Military Technical College
Kobry El-Kobba
Cairo, Egypt

12-th International Conference on Aerospace Sciences & Aviation Technology

DESIGN AND IMPLEMENTATION OF NEW AUTOMATIC TESTING SYSTEM FOR DIGITAL CIRCUITS BASED ON THE SIGNATURE ANALYSIS

MOHAMED H. EL-MAHLAWY [1], AND AHAMED SEDDIK [1]

ABSTRACT

This paper presents the design and implementation of a new automatic test equipment (ATE) for digital circuits based on the signature analysis. This testing system approach is designed to apply the test pattern generated either by a pseudorandom test pattern generator (PRTPG) or by a deterministic test pattern generator (DTPG) to the circuit under test (CUT). The response of the CUT is compacted by signature analyzer. The generated signature is automatically transferred to the personal computer (PC) through the parallel port of the personal computer (PC). This signature is stored on a file and displayed on the screen of the PC. The timing controller card and the control port of the parallel port generate all control signals to control all steps of the test cycle for proper operation. The PC equipped with a software part. It is designed to receive the signature automatically. Good signatures are generated from the good CUT and the measured signatures are generated from the CUT. Determination of whether the measured signature is good or bad is then made by comparing with those of the golden unit. Bad signatures are then traced back to locate the source of the fault.

KEYWORDS: Automatic Test Equipment, Testing of electronic digital circuits.

1. INTRODUCTION

Automatic Test Equipment (ATE) are widely used to detect and locate faults in electronic systems. ATE can minimize errors, omissions, faulty judgment, erroneous measurements, and other such drawbacks of human control. In addition, they eliminate many repetitious and tedious steps from tests, leaving only the steps that require human judgment and involvement [1-4].

Signature analysis is a technique pioneered by Hewlett-Packard Ltd., which detects errors in data streams, caused by hardware faults in digital systems [5-6]. This compaction technique can map lengthy bit streams into a *short signature*. This signature is a characteristic number representing time dependent logic activity during

[1] **Egyptian armed forces**

a specified measurement interval for a particular circuit node. Any change in the behavior of this node will produce a different signature indicating a probable circuit malfunction. Troubleshooting method by signature analysis can detect the faults in complex digital circuits with very high accuracy. The error will appear in the data stream only if the test pattern generator (TPG) generates the test pattern that detects the fault. Therefore, the basic steps of test technique are *circuit generated stimulus* and *data compaction*.

The ATE based on signature analysis capability can learn the signature at each node of interest on a good board and store it [2-4]. The signature analyzer is considering a test response compactor that generates the signature. It is realized by a linear feedback shift register (LFSR) which is simple in its implementation [7-8]. Fig. 1 and Fig. 2 illustrate the implementation of type 1 and type 2 LFSRs used for this purpose respectively. The autonomous LFSR (ALFSR) is modified to accept an external input in order to act as a polynomial divider [9-11]. The input sequence, represented by a polynomial, is divided by the characteristic polynomial of the LFSR. As the division proceeds, the quotient sequence appears at the output of the LFSR and the remainder is kept in the LFSR. Once testing is completed, the content of the LFSR can be treated as a signature.

The sequence generated by an *n*-stage ALFSR is less than or equal to 2^n-1. If the sequence, generated by an *n*-stage ALFSR, has period 2^n-1, it is called a *maximum length sequence*. The characteristic polynomial of a maximum length sequence is called a *primitive polynomial*. It can be rewritten as:

$$p(x) = 1 + c_1 x + c_2 x^2 + \ldots + c_{n-1} x^{n-1} + x^n \quad (1)$$

Fig. 1, and Fig. 2 have c_i's as binary constants in equation (1), c_i = 1 implies that a connection exists (feedback path), while c_i = 0 implies that there is no connection.

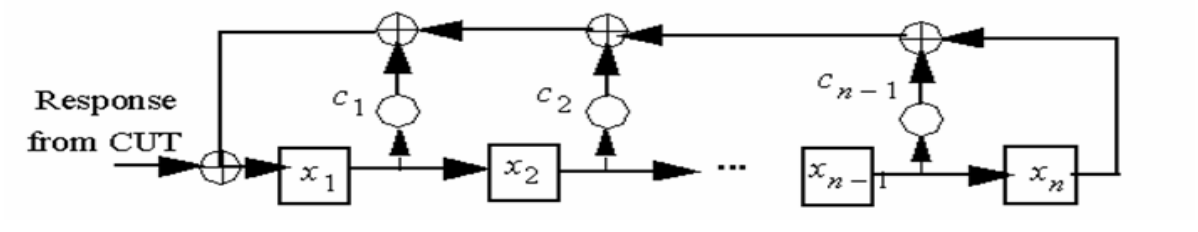

Fig. 1. Type 1 LFSRs as compactors.

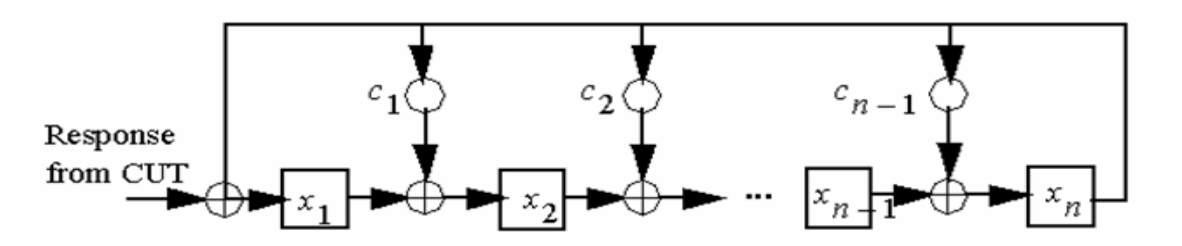

Fig. 2. Type 2 LFSRs as compactors.

In this paper, the system is designed to generate test pattern to hot exercising the CUT, receive the response from the CUT, and compact the response from a CUT.

The reference signatures are stored for the comparison with the measured one, and then determine the source faulty node(s).

This paper begins with the concept of testing techniques in section 2. Section 3 presents the concept of this testing approach. The block diagram of this new system is in section 4. The experimental results and the conclusion will be discussed in section 5 and section 6, respectively.

2. BASIC CONCEPT OF THE TESTING TECHNIQUES

Printed circuit boards are tested using *automatic test equipment (ATE)* that produce input test patterns and check binary outputs for correctness. A number of ATE systems are available, the variations are extensive, but most of them fall into two main categories [1-3, 13]: Functional tester, and In-circuit tester. An in-circuit tester examines each component on the board individually. The in-circuit tester requires a special kind of test fixture to provide electrical connections to all component pins [16]. A functional tester is important to the final user of the circuit board, it verifies that the board meets its performance specifications, i.e. it performs the functions it was designed for. Only board inputs and outputs need to be tested. The required test fixture is the edge connector that it is inexpensive and simple to construct.

With the advent of the complex integrated circuits, troubleshooting engineer for digital system finds himself dealing more with long digital data patterns than with waveforms. As packaging density increases fewer test points are available. The data streams at the available test points become very complex. Testing digital circuits for correct operation after manufacturing is an important problem. The problem is how to apply the required test patterns to the CUT and to analyze the response to locate the faulty component so that it can be replaced and the circuit board returned to service. When testing a combinational circuit, the problem can be viewed as shown in Fig. 3. During testing, a sequence of input test patterns is applied to the CUT to test whether the circuit operates properly. For each input test pattern, the binary output is compared to known correct binary value. If they do not match, an error has been found.

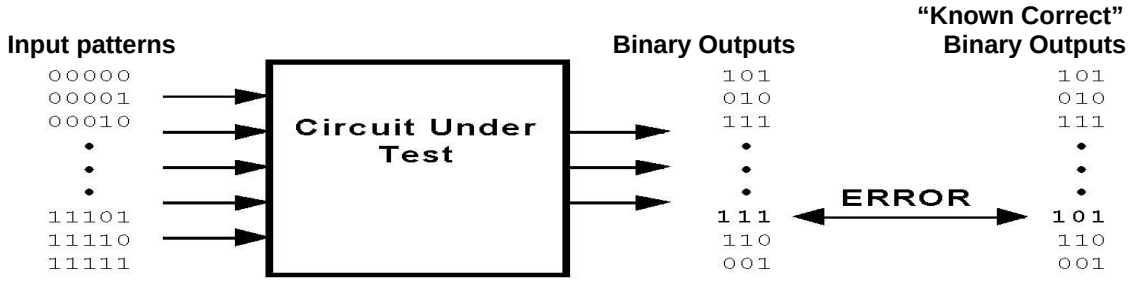

Fig. 3. Digital testing of the CUT.

The TPG that exercises the CUT is required to detect the faults in it. Test generation concept [10, 11, and 14] let fault *f* change the output function $Z(X)$ of a CUT to $Z_f(X)$. A test pattern *t* detects fault *f* if $Z(t) \neq Z_f(t)$. The fault *f* is undetectable (redundant) if $Z(t) = Z_f(t)$ for all *t*. A test set is complete if it detects (covers) all detectable faults. The easiest way to test a small combinational circuit is just to apply input test patterns for all possible input values. This is known as *exhaustive testing* [4]. As each input pattern is provided to the CUT, the resulting binary output is compared to the

corresponding row in the circuit's truth table, which represents the known correct output value. However, since exhaustive testing requires 2^n vectors for *n* inputs, this approach is not practical for large circuits.

The more common approach for testing is to use computer algorithms for *automatic test pattern generation* (ATPG) [11]. An ATPG algorithm models circuit failures as stuck-at faults – faults where a logic signal is either *stuck-at-one* or *stuck-at-zero*. For each fault, the ATPG algorithm searches for an input test pattern that detects this fault. ATPG algorithms are effective at finding sequences of test patterns that can detect in excess of 99% of these faults [11]. However, they are not practical for Built-In Self-Test (BIST) architecture. Instead, simple circuits are used to generate arbitrary test sequences that can detect a smaller set of faults. In pseudorandom testing [17], test patterns have the characteristics of random patterns in spite of being generated deterministically. Pseudorandom testing is widely used for testing digital integrated circuits. Pseudorandom test patterns can be generated by simple hardware circuits, for example, an autonomous LFSR (ALFSR).

The main concern with the pseudorandom testing approach is to choose the right test pattern generator to provide high fault coverage, while keeping the test set length short. For a given pseudorandom test set, fault coverage is generally obtained by simulation (fault coverage can be determined by fault simulation). If the fault coverage is found unacceptable, the length of the test set is increased. For large circuits, when simulation is expensive, probabilistic measures are used to compute the length of the test set for the desired fault coverage. The length of test set for a desired level of fault coverage should be much shorter than that required for exhaustive testing [9-11].

The data streams at the available test points become very complex. Signature analysis is a compaction technique that detects errors in data streams caused by hardware faults [4]. The *signature analyzer* (SA) compacts the output response for each output of the CUT into a shorter sequence. A smaller number of bits reduce the number of bits that must be stored and compared. This addresses the problem of the storing and comparison. Fig. 4 illustrates as an example of the schematic of a 4-stage simple signature analyzer. This circuit resembles an LFSR but has an additional PROBE input that is connected to an output in the CUT. During each clock cycle, an input test pattern is applied to the CUT and the resulting binary output is XOR'd with the feedback bits from the shift register. The result of this XOR is fed into the serial input of the shift register. After all input test patterns are applied, the shift register will contain a *signature*. A signature can be computed for known correct output values and then compared to the measured signature. When the good signatures and measured signatures are differed, a fault has been detected. The SA can detect almost all faults in the output of a CUT while requiring the storage of far fewer bits.

The aliasing probability of the signature analyzer is often estimated based on probabilistic models of erroneous sequences generated by faulty circuits. For an input data sequence of length *m*, if all possible errors in a sequence are equally likely, then the aliasing probability of an *n*-stage signature generator is equal to $(2^{m-n}-1)/(2^m-1)$, which for *m* >> *n*, approaches 2^{-n} [12]. Let a 23-stage signature generator be used. The worst-case probability of detecting an error as long as the error appears in the data stream, and the error bit(s) have been clocked in the SA

equals to 99.999988 %. The error will appear in the data stream only if the TPG generates the test pattern that detects the fault. Therefore, the basic steps of test technique are *circuit generated stimulus* and *data compaction*. Increasing the length of the signature analyzer increases the accuracy of it.

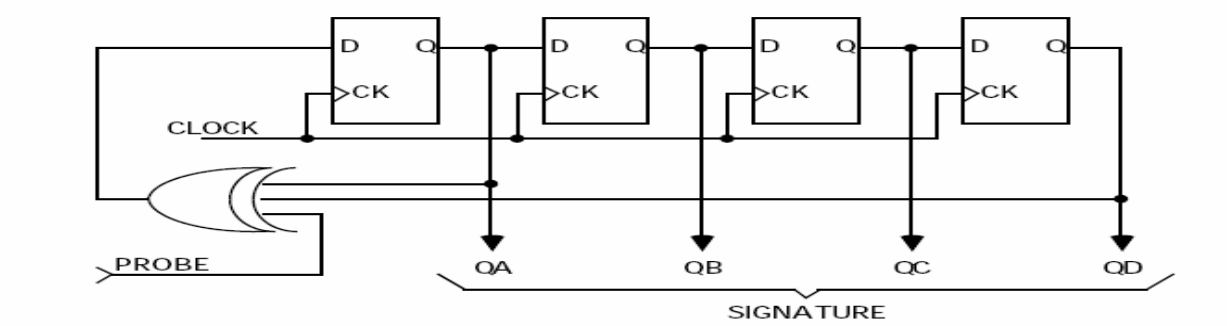

Fig. 4. Simple Signature Analyzer.

HP Company was the pioneer company in using signature analysis. The measurement window (gate), framed by the START and STOP signals generated from the CUT [5-6], must be unique and synchronized to all the nodes tested to produce signatures. The number of clock edges enclosed must be a constant for each test set up condition. Data must be synchronous and stable during the triggering edge of the clock. The SA based HP company requires operator with a high level of training. It is designed to act as a testing tool for HP cards, where the points to connect the START, STOP, and CLOCK signal is marked. For non-HP cards, it is required to specify the points on the card to connect the START, STOP and CLOCK signals. If the troubleshooter fails to determine the START and STOP signals from the CUT or the CUT has not have its own clock, the SA based HP Company fails to test the CUT properly. The gate has some situations that cannot open or close. If the gate does not open or close as expected, the measured signature is unstable. All these drawbacks are solved with the presented work in this paper that introduces a complete framework for the testing of the printed circuit boards (PCBs) of the digital integrated circuits.

In this paper, the system is composed of the test pattern generator (TPG), test response compactor based on the signature analysis. Test patterns, needed for the CUT, are generated either pseudorandomly or determistically. All control signals for the CUT and ATE system are generated either from the controller card or from the control port of the parallel port of the PC. The generated signatures are compared with the good one. The system represents an interactive system providing user (trouble-shooter) with easy access to an algorithmic model to detect and locate the faulty node(s) propagating their effect to different other nodes. The designed system is based on an operator with a less level of training. The system is designed to act as a testing tool for different digital circuit cards. It is not required to specify the points on the card to connect the START, STOP, and CLOCK signals as do in the signature HP analyzer [5-6].

3. CONCEPT OF THE SYSTEM DESIGN

The objective of the design of this new system is to make the generated signature repeatable (the gate is open for the same number of clock edges) and stable (stable

signature occurs when two adjacent signature measurements result in the same signature for all measurements). The controlling processing of the test cycle is required to achieve this objective. The stable approach is to control the opening and closing the gate to include same numbers of CLOCK edges. The DATA bit from a CUT is compacted. This insures repeatable and stable signatures (Fig. 5). The testing operation of our system is based on the rejection of the START and STOP connection. The gate with constant clocks is not dependent on the CUT. The test pattern generator (TPG) requires stimulating all nodes in the CUT. The stimulated node makes the SA compact all data bits on it into a *signature*. This signature contains all the information about the correct function of the device driving the node. By supplying known input test patterns to a CUT, unique signature can be generated at various nodes in the circuit. Measuring an incorrect signature during troubleshooting will accurately indicate an incorrect waveform for that node as long as the error appears in the data stream and clocks in the SA. The error will appear in the data stream only if the TPG generates the test pattern that detects the fault.

The CLOCK1, generated from CLOCK input, is used to synchronize and control the sample rate of the probe input DATA and the TPG as shown in Fig. 5, Fig. 6, and Fig. 7. So, the input DATA to the SA is processed every CLOCK1 cycle within the gate interval. Using the suitable triggering edges of the CLOCK1, the synchronization and processing operation are achieved. The gate can be accurately controlled by the control signals. It is generated for proper operation of the signature measurement. CLOCK1 is the gated clock inside the gate (99999 clocks). CLOCK2OS is the signal that clears the pseudorandom test pattern generator in the beginning of the gate. CLOCK4OS is the signal that clears the SA in the beginning of the gate (Fig. 6). The SA takes a new signature measurement. The signature from the previous measurement cycle remains displayed. After 99999 clocks, the gate is closed as shown in Fig. 7 and the SA stops running and displays the new *signature*.

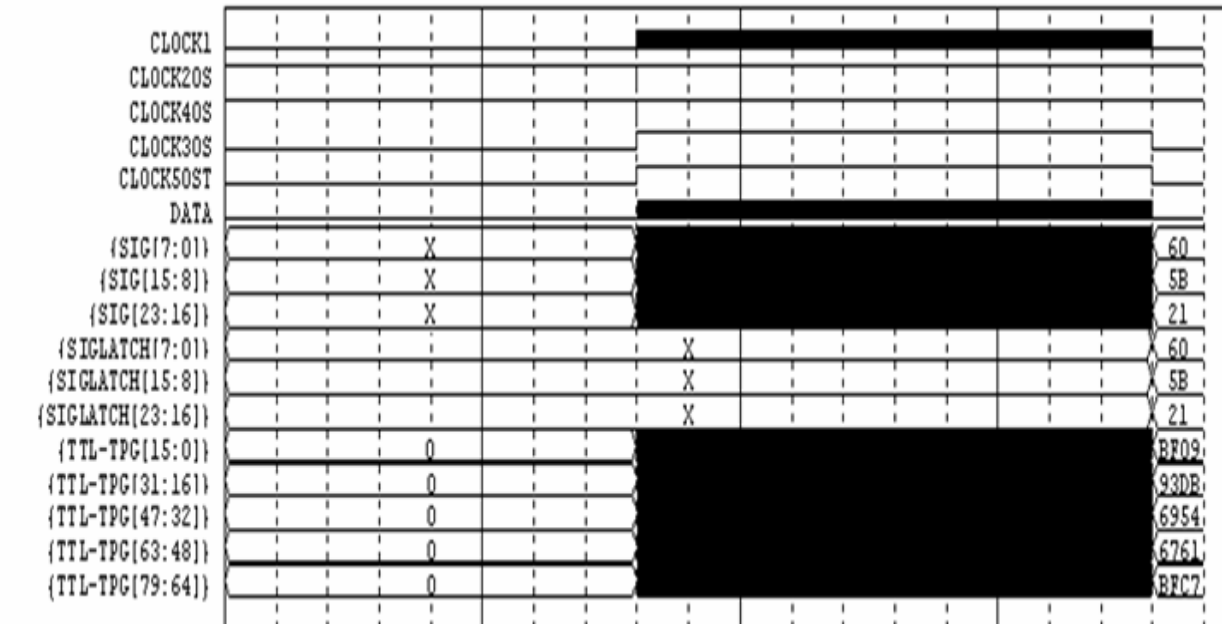

Fig. 5. Basic gate operation.

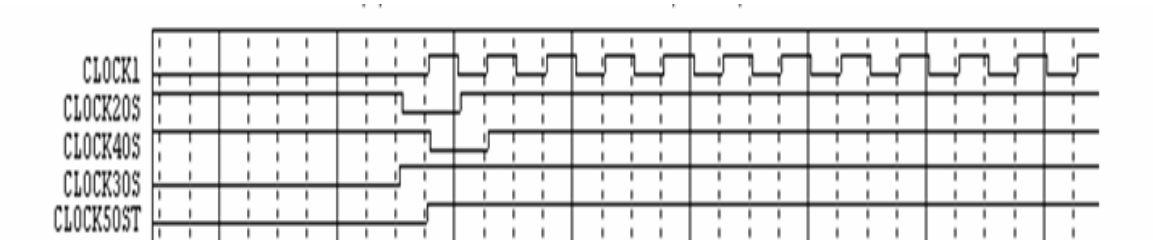

Fig. 6. Start gate window.

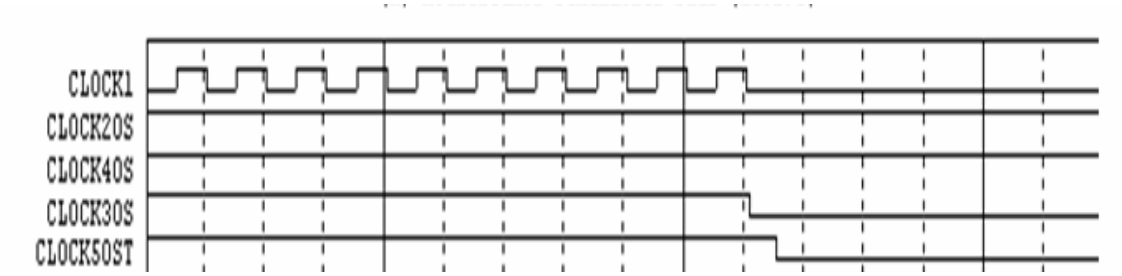

Fig. 7. Close gate window.

4. NEW AUTOMATIC TESTING SYSTEM

This paper presents a new automatic test equipment for fault detection and fault location to the net level on the printed circuit board (PCB), containing digital ICs. It performs the hot functional test for TTL and CMOS digital IC technologies. This system consists of two main integrated parts hardware, and software. The hardware part is equipped with all basic parts of the generalized ATE. The basic block diagram of the hardware part is shown in Fig. 8. It is composed of the TPG to exercise the CUT as a stimulus, the SA to compact the response of the measured net as a test response compactor, the timing controller synchronized with either internal or external CLOCK to control each part of the system accurately, and a simple test fixture that provides the system with an electrical and mechanical interface with the CUT. In addition, it is equipped with the interface unit to enable the automatic signature transfer to the PC and generate the deterministic test patterns as will be presented later.

4.1 Test Pattern Generator of the System

The hardware part of this system is equipped with a TPG as a stimulus. It is requires to get a TPG with high fault coverage. In this system, the TPG is based on either pseudorandom testing or deterministic testing. The TPG based on pseudorandom testing is ALFSR with primitive polynomial $1 + x^9 + x^{79}$ [8] and test length 99999 test patterns. This test length guarantees 95 % fault coverage of stuck-at fault model. The TPG based on deterministic testing is called deterministic TPG (DTPG). It has arbitrary test length according to the test patterns calculated from algorithmic methods. These algorithms support the detection of different fault models [9-11]. The software part with the interface unit is responsible to automatically transfer these calculated test patterns to the CUT. The output of both TPGs can drive different technologies such as transistor transistor logic (TTL) and complementary metal oxide semiconductor (CMOS) digital ICs as shown in Fig. 8.

This system has four basic modes of operation:

1. **Pseudorandom testing mode:** In this mode of operation, the TTL (CMOS) pseudorandom test pattern generator (PRTPG) is controlled by CLOCK1 and CLOCK2OS. It is run in either free running or one shot as will be explained next.
2. **Hybrid between pseudorandom testing and deterministic testing mode:** In this mode of operation, all required control signals are utilized as in the pseudorandom testing mode beside the utilization of the single deterministic test pattern generated from TTL (CMOS) DTPG.

3- Deterministic testing mode: In this mode of operation, the TTL (CMOS) DTPG is utilized with the deterministic CLOCK (DCLK) and deterministic RESET (DRESET).

4- Monostable multivibrator mode: In this mode of operation, the TTL (CMOS) DTPG is utilized in deterministic mode to generate all required deterministic test patterns to trigger the monostable multivibrator circuits. The probe connection is switch from signature probe to monostable multivibrator probe to measure the signature of the output of the monostable multivibrator. All required details about this new approach are presented in paper [15].

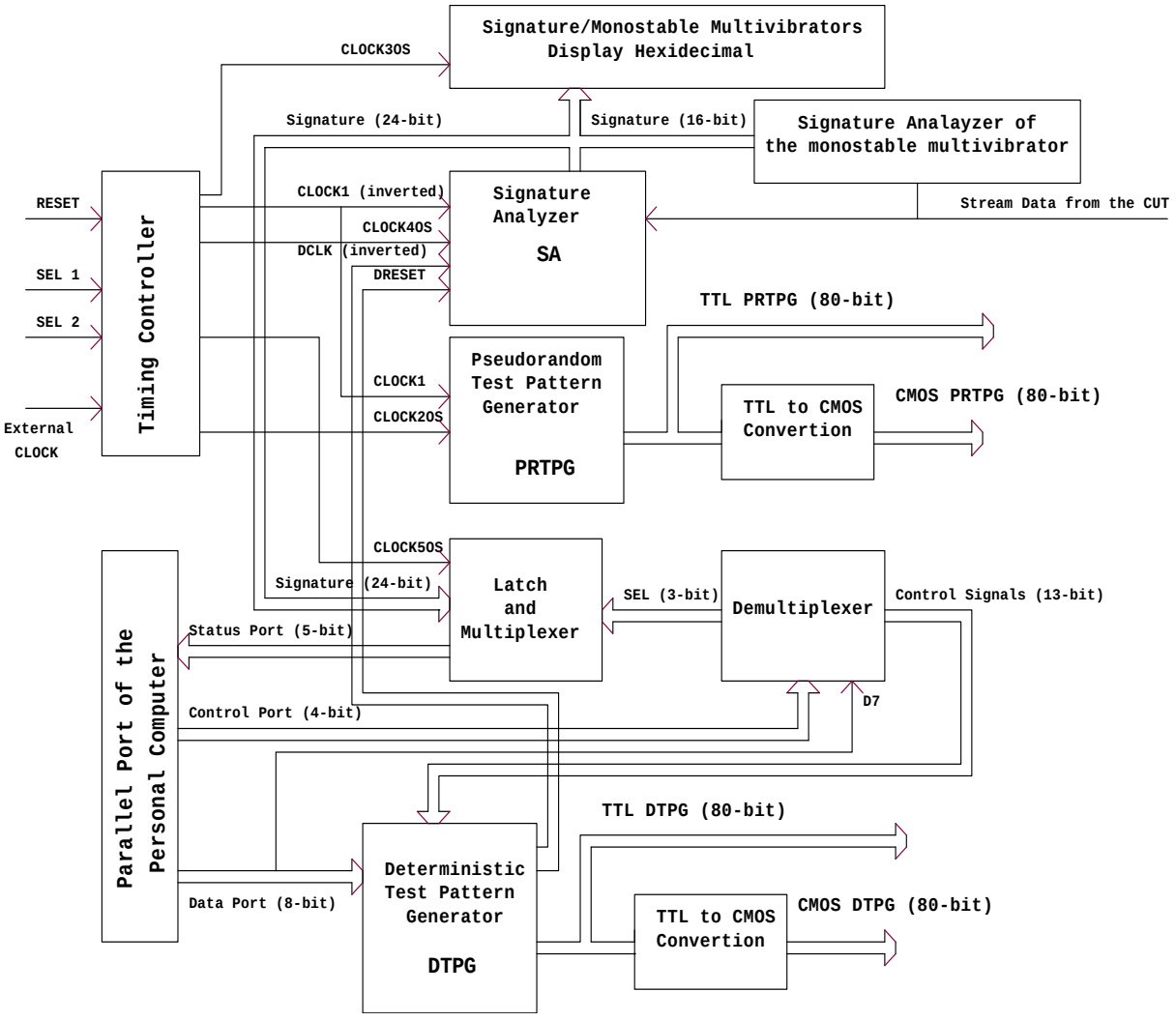

Fig. 8. Block Diagram of the basic architecture of the hardware part.

In pseudorandom testing mode, both internal and external clock are utilized using the SEL1. When SEL1 sets LOW, the internal clock which is 1 MHz is the synchronized clock. When SEL1 sets HIGH, the external clock is the synchronized clock. The system has master RESET. The GATE window is controlled by the control signal CLOCK3OS, which is asserted at the falling edge of CLOCK in starting of the gate and the closing of it. The PRTPG operates in two mode of operation; free-running mode and one-shot mode. In free-running mode, the gate is continuously opened and closed every 100 ms. The stability of the signature on the hexadecimal display ensures the proper operation. In one-shot mode, the gate is opened once and then it is closed. If you want to open the gate to calculate new signature, you need to press RESET. The switching between the free-running mode and one-shot mode is done using SEL2. When SEL2 sets LOW, the free-running mode is used. When SEL2 sets HIGH, the one-shot mode is used.

From Fig. 9, the CLOCK1, generated from CLOCK, is used to synchronize and control the sample rate of the probe input DATA and the PRTPG. So, the input DATA to the SA is processed every CLOCK1 cycle within the gate interval. In Fig. 10, the output of the PRTPG is asserted in the rising edge of CLOCK1 and the SA is asserted in the falling edge. This provides the test pattern, generated from the PRTPG, enough time to propagate through the ICs built in the electronic card before the acquisition of the DATA. The gate can be controlled accurately by timing controller card, which makes the generated signature stable, repeatable, and correct. CLOCK1 is the gated clock inside the gate window (99999 clocks). CLOCK2OS is the signal that clears the PRTPG in the beginning of the gate. CLOCK4OS is the signal that clears the SA in the beginning of the gate. After 99999 clocks, the gate is closed. Then, the outputs of the PRTPG has constant binary values as shown in Fig. 9 and Fig. 11 and the outputs of the SA has proper signature “67897A” which displays on the hexadecimal display.

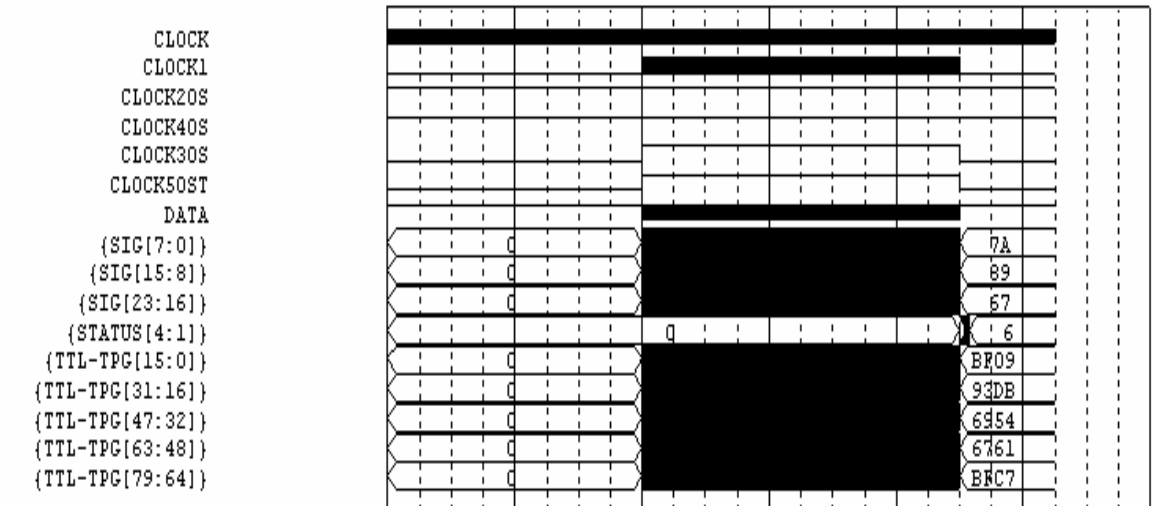

Fig. 9. Pseudorandom Testing mode.

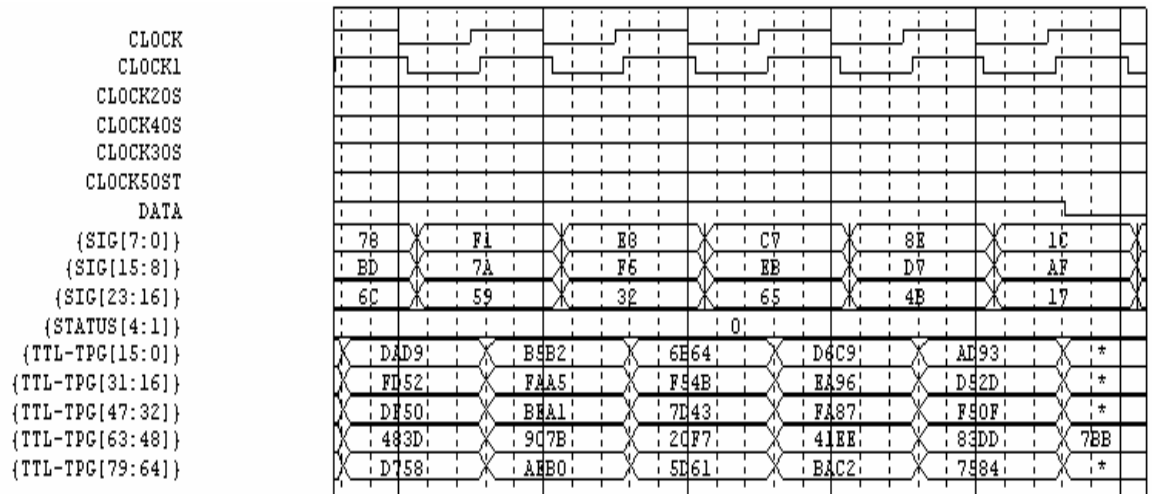

Fig. 10. Interaction between the TPG and SA.

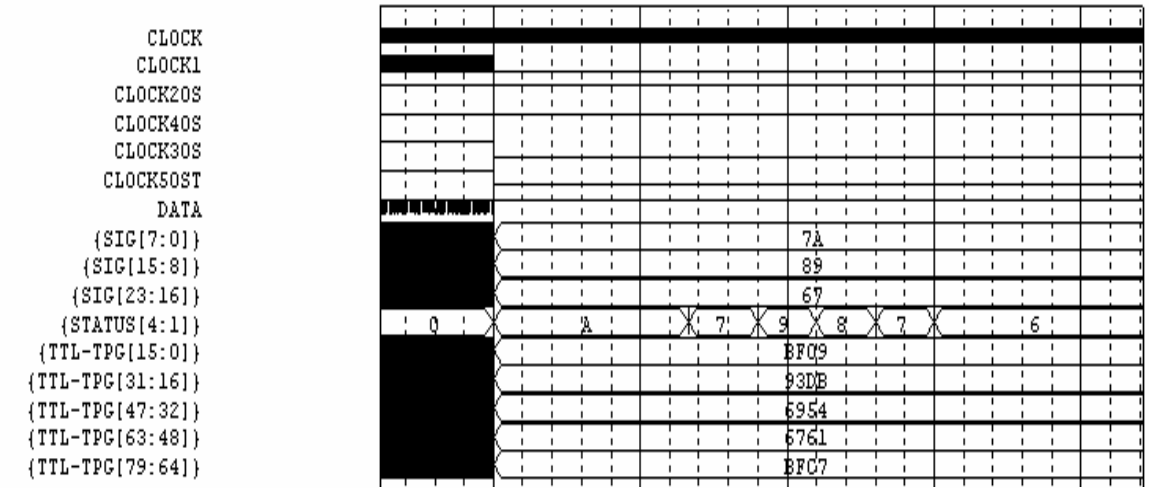

Fig. 11. Automatic transfer of the generated signature to the PC.

In the deterministic testing mode, the deterministic test patterns are generated from the personal computer through the parallel port (Fig. 8). The start, stop, and all control signals are generated from the personal computer. In this case, the number of the clocks inside the gate is variable depends on the required test patterns to deterministically exercise the CUT. According to the calculated test patterns from algorithmic methods such as path sensitization, D algorithm, PODEM, FAN, … so [9-11]. These algorithms support the detection of the stuck-at fault. Other algorithms support the detection of different faults such as bridging faults and stuck-open faults [9-11]. The calculated test patterns are stored in a file. The software part is responsible to retrieve the test patterns from this file and generates all required control signals to automatically transfer these patterns to the CUT through the DTPG. The number of the DTPG outputs is 80 pins.

From Fig. 12, the DCLK is used to synchronize and control the sample rate of the probe input DATA and the DTPG. So, the input DATA to the SA is processed every DCK cycle within the gate interval. The output of the DTPG is asserted in the rising edge of DCLK and the SA is asserted in the falling edge. This provides the test pattern, generated from the DTPG, enough time to propagate through the ICs built in the electronic card before the acquisition of the DATA. The interface unit, discussed

in section 4.4, can accurately control the gate. The designed of the interface unit eliminates glitches and provides stable operation. DRESET is the signal that clears the SA in the beginning of the gate. After the gate is closed, the proper signature is displayed on the hexadecimal display. The generated signature from either pseudorandom testing or deterministic testing is automatically transferred to the PC through the interface unit.

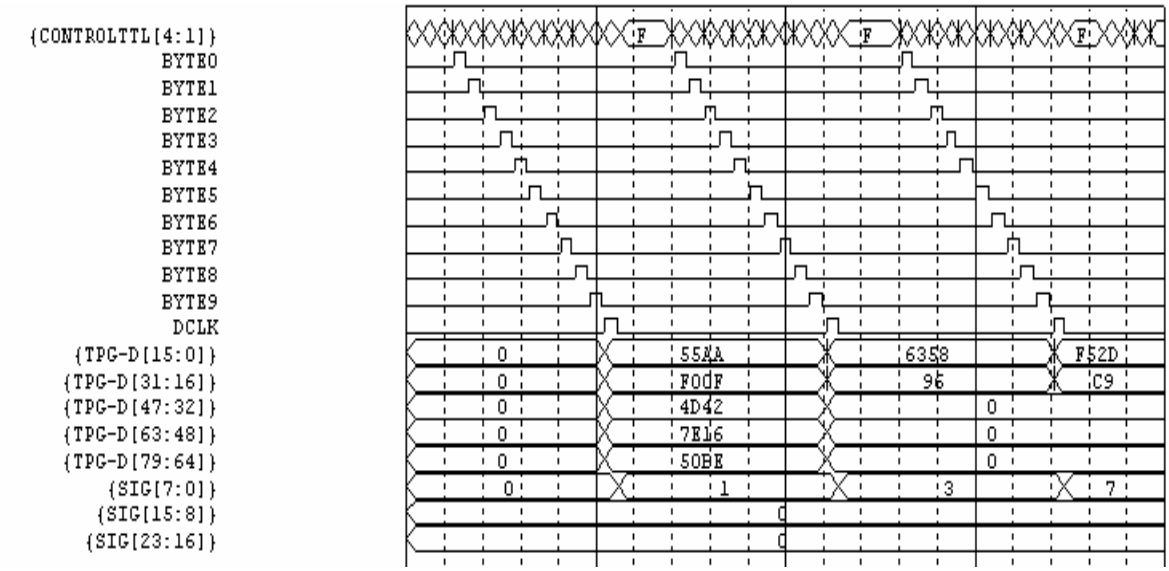

Fig. 12. Deterministic testing mode.

Some CUT inputs of the electronic cards needs fixed logic state and the other can be tested using the random binary signals. The hybrid between pseudorandom testing and deterministic testing mode is divided the CUT inputs into two sets. One set of CUT inputs that needs fixed logic state is connected to single deterministic test pattern generated from DTPG and the other set is connected to the PRTPG. The test patterns applied to CUT inputs is the concatenation of the PRTPG and DTPG.

4.2 Signature Analyzer

The SA is based on the LFSR circuit as shown in Fig. 1a. This analyzer has 23 stage with aliasing error probability = 2^{-23} = 0.000012 %. A 23-bit LFSR compactor is used in the implementation of the SA with primitive polynomial $1 + x^5 + x^{23}$ [8]. It supports TTL and CMOS logic level inputs. The measurement technique based on SA gives a 99.999988 % worst-case probability of detecting an error as long as the error appears in the data stream of that node, and the error bit(s) have been clocked in the SA.

The error will appear in the data stream only if the TPG generates the test patterns that detect the faults. The basic steps of this testing technique are *circuit generated stimulus* and *data compaction*. TPG is used to stimulate all nodes in the CUT. By supplying a known input test pattern to a CUT, unique signature can be generated at various nodes in the circuit. If the correct signature is repeatable, then measuring an incorrect signature during troubleshooting will accurately indicate an incorrect waveform for that node. Data compaction is achieved by probing a test node from which data is input for each clock within the time period of the gate. Bits of sequence being measured are summed in modulo-2 with the LFSR feedbacks. Input sequence

may be with different lengths but at the end of the measurement, only the signature is the residue of the SA.

From Fig. 8, the SA module has four input signals. CLOCK1 and CLOCK4OS are used in the pseudorandom testing mode. CLOCK1 that clocks the PRTPG at the rising edge clocks the SA at the falling edge. CLOCK4OS clears the SA at the start of each gate interval as shown in Fig. 6. In the deterministic testing mode, DCLK and DRESET are used. DCLK that clocks the DTPG at the rising edge clocks the SA at the falling edge as illustrated in Fig. 12. DRESET clears the SA at the start of the operation. At the end of the gate interval, the signature is displayed on the hexadecimal display. Each gate operation needs 200 ms. So, this operation repeats 5 times per second in the free-running mode and repeats once in the one-shot mode. The stability of the signature in the free-running mode on the display assures the repeatability and the good stability of the SA. Also, the signature is automatically transferred to the PC through the interface unit as will be shown in section 4.4.

4.3 Timing Controller

The timing controller module is responsible for the time synchronization of the system in the pseudorandom testing mode and the hybrid between pseudorandom testing and deterministic testing mode. This module has four input signals and five output signals. SEL1, SEL2, RESET, and External clock are the input signals. SEL1 is the signal that switches between the internal and external synchronized clock. The PRTPG operates in two mode of operation, free-running mode, and one-shot mode. The switching between the free-running mode and one-shot mode is done using SEL2. RESET is used to initialize the hardware system. It is used also in the one-shot mode that opens the gate once to calculate the new signature.

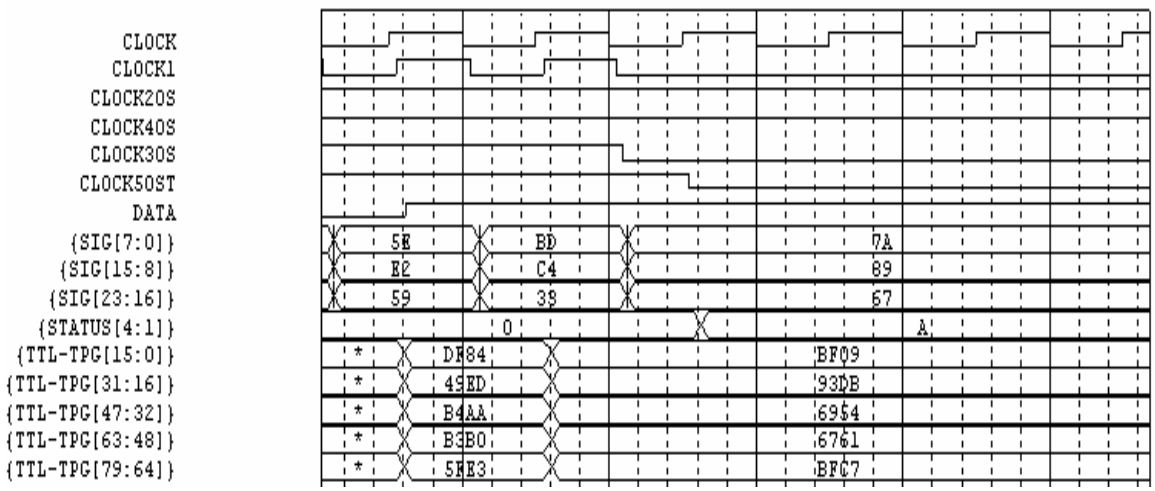

Fig. 13. CLOCK5OST is asserted half clock after the assertion of CLOCK3OS

CLOCK1, CLOCK2OS, and CLOCK4OS were explained carefully in section 4.1 and section 4.2. Fig. 6 and Fig. 7 illustrate their timing generation that assures the proper operation of the system through the testing cycle. CLOCK3OS controls the time interval of the gate. It is asserted at the falling edge of CLOCK in starting of the gate and the closing of it as shown in Fig. 6, Fig. 7, and Fig. 13. CLOCK5OST is the half clock shift version of CLOCK3OS as shown in Fig. 6, Fig. 7, and Fig. 13. It controls the time to properly latch the signature in the interface unit after the testing

processing finishes with half clock. In addition, CLOCK3OS is connected to hexadecimal display to display the signature only when CLOCK3OS is asserted low.

4.4 Interface Unit with the Personal Computer

The interface unit of this system is composite of the latch, a multiplexer module and a demultiplexer module interfaced through the parallel port of the PC. The parallel port is found commonly on the back of your PC as a D-Type 25 pin female connector. The parallel port is accessed via 3 consecutive 8-bit ports in the processor's I/O space.

- 8 output pins accessed via the *DATA Port*
- 5 input pins (one inverted) accessed via the *STATUS Port*
- 4 output pins (three inverted) accessed via the *Control Port*
- The remaining 8 pins are grounded

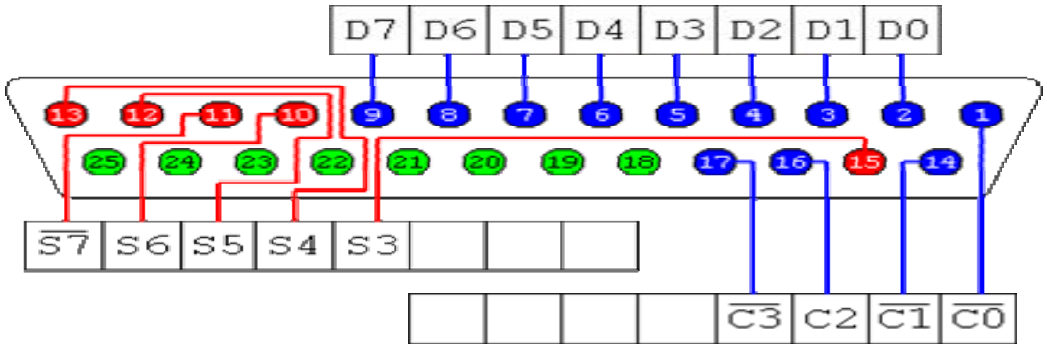

Fig. 14. 25-way Female D-Type Connector

From Fig. 8, the demultiplexer is accepted the four input signals from control port. The generated output signals from the demultiplexer are the encoded signals that control the proper operation of the DTPG module and the multiplexer. The control port and the demultiplexer generate all required control signals. To compensate the different delay of the signals of the control port that generates glitches in the outputs of the demultiplexer, D7 generated from data port is controlled to disable the demultiplexer in the time of the changing states.

The multiplexer is controlled to automatically transfer the signature to the PC after it is latched. Due to the limited number of the input signals of the status port, the signature is multiplexed as shown in Fig. 11. The software part of the system is used to receive this signature and generate all required control signals via parallel port for signature transfer and deterministic test pattern generation. The signature is stored in a file for fault detection and fault location processing.

4.5 The Software Part of the ATE System

The system is designed to make the overall timing control, generate test pattern to hot exercising the CUT, receive the response from the CUT, compact the response from a golden reference CUT in programming mode or from the malfunction CUT in the testing mode, and store the reference signature to a database file for that CUT. The software is designed to characterize CUT by transferring the circuit diagram to specified database files. It automatically makes the comparison between the reference signature and the measured one, carries out test analysis to determine the

source faulty node(s), and finally generate the test result reports. This system represents an interactive testing system providing the user (troubleshooter) with an easy access to an efficient algorithmic decision model for locating the source faulty node(s) propagating its (their) effect to different other nodes in the circuit making them behave as if they are faulty.

5. CASE STUDY

- Circuit under test: as shown in the Fig. 15.
- Installations : In this case, we have:
  a) 22 inputs to the first 22 output signal from the PRTPG.
  b) 48 nodes (VCC, GND, TTL-TPG0 – TTL-TPG21, TP1 – TP24)
- Get good signatures :
  The resulting signatures for 48 nodes are as in Table 1. These signatures are stored as the GOOD SIGNATURES in the corresponding file.
- Generate an artificial fault as shown in Fig. 15.
- Measure and store the signatures of the faulty circuits in the corresponding file.
- Run the comparison subroutine and analyze the output reports. Table 1 shows the Status Node Report of the fault.
- Test analysis and results :
  a) Look for the faulty output nodes.
  b) Using back tracing technique, determine the dependent nodes for each faulty node.
  c) Repeat step b, for the new faulty nodes.
  d) Continue back tracing until finding a faulty output node with all input nodes good "PASS", this IC is faulty.
- U3A, U3B, U4A are the source faulty node.

Table 1. Status node report of the fault

NODE	Good SIGN	Measured SIGN	Status	NODE	Good SIGN	Measured SIGN	Status
VCC	299BD5	299BD5	PASS	*TP1*	2BEC8B	2BEC8B	PASS
GND	000000	000000	PASS	*TP2*	68A07A	68A07A	PASS
TTL-TPG0	02775E	02775E	PASS	*TP3*	090602	090602	PASS
TTL-TPG1	413BAF	413BAF	PASS	*TP4*	7E8262	7E8262	PASS
TTL-TPG2	209DD7	209DD7	PASS	*TP5*	48C13E	48C13E	PASS
TTL-TPG3	504EEB	504EEB	PASS	*TP6*	67897A	67897A	PASS
TTL-TPG4	682775	682775	PASS	*TP7*	478208	478208	PASS
TTL-TPG5	3413BA	3413BA	PASS	*TP8*	47997E	47997E	PASS
TTL-TPG6	5A09DD	5A09DD	PASS	*TP9*	0930E9	0930E9	PASS
TTL-TPG7	6D04EE	6D04EE	PASS	*TP10*	34E6DC	34E6DC	PASS
TTL-TPG8	768277	768277	PASS	*TP11*	41C3B7	41C3B7	PASS
TTL-TPG9	7B413B	7B413B	PASS	*TP12*	3C5BBE	61A7AC	FAIL
TTL-TPG10	7DA09D	7DA09D	PASS	*TP13*	7D3972	299BD5	FAIL
TTL-TPG11	3ED04E	3ED04E	PASS	*TP14*	34A313	52DAEE	FAIL
TTL-TPG12	5F6827	5F6827	PASS	*TP15*	4E83B3	496F9F	FAIL
TTL-TPG13	6FB413	6FB413	PASS	*TP16*	3449B5	299BD5	FAIL
TTL-TPG14	37DA09	37DA09	PASS	*TP17*	6F7C6A	543B48	FAIL

NODE	Good SIGN	Measured SIGN	Status	NODE	Good SIGN	Measured SIGN	Status
TTL-TPG15	5BED04	5BED04	PASS	*TP18*	010EDD	2C6753	FAIL
TTL-TPG16	6DF682	6DF682	PASS	*TP19*	5498C5	5498C5	PASS
TTL-TPG17	76FB41	76FB41	PASS	*TP20*	452983	452983	PASS
TTL-TPG18	3B7DA0	3B7DA0	PASS	*TP21*	70F39E	244BE3	FAIL
TTL-TPG19	5DBED0	5DBED0	PASS	*TP22*	033CF6	299BD5	FAIL
TTL-TPG20	6EDF68	6EDF68	PASS	*TP23*	543873	000000	FAIL
TTL-TPG21	376FB4	376FB4	PASS	*TP24*	1729AC	1729AC	PASS

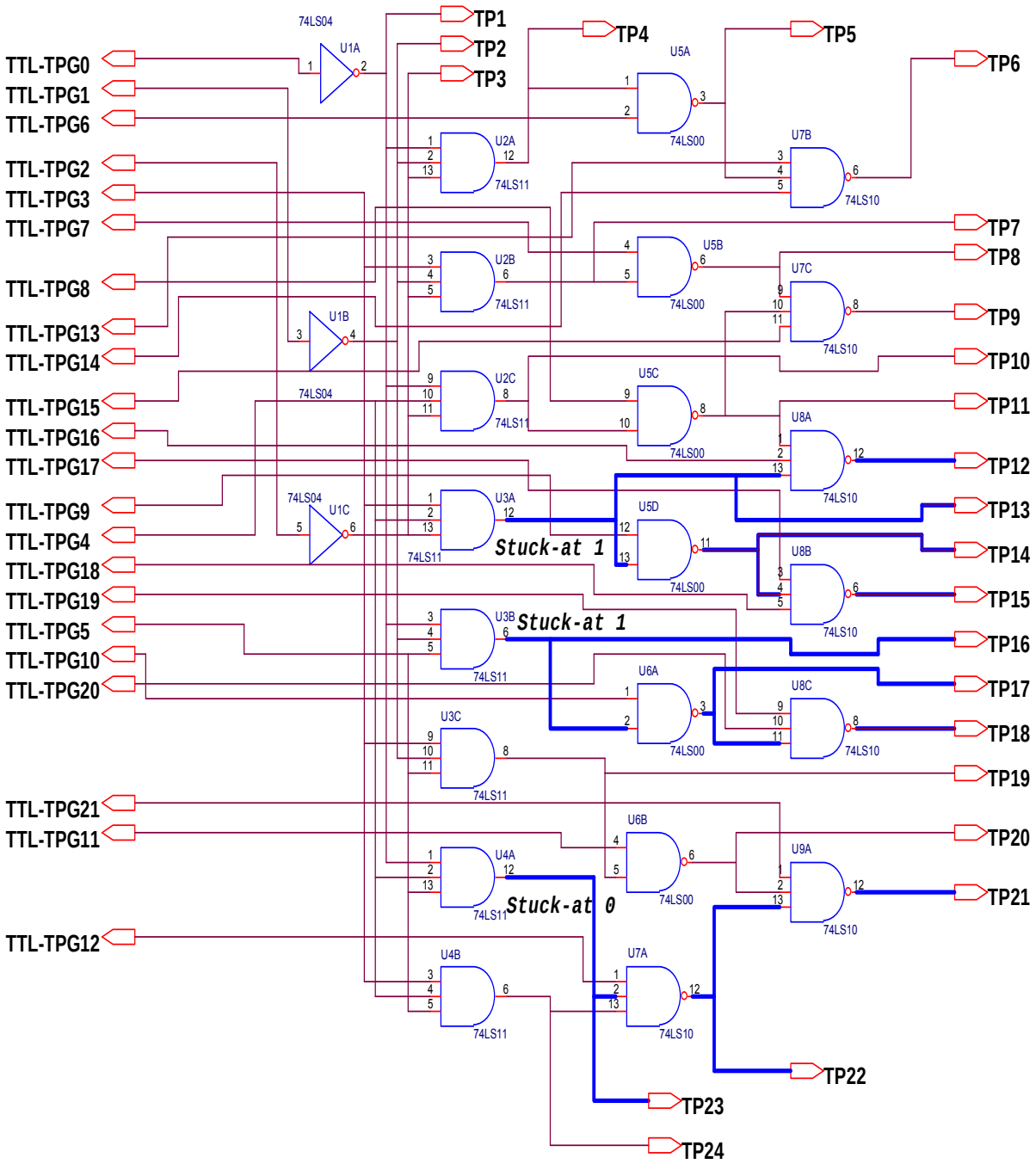

Fig. 15. Circuit under test.

6. CONCLUSION

In this paper, the new automatic testing approach for digital ICs is developed. This system is capable of performing the following main tasks: CUT characterization, CUT exercising, and fault detection and source fault location. This system has test pattern generation scheme that generates test patterns pseudorandomly, determistically or hybrid between pseudorandomly and determistically. The response of the CUT is evaluated by the signature analysis technique. All control signals are generated either from the timing controller card in the pseudorandom mode or from the control port of parallel port of the PC in the deterministic mode. In this new design approach, direct interface between the CUT and the computer, as well as accuracy and speed of the testing operation are achieved. The accuracy is guaranteed by controlling opening and closing of the gate, where precise signature is generated. The gate needs to open only as long as it stimulates the measured node through its functional states. Then, the SA compacts data on a node into a signature that contains all information about the correct function of the device driving the node.

The measured signatures in both modes are compared with the good one. The fault detection and location are achieved as long as the error exists in the waveform and the stimulus causes it to be there. The system is designed to act as a testing tool for different digital circuit cards. This system allows the trouble-shooting of digital circuits in the different fields.

This system performs:

- Hot functional test to the printed circuit board (PCB) level.
- Fault detection and source fault location to the node level in the PCB.
- Testing TTL, and CMOS, digital circuit technologies.
- Storing good signatures in a computer file in the programming mode.
- Minimum programming preparation time and reduced time to test and diagnose
- Minimum adaptation between the system and the CUT, (only edge connector).
- Test pattern generation
  a- The test pattern generator based on pseudorandom testing operates in two mode of operation: Free-running mode and one-shot mode.
  b- The primitive polynomial of this generator is $1 + x^9 + x^{79}$.
  c- Test length is 99999 test patterns in the case of pseudorandom testing.
  d- Test length is arbitrary in the case of deterministic testing.
  e- Derive up to 80-input pins of the CUT in the case of deterministic testing.
  f- Great Fault coverage.
  g- Either internal or external clock is utilized.
  h- Has master reset for CUT.
- Signature Analysis
  a- Use linear Feedback Shift Register (LFSR) circuit as the signature analyzer.
  b- Number of stages is 23 stages.
  c- Aliasing error probability = 2^{-23} = 0.000012 %.
  d- Generate the signatures of different nodes and storing them for post-processing by the PC.
  e- The signature analysis measurement gives a 99.999988 % worst-case probability of detecting an error as long as the error appears in the waveform for that node, and the error bit(s) clocks into the signature analyzer.

- The interface between the ATE system and the computer. This interface depends mainly on the parallel port interface, which produces control signals to control the testing cycle in the deterministic testing mode.

REFERENCES

[1] A. Cstover, “Automatic Test Equipment”, McGraw-Hill inc. (1984).

[2] A. Seddik, “Microcomputer Based Automatic Testing System through Signature Analysis”, M.Sc. Degree, MTC, Cairo, (1992).

[3] Mohamed H. El-Mahlawy, “Automatic Measurment of Digital Circuits”, M.Sc. degree, MTC, Cairo, 1995.

[4] M. El Said Gohniemy, S. Fadel Bahgat, Mohamed H. El-Mahlawy, and E. E. M. Zouelfoukkar, “A Novel Microcomputer Based Digital Automatic Testing Equipment using Signature Analysis”, IEEE conference on Industrial Applications in Power Systems, Computer Science and Telecommunications, Bari, Italy, 13-16 May (1996).

[5] Application note 222-2, “Application Articles on Signature Analysis”, Hewlett-Packard, (1977).

[6] Application note 222-4, “Guidelines for Signature Analysis Understanding the Signature Measurement”, Hewlett Packard, (1977).

[7] Paul H. Bardell, Willian H. McAnney, Jacob Savir, “Built-In test for VLSI: pseudorandom techniques”, John Wiley and Sons, (1987).

[8] V. N. Yarmolik, “Fault Diagnosis”, John Wiley and Sons, (1990).

[9] Miron Abramovici, Melvin A. Breuer, Arthur D. Friedman, “Digital systems testing and testable design”, IEEE PRESS the Institute of Electrical and Electronics Engineers, Inc., New York, (1990).

[10] Rochit Rajsuman, “Digital hardware testing: Transistor-level fault modelling and testing”, Artech House, (1992).

[11] Parag K. Lala, “Digital circuit testing and testability”, Academic Press, (1997).

[12] Thomas W. Williams, Wilfried Daehn, Matthias gruetzner, and Cordt W. Starke, “Bounds and analysis of aliasing errors in linear feedback shift register”, IEEE transactions on computer-aided design, Vol. 7, NO. 1, pp 75-83, Jan. (1988).

[13] Mohamed H. El-Mahlawy, “Pseudo-Exhaustive Built-In Self-Test for Boundary Scan”, Ph.D. thesis, Kent University, U.K., (2000).

[14] Janusz Rajski, Jerzy Tyszer, “Arithmetic Built-In Self-Test for Embedded Systems”, Prentice Hall PTR, (1998).

[15] Mohamed H. El-Mahlawy, “A novel testing method for monostable multivibrators”, 5th International Conference of the Electrical Engineering, Military Technical College, Egypt, pp. EM-6-1- EM-6-11, May. (2006).

[16] Sherif Anas Mohamed, “In-Circuit Testing for Electronic Board”, M.Sc. degree, MTC, Cairo, (2006).

[17] Paul H. Bardell, Willian H. McAnney, Jacob Savir, “Built-In test for VLSI: pseudorandom techniques”, John Wiley and Sons, (1987).