDT.BAT as required to make it point to your favourite DOS text editor. Type the name of the file you want to process after '*EDT*'. In case 'VECT.TVC' is too large for your wordprocessor, consider creating the part to be appended as a separate file. Once the new test-vector

Figure 9. Example illustrating the operation of a test vector script. The diagram and the script are all you need to put the 4040 12-bit counter through its paces.



980029 - 2 - 19

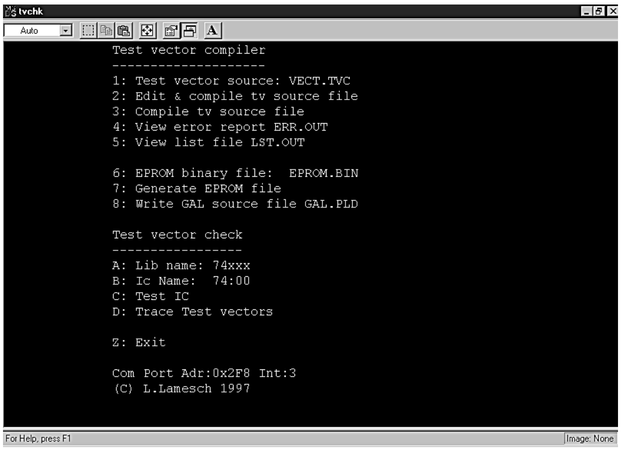


Figure 10. The TVCHCK program offers a useful shell around the utilities for test vector testing and compiling.

scripts have been debugged, they may be appended to VECT.TVC without using TVCHK. Next, the option 'Compile tv source' from TVCHK is used to compile the complete file for burning an EPROM.

A good start for practising is the file 'SMALL.TVC'. To get the feel, try replacing 'VECT.TVC' by 'SMALL.TVC' under menu option 2 in TVCHK.

The main menu of TVCHK comprises the following options:

Test Vector Compiler
1: Test vector source: VECT.TVC
Indicate name of test vector source file.
2: Edit & Compile tv source file.
3: Compile tv source file.
4: View error report ERR.OUT.
5: View list file LST.OUT.
6: EPROM binary file: EPROM.BIN
Indicate the name of the binary file to be burned into the EPROM.
7: Generate EPROM file.
Remember, the output format is binary!
8: Write GAL source file GAL.PLD.

Test Vector Checking
A: Lib name: 74xxx
Indicate name of the library containing the test-vector set to be tested/debugged.
B: IC name: 74:00
Indicate name of test-vector set to be tested/debugged.
C: Test IC
This may take a while when many test vectors have to be applied (e.g., more than 10 minutes when testing a script for the 4020).
D: Trace Test Vectors
This is the actual debugging tool. For each test vector, the expected logic states and the ones actually measured on the DUT are indicated. Errors are highlighted in red. Pin 1 is always indicated at the left-hand side, and the pin with the highest pin number, at the right-hand side. REPEAT…ENDR loops may be skipped by pressing the 's' key.
Z: Exit
Quit the program.
COM Port Addr: xxx Int: y
Tell the PC which COM port to use for communication with the IC Tester.

Larger eproms

Those of you who wish to add test vector scripts for ICs not supported by the default system EPROM may soon find that the size of the .BIN file generated by TVCHK option '7' exceeds the capacity of a 512-kbit (64-kByte) EPROM like the 27(C)512. That is not a problem, however, because the IC tester hardware accommodates larger EPROMs like the 2-Mbit (256-kByte) 27020 without problems. Because these giant EPROMs are divided into 64-kByte banks, bank-switching then has to be implemented by means of outputs B6 and B7 of PIO device IC2. Address line A17 is then also required, so you have to set jumper JP1 to the 'not-A' position when using a 27C020 EPROM.

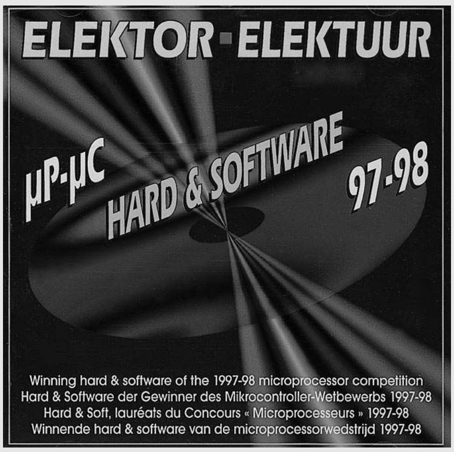
Conclusion

You have been reading an article discussing a test instrument which, in its standard version, allows a vast number of integrated logic circuits from the 74 (TTL) and 4000 (CMOS) series to be subjected to some pretty thorough testing. If you are not satisfied with the range of ICs that can be tested, a number of powerful software tools are available to 'roll your own', just using a common-or-garden PC (running plain old DOS) and, optionally, an EPROM programmer capable of handling EPROMs with a capacity of at least 512 kbits. Happy testing!

(980029-2)

The 100% DIY approach

Although the best guarantee to successful construction of this project is to order a ready-programmed GAL and EPROM from the Publishers, together with a PCB and a floppy disk (order as a set, order code 980029-C), there is are two alternative, cheaper, ways for the more audacious.
The CD-ROM entitled 'µP-µC Hard & Software 97-98' (order code 986001-1) contains THE WORKS, i.e., all files to program your own GAL and EPROM for this project, in addition to the 'master' test vector file, the 535's executable code, source code files in C and assembly language, and all software utilities mentioned in this instalment. So, if you are completely self-supporting, that is, have access to a PC, a GAL programmer and an EPROM programmer, we suggest buying just the PCB and the CD-ROM. The relevant subdirectory on the CD-ROM is /INT. Remember, you can not run the test vector utilities from the CD-ROM since they need to create files!



The second option for the more advanced among you is to buy only the PCB and the floppy disk. The floppy contains a sub-set of the files on the CD-ROM: not included are the assembler files and the PCB artwork and circuit diagrams as originally supplied by the author. For the rest, everything is included to create and debug your own test vector files, and prepare a binary file for burning into an EPROM, as described in this article. The .JED file for programming your own GAL is also included.
Whichever option you choose, remember that any hardware or software component needed to build (and understand) this project is available separately through our Readers Services.

IC Tester

ELEKTOR ELECTRONICS

Elektor Electronics
March & April 1998

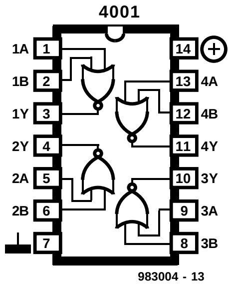
DATASHEET 4/98

IC Test Vectors (illustrated examples) Extract from: vect.tvc

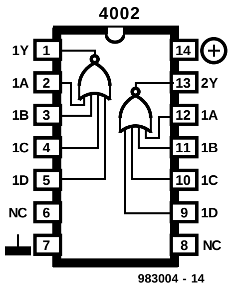
Note: tabbed formatting and IC functional diagrams added for clarity's sake, these are not available in file 'vect.tvc'.

1 = logic high   0 = logic low   G = ground   V = positive supply voltage
I = input   O = output   nc = not connected   XX = don't care
** = vectors not yet verified with a correctly operating IC*
iccl = measure supply current   icch = measure supply current

name	4001													
pins	14													
pinorder	1	2	3	6	5	4	13	12	11	8	9	10	7	14
pindef	I	I	O	I	I	O	I	I	O	I	I	O	G	V
vect	0	0	1	1	1	0	1	1	0	1	1	0	0	1
vect	1	0	0	1	1	0	1	1	0	1	1	0	0	1
vect	0	1	0	1	1	0	1	1	0	1	1	0	0	1
vect	1	1	0	0	0	1	1	1	0	1	1	0	0	1
vect	1	1	0	0	1	0	1	1	0	1	1	0	0	1
vect	1	1	0	1	0	0	1	1	0	1	1	0	0	1
vect	1	1	0	1	1	0	0	0	1	1	1	0	0	1
vect	1	1	0	1	1	0	0	1	0	1	1	0	0	1
vect	1	1	0	1	1	0	1	0	0	1	1	0	0	1
vect	1	1	0	1	1	0	1	1	0	0	0	1	0	1
vect	1	1	0	1	1	0	1	1	0	0	1	0	0	1
vect	1	1	0	1	1	0	1	1	0	1	0	0	0	1
vect	1	1	0	1	1	0	1	1	0	1	1	0	0	1
iccl														
vect	0	0	1	0	0	1	0	0	1	0	0	1	0	1
icch														



name	4002													
pins	14													
pinorder	2	3	4	5	1	12	11	10	9	13	7	14	6	8
;	1a	1b	1c	1d	1y	2a	2b	2c	2d	2y	gnd	vcc	nc	nc
pindef	I	I	I	I	O	I	I	I	I	O	G	V	O	O
vect	1	0	0	0	0	0	0	0	0	1	0	1	X	X
vect	0	1	0	0	0	0	0	0	0	1	0	1	X	X
vect	0	0	1	0	0	0	0	0	0	1	0	1	X	X
vect	0	0	0	1	0	0	0	0	0	1	0	1	X	X
vect	0	0	0	0	1	1	0	0	0	0	0	1	X	X
vect	0	0	0	0	1	0	1	0	0	0	0	1	X	X
vect	0	0	0	0	1	0	0	1	0	0	0	1	X	X
vect	0	0	0	0	1	0	0	0	1	0	0	1	X	X
vect	0	0	0	1	0	0	0	0	1	0	0	1	X	X
iccl														
vect	0	0	0	0	1	0	0	0	0	1	0	1	X	X
icch														



AD557

ELEKTOR ELECTRONICS

Integrated circuits
Special Applications

DATASHEET 4/98

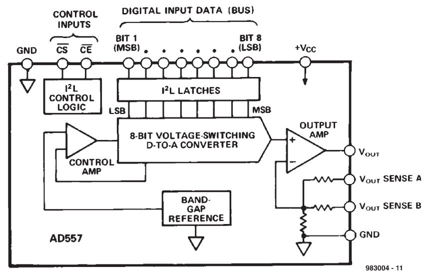
AD557
DACPORT, Low-Cost Complete µP-Compatible 8-bit DAC

Manufacturer
Analog Devices, One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, USA. Tel. (617) 329-4700, fax (617) 326-8703.
Internet: www.analog.com.

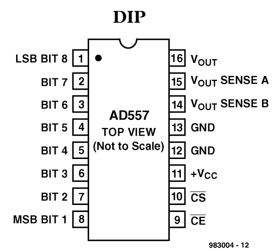
Features
Complete 8-Bit DAC
Voltage Output: 0 V to 2.56 V
Internal Precision Band-Gap Reference
Single-Supply Operation: + 5 V (± 10%)
Full Microprocessor Interface
Fast: 1 µs Voltage Settling to ± ½ LSB
Low Power: 75 mW
No User Trims Required
Guaranteed Monotonic Over Temperature
All Errors Specified T_{MIN} to T_{MAX}
Small 16-Pin DIP or 20-Pin PLCC Package
Low Cost

General description
The AD557 DACPORT® is a complete voltage-output 8-bit digital-to-analogue converter, including output amplifier, full microprocessor interface and precision voltage reference on a single monolithic chip. No external components or trims are required to interface, with full accuracy, an 8-bit data bus to an analogue system.
The low cost and versatility of the AD557 DACPORT are the result of continued development in monolithic bipolar technologies.
The complete microprocessor interface and control logic is implemented with integrated injection logic (I^2L), an extremely dense and low-power logic structure that is process-compatible with linear bipolar fabrication. The internal precision voltage reference is the patented low-voltage band-gap circuit which permits full-accuracy performance on a single + 5 V power supply.
Thin-film silicon-chromium resistors provide the stability required for guaranteed monotonic operation over the entire operating temperature range, while laser-wafer trimming of these thin-film resistors permits absolute calibration at the factory to within ± 2.5 LSB; thus, no user-trims for gain or offset are required. A new circuit design provides voltage settling to ± ½ LSB for a full-scale step in 800 ns.
The AD557 is available in two package configurations. The AD557JN is packaged in a 16-pin plastic, 0.3"-wide DIP. For surface mount applications, the AD557JP is packaged in a 20-pin JEDEC standard PLCC. Both versions are specified over the operating temperature range of 0ºC to + 70ºC.

Application Example
PC-aided BJT transistor tester revisited, *Elektor Electronics* April 1998.



functional block diagram



pin configuration (DIP)

AD557

Integrated circuits
Special Applications

ELEKTOR ELECTRONICS

DATASHEET 4/98

Electrical Specifications

(@ T_A = + 25°C, V_{CC} = + 5 V unless otherwise noted)

Model		Min	Typ	Max	Units
RESOLUTION				8	Bits
RELATIVE ACCURACY[1]	0 to + 70°C		± 1/2	1	LSB
OUTPUT	Ranges Current Source Sink	 + 5	0 to + 2.56 Internal Passive Pull-Down to Ground[2]		V mA
OUTPUT SETTLING TIME[3]			0.8	1.5	µs
FULL-SCALE ACCURACY[4]	@ + 25°C T_{MIN} to T_{MAX}		± 1.5 ± 2.5	± 2.5 ± 4.0	LSB LSB
ZERO ERROR	@ + 25°C T_{MIN} to T_{MAX}			± 1 ± 3	LSB LSB
MONOTONICITY[5]	T_{MIN} to T_{MAX}		Guaranteed		
DIGITAL INPUTS	T_{MIN} to T_{MAX} Input Current Data Inputs, Voltage — Bit On—Logic "1" Bit On—Logic "0" Control Inputs, Voltage — On—Logic "1" On—Logic "0" Input Capacitance	 2.0 0 2.0 0 	 4	 ± 100 0.8 0.8 	 µA V V V V pF
TIMING	t_W Strobe Pulse Width T_{MIN} to T_{MAX} t_{DH} Data Hold Time T_{MIN} to T_{MAX} t_{DS} Data Setup Time T_{MIN} to T_{MAX}	225 300 10 10 225 300			ns
POWER SUPPLY	Operating Voltage Range (V_{CC}) 2.56 Volt Range Current (I_{CC}) Rejection Ratio	 + 4.5	 15	 + 5.5 25 0.03	 V mA %/%
POWER DISSIPATION, V_{CC} = 5 V			75	125	mW
OPERATING TEMPERATURE RANGE		0		+ 70	°C

1 Relative Accuracy is defined as the deviation of the code transition points from the ideal transfer point on a straight line from the zero the the full scale of the device.
2 Passive pull-down resistance is 2 kΩ.
3 Settling time is specified for a positive-going full-scale step to ± ½ LSB. Negative-going steps to zero are slower, but can be improved with an external pull-down.
4 The full-scale output voltage is 2.55 V and is guaranteed with a + 5 V supply.
5 A monotonic converter has a maximum differential linearity error of ± 1 LSB.
Specifications subject to change without notice.

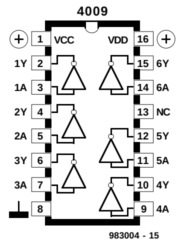
IC Tester

Elektor Electronics
March & April 1998

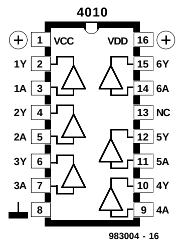
ELEKTOR ELECTRONICS

DATASHEET 4/98

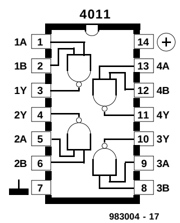
```
name 4009*
pins      16
pinorder 3   2   5   4   7   6   9   10  11  12  14  15  8   11  16  13
pindef   I   O   I   O   I   O   I   O   I   O   I   O   G   I   V   O
vect     1   0   1   0   1   0   1   0   1   0   1   0   0   1   1   X
vect     0   1   1   0   1   0   1   0   1   0   1   0   0   1   1   X
vect     1   0   0   1   1   0   1   0   1   0   1   0   0   1   1   X
vect     1   0   1   0   0   1   1   0   1   0   1   0   0   1   1   X
vect     1   0   1   0   1   0   0   1   1   0   1   0   0   1   1   X
vect     1   0   1   0   1   0   1   0   0   1   1   0   0   1   1   X
vect     1   0   1   0   1   0   1   0   1   0   0   1   0   1   1   X
vect     1   0   1   0   1   0   1   0   1   0   1   0   0   1   1   X
iccl
vect     0   1   0   1   0   1   0   1   0   1   0   1   0   1   1   X
icch
```




```
name 4010*
pins      16
pinorder 3   2   5   4   7   6   9   10  11  12  14  15  8   1   16  13
pindef   I   O   I   O   I   O   I   O   I   O   I   O   G   I   V   O
vect     1   1   1   1   1   1   1   1   1   1   1   1   0   1   1   X
vect     0   0   1   1   1   1   1   1   1   1   1   1   0   1   1   X
vect     1   1   0   0   1   1   1   1   1   1   1   1   0   1   1   X
vect     1   1   1   1   0   0   1   1   1   1   1   1   0   1   1   X
vect     1   1   1   1   1   1   0   0   1   1   1   1   0   1   1   X
vect     1   1   1   1   1   1   1   1   0   0   1   1   0   1   1   X
vect     1   1   1   1   1   1   1   1   1   1   0   0   0   1   1   X
vect     1   1   1   1   1   1   1   1   1   1   1   1   0   1   1   X
icch
vect     0   0   0   0   0   0   0   0   0   0   0   0   0   1   1   X
iccl
```




```
name  4011
pins      14
pinorder 1   2   3   6   5   4   13  12  11  8   9   10  7   14
pindef   I   I   O   I   I   O   I   I   O   I   I   O   G   V
vect     0   0   1   1   1   0   1   1   0   1   1   0   0   1
vect     0   1   1   1   1   0   1   1   0   1   1   0   0   1
vect     1   0   1   1   1   0   1   1   0   1   1   0   0   1
vect     1   1   0   0   0   1   1   1   0   1   1   0   0   1
vect     1   1   0   0   1   1   1   1   0   1   1   0   0   1
vect     1   1   0   1   0   1   1   1   0   1   1   0   0   1
vect     1   1   0   1   1   0   0   0   1   1   1   0   0   1
vect     1   1   0   1   1   0   0   1   1   1   1   0   0   1
vect     1   1   0   1   1   0   1   0   1   1   1   0   0   1
vect     1   1   0   1   1   0   1   1   0   0   0   1   0   1
vect     1   1   0   1   1   0   1   1   0   0   1   1   0   1
vect     1   1   0   1   1   0   1   1   0   1   0   1   0   1
vect     1   1   0   1   1   0   1   1   0   1   1   0   0   1
iccl
vect     1   0   1   1   0   1   1   0   1   1   0   1   0   1
icch
```

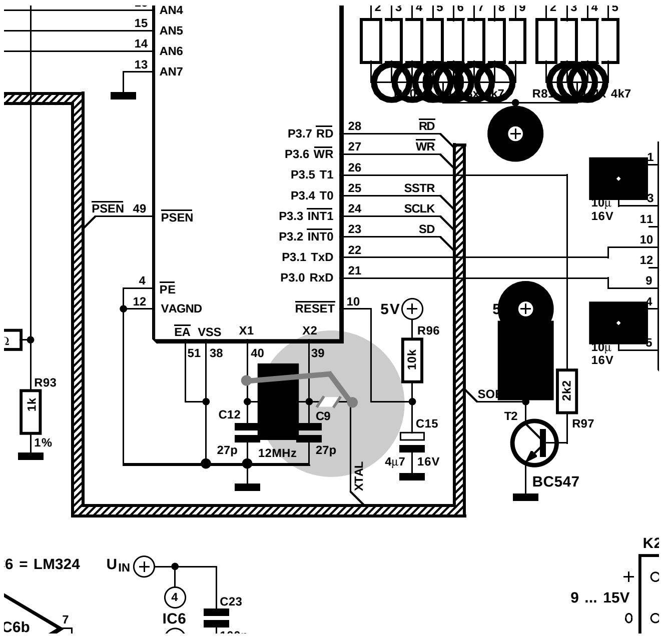


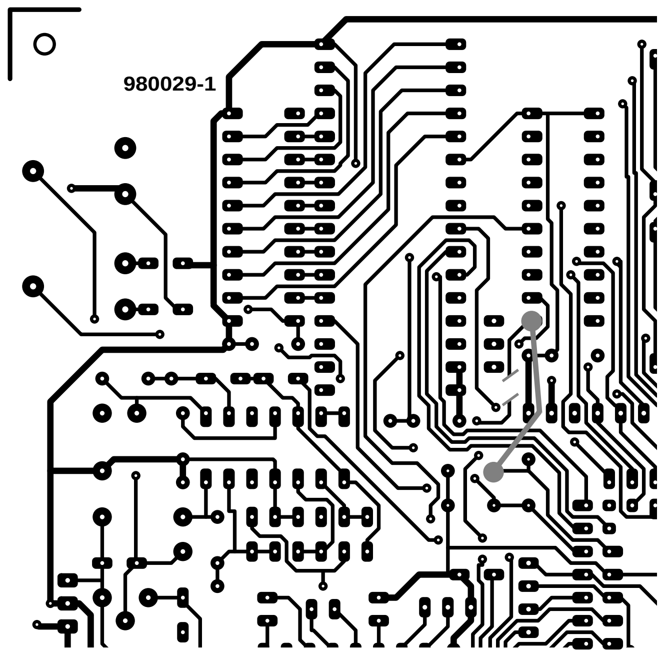
Intelligent IC Tester

March 1998, 980029

Pin 39 of the processor (IC3) is connected to pin 1 of the firmware GAL (IC5). Due to specific signal load conditions, this connection may cause problems with certain GAL brands.

The remedy is simple: instead of pin 39, use pin 40 of the processor as the oscillator output. The two PCB drawings show how the modification is made. Using a sharp hobby knife, a track is cut. Next, the new connection is made using a short length of thin, insulated wire.





SWITCHBOARD

Switchboard allows PRIVATE READERS of *Elektor Electronics* one FREE advertisement of up to 106 characters, including spaces, commas, numerals, etc., per month. The advertisement MUST relate to electronics, and it MUST INCLUDE a private telephone number or name and address; post office boxes are NOT acceptable.
Elektor Electronics (Publishing) can not accept responsibility for any correspondence or transaction as a result of a free advertisement or of any inaccuracy in the text of such an advertisement.
Advertisements will be placed in the order in which they are received. Elektor Electronics (Publishing) reserve the right to refuse advertisements without giving reasons or without returning them.

FOR SALE Handheld Dataman S4, complete with mains charger, manual, s/w and MCS51 ROM. £175. Phone Mike on (01749) 670016.

FOR SALE Kits of 1992/93 Elektor 1.2GHz Multifunction Frequency Meter, complete with case, front panel foil and all parts. Superb design and project, few available, £75 plus P&P. Anita, Sibberkerkstr. 100, NL-6301-AW, Valkenburg, Netherlands. Email techtext@worldonline.nl.

FOR SALE Ex-hobbyist's tools, oscilloscope, transformers, many components. Phone Rae Kozary on (0181) 9429377.

FOR SALE Graphtec MP3100 plotter (A3). WANTED Gould OS300 mains transformer (or dead scope). Phone G. Coast on 01634 314999

WANTED Book 2 of 'Formant Music Synthesizer', by M. Aigner. Klaus Nielsen, Viborggade 4, 4th. DK-2100, Coperhagen, Denmark.

LAB CLEAROUT Surplus components for sale: digital & analogue ICs, resistors, capacitors, switches, pots, bare PCBs (all unused) plus test equipment, incl. Microprocessor development system, PROM programmer, UV light, etc., Send SAE for list to D. Fittes, 8 Elisabeth Court, Warwick CV34 6QB, or tel. 01926 493092.

FOR SALE Lotus Smartsuite for Windows (123 Spreadsheet, Amipro word processor & Harvard Graphics), £20. Blackwells Idealist for Windows, database and retrieval system, unopened, unregistered, £20. All original disks with manuals. J. Hopkins, tel. 01243 784159.

WANTED A VHF to UHF convertor, or circuits, as I have a display unit with a VHF output, and I wish to connect a TV. Mr. T. Collins, 215 Arlott Crescent, Oldbrook, Milton Keynes MK6 2QT.

FOR SALE EPROM programmer GP EP8000 £75, Spectron D-586 datascope £185, Intel MDS + ICE with manuals etc. £100. P. Clark (01344) 868985.

FOR SALE Due to workshop clearance: 3 multimeters, transistor tester, frequency counter, valve voltmeter. Phone for detailed list. Ken Phillips, phone (01376) 323164 (Essex).

FOR SALE Sony SMC-70GP video titler with books, software, etc. Twin 3.5" drives, integral keyboard, genlocker, PAL superimposer and separate colour monitor. Job lot: untested and sold 'as seen'. First £50 secures. Trevor Wiltshire, Reading (0118) 9701163.

FOR SALE PSU PCB's, 45W, I/P 240VAC, O/P DC + 5V (x2), + 12V, –12V, £5 each. Tel. Paul on (01942) 706769 after 6pm.