

## Ново приложение на концентратори в схеми на конвейерни умножители

Димитър С. Тянев, Стефка И. Попова, Александър И. Иванов, Драгомир В. Янев

**Резюме:** Представено е едно ново приложение на концентратори от тип 3:1 в схеми на конвейерни умножители, с помощта на които скоростта се увеличава двойно. Показано е, че приложението на концентраторите от този тип, е най-доброто в сравнение с известните алгоритми за умножение на 2 разряда едновременно. Представена е оригинална принципна логическа схема на комбинационната част за всяко ниво в умножителя, както и общата логическа структура на конвейерния умножител. Работоспособността и качествата на предлаганата схема за построяване на конвейерни умножители са експериментирани и показани върху продукти на фирма Xilinx.

### Conveyer multiplier with concentrators

Dimitar S. Tyanev, Stefka I. Popova, Aleksandar I. Ivanov, Dragomir V. Yanev

**Abstract:** A new application of 3:1 concentrators in schemes of conveyer multipliers are introduced to double acceleration of the speed. It is showed that the application of such concentrators is the best in comparison with well-known algorithm for two bits together multiplication. An original logical scheme of the combinational part for each level in the multiplier is introduced, as well as the general logical structure of the conveyer multiplier. The efficiency and qualities of the proposed scheme for building conveyer multipliers are tested and showed through Xilinx products.

#### 1. Въведение

Операция умножение на цели числа е с висока честота на изпълнение в компютърните изчисления. Съвременните цифрови процесори се характеризират с това, че тяхната операционна част е конвейерно организирана. Комбинационните схемни умножители, въпреки изключително прецизния си логически синтез [1÷9], предвид голямата дължина на операндите, внасят закъснение, което е съществено по-голямо от периода за тактуване на командния конвейер. Ето защо е актуална конвейерната организация на схемния умножител [13÷16].

#### 2. Същност на конвейерния умножител

Възможната логическа структура на конвейерния умножител е представена подробно в [10]. Тази структура се характеризира с това, че съдържа  $n$  на брой нива (фиксатори) за едновременно движещите се междинни суми и съответните им двойки съмножители, представени в  $n$ -битова разрядна мрежа. Логическа структура може да поеме конвейерното изпълнение на множество последователни операции умножение, от които в конвейера ще се изпълняват паралелно във времето  $n$  на брой от тях. Така на всеки такт от конвейера ще слиза поредната двойка съмножители и съответното ѝ произведение. Закъснението на дадено произведение по отношение момента, в който съмножителите се зареждат в конвейера, е равна на  $n$  такта. Ако приемем, че едно последователно умножение на 2 числа отнема  $n$  на брой такта, то последователното изпълнение на  $n$  на брой операции умножение би отнело  $n^2$  на брой такта. Същият брой умножения, с помощта на конвейерно организиран схемен умножител, би отнело  $2n$  на брой такта. Така се разбира, че конвейерната организация, в сравнение с последователната, е в състояние да повиши производителността на процесора при последователно изпълнение на операции умножение, до  $n/2$  пъти, т.е. примерно, при 32-битова разрядна мрежа, увеличението достига 16 пъти.

В настоящата работа се представя проектирането и изследването на оригинална логическа структура на конвейерен умножител, чиято организация се отличава от известните по това, че формира междинните суми като паралелни суми от 3 числа, с помощта на концентратор

от типа (3:1), чийто синтез и изследване са изложени в [11,12]. Традиционно при умножение (на  $n$ -битови числа без знак) междинните суми се получават както следва:

$$S_{i+1} = S_i + X.y_{i+1}.2^{i+1}, \quad i = \overline{0, n-1} . \quad (1)$$

За да натоварим обаче три-входовия суматор, който приемаме да използваме като оператор на дадено ниво в конвейера, трябва да формираме междинната сума както следва:

$$S_{j+1} = S_j + X.y_{i+1}.2^{i+1} + X.y_i.2^i . \quad (2)$$

От горния израз се вижда, че във всяка нова междинна сума се натрупват две последователни поразрядни произведения. От тук веднага следва изключително положителният извод, че броят на формираните с помощта на такива суматори междинни суми, ще бъде два пъти по-малък и за техният индекс  $j$  можем да запишем закона:  $j = \overline{1, n/2}$

За логическия синтез на суматора, който ще реализира сумата (2), е полезно тя да бъде илюстрирана със следната схема:

		$q^{(i)}_{n-1}$	$q^{(i)}_{n-2}$	...	...	$q^{(i)}_1$	$q^{(i)}_0$	0	...	0	0	$=X.y_i.2^i$
+		$q^{(i+1)}_{n-1}$	$q^{(i+1)}_{n-2}$	...	$q^{(i+1)}_1$	$q^{(i+1)}_0$	0	0	...	0	0	$=X.y_{i+1}.2^{i+1}$
+	$s^{(j)}_k$	$s^{(j)}_{k-1}$	$s^{(j)}_{k-2}$	...	...	...	...	...	...	$s^{(j)}_1$	$s^{(j)}_0$	$=S_j$
	$s^{(j+1)}_{k+2}$	$s^{(j+1)}_{k+1}$	$s^{(j+1)}_k$	$s^{(j+1)}_{k-1}$	...	...	...	...	...	$s^{(j+1)}_1$	$s^{(j+1)}_0$	$=S_{j+1}$

Фиг. 1 Схема за натрупване на поредната междинна сума

Схемата показва, че младшите разряди (на брой  $2.j$ ) в  $j$ -тата междинна сума са окончателно получени при предходните събирания и следователно могат да бъдат пропуснати и изключени от текущото събиране. От тук следва, че всички концентратори употребени в отделните нива на конвейера следва да бъдат с една и съща дължина и съответно изместени наляво един спрямо друг, в съответствие с нарастващия порядък на използваните битове от множителя (тук се има предвид методът за умножение с младшите разряди напред). Синтезът на принципната логическа схема на тук прилаганите концентратори се различава от този, изложен в [11] и в [12], само по това, че трите числа са разместени едно спрямо друго на един бит, във съответствие с формула (2). Всички останали параметри на концентратора са подробно изложени в посочената литература.

След така изложените съображения следва, че получената структура на конвейерния умножител, реализиращ натрупването (2), ще съдържа два пъти по-малко на брой нива (фиксатори), т.к. на всяко ниво тя консумира едновременно по два бита от множителя. Така, структурата на конвейерен умножител с концентратори от типа (3:1) ще има вида, показан на фигура 2.

Както се вижда от фигура 2, първата междинна сума се формира като сума от следните три поразрядни произведения:

$$S_1 = X.y_0 + X.y_1.2^1 + X.y_2.2^2 , \quad (3)$$

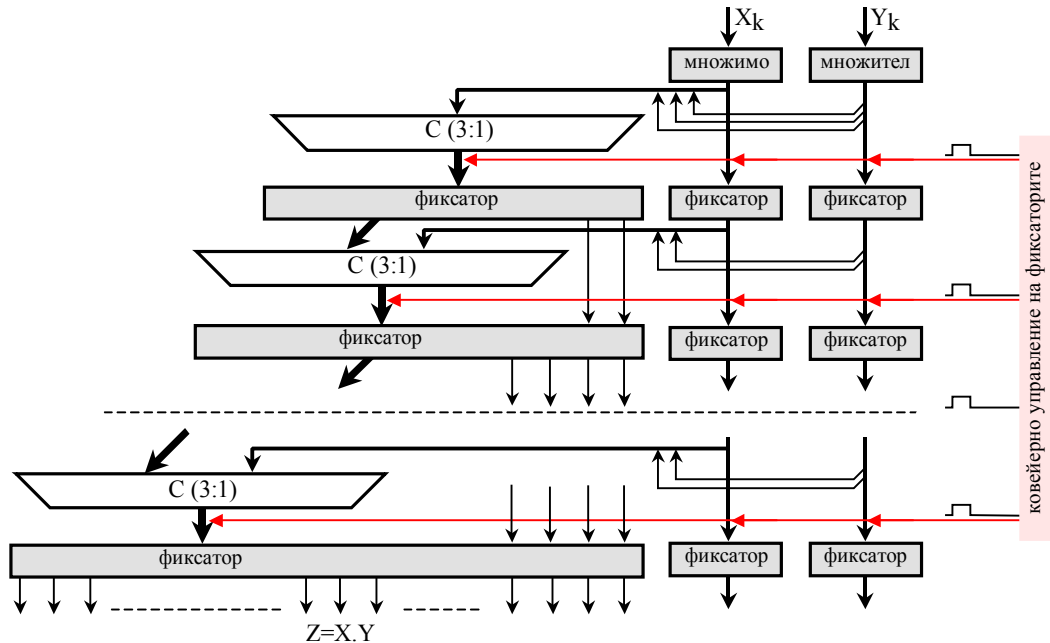
а всички останали – според уравнение (2). Вижда се още, че от всяко ниво на конвейера окончателно се получават по две младши цифри на произведението, а сумиращите схеми имат една и съща дължина.

В случаите, когато дължината  $n$  на разрядната мрежа е четно число, в последната степен на конвейера ще се събират само две числа, от което следва, че в последната степен ще бъде употребен обикновен двоичен суматор. В случай, че дължината на разрядната мрежа е нечетно число, тогава всички сумиращи схеми ще бъдат еднакви.

Представената конвейерна структура може да се прилага и за умножение на числа със знак, представени в допълнителен код. Единственото изменение, което следва да се направи, се състои в това, че в последната степен следва да се изпълни операция изваждане на множимото, която се налага за корекция на произведението. Това може да бъде реализирано

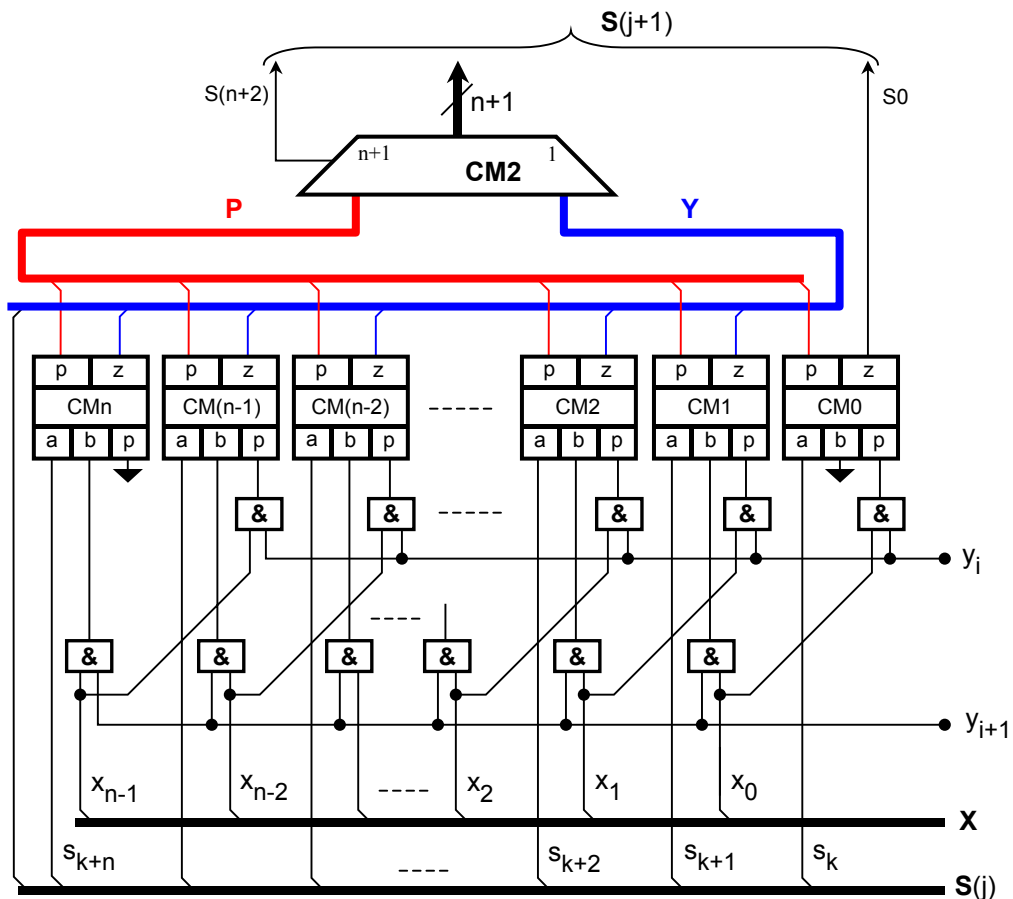
автоматично, чрез управление на входния за суматора мултиплексор с помощта на знаковия бит на множителя  $y_{n-1}$  в съответствие с логиката на следния оператор:

$$\begin{aligned} \text{if } y_{n-1} = 0 \text{ then } S_{j+1} &= S_j + X.y_{n-1}.2^{n-1} + X.y_{n-2}.2^{n-2} \\ \text{else } S_{j+1} &= S_j + \overline{X}.y_{n-1}.2^{n-1} + X.y_{n-2}.2^{n-2} + y_{n-1} \cdot \end{aligned} \quad (4)$$



Фиг. 2 Логическа структура на конвейерния умножител

В съответствие с изложените до момента съображения, принципната логическа схема на необходимия за конвейерното приложение концентратор ще има вида, показан на фигура 3.



Фиг. 3 Логическа схема конвейерния концентратор

### 3. Допълнителни изводи

Всички известни схеми за умножение, за разлика от представената тук, се характеризират с използване на двоичен суматор, в който междинната сума се събира с някакъв еквивалент на поразрядно произведение. Този еквивалент се избира като се мултиплексира по единия от входовете на суматора измежду няколко достатъчно сложни за получаване резултата. Броят на тези резултати за всяко ниво в конвейера се движи от 5 нагоре (тук се имат предвид схеми, които се получават при използване на алгоритми поне за умножение едновременно с 2 разряда от множителя, с които е естествено да се сравнява тук предложената схема) – [1, 16]. Това означава, че апаратните разходи за реализация на комбинационната част на всяко ниво в конвейерния умножител са най-малко 2 пъти по-вече в сравнение с тези в тук предложената.

По същия начин стои въпросът с оценката на бързодействието на тези схеми. От тук логично следва, че тактовата честота за надеждно и стабилно управление на предложения конвейер с концентратори, ще бъде възможно повишена до 2 пъти.

### 4. Заключение

С експериментална цел описаната структура е проектирана конкретно за умножение на 8-битови числа. Този примерен вариант на проекта е реализиран в технологичната среда WebPack ISE на фирма Xilinx, чрез HDL-езика Verilog, като е предназначен за имплементиране в FPGA-матрица от фамилията Spartan II на същата фирма. На моделната времедиаграма, показана по-долу, може да се види конвейерното умножение на последователните двойки съмножители:

7.23=161 ;  
10.22=220 ;  
13.21=273 ;  
16.20=320 ;  
19.19=361 ;  
22.18=396 .

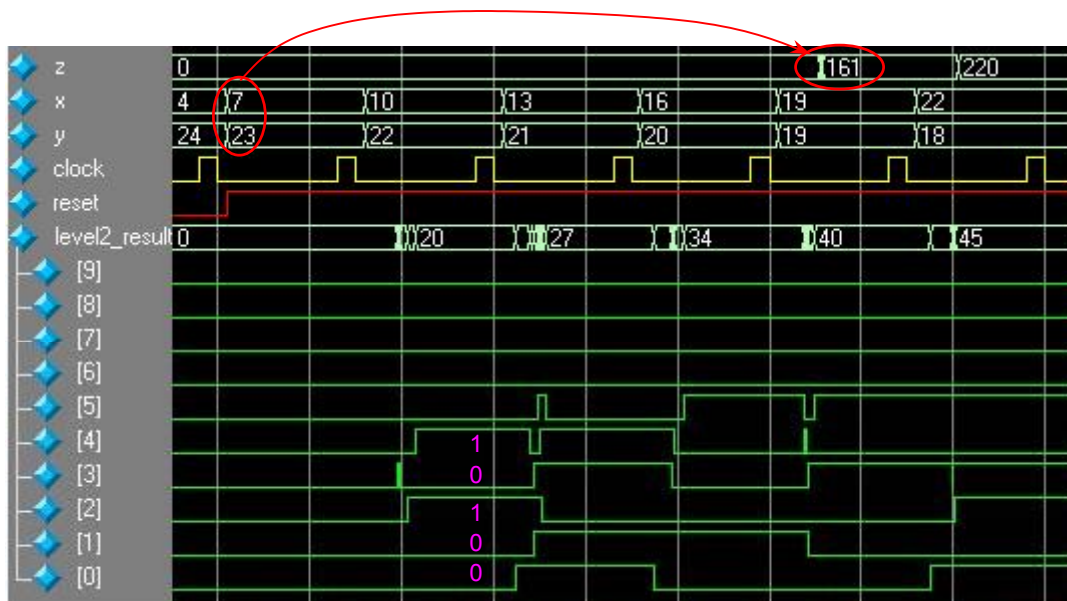
На всеки такт от конвейера слизат посочените произведения (вижте фигура 4). Върху изходите на началните нива на конвейера, т.е. в дълбочина, в десетична бройна система са представени стойностите на междинните суми, до които е достигнало като междинна сума съответното произведение в конвейера.



Фиг. 4 Изходни и междинни резултати в умножителя

На следващата фигура 5 е показан преходният процес на изходите на концентратора във второ ниво на конвейера. Така, при движение на двойките съмножители и преминаването им през това ниво, на неговия изход последователно се получават междинните суми 20, 27, 34, 40 и 45.

Тъй като периодът на тактовата последователност в случая е 15[ns] и ясно се вижда, че е възможно да бъде намален двойно, то можем да твърдим, че такъв конвейер ще генерира 16-битово произведение на всеки 7,5[ns]. С други думи повече от  $133 \cdot 10^6$  числа в секунда.



Фиг. 5 Закъснения на второ ниво в конвейера

## 5. Литература

- [1]. Wallace C. S., *A suggestion for a fast multiplier*, IEEE Trans. On Computers, vol.13, p.14-17, 1964.
- [2]. Dadda L., *Some schemes for parallel multipliers*, Alta Frequenza, vol. 34, p.349-356, 1965.
- [3]. Waser S., *High-Speed Monolithic Multipliers for Real-Time Digital Signal Processing*, Computer, №10, 1978.
- [4]. Baugh Ch., Woley B., *A Two's Complement Parallel Array Multiplication Algorithm*, IEEE Transaction on Computers, C22, №12, 1973.
- [5]. Rubinfeld L. P., *Proof of the Modified Booth's Algorithm for Multiplication*, IEEE Transaction on Computers, C22, №10, 1975.
- [6]. Takagi N., Yasuura H., Yajima S., *High-Speed VLSI Multiplication Algorithm with a Redundant Binary Addition Tree*, IEEE Transaction on Computers, C34, №9, 1985.
- [7]. Карцев М. А., Брик В. А., *Вычислительные системы и синхронная арифметика*, Издательство "Радио и связь", 1981.
- [8]. Bickerstaff K., Swartzlander E., Schulte M., *Analysis of column compression multipliers*, 15<sup>th</sup> IEEE Symposium on Computer Arithmetic, p.33-39, 2001.
- [9]. Swartzlander E., Goto G., *Computer arithmetic*, ed. Boca Raton, CRC Press, 2002.
- [10]. Тянев Д. С., *Електронни цифрови машини*, ТУ-Варна, ISBN 954-20-0016-2, 1995.
- [11]. Тянев Д. С., *Организация на компютъра (цифрова аритметика)*, ТУ-Варна, ISBN 954-20-0258-0, 2004.
- [12]. Тянев Д. С., Попова С. И., Иванов А. И., Янев Д. В., *Синтез и сравнителен анализ на паралелни многовходови суматори*, сп. "Компютърни науки и технологии" ISSN 1312-3335, №2-2005, стр. 51-61. Интернет публикация: [http://www.tyanev.com/resources/docs/Document\\_V\\_37\\_ENG.pdf](http://www.tyanev.com/resources/docs/Document_V_37_ENG.pdf).
- [13]. Kiefer G., Waker A., Thumm A., *Rechnerorganisation*, Institut fur Informatik, Stuttgart, 2001.
- [14]. Schiller, Jochen, *Rechnerstrukturen*, Freie Universitat, Berlin, 2003.
- [15]. *Design FPGA-Based DSPs for Performance and Power*, Jim Simkins and Ben White, "Chip Design Magazine", February, 2005.
- [16]. *Multiplication in FPGAs* - [www.fpga-guru.com/multipli.htm](http://www.fpga-guru.com/multipli.htm).