Short Note

On the Effect of Size of Fault Word in Parallel Fault Simulation

YUZO TAKAMATSU* and KOZO KINOSHITA**

The effect of the size of a fault word in parallel fault simulation is evaluated. Experimental results indicate that a larger fault-word size is more efficient so that the total CPU time of simulation and the number of generated test patterns are minimized. However, the effect of increasing the size of the fault word disappears when the number of average faults simulated per pass becomes less than about $5 \sim 10$.

1. Introduction

Fault simulators have been widely used for logic verification and test generation of logic circuits. Currently well-known fault simulators are classified into three categories, the parallel[1], the deductive[2], and the concurrent[3] simulators. Parallel fault simulation is the simulation in parallel of both the fault-free circuit and a number of faulty circuits. In the fault simulator, a test pattern is given and faults are simulated. This processing faults simultaneously is called a pass. The parallel simulator is faster for small circuits than the deductive and the concurrent simulators, but is inefficient for large circuits because of the increased number of passes for simulation[4]. This depends on the number of bits of the host computer used. Therefore, a variable-width-fault word is utilized so that the CPU time of simulation and the number of randomly generated test patterns are minimized. Parallel simulators using a variable-width-fault word have been proposed[4]-[7], but the effect of the size of the variable-width-fault word has not been evaluated. The size of the fault word has effects on both the CPU time of simulation and the number of randomly generated test patterns. If a large fault-word size is used for a circuit having a small number of faults, it will have an adverse effect on CPU time.

In this paper, we have implemented a parallel fault simulator using a variable-width-fault word. In this simulator the size of the fault word which represents the value of each signal line in the circuit is variable up to the maximum word M. By using this parallel simulator, we have evaluated the relationship between the number of faults in the circuit and the size of the fault word using circuits ranging in size from 160 to 3500 gates, and from 500 to 7500 faults. Experimental results show that a large size M is more efficient. Therefore, the total CPU time of simulation and the number of randomly generated test patterns are minimized. However, the effect of increasing the size of the fault word disappears when the value $N(f)$ becomes less than about $5 \sim 10$. Where $N(f)$ represents the number of average faults simulated per pass and is defined by $N(f)=F/N$. Here, F and N are the number of faults in the circuit and the number of maximum faults simulated per pass, respectively. In parallel fault simulation, this result is applicable to decide the effective size of the fault word for circuits having F faults.

*Dept. of Electronic Engineering, Faculty of Science and Engineering, Saga University, Saga, 840 Japan.
**Dept. of Information and Behavioral Sciences, Faculty of Integrated Arts and Sciences, Hiroshima University, Hiroshima, 730 Japan.

2. Variable-width-fault Word in Parallel Fault Simulation

Combinational circuits in this paper consist of AND, OR, NAND, NOR, NOT and EXCLUSIVE-OR gates. Faults under consideration are the stuck-at-0(*s-a-*0) and the stuck-at-1(*s-a-*1) faults. The parallel fault simulator implemented allows only a zero-delay gate model and simulates using a 2-valued logic for both the fault-free circuit and the faulty circuit. One faulty circuit has a single stuck fault in the circuit. In order to simulate effectively, level-ordering and variable-width-fault word techniques are employed in this parallel simulator.

[Technique of the Variable-width-fault Word]

One fault word is associated with each signal line in the circuit at the beginning of simulation. The signal value of the fault-free circuit is represented by the right most bit (bit position 0) in the first fault word. During a pass of fault simulation, faults are inserted in the fault word. The host computer used has 32 bits in a word, then 31 faults can be inserted in the first fault word.

When the number of faults inserted is greater than 31, we add the second fault word to the line for inserting the fault of the line. Thus fault words are automatically incremented up to the number M, where M is the maximum size of the fault word.

For example, consider the circuit of Fig. 1(a). When we insert the stuck-at-1 fault of line A, we add the second fault word to line A if 31 faults have already been inserted in the first fault word of line A. Then the stuck-at-1 fault of line A is inserted in the bit 0 position of the second fault word as shown in Fig. 1(b). Adding the second fault word to line B, the signal value of line C, which is represented by two fault words, can be calculated.

The parallel fault simulator implemented can increase the fault word up to the maximum number M, automatically. In the present experiments, we set $M=1(N=31$ faults), $M=10(N=319$ faults), $M=20(N=639$ faults), and $M=30(N=959$ faults). Here N is the number of maximum faults simulated per pass.

3. Experimental Results

We have implemented the parallel fault simulator described above by a FORTRAN program on a FACOM M-380 system. We evaluated the effect of the size of the fault word using circuits ranging in size from 160 to 3500 gates, and from 500 to 7500 faults. Characteristics for these sample circuits[8] are shown in Table 1. In the present experiments, a random pattern is generated and faults are simulated. The simulation is terminated when fault coverage is greater than 90 percent, or when the number of faults detected by 20 consecutive patterns is 0.

[Effect of the Size of Fault Words]

The total CPU time and fault coverage for simulation are listed in Table 2 for different sizes of the fault word. Also in this table, the number of randomly generated test patterns and of random trial patterns are given. The sizes of the fault word used are $M=1$, $M=10$, $M=20$ and $M=30$. Experimental results show that a larger size fault word is more efficient so that the total CPU time and the number of randomly generated test patterns are minimized. However, a larger size fault word has an adverse effect on simulation time, for sample circuits such as C880 and C1355 in the cases of

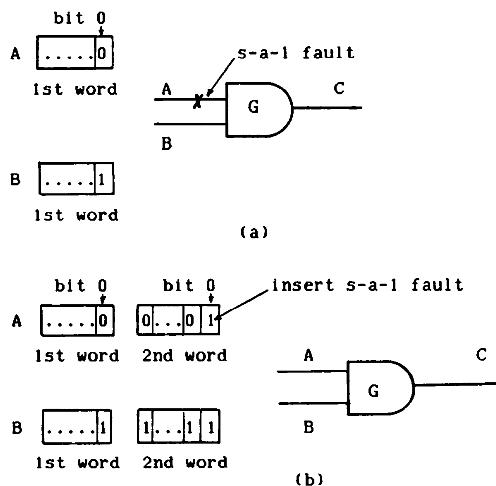

Fig. 1 Increment fault word.

Table 1. Characteristics of Circuits.

Circuit Name	Total Gates	Total Lines	Primary Inputs	Primary Outputs	Number of Faults
C432	160	432	36	7	524
C499	202	499	41	32	758
C880	383	880	60	26	942
C1355	546	1355	41	32	1574
C1908	880	1908	33	25	1879
C2670	1193	2670	233	140	2747
C3540	1669	3540	50	22	3428
C5315	2307	5315	178	123	5350
C6288	2406	6288	32	32	7744
C7552	3512	7552	207	108	7550

Table 2. Experimental Results.

Circuit name	Maximum word (M)	Fault coverage (%)	CPU time (sec.)	Randomly generated tests	Random trial patterns
C432	1	90.1	0.45	62	98
	10	90.1	0.51	60	108
	20	90.1	0.41	55	83
C499	1	90.2	0.98	136	179
	10	90.2	0.60	32	70
	20	90.2	0.53	38	56
C880	1	90.0	3.13	316	377
	10	90.2	0.75	53	65
	20	90.0	1.11	57	97
C1355	1	90.0	8.51	452	682
	10	90.3	2.86	68	151
	20	90.2	2.92	63	139
C1908	1	90.1	26.93	963	1690
	10	90.3	9.82	166	372
	20	86.9	7.67	84	278
C2670	1	90.0	92.98	1907	3423
	10	82.0	10.25	142	247
	20	82.2	9.89	89	247
C3540	1	90.1	91.10	1947	3048
	10	90.1	22.59	367	423
	20	90.2	19.61	185	360
	30	90.1	14.61	142	257
C5315	1	—	—	—	—
	10	90.5	31.03	298	298
	20	90.1	16.89	116	122
	30	90.9	14.22	78	94
C6288	1	—	—	—	—
	10	91.9	11.21	113	113
	20	91.7	5.68	42	42
	30	90.3	4.47	26	26
C7552	1	—	—	—	—
	10	90.0	132.7	1107	1107
	20	90.2	53.15	299	347
	30	90.0	41.21	198	237

$M=10$ and $M=20$.

The ratio of CPU time of simulation versus $N(f)$ is shown in Fig. 2. Where $N(f)$ represents the number of average faults simulated per pass and is defined by $N(f) = F/N$. F and N represent the number of faults in the circuit and the number of maximum faults simulated per pass, respectively. Fig. 2 indicates that the effect of increasing the size of the fault word M disappears when $N(f)$ is less than about $5 \sim 10$.

The distribution of fault words of randomly generated test patterns is shown in Fig. 3 for circuit C3540. In this case (1669 gates and 3428 faults), test patterns generated by maximum words are 87.5%, 39.5% and 11.3% for $M=10(N(f)=10.7)$, $M=20(N(f)=5.36)$ and $M=30(N(f)=3.57)$, respectively. Fig. 3 shows that the effect of increasing the size of the fault word disappears when M is greater than 30.

Fig. 4 shows the fault coverage plot for circuit C1355. From this graph it can be seen that the fault word $M=10$ is enough for a small circuit such as circuit C1355. Here $N(f)=4.93$ for circuit C1355 when $M=10$.

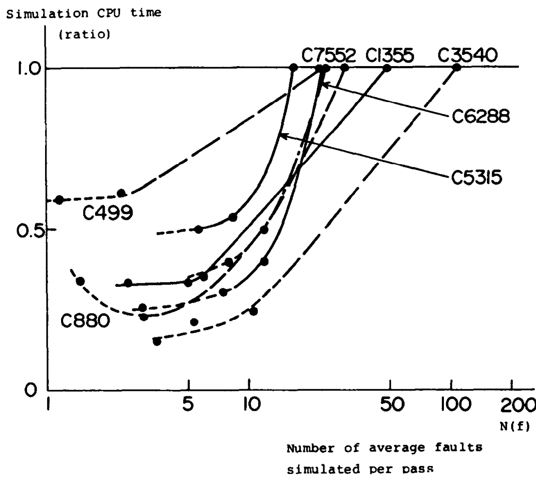

Fig. 2 Simulation CPU time-$N(f)$.

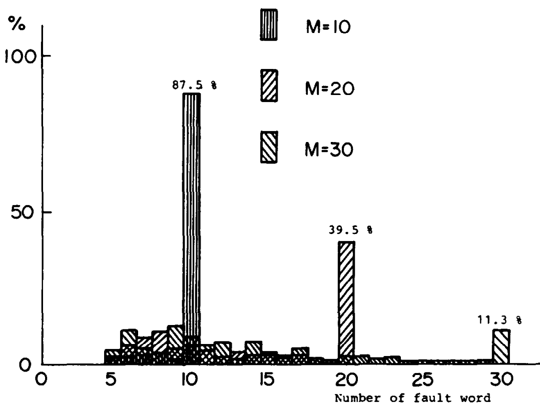

Fig. 3 Distribution of fault words of randomly generated test patterns for C3540.

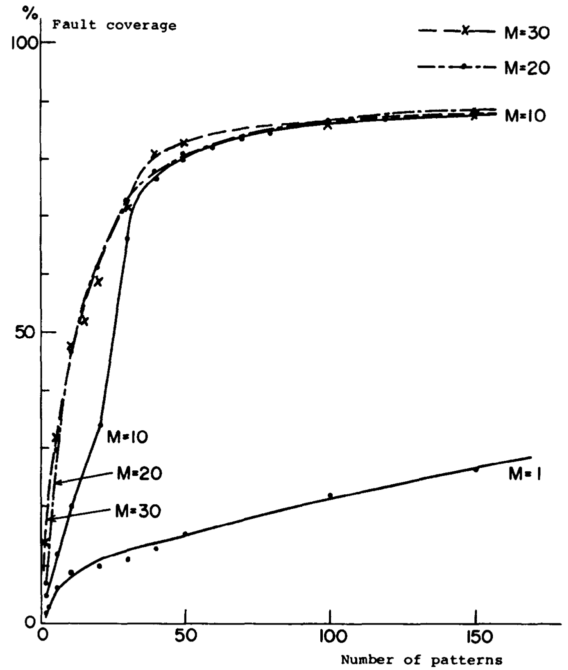

Fig. 4 Fault coverage for C1355.

4. Conclusion

We have studied the relationship between the number of faults in the circuit and the size of the fault word in parallel fault simulation. From our experimental results it can be seen that a larger fault-word size is more efficient but that the effect of increasing the size of the fault word disappears when the number of average faults simulated per pass becomes less than about $5 \sim 10$. This result is applicable to decide the effective size of the fault word in parallel fault simulation.

References

1. THOMPSON, E. W. and SZYGENDA, S. A. Digital logic simulation in a time-based, table-driven environment, Part 2. Parallel fault simulation, *IEEE Computer*, **8**, 3 (March 1975), 38–49.
2. ARMSTRONG, D. B. A deductive method for simulating faults in logic circuits, *IEEE Trans. Electron. Comput.*, EC-**21**, 5 (May 1972), 464–471.
3. ULRICH, E. G. and BAKER, T. The concurrent simulation of nearly identical digital networks, *Proc. of 10th DAC*, (June 1973), 145–150.
4. CHANG, Y. H., CHAPPELL, S. G., ELMENDORF, C. H. and SCHMIDT, L. D. Comparison of parallel and deductive fault simulation methods, *IEEE Trans. Comput.*, C-**23**, 11 (Nov. 1974), 1132–1138.
5. CHAPPELL, S. G., ELMENDORF, C. H. and SCHMIDT, L. D. LAMP; Logic-circuit simulators, *Bell S. T. J.*, **53**, (Oct. 1974), 1451–

1476.
6. YAMADA, A., WAKATSUKI, N. and TOMITA, K. Parallel fault simulator based on circuit delay and selective tracing of asynchronous logic circuit, *J. of IPS Japan*, **17**, 7 (July 1976), 569-576.
7. KIKUCHIHARA, H., AMANO, A., MURAKAMI, M. and HIRAKAWA, K. Acceleration of LSI fault simulator, *Proc. 20th Annual Convention IPS Japan*, 4B-8 (March 1974), 597.
8. BRGLEZ, F. and FUJIWARA, H. A neutral netlist of 10 combinational benchmark circuits and a target translator in FORTRAN, Special Session on ATPG and fault simulation, *Proc. 1985 IEEE Int. Symp.*, Circuit and Systems (June 1985), 663-698.

(Received April 16, 1986; revised September 3, 1986)