Military Technical College
Kobry Elkobbah,
Cairo, Egypt

8th International Conference
on Aerospace Sciences &
Aviation Technology

TOWARDS A NEARLY – MINIMAL TEST FOR MICROPROGRAMMABLE-MICROPROCESSORS A BIT – SLICED CASE STUDY

M. A. EL- LITHY *

ABSTRACT

Conventional testing methods of digital systems are not practical for microprocessors due to:

- Their high density of integration (VLSI) , lack of information about their internal physical structure , internal test points are not accessible , impossibility of inserting new test points in an integrating chip with a limited number of external leads (pins) and due to the nonprecision of hardware failure modes of new technologies such as CMOS,bipolar, ...etc.
- Moreover, the conventional test methods have two serious limitations concerning their application for the verification of LSI circuits:
- They consider the SSI components (gates, flip-flops, ...) as the primitive test elements.
- They apply fault models based primarily on the s-a-0 & s-a-1 model which associates faults to the connections between those basic elements.

Bit-sliced microprocessors represent an important class of LSI components (microprogrammable). A microprogrammed system can be generally tested by a suitable microprogram added to its normal operation microprograms .

This paper presents a methodology for generating a test microprogram for one of the recent powerful bit-sliced microprocessors (the superslice Am 2903).The test procedure is based on a fault model that takes into account a large variety of faults encountered with such microprocessors . The fault model is formulated at a higher level in terms of functional modules such as the ALU, memory and multiplexers. The derived test is nearly – minimal and its execution does not speed down the system operation

KEY WORDS

Computer reliability, Fault – tolerance, Micro diagnostics, Bit – sliced Microprocessors

---

*Associate Professor, Dpt. Of Elect Eng,.BHIT (Benha High Institute of Tecnology) Benha,Egypt

1-INTRODUCTION

The urgent need for digital systems reliability and the monotonic increase of implementing microprocessors in up– to-date designs justify the necessity for thorough and efficient testing methods for microprocessors. Testing of microprocessors would be envisaged during design, within the useful life period of the system operation, during scheduled maintenance and when inspecting new components (spare parts).

A microprocessor may be viewed as a circuit with very great complexity for which the classical testing methods of digital systems are no longer practical due to:

- inaccessibility of internal testing points and limited number of pins available for accessing the internal structure .
- hardware physical failure modes are not defined for new technologies such as the bipolar technology.
- limitations of conventional testing methods as they recognize only small devices such as gates and flip-flops and moreover , they use low level fault models such as stuck-at-zero (s.a.0.) and (s.a.1.) .
- the number of gates on a chip is very large in such a way that it will be very tedious to construct a comprehensive test set using such testing methods.

2-TESTING OF MICROPROCESSORS

2-1 Monolithic Microprocessors

They are those microprocessors (μ P's) where both control and data processing units are integrated on the same chip such as M6800, Intel 80386,.....Several research works have considered the problem of testing such μ P's [1,2] . They did not take into account the faults due to interaction between both control and data processing units .The control unit (CU) is used for testing the data processing unit (DPU) and is assumed to be fault – free (or separately tested at a higher level) which is not practically justified .

2-2 Bit-sliced Microprocessors

They differ mainly in that the control unit (CU) is not integrated on the same chip . Their architectures are microprogrammable.

A general test procedure for an LSI component may comprise three phases:

a - Parametric testing: checking of electrical parameters , e.g. voltages, currents ,.....

b - Dynamic testing : checking of propagation delays and response times under normal operating conditions

c - Functional testing : to check its logical behavior.

Generally, an IC is submitted to tests (a,b) at the end of its production line. It remains to test it at a functional level to check its adequate performance . This is quite justified for a bit-sliced μP as its architecture is synchronously micro – controlled at different points of its functional structure A change of one or more of its parameters (DC or time parameters) will affect the normal execution of at least one microfunction which would result in an erroneous output.

Therefore , functional testing is considered as a necessary and sufficient test for checking the proper state of a bit-sliced μP. Otherway – said , a bit-sliced μP performing all its microfunctions correctly , would be undoubtedly failure free.

A previous work ,in this aspect [3] has treated a bit – sliced cell of 1- bit and assumed that the same test generation method would be equally applicable to any slice of m- cells . This is not quite justified as it ignores the interaction effect between cells of one slice

3-FUNCTIONAL MODEL OF μP

At the functional level , a bit- sliced μP is considered as an assembly of several functional modules such as MUX, ALU, ... etc . Such modules are treated as black boxes which are described by functional tables provided by the manufacturer . This consideration is quite reasonable since their physical structure is not known and they are not accessible through the external leads of the chip . For derivation of a test program , the real structural diagram is considered the functional model of the μP as given in Fig.1 .

4-FAULT MODEL OF μP

The present paper deals with a real functional structure of one of the commercially – available and widely used bit sliced μP's (4-bit Am2903). The derived fault model is based on a functional level approach which considers the μP as an interaction of smaller functional modules (**MUX , ALU , ...**) of which the fault models are specified and presented . Such fault models take into account different physical failures (stuck – at , bridging , change of **DC** or time parameters , couplings , ...). Under the effect of a physical failure , a functional module will perform one or more of its functions inadequately. Failure modes of functional modules are defined in the following :

4-1 Input / Output leads :

F1 : one or more leads are s-a-o/1
F2 : one or more leads form a wired- OR , wired – AND as a result of shorts or spurious coupling

4-2 Multiplexers :

F3 : an incorrect selection
F4 : double selection where the output is a wired – OR or wired- AND of both sources

4-3 Buffers :

Functionally , a buffer is a switch circuit . Therefore :
F5 : an incorrect switching function

4-4 Shift Modules :

A shifter realizes a data shift of one position to the right , to the left or affects no shift . A shifter is generally realized by four multiplexers , one for each bit , with two buffers at its both sides. Therefore :
F6 : reversed shift sense
F7 : one or more MUX fail to function properly
F8 : one or more buffers perform uncorrectly

4-5 ALU :

It performs 16 different functions . Therefore ,
F9 : a function , other that instructed , is performed
F10: the ALU operates in a bit – wised sense... i.e. it allocates different Functions for different data bits
F11: one or more status signals are faulty

4-6 Data Storage :

It is a RAM of (16×4) memory cells. The following failure modes are proposed.
F12: one or more of cells are s-a-o/1
F13: one or more of cells fail to undergo a transition (0 ⟶ 1) or (1 ⟶ 0)
F14: two or more cells are coupled i.e. a transition in one cell (say i) changes the state of the other (or more) cells (say j , k , ...) .

4-7 Latches and Register (Q) :

may be treated as a one-word memory . Therefore , the failure modes F_{12-14} are valid

4-8 Instruction Decoder :

It elaborates the control commands for different modules as instructed by (I_{0-8} , MSS , LSS , IEN)
F15: an incorrect decoding , leading to the execution of an instruction other that instructed .
F16: an illegal instruction (not among the instruction set) is executed.
F17: improper programming of the slice position in the array of processors.

4-9 Clock Lines :

Failure of clock lines are covered by the fault model of data storage . It is noticed that the proposed fault model provides a complete coverage for the most – likely physical failures . Moreover , a single fault assumption is not restricted at this functional level of modeling .

5- TEST GENERATION PROCEDURES

Test generation is based on the analysis of the circuit functional diagram . The general strategy of test is to define the test vectors and store them in a PROM memory. Each vector comprises , besides the instructions, control commands and direct inputs , the expected correct response of the fault – free microprocessor. The defined test sequence is then applied to the processor under test and the obtained response is compared to the stored correct one . Any mismatch between them indicates the existence of some failures in the tested processor . Fault localization is not of our interest since the repair is not possible . Test procedure for each functional module is given in the following .

5-1 Memory Module

A memory functional test must be able to check for :
- s – a – 0/1 at each memory cell .
- the ability of each cell to undergo transitions (0 $\longrightarrow$ 1)& (1 $\longrightarrow$ 0)
- coupling between each pair of memory cells .

The general test procedure is based on memory partitioning into equal parts and then check for coupling between cells of both parts . Thereafter, each part is repartitioned into two parts and the same procedure continues up to the level at which each part cantains only one memory word.The test procedure for any two equal parts is given in the following table :

Step	Initial	1	2	3	4
Part (1)	0	↑	1	↓	0
Part (2)	0	0	↑	1	↑

For memory having n – words, the total sequence necessary is given by :
$N P = 4n . \log_2 n$
In our case , n = 16 ⟶ N P = 256
To check for coupling between cells of one memory word , a sequence of 8 test patterns has been elaborated for each word i . e 128 additional test steps are required for the memory of 16 word.

Step / Word bit	Initial	1	2	3	4	5	6	7	8
0	0	↑	1	↓	0	↑	1	↓	0
1	0	↑	1	↓	0	0	↑	1	↓
2	0	0	↑	1	↓	↑	1	↓	0
3	0	0	↑	1	↓	0	↑	1	↓

The total number of test steps is :

$$N_T = N_P + 128 + N_{init} = 256 + 128 + N_{init} = 384 + N_{init}$$

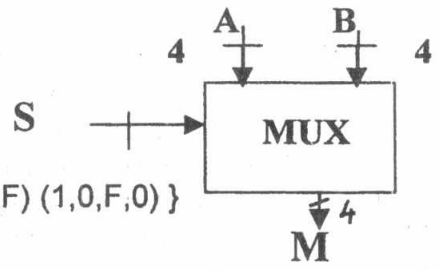
Where : N_{init} : number of steps necessary for initialization of the memory

5-2 Q-Register

It is treated as a one – word memory , for which a sequence of 10 test steps has been elaborated

5-3 Multiplexers

A test sequence of 4 test patterns has been found sufficient for detecting the failure modes defined by the fault model



$T = (T_0 , T_1 , T_2 , T_3) = \{ (0,0,F,F) , (0,F, 0,0) (1,F,0,F) (1,0,F,0) \}$
where $T_i = (S_i , A_i , B_i , M_i)$

5-4 ALU

A test sequence of 44- tests has been elaborated . This test sequence is sufficient to verify the correct execution of ALU function (fo,....., f_F) and to avoid ambiguities in the test results . As an example , the 4- steps test given in the following table is capable of testing and distinguishing between the logic functions on the basis of their output sequences .

Test	Operands		OR	NOR	AND	NAND	EX-OR	EX-NOR	R.S
	S	R	f_F	f_D	f_C	f_E	f_B	f_A	f_G
To	O	O	O	1	O	1	O	1	O
T1	O	1	1	0	0	1	1	0	0
T_2	1	0	1	0	0	1	1	0	1
T_3	1	1	1	0	1	0	0	1	0

5-5 Destination Control

The processor performs 16 – different destination control functions . Based on the same criterion (verification of adequate functioning and distinguishability between different functions) , a test sequence of 22- test patterns has been elaborated .

5-6 Special Functions

Nine special functions are assigned to the processor through the analysis of their mechanisms and the most likely failure modes of the processor .
A test sequence of 20-test patterns has been derived and the results have proved its ability to verify and check for the correct functioning .

5-7 Instruction Decode

Failure modes F 15 , F 16 are implicitly covered through testing of other modules while F 17 is tested by a sequence of 4-test patterns which verifies the proper programming of the slice according to its position in the array of processor .

6- MINIMZATION OF TEST SET

The general rules are :
a- Start – small approach
b- Using results of the precedent test stored in the RAM , as operands for the present test .
c- Peculiarities of bit – sliced architecture are well – exploited for test minimization .
Among them , we mark :-

* two – port RAM , allowing double accessing of memory
* two input / output ports (Y,DB)
* six status signals (G,P,N OVR,Z,Cn+ 4)
* two ports of direct data inputs (DA,DB)
* parallel access of shift modules (ALU,Q)

d- A test matrix M_T is established , associating each test with the activated functional modules, functions and failure modes as shown in Fig.2 .
Tests T_k , T_{k+1} described in M_T are represented diagrammatically in Fig3 .

e- The test sequence is minimized by merging of separate tests to a minimal set realizing a coverage for the prescribed failure modes . As a result , a sequence of 256 – test patterns has been obtained .

7- CONCLUSION

This paper provides a test sequence for testing the super bit – sliced processor (Am 2903) . The derived test is nearly – minimal (256 test patterns) and provides fault coverage for the processor failure modes . The derived test set may be applied for testing the microprocessor slice in one of the following cases
a- Isolated processor , as a new component or spare part . A proposed scheme is given in Fig .4
b- During a maintenance process (periodic or preventive)
c- Concurrently during its normal operation (micro-diagnostics)

The derived test set is applied in a microprogrammed control system for on-line testing of a μp operating with 10MHZ clock. The time required was
26 μ sec which never speeds-down the system performance .

An important aspect of the derived test set to be high-lightened is its applicability to an array of processors .
Only the length of test vectors has to be adjusted according to number of slices in the array .

In conclusion, we must stress on the fact that microprogrammable processors have been widely used in the implementation of modern control system and especially the dedicated ones such as distributed and parallel processing systems, high-speed algorithms systems (FFT, ...) , fault-tolerant systems , DAS, ...etc [5] , [6] , [7]
Thus, such derived test proves necessary and useful for concurrently verifying the adequate performance of bit-sliced microprocessors without having any impact on the system operation speed [8] .

8- REFERENCES

[1] ROBACH, C., SAUCIER,G and AL EONARD ,C
"Microprocessor system Testing; a review and future prospects"
Euromicro Journal 5.1979 , pp31-37

[2 THATTE , S.M. and . ABRAHAM , J.P.
"Test generation for general microprocessor architectures"
Digest symp. , FTC-9,1979 , PP 204-210

[3] SRIDHAR,T. and HAYES , J.P.
"Testing Bit-Sliced Microprocessors"
Proceeding of FTCS-9, 1979 , PP 211-218

[4] AMD , " The Am 2900 Family data book "
Advanced Micro Devices, 1978

[5] AHSON, SI . " Microprocessors with Applications in Process Control
Tata Mc Graw-Hill , New Delhi, 1984

[6] LIEBOWITZ , B.H. and CARSON,J.H. " Multiple processor systems
for Real – Time Applications" , Prentice-Hall, 1985

[7] EL-REWINI,H. and LEWIS,T.G. "Distributed and parallel
computing" , MANNING publications Co. , 1998

[8] HOCKNEY,R.W and JESSHOPE, C.R " Parallel Computers 2;
Architecture, Programming and Algorithms" IOP Publishing Ltd,
1988

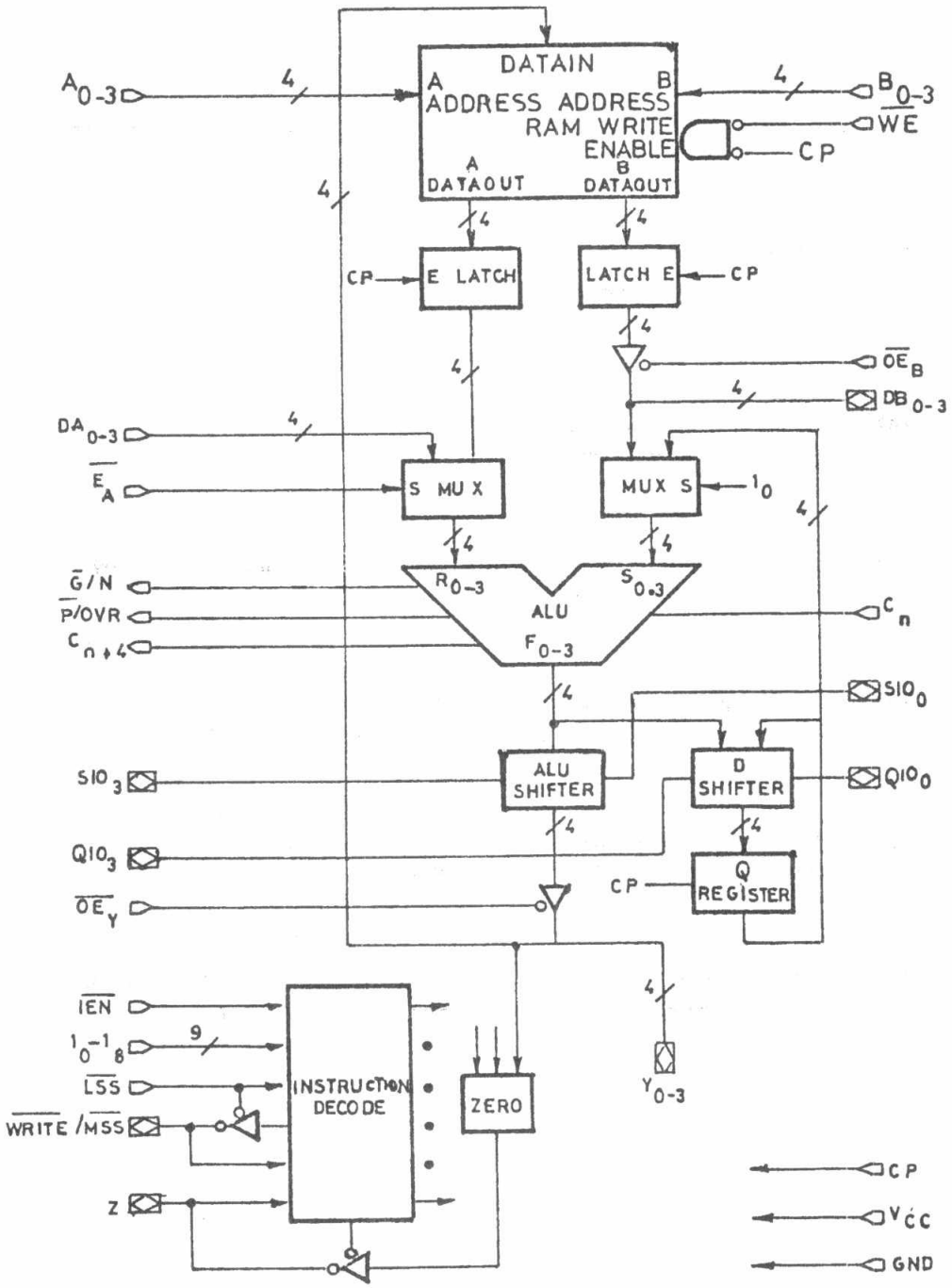


Fig. 1 Microprocessor Functional Model

Test	I/0 Lends		MUX (R)		MUX (S)		Buffer DB Y		ALU Shifter			Q Shifter			ALU	ALU			Destinat Control	RAM Memory			Q Register			Instruction Decode			Ex. & F	Ram (Q) Operation
	F1	F 2	F 3	F 4	F 3	F 4	F 5	F5	F 6	F 7	F 8	F 6	F 7	F 8	Executed Function	F 9	F 10	F 11	Executed Function	F 12	F 13	F 14	F 12	F 13	F 14	F 15	F 16	F 17		
T_O	×							×		×			×		F 8	×		×	D 7							×				Initialize Mo (0) Initial . Q
T_1 . . .	×		×	×	×	×	×	×		×			×		F 4	×		×	D 6	×			×			×	×			Read M_o (0) Read Q (0)
T_K	×	×	×	×	×	×	×	×		×			×		F E	×		×	D7	×	×		×			×	×			Read Mi M_j Write M_j (0) 0 → Q Read M_i M_j
T_{K+1}	×	×	×	×			×	×		×			×		F 6	×		×	DC	×			×			×	×			Write Mi (0)

Fig . 2 Test Matrix – A Segment .

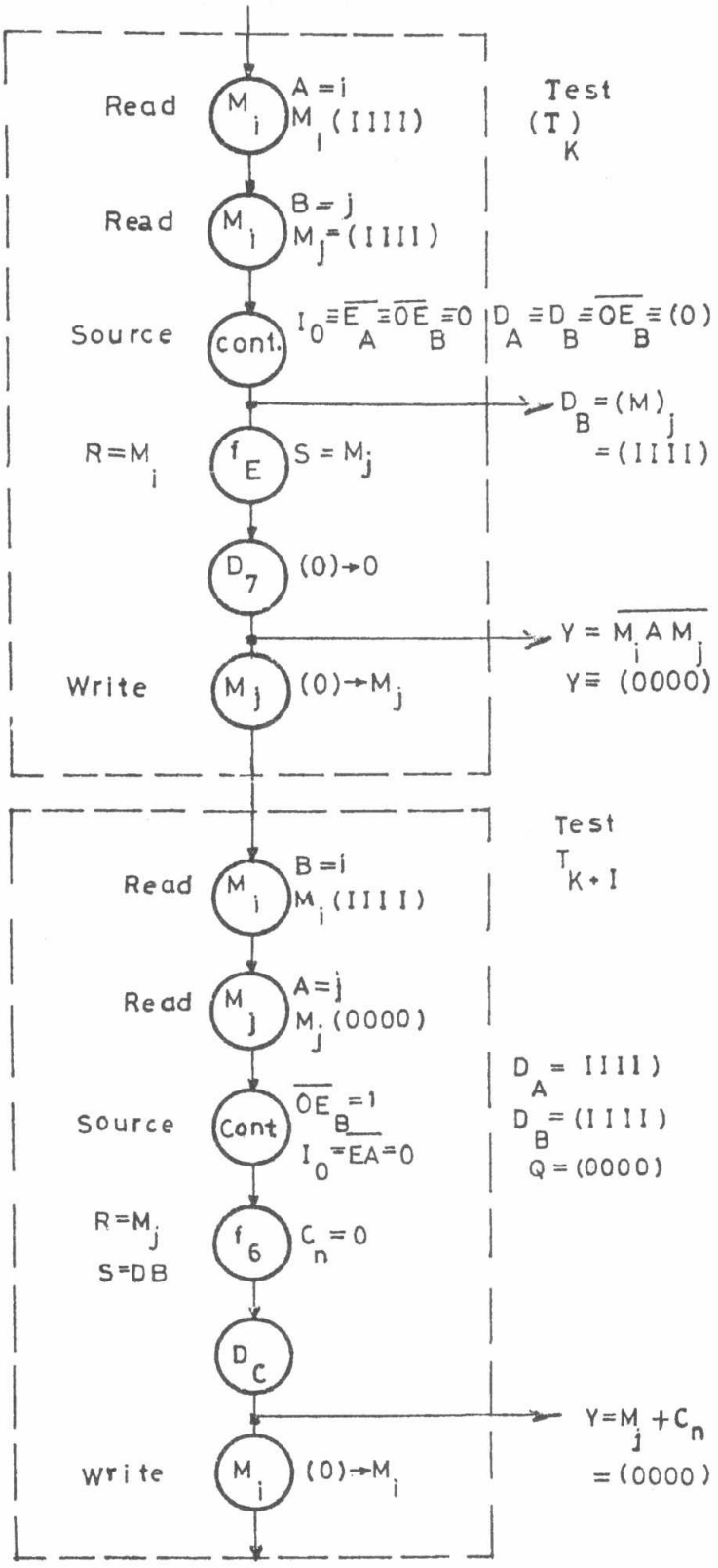

Read
A = i
Test
(T) K
Read
B = j
Source
cont.
R = M i
S = M j
D 7
(0) → 0
Write
(0) → M j
Test
T K + I
Read
B = i
Read
A = j
Source
Cont
R = M j
S = DB
f 6
C n = 0
D C
Write
(0) → M i
Y = (0000)
D A = IIII)
D B = (IIII)
Q = (0000)
Y = M j + C n
= (0000)

Fig. 3.

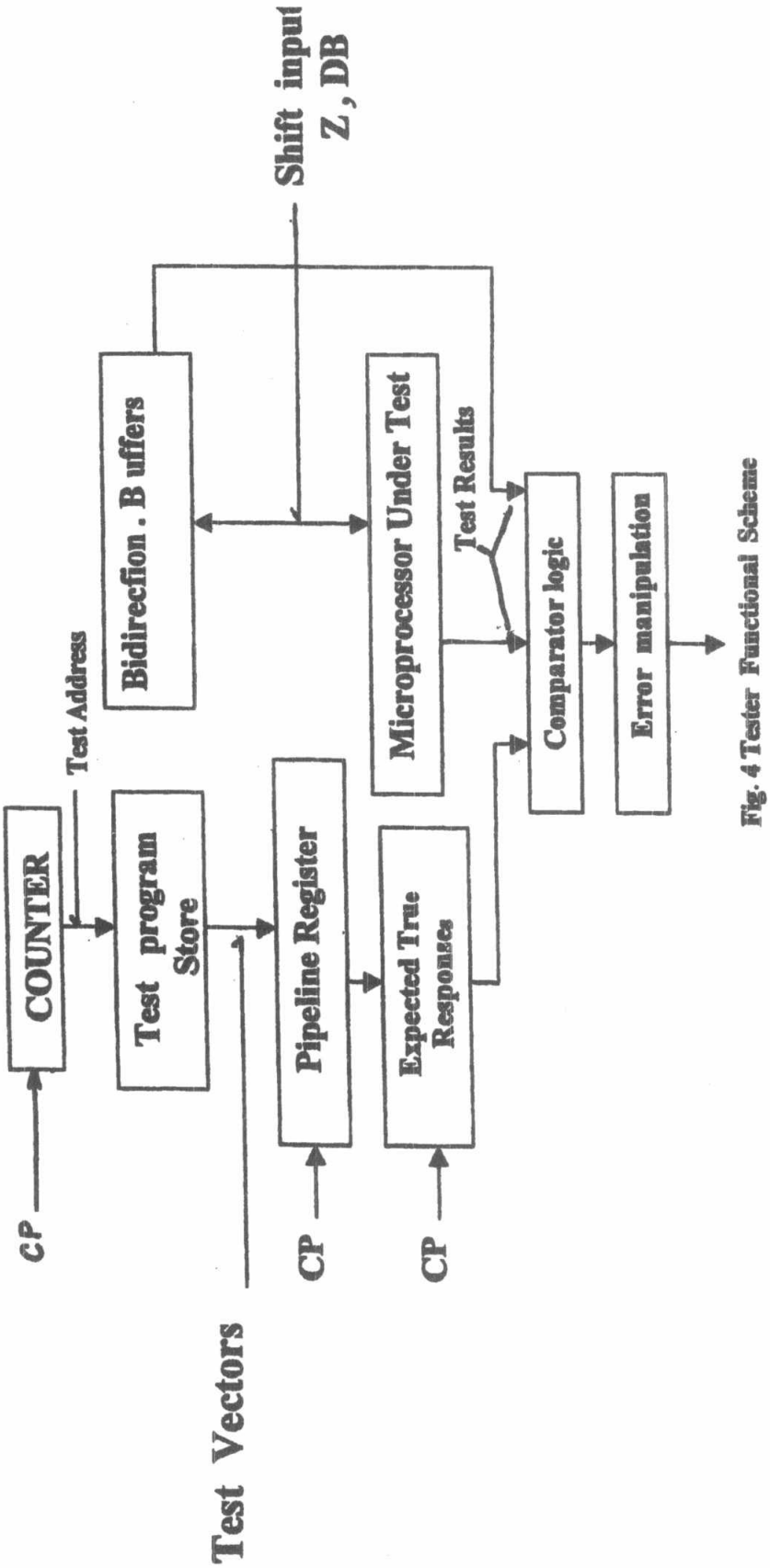


Fig. 4 Tester Functional Scheme