Efficient Synthesis of Fault-Tolerant Controllers

R. Rochet, R. Leveugle, G. Saucier

Institut National Polytechnique de Grenoble / CSI
46, avenue Félix Viallet - 38031 GRENOBLE Cedex - FRANCE
Phone: (33) 76.57.46.86 Fax: (33) 76.50.34.21 E-mail: Raphael.Rochet@imag.fr

Today, there is an increasing need for fault tolerance capabilities in integrated circuits used in critical applications such as aircraft control. The classical way to achieve fault tolerance in a logic block is to triplicate it and to implement a majority voting block on the outputs (Triple Modular Redundancy, or TMR). The Single Independent Decoder (SID) architecture was defined in order to achieve with a lower hardware overhead the tolerance of faults in the circuit control part (the Finite State Machine), and more precisely in the sequencing logic (next-state logic and state register). A dedicated synthesis tool (ASYL-SdF) has been developed and the results obtained on a large set of examples in terms of silicon area and dependability evaluation have shown its efficiency, especially compared with the TMR implementation of the sequencing logic (TMR Seq).

SID architecture

The SID architecture is based on the concept presented by Armstrong [1]. This concept basically consists in using an error correcting code when choosing the set of binary values to be assigned to the FSM states. To achieve single fault tolerance in the state register, a SEC code (Single Error Correcting code) with a minimum Hamming distance equal to 3 must be used. Several specific state assignment procedures have been defined. Here, we consider a specific state assignment performed in two steps: a classical, optimized, distance one state assignment, followed by the Hamming check bit generation. Such an encoding optimizes the functions which compute the information bits of the next-state code (not the check bits), the output logic and the decoder logic.

Then, to guarantee also the tolerance of a fault in the next-state logic, the implementation avoids any gate shared among next-state functions. Finally, the fault tolerance is achieved by an independent logic block connected on the outputs of the state register (Figure 1), so that an erroneous state code is corrected before it is used to compute the next state and output functions. This block is carefully optimized according to the Hamming code properties.

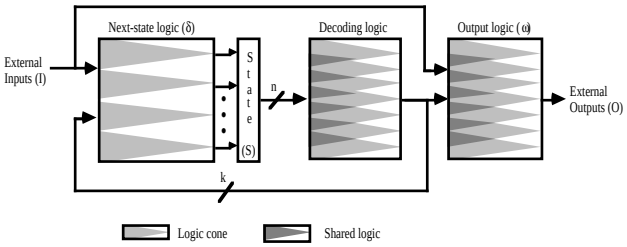

Figure 1: Architecture SID.

Implementation results [2]

About one hundred FSMs have been implemented with ASYL-SdF. For 93.4% of them, the SID architecture leads to lower area penalties (after placement and routing) than the TMR approach. The average gain is 16.3% with gains up to 38.65%.

From the reliability point of view, the Mission Time (for a reliability equal to 0.8), and the Mean Time to First Failure have been computed for a representative subset of the examples. In the worst case hypothesis (no fault tolerated in the decoder block), MT (resp. MTFF) is better with the SID implementation for 62.5% (resp. 75%) of the examples. Moreover, even when the SID implementation does not give the best MT, it often leads to a very good trade-off between area and dependability (Table 1).

Table 1: Area (μm^2), MT and MTFF (10^3 hours)

Name	TMR Seq			SID		
	Area	MT	MTFF	Area	MT	MTFF
Cpt100	**1 330 464**	**31 747**	**80 331**	863 089	**37 815**	98 609
Dk16	743 192	49 711	134 379	539 987	**52 596**	149 143
Imec10	2 677 443	6 233	24 754	2 599 707	**5 834**	23 642
Jay	1 022 791	34 094	95 102	790 428	**33 949**	97 537
Kirkman	354 110	62 884	220 178	323 447	**55 048**	203 736
Log	514 792	**64 878**	184 894	365 063	**68 243**	208 062
Scf	2 613 175	**13 995**	38 049	1 817 656	**16 127**	45 738
Zeegers	4 006 714	**7 126**	22 011	3 103 853	**7 158**	23 621

References

[1] D. B. Armstrong, "A general method of applying error correction to synchronous digital systems", The Bell System Technical Journal, vol. 40, no. 2, March 1961, pp. 577-593

[2] R. Rochet, R. Leveugle, G. Saucier, "Alternative implementations of fault-tolerant controllers", CSI Research Report CSI-SdF-94.2, October 1994