

# Progress Report

Kan Shi

October 2013

## 1 Design of Digit-parallel On-line Multiplier

$$\begin{cases} H[j] &= 2^{-3}(x_{j+4} \cdot Y[j+1] + y_{j+4} \cdot X[j]) \\ W[j] &= 2P[j] + H[j] \\ z_j &= SEL(\widehat{W[j]}) \\ P[j+1] &= W[j] - z_j \end{cases} \quad (1)$$

## 2 Probabilistic Model of Overclocking Error in On-line Multiplier

### 2.1 Annihilation of the Propagation Chain

While the delay in a digit-parallel on-line multiplier might be derived from many sources such as the computation delay to generate outputs, the overall delay will eventually be determined by the longest propagation delay between stages with increasing operand word-lengths. Let the propagation delay between two adjacent stages in a  $N$ -digit olMult be denoted by  $\mu$ , hence the delay of the longest chain which propagates from the MSD to the LSD is given by  $d_w$  as shown in (2).

$$d_w = (N + \delta - 1) \cdot \mu \quad (2)$$

However, we note that the chain is annihilated at a certain stage if the propagation inputs of this stage change while the propagation outputs keep stable. This will shrink the value of  $d_w$ , and there are two possible cases specifically. As an example for the first case, assume at time  $t$  ( $t > \mu$ ) the value of propagation inputs and outputs of a stage  $S_i$  to be  $Pin(t)$  and  $Pout(t)$  respectively. After delay  $\mu$ , the input value changes to  $Pin(t + \mu)$  and stabilizes thereafter, while the output of this stage remains  $Pout(t)$ . Hence for the next stage  $S_{i+1}$ , the input of which varies from  $Pout(t - \mu)$  to  $Pout(t)$ , as illustrated in Figure.xxx. Under this situation, the chain delay to  $S_{i+1}$  is reduced by  $\mu$  because of the annihilation.

For the second case, the current chain will be completely annihilated and a new chain will be generated at a given stage if  $Pout(t) = Pout(0)$  for  $t = \mu, 2\mu, \dots$ . Therefore the worst case delay is given by (3) where  $d_p$  denotes the delay of the  $p^{th}$  propagation chain.

$$d'_w = \max(d_1, d_2, \dots) \quad (3)$$

In the following of this section, detailed analysis for both cases will be described and the worst-case delay of the olMult will be discussed.

## 2.2 Worst-case Analysis in On-line Multiplier

From (1) several observations can be made under the assumption that all signals are reset to 0 initially.

**Observation 1.** *The two integer bits of  $W[j]$  and  $2P[j+1]$  are either 00 or 11.*

This observation can be justified by contradiction. All the combinations of  $\widehat{W[j]}$  and the corresponding  $z_j$ , the most significant 3 bits of  $P[j+1]$  and the 2 integer bits of  $2P[j+1]$  are listed in Table 1. For  $j = -3$ , we have (4) according to (1).

$$W[-3] = 2^{-3}(y_1 \cdot X[-3]) \quad (4)$$

Hence  $\widehat{W[-3]}$  is either 11.1 or 00.0, with the corresponding  $2P[-2]$  being 00 or 11 which is the propagation input for the next stage. For  $j > -3$ , the 3 MSBs of  $H[j]$  in (1) is either 00.0 or 11.1 due to the shift of binary point. Also as seen in Table.xxx, if the two integer bits of  $\widehat{W[j]}$  are identical, the integer bits of  $2P[j+1]$  will be the same. Therefore the propagation of  $2P[j+1]$  will not generate diverse bits in the integer part of  $W[j]$ .

Table 1: All the combinations of  $\widehat{W[j]}$  and the corresponding  $z_j$ ,  $P[j+1]$  and  $2P[j+1]$ .

$\widehat{W[j]}$	$z_j$	$P[j+1]$ (3 MSBs)	$2P[j+1]$ (2 MSBs)
00.0	0	00.0	00
00.1	1	11.1	11
01.0	1	00.0	00
01.1	1	00.1	01
10.0	$\bar{1}$	11.0	10
10.1	$\bar{1}$	11.1	11
11.0	$\bar{1}$	00.0	00
11.1	0	11.1	11

**Observation 2.**  *$P[j+1]$  will not change if only the integer bits of  $W[j]$  change to a new value.*

According to Observation 1, there are only four possible cases for the change of  $W[j]$  as summarized in Table.xxx. This observation describes the situation where a propagation chain annihilates but not necessarily generating a new chain, because  $2P[j+1]$  may not maintain its initial value at all times.

Table 2: Ob2.

$\widehat{W[j]}$	$z_j$	$P[j+1]$ (3 MSBs)	$2P[j+1]$ (2 MSBs)
00.0 $\rightarrow$ 11.0	0 $\rightarrow$ $\bar{1}$	00.0	00
00.1 $\rightarrow$ 11.1	1 $\rightarrow$ $\bar{0}$	11.1	11
11.0 $\rightarrow$ 00.0	$\bar{1}$ $\rightarrow$ 0	00.0	00
11.1 $\rightarrow$ 00.1	0 $\rightarrow$ 1	11.1	11

As stated in Section 2.1, the annihilation of propagation chain would leads to a reduction of the propagation delay. It is worthwhile exploring the conditions of occurrence of Observation 2. For instance in a 3-digit olMult, there are 6 stages in total. If  $x_{-3} \neq 0$  and  $y_{-3} \neq 0$ , no more than 6 MSBs of  $P[-2]$  will change with respect to the initial value at  $t = 0$ . This is because in (4)  $y_1 \cdot X[-3]$  contains only 1 fractional bit and 2 integer bits. Hence maximally 5 MSBs of  $2P[-2]$  will be updated. It is worth noting that this is irrelevant to the word-length of the olMult. Besides, at  $t = 0$  the computation of  $H[j]$  in (1) is finished, as assumed previously in the timing model. For  $t > 0$ , the variation of  $2P[j]$  propagates from *Stage0* to *Stage1*  $\cdots$  *Stage4* with the number of altered MSBs being 5 to 2, from  $t = \mu$  to  $t = 4\mu$  respectively. Also note that this changing procedure is irrelevant to the value of  $x_{j+4}$ ,  $y_{j+4}$ ,  $X[j]$  and  $Y[j + 1]$  for *Stage1*  $\cdots$  *Stage4*. Hence for *Stage4*, we have  $Pout(4\mu) = Pout(3\mu)$  in accordance with Observation 2. This stands for an annihilation of the chain, and therefore the propagation delay from *Stage0* to *Stage5* is  $4\mu$ . The above process is illustrated in Fig.xxx.

This changing process happens in a 3-digit olMult when  $P[-2] \neq 0$ , i.e. it is only determined by the inputs of the first stage regardless of the other stages. Actually in other situations, the overall propagation delay will not be increased. For example if  $P[-2] = 0$  while  $P[-1] \neq 0$ , similarly we have maximally 7 MSBs of  $2P[-1]$  will change at  $t = 0$ . This will propagates to *Stage5*, the propagation inputs of which change 2 MSBs, and finally yields a delay of  $4\mu$ . This value shrinks to  $3\mu$  if  $P[-2] = P[-1] = 0$  while  $P[0] \neq 0$  based on a similar analysis.

### 2.3 Probability of Overclocking Error in On-line Multiplier

### 2.4 Magnitude of Overclocking Error in On-line Multiplier

## 3 Probabilistic Model of Truncation Error in On-line Operators