# EFFICIENT METHOD OF FAILURE DETECTION IN ITERATIVE ARRAY MULTIPLIER

Alexander Drozd
Department of Computers, Odessa State Polytechnic University, Odessa, Ukraine
apdroz@favt.ospu.odessa.ua

## Abstract

*In this paper we present a method for on-line testing of multiplier. The method is based on the time and information natural redundancy and provides design of the simple self-checking checker for hard failure detection in **8**-bit and **16**-bit array multiplier.*

## 1. Hard failure detection

Traditionally the hard failure identification process is executed by errors detection and calculation their number on the given time. It requires the use of check method with high probability of error detection (for example, residue checking) and leads to design of complicated checker.

We propose to detect the hard failure by the various probabilities of detection of random fault and hard failure. It needs a check method with small probability $P$ and time of hard failure detection $\tau = -\ln(2 - P) / (\ln(1 - P))$.

## 2. The Check Method

The method based on Fermat (1601-1665) supposition, that all numbers in aspect of $2^n+1$, $n=2^x$ ($x$ is natural number) are prime numbers, and next theorem uses also.

**Theorem 1**. Double code vectors $G(n, n)$ are unallowed value of $n$-bit multipliers product, except null vector, if $2^n+1$ is prime number.

**Proof.** Code vectors $G(n, n)$ have aspect $k(2^n+1)$, where $k=1 \div 2^n-1$. Number $2^n+1$ is not $n$-bit multiplier, for the reason of its $n+1$ capacity, and does not factorable as prime number. Consequently, this number and numbers $k(2^n+1)$ are unallowed value of $2n$-bit product. **Q. E. D.**

Euler (1707-1783) refuted of Fermat statement for $x=5$, however the statement and theorem 1 are true for $x<5$. The cases of $x=3$ and $x=4$ are conform to $n=8$ and $n=16$.

The method detects error if for multipliers $A\{1 \div n\}$, $B\{1 \div n\}$ and product $V\{1 \div 2n\}$ only one of two conditions performs: $(A\{1 \div n\} \neq 0) \& (B\{1 \div n\} \neq 0)$; $V\{1 \div n\} = V\{n+1 \div 2n\}$.

**Theorem 2**. Every characteristic hard failure of iterative array multiplier is detected at least on one input word.

Characteristic errors of the iterative array multiplier have aspect $\pm 2^{r-1}$, where $r$ is number of product bit, $r=1 \div 2n$. Proof of theorem is based on the factorization of the assumption formula $k(2^n+1) \pm 2^{r-1}$ on multipliers $A$ and $B$.

The checker is shown on Figure 1. It consists of two blocks and forms two-bits check code $E\{1, 2\}$:

$$E\{1\} = ((A\{1 \div n\} \neq 0) \; \& \; (B\{1 \div n\} \neq 0));$$
$$E\{2\} = (V\{1 \div n\} = V\{n + 1 \div 2n\}).$$

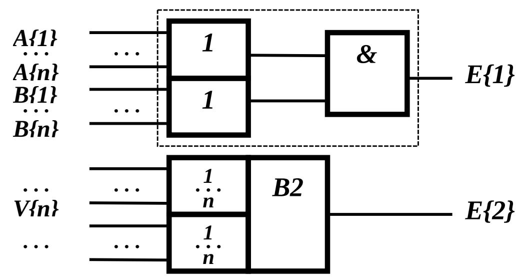


**Figure 1**: Checker of multiplier

The first block $B1$ consists of two $n$-bits disjunctors which check the conditions $A \neq 0$, $B \neq 0$, and conjunctor computes the bit $E\{1\}$ from condition, that both of the multipliers are not zero. The second block $B2$ is comparator of the low and high product bits. It computes the bit $E\{2\}$. The values $00$, $11$ of the code $E\{1, 2\}$ detect fault of multiplier, and the values $01$, $10$ are ensured correct work of the device.

The checker is self-checking circuit for any fault in one of its block because of separate computation of the bits of code $E$. Probability $P=3 \cdot 2^{-n}$ of error detection is estimated by ratio of input words number which lead to error detection to total the input words number. The hard failure detection time $\tau=59$ for $n=8$ and $\tau=15142$ for $n=16$. The hardware overhead reduces comparing to residue checker in $6.6$ times.

## 3. Conclusions

We have proposed the on-line testing method for hard failure detection in **8** and **16** bits multiplier. The method uses the time redundancy $\tau$ and the natural information redundancy of product and permits to build the simple self-checking checker.