

**ИЗСЛЕДВАНЕ ВЛИЯНИЕТО НА ПАРАМЕТРИТЕ НА ЛОГИЧЕСКАТА  
АРХИТЕКТУРА ВЪРХУ ЗАКЪСНЕНИЯТА И ПЛОЩТА НА FPGA  
СХЕМИТЕ**

**A STUDY ON THE EFFECT OF LOGICAL ARCHITECTURE  
PARAMETERS ON FPGA'S PERFORMANCE AND DENSITY**

**Петър Минев**

*ТУ-Габрово*

**Валентина Куценска**

*ТУ-Габрово*

**Abstract**

*In this report we present a new analytical model for determining the parameters in the logical architecture of the FPGA circuits. We did a study on the effect of logic block functionality on FPGA performance and density based on this model. In particular, in the context of a lookup table, cluster-based island-style FPGAs, we explore the effect of lookup table (LUT) size and cluster size (number of LUTs per cluster) on the speed and the logic density of a FPGA. The purpose of the study is to demonstrate the applicability of the developed model in the designing of new FPGA architectures.*

**Keywords:** Field-programmable Gate-Array (FPGA), Computer-Aided Design (CAD), Architecture, VLSI, Look-up Table (LUT), Basic Logic Element (BLE), Logic Block (LB).

**ВЪВЕДЕНИЕ**

FPGA приборите намират широко приложение в проектирането и производството на електронни схеми и устройства. Те са предпочитани поради ниската си цена и възможността за програмиране на място от проектанта. Все по-широкото използване на тези прибори налага постоянното им усъвършенстване с цел повишаване на тяхната производителност, намаляване на консумираната мощност и намаляване на заеманата площ. Това изисква изследване и промяна на тяхната логическа и опроводяваща архитектура.

При разработката на нови FPGA ИС се прави избор на различни архитектурни параметри, свързани с тяхната логическа и опроводяваща структура. Обикновено тези параметри се избират експериментално чрез експериментална методика, която да покаже при какви стойности се получава най-добро съотношение между заемана площ и производителност на новите FPGA прибори [3] [12]. Също така, съществуват аналитични методи за определяне на някои от логическите и трасиращите архитектурни параметри [1] [6] [8].

В настоящият доклад е представен нов аналитичен модел (1), за определяне на един от важните параметри в логическата архитектура на FPGA схемите. Моделът е изведен на база правилото на Рент [18] и показва зависимостта на броя входове на един логически блок от размера и броя на LUT елементите, които го изграждат.

$$I = N^p (K + 1) - N, \quad (1)$$

където  $K$  е размера на LUT елемента,  $N$  е брой LUT елементи в логическия блок,  $p$  е експонентата на Рент.

Този модел дава резултати близки до резултатите на модела (2), който е експериментално изведен и доказан в [1]. За разлика от (2) в настоящия модел участва и експонентата на Рент, което го прави с по-универсално приложение. Експонентата на Рент показва степента на сложност на схемите, които ще се реализират в FPGA. За ASIC тя се избира между  $0,5 \div 0,75$  [3], докато при FPGA приборите е в порядъка от  $0,75 \div 0,85$  [12].

$$I = \frac{K}{2} (N + 1), \quad (2)$$

В съвременните фамилии FPGA схеми се цели постигане на максимална (близка до 100%) използваемост на логическите ресурси в чипа. Затова най-често броя входове на един логически блок се определя като произведение от броя входове на изграждащите го LUT елементи и общият им брой в него, т.е.

$$I = K \cdot N \quad (3)$$

Това обаче води до ненужно увеличаване на размера на входните мултиплексори, а оттам и заеманата площ от един логически блок и закъсненията в него.

В настоящият доклад се доказва, че и при по-малък брой входове определен спрямо (1) също се получава близка до 100% използваемост на логическите ресурси в схемата.

За да се докаже, че (1) е приложим при проектирането на нови FPGA архитектури е необходимо решаването на следните задачи:

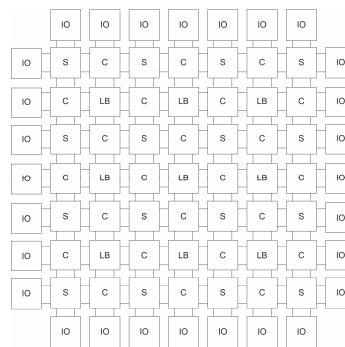
1. Намиране на такова  $p$ , при което използваемостта на BLE е най-близка до 100%;
2. Определяне на ефекта на  $K$  и  $N$  при използване на (1) върху площта на FPGA схемите;
3. Определяне на ефекта на  $K$  и  $N$  при използване на (1) върху закъсненията в FPGA схемите.

## ИЗЛОЖЕНИЕ

### А. Обща архитектура на FPGA схемите

Архитектурата на съвременните SRAM базирани FPGA интегрални схеми се състои от логически блокове (LB) разположени симетрично в матрица от редове и колони, заобиколени от свързващи канали. Каналите съдържат определен брой програмируеми връзки, посредством които се осъществява свързването между отделните логически блокове (фиг. 1).

В периферията на архитектурата се намират входно-изходни блокове (IO), които свързват FPGA логическите блокове с външни устройства.



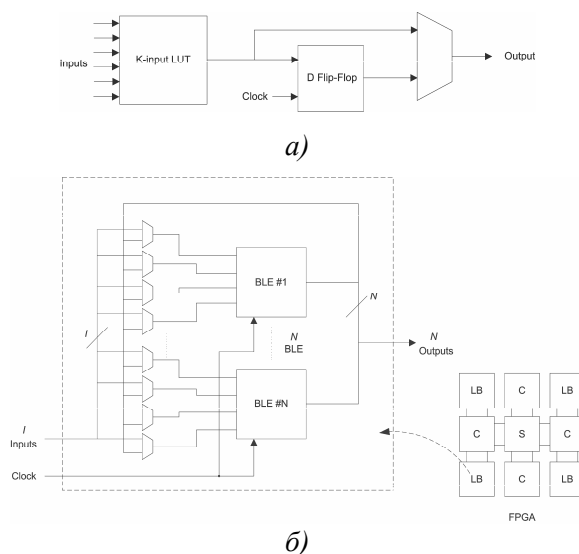
Фиг. 1. FPGA архитектура с матрична структура

### Б. Архитектура на логическите блокове

Логическите блокове са изградени от  $N$  на брой базови логически елементи (BLE), които формират една логическа група (фиг. 2).

Броят на входовете на логическата група е означен с  $I$ , а броят на изходите е равен на броя BLE в нея, т.е.  $N$ . Всеки BLE съдържа един LUT елемент с  $K$  входа и един D-тригер. Елементът LUT представлява логическа структура, която позволява реализация на всяка една логическа функция с  $K$  на брой променливи. С двувходивия мултиплексор се избира дали изхода на BLE да се взема от D-тригера или не (фиг. 2 (а)).

Всяка логическа група е „напълно свързана“, което означава, че всеки  $I$ -вход и всеки  $N$ -изход могат да се свържат с всеки  $K$ -вход на LUT елемента. За тази цел се използват мултиплексори свързани към входовете на BLE както е показано на фиг. 2(б).



Фиг. 2. Структура на BLE (а) и логическа група (б)

### В. Методология на експериментите

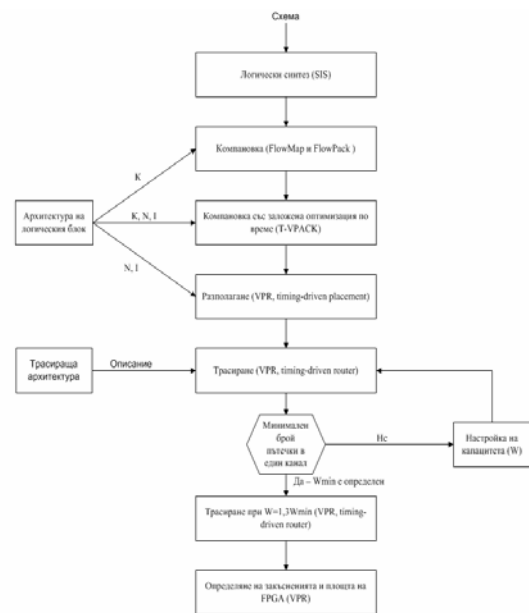
За да отговорим на поставените в началото на това изследване задачи е използвана утвърдена академична методика за изследване на FPGA архитектури [3]. При нея експериментално се синтезират реални схеми с помощта на описаните в [3] софтуерни инструменти за различни FPGA архитектури, които са от значение за конкретното изследване, в резултат на което може да се определи заеманата площ и закъсненията за всяка от синтезираните схеми във всяка от използваните архитектури [3] [6] [11].

Етапите и софтуерните инструменти в тази методика са илюстрирани на фиг. 3. Първоначално, всяка схема преминава през логическа оптимизация, която е независима от използваната технологична база. На този етап се използва програмата за логически синтез *SIS* [16]. Като изключим този етап, всички останали етапи в избраната методика извършват оптимизация по време. Оптимизацията по време е необходима, защото целта ни е да направим заключение относно производителността на архитектурите, които изследваме. Втората стъпка е етапа компоновка (преобразуване на оптимизираната схема/нетлист с логически елементи в нетлист с  $K$ -входни LUT елементи). Тя се извършва с помощта на програмите *FlowMap* и *FlowPack* [7]. На следващият етап се извършва декомпозиция, като всички регистри и LUT елементи се декомпозират (свързват) в логически клъстери, посредством алгоритъма за декомпозиция със заложен оптимизация по време *T-VPACK* [13]. Следващите две стъпки в методиката са разполагане и трасиране. Те също извършват оптимизация по време и се реализират с помощта на програмата *VPR* [2].

Един от основните параметри, които са свързани с моделирането на необходимата площ за всяка една схема при нейната имплементация в FPGA е  $W_{min}$ . Този параметър показва минималния брой пътечки в един канал, необходими за успешното трасиране на дадена схема в зададена FPGA архитектура. Както е описано в [8]  $W_{min}$  заема съществен дял при определяне на логическата плътност за дадена архитектура, защото той определя необходимите ресурси за трасиране за всяка една схема при всяка

една архитектура. За да се определят минималния брой пътечки в канал за успешно трасиране на дадена схема, тя се трасира многократно, като при всяко следващо трасиране се премахват пътечки от архитектурата, докато поредното трасиране се провали.

Ситуацията, при която в FPGA архитектурата се използват само минимално необходимия брой пътечки в канал за трасиране на дадена схема, не е оптимална за измерване нейната производителност, тъй като се работи на ръба на възможностите за трасиране. За да се създадат оптимални условия за измерване на производителността на схемите се добавят близо 30% повече пътечки към минималния брой, след което могат да се измерят закъсненията в схемата. Тези закъснения заедно с заеманата площ се определят от изходните данни, извеждани от програмата *VPR* след финалното трасиране при  $W = 1,3 \cdot W_{min}$ .



Фиг. 3. Етапи и използвани инструменти при експериментите

#### В.1. Моделиране на FPGA архитектурата.

В тази точка се дава кратко описание на модела на площта и модела на закъсненията, разработени от Бетц и описани в [3]. При по-ранни изследвания площта се моделира като се използват параметри, свързани с размера на проводниците [15] или броя на пиновете [9], използвани при трасиране от логическите блокове. На тяхна база се

определя общата площ, както и делът на заеманата от логическите блокове и програмируемите свързващи вериги площ. В ранните изследвания, закъснението се моделира на база броя програмируеми ключове в дадена свързваща верига или чрез използване на опростен модел на LUT елемента [17].

В това изследване всички параметри на схемите и устройствата са моделирани посредством SPICE симулация по 0,35  $\mu\text{m}$  CMOS процес. Моделът е заложен в използвания CAD софтуер (VPR). Направени са следните допускания относно базовата матрична архитектура:

- броя на трасиращите пътечки на всеки канал между логическите блокове е еднакъв за целия FPGA прибор;
- всяка схема се разполага в квадратна мрежа (MxM) от логически клъстери, с най-малкия възможен за разполагане на схемата размер. При изчисляване на заеманата площ вместо (MxM) логически клъстера се използва точния брой на клъстерите, необходими за имплементиране на схемата. Например, ако са необходими 800 логически блока, то мрежата от клъстери ще бъде 29x29, което прави 841 блока в FPGA прибора. За да определим площта обаче използваме само 800 клъстера, вместо 841.

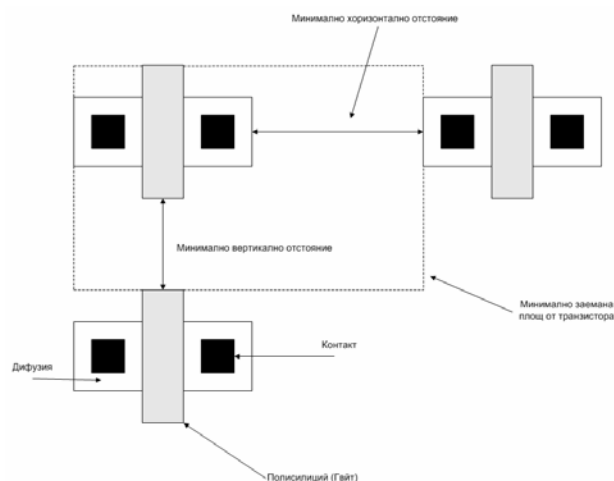
### В.2. Модел на площта

При моделиране на площта [3] се създават подробни транзисторни схеми на всички логически и трасиращи вериги в FPGA прибора. Те включват схеми на LUT елементи, тригери, мултиплексори в логическия блок, мултиплексори в програмируемите свързващи вериги и блоковете за комутация и SRAM клетки. Приема се, че общата площ на FPGA зависи основно от транзисторите и не зависи от свързващите пътечки, тъй като те се разполагат в различни и много на брой слоеве. Двата най-големи производители на PLD потвърждават това предположение.

Процесът на проектиране включва правилно оразмеряване на всички логически елементи и буфери, включително pass-транзисторите в програмируемите свързващи

вериги. В [3] се използва метриката "минимална площ заемана от транзистор" за определяне на площта. Нейната дефиниция е: най-малката възможна площ, заемана от един транзистор, произведен по определена технология плюс минималната площ оставяна около транзистора (фиг. 4).

Всички транзистори в схемата, с размери по-големи от минималния се считат за по-голям брой транзистори с минимална заемана площ, като се взема предвид факта, че двойният размер на транзистора не удвоява заеманата от него площ. Предимството на тази метрика е, че площта се изчислява независимо от технологичния процес на производство на FPGA приборите.



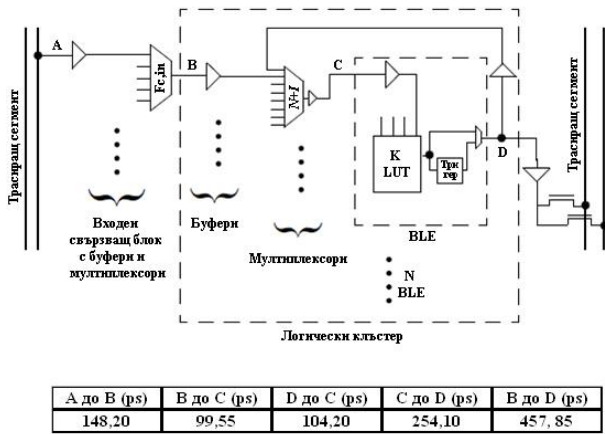
Фиг. 4. Определяне на минимално заеманата площ от транзистор

### В.3. Модел на закъсненията

Създаването на подробни транзисторни схеми е необходимо също и за точно определяне на закъсненията при окончателното разполагане и трасиране на схемата. При сегашните технологични процеси за производство на ИС, които са на подмикронно ниво, ефектът от съпротивлението и капацитета на проводниците става преобладаващ. Този ефект е взет под внимание в използвания модел на закъсненията.

На фигура 5 е показана подробна схема на логическия блок [3].

Времеви стойности показани на фиг. 5 се базират на направени симулации със SPICE за 0,35  $\mu\text{m}$  CMOS технологичен процес на производство [3].



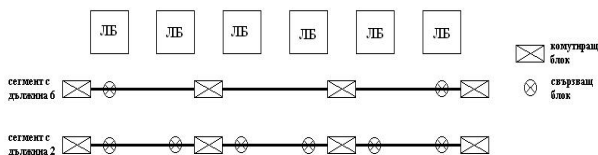
Фиг. 5. Структура на логическия блок и точки за определяне на закъсненията в него

#### В.4. Трасираща архитектура.

За всички експерименти се използва една и съща трасираща архитектура (фиг. 6) със следните параметри:

- трасиращите сегменти са два вида: сегменти с дължина 6, които са 75 % от общия брой сегменти и сегменти с дължина 2, които са 25 % от общия брой сегменти.

- всички сегменти използват „tree state buffers” (буфери с три състояния) като програмируеми ключове.



Фиг. 6. Използвана трасираща архитектура

Избраната трасираща архитектура отговаря на съвременните архитектури, използвани в FPGA приборите на производителя Xilinx.

Съществуват някои нови изследвания [3] [8] [12], които доказват, че при използване на описаната трасираща архитектура се получава добро съотношение между заемана площ и бързодействие за FPGA.

#### Г. Експериментални резултати

За направата на експериментите са използвани 10 еталонни схеми, които са част от утвърдения еталонен набор схеми за тестване и експерименти MCNC [19]. В та-

блица 1 са представени 10-те схеми, като е посочен броя на връзките и броя на съдържащите се в тях 4-входни LUT елементи.

За всяка от представените схеми се извършва декомпозиция, разполагане и разместване, като се променя размера на LUT елемента (броя входове)  $K = 4 \div 7$  и размера на клъстера (броя LUT елементи в логическия блок)  $N = 4, 6, 8$  и  $10$ . Така с 4 различни размери на LUT елемента и 4 различни размери на клъстера се получават общо 16 логически архитектури за FPGA.

#### Г.1. Избор на стойност за експонентата на Рент.

Както бе посочено в началото на доклада, целта е да се потвърди приложимостта на аналитичния израз (1), чрез който могат предварително да се определят броя входове на логическия блок, без за това да са нужни експериментални изследвания. Практиката, която съществува при определяне на броя входове на LB е да се използва напълно свързан клъстер, което означава че броя входове е  $I = K \times N$ . Така се постига пълна използваемост на логиката вътре в логическия блок. От друга страна по-големия брой входове означава по-големи и бавни входни мултиплексори и повече програмируеми ключове, необходими за връзка между логическите блокове, което внася допълнителни закъснения в схемите и увеличава заеманата площ. За постигане на високо ниво на използваемост на BLE в LB не е необходима пълна свързаност ( $K \times N$ ), както това е отбелязано в [4] [5] [10].

Таблица 1. Описание на използваните еталонни схеми от тестовия набор MCNC.

| Схема  | Брой 4-входни LUT елементи | Брой на връзките |
|--------|----------------------------|------------------|
| apex4  | 1262                       | 1271             |
| des    | 1591                       | 1847             |
| diffeq | 1497                       | 1561             |
| dsip   | 1370                       | 1599             |
| ex1010 | 4598                       | 4608             |
| ex5p   | 1064                       | 1072             |
| frisc  | 3556                       | 3576             |
| pdc    | 4575                       | 4591             |
| s38417 | 6406                       | 6435             |
| spla   | 3690                       | 3706             |

Една от задачите, която следва да се реши за постигане на поставената цел е да бъде открито такова  $p$ , при което използваемостта на BLE (базовите логически елементи) е достатъчно висока. Приемаме използваемостта да бъде най-малко 98 %. Това ниво на използваемост води до постигането на добра ефективност по отношение на заеманата площ [4].

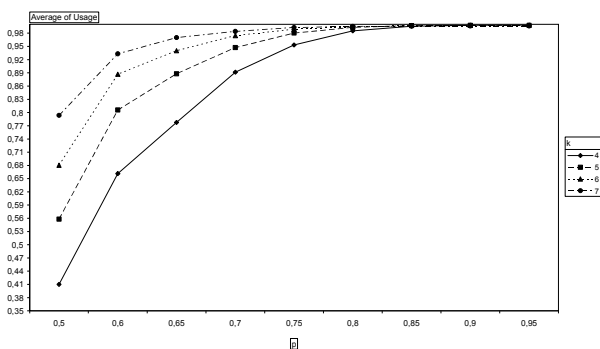
За решаване на тази задача за всяка от 10 еталонни схеми се извършва декомпозиция с програмата T-VPACK при промяна на параметрите  $K$ ,  $N$  и  $I$ .  $I$  се изчислява според (1), като за всяка комбинация за  $K$  и  $N$ ,  $p$  се променя от 0,5 до 0,95 през 0,05. Така се получават общо 144 архитектури, с които се декомпозира всяка от 10-те еталонни схеми. На фиг. 4.1 графично са представени обобщените осреднени резултати от извършените експерименти.

От направените експерименти и обобщените резултати представени на фиг. 7 се вижда, че използваемост над 98% се постига при  $p=0,8$ . Ето защо за следващите експерименти приемаме да работим с константата  $p=0,8$ .

Този резултат съвпада с други изследвания в тази област [3] [6] [15], където е посочено, че за FPGA прибори, най-подходящата стойност за  $p$  е в диапазона от 0,75 до 0,85.

### Г.2. Зависимост между заеманата площ от програмируемите свързващи вериги и параметрите $K$ и $N$ .

В тази точка са обобщени и анализирани експерименталните резултати, които показват зависимостта на площта на FPGA прибора като функция на  $N$  и  $K$ . Тук стойността на  $I$  се определя от (1).

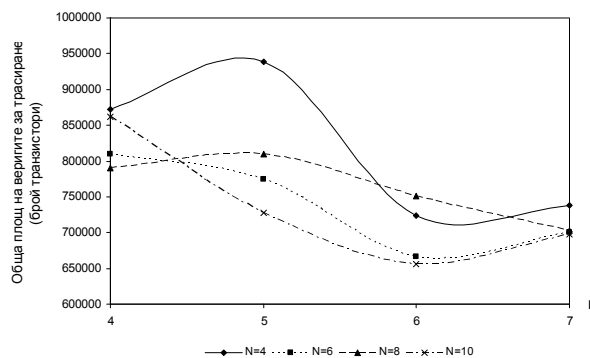


Фиг. 7. Зависимост между използваемостта на BLE и експонентата на Рент

Представените резултати са осреднени геометрично за 10-те използвани еталонни схеми.

Площта се измерва като се отчита общият брой на транзисторите, необходими за реализиране на логическите и трасиращите схеми.

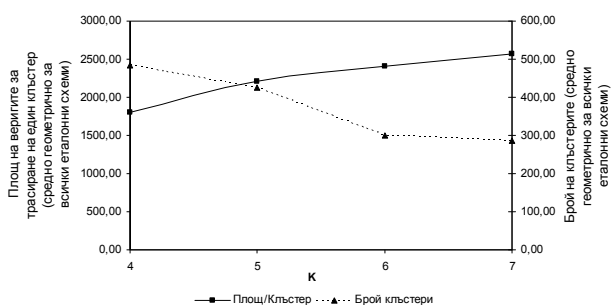
Тъй като площта на веригите за трасиране заема от 80 до 90% от общата площ на FPGA прибора [12], тя е определяща за неговия размер. Фиг. 8 показва, че площта заемана от трасиращите вериги намалява с увеличаване на размера на LUT елемента и достига минимална стойност при  $K=6$ . При  $N=4$  се забелязва постигане на максимум когато  $K=5$ . Този резултат е различен от предишни изследвания, които показват постигане на минимум при  $K=3$  и  $K=4$  [15] или слабо линейно намаление с увеличаване на  $K$  [1].



Фиг.8. Обща площ, заемана от програмируемите свързващи вериги като функция на  $K$  и  $N$

За да обясним този резултат, разлагаме общата площ, заемана от трасиращите ресурси на две компоненти: брой на клъстерите и площ за трасиране, която се пада на един клъстер. Това е показано на фиг. 9. Кривите на тази фигура показват поведението на двете компоненти в зависимост от размера на LUT елемента само при  $N=4$ . Те обаче са представителни и за останалите стойности на  $N$ . Произведението от двете криви дава общата заемана площ от веригите за трасиране. Причината за тенденцията към намаляване и минимума, който се получава при  $K=6$  е в рязкото намаляване на броя клъстери когато  $K$  стане 6. След което следва сравнително бавно намаляване на този брой.

От друга страна заеманата площ за трасиране от един клъстер нараства почти линейно и сравнително бавно. Разликата с цитираното изследване [15] е в използването на по-усъвършенствани програми и алгоритми, които подобряват качеството на трасиране, разполагане и раз местване, което от своя страна води до значително намаляване на необходимото опроводяване. Друга разликата с [15] е че тук използваме различна архитектура за трасиране, която е съставена от два вида трасиращи сегменти, при които се получава по-ефективно опроводяване в FPGA.

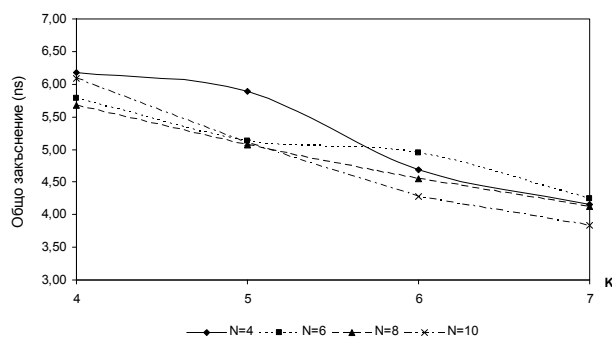


Фиг. 9. Брой на клъстерите и площ на веригите за трасиране за един клъстер като функция на  $K$  (при  $N=4$ )

### Г.3. Зависимост между закъсненията и параметрите $K$ и $N$ .

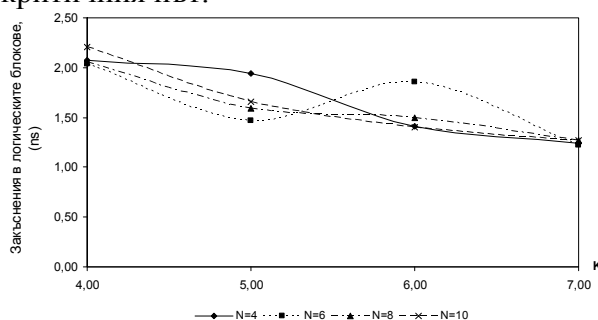
Вторият основен критерий за оценка на ефективността на FPGA архитектурата са закъсненията, които се получават в критичните пътища. Те дават информация за бързодействието и производителността на FPGA. Общото закъснение е сума от закъснението в логическия блок (клъстер) и закъснението в трасиращите вериги. На фиг. 10 е показано средното геометрично общо закъснение на десетте синтезирани еталонни схеми като функция на размера на клъстера ( $N$ ) и LUT елемента ( $K$ ). С увеличаване на  $N$  или  $K$  общото закъснение намалява. Налице са две тенденции, които обясняват това поведение. Когато размера на LUT елемента и размера на клъстера нарастват:

- закъснението в LUT елемента и закъснението в един клъстер се увеличава;
- броя на LUT елементите и броя на клъстерите, участващи в критичните пътища намалява.



Фиг. 10. Обща производителност като функция на  $K$  и  $N$

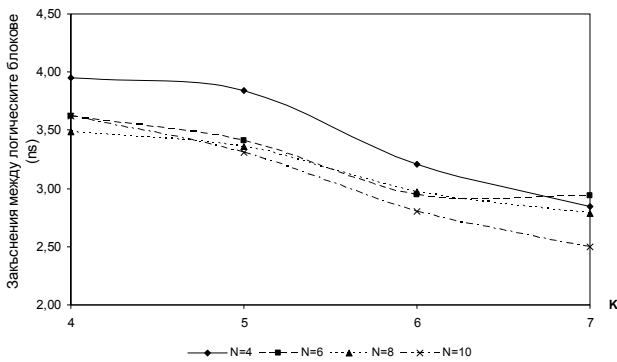
Нека разделим общото закъснение на два компонента: закъснение в логическия блок (което включва закъснението в мултиплексорите и LUT елементите) и закъснение между логическите блокове. На фиг. 11 е показано закъснението, което се получава вътре в логическия блок като функция на  $K$  и  $N$ . С увеличаване на размера на LUT елементите закъсненията намаляват. Това се дължи на по-малкия брой BLE, които участват в изграждането на логическата схема и съответно намаления брой BLE в критичния път.



Фиг. 11. Закъснения в логическия блок като функция на  $K$  и  $N$

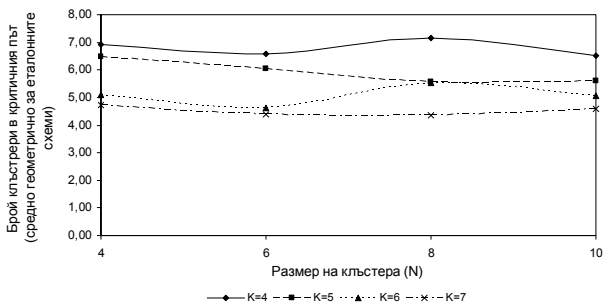
На фиг. 12 е показано закъснението между логическите блокове като функция на  $K$  и  $N$ .

Когато  $K$  се увеличава намалява броя на LUT елементите, участващи в образуването на критичен път, което води до по-малко връзки между логическите блокове (клъстерите). Така се намаляват закъсненията между тях. По същият начин, когато  $N$  нараства, повече връзки се реализират в логическия блок за сметка на връзките между логическите блокове, което води до намаляване на закъсненията в свързващите вериги между клъстерите.



Фиг. 12. Закъснения между логическите блокове като функция на  $K$  и  $N$

Принципно, закъсненията между логическите блокове е по-голямо от закъсненията вътре в клъстера, така че то е определящо за общото закъснение. Вижда се обаче, че с увеличаване на размера на клъстера закъсненията между LB не се променят значително. Това може да се обясни като се разгледа фиг. 13, която показва зависимостта на броя клъстери участващи в критичните пътища от размера на клъстера.

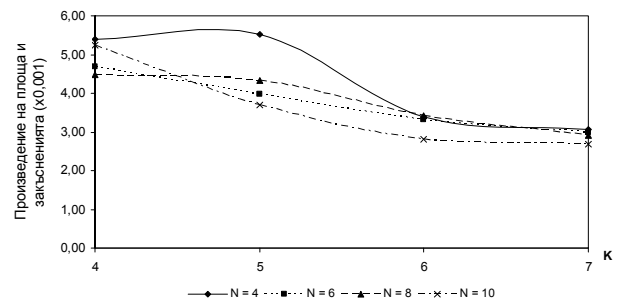


Фиг. 13. Брой на клъстерите в критичния път.

Като обобщение може да се каже, че увеличаването на размера на клъстера има малък ефект върху производителността.

#### Г.4. Обща оценка на зависимостта на площта и закъсненията от параметрите $K$ и $N$ .

Тъй като много често при проектирането и производството на ИС се прави компромис между заемана площ и производителност, за да отговорим на поставените задачи в настоящата работа ще разгледаме влиянието на  $K$  и  $N$  върху производението от площта (измервана в брой транзистори с минимална ширина) и общото закъснение. Тази зависимост е показана на фиг. 14.



Фиг. 14. Зависимост между произведението на заеманата площ и закъсненията от  $k$  и  $N$

Трябва да отбележим, че тук заеманата площ е само от веригите за трасиране. Това води до тенденцията за постоянно намаление на произведението при увеличаване на размера на LUT елемента и размера на клъстера. Може да се очаква че при използване на общата площ, заемана от трасиращите вериги и логическите блокове в това произведение ще се получи леко увеличение на при  $K > 7$  и  $N > 6$ , тъй като размера на логическия блок при тези стойности става значителен фактор.

## ЗАКЛЮЧЕНИЕ

В настоящия доклад е представено едно изследване на влиянието на логическите архитектурни параметри на FPGA приборите върху заеманата площ и закъсненията в тях. Обобщените резултати, които бяха представени в доклада до голяма степен съвпадат с други подобни изследвания в областта [1]. Като тук се забелязва тенденция на непрекъснато, макар и минимално, намаление на закъсненията при увеличаване на  $K$  и  $N$ . Това ни дава основание да твърдим, че изведената на база правилото на Рент аналитична зависимост (1) е подходяща за приложение при проектиране на нови фамилии FPGA прибори. Изчисляването на броя входове на LB, посредством (1) дава една оптимална стойност, която не намалява използваемостта на BLE вътре в LB, като същевременно не се налага този брой да бъде ненужно голям (3). Така се намалява заеманата от входните мултиплексори площ и същевременно се увеличава бързодействието им, а от там и общото бързодействие на прибора.



## ЛИТЕРАТУРА

- [1] Ahmed E., "The Effect of LUT and Cluster Size on Deep-Submicron FPGA Performance and Density", University of Toronto, 2001
- [2] Betz V. et All, VPR and T-VPack1 User's Manual, University of Toronto 2008.
- [3] Betz V., Rose J., and Marquardt A.. "Architecture and CAD for Deep-Submicron FPGAs". Kluwer Academic Publishers, New York, 1999.
- [4] Betz V. and Rose J., "Cluster-Based Logic Blocks for FPGAs: Area-Efficiency vs. Input Sharing and Size", IEEE Custom Integrated Circuits Conference, Santa Clara, CA, 1997, pp. 551-554.
- [5] Betz V. and Rose J., "How Much Logic Should Go in an FPGA Logic Block?", IEEE Design and Test Magazine, Spring 1998, pp. 10-15.
- [6] Brown S., et All, "Field-Programmable Gate Arrays", Kluwer Academic Publishers, 1992.
- [7] Cong J. and Ding Y., "FlowMap: An Optimal Technology Mapping Algorithm for Delay Optimization in Lookup-Table Based FPGA Designs", IEEE Trans. on CAD, Jan. 1994, pp.1-12.
- [8] Fang W., A Modeling Routing Demand for Early-Stage FPGA Architecture, Master Thesis, University of Toronto, 2007.
- [9] Hill D. and N-S Woo, "The Benefits of Flexibility in Look-up Table FPGAs", in FPGAs, W. Moore and W. Luk Eds., Abingdon 1991, edited from the Oxford 1991 International Workshop on FPGAs, pp. 127-136.
- [10] Kaptanoglu S., et All, "A new high density and very low cost reprogrammable FPGA architecture", FPGA'99, Monterey, CA, 1999, pp. 3-12.
- [11] Kouloheris J. and El Gamal A., "FPGA Performance vs. Cell Granularity", Proc. of Custom Integrated Circuits Conference, May 1991, pp. 6.2.1 - 6.2.4.
- [12] Kuon I., FPGA Architecture: Survey and Challenges, Now Publishing, 2007.
- [13] Marquardt A., Betz V., and J. Rose. "Using Cluster-Based Logic Blocks and Timing-Driven Packing to Improve FPGA Speed and Density". In ACM/SIGDA FPGA, 1999.
- [14] Maxfield C., The Design Warrior's Guide to FPGAs, Newnes, 2004.
- [15] Rose J., et All, "Architecture of Field-Programmable Gate Arrays: The Effect of Logic Functionality on Area Efficiency", IEEE Journal of Solid-State Circuits, 1990, pp. 1217-1225.
- [16] Sentovich E.M. et All., "SIS: A System for Sequential Circuit Analysis", Tech. Report No. UCB/ERL M92/41, University of California, Berkeley, 1990.
- [17] Singh S., et All, "The Effect of Logic Block Architecture on FPGA Performance", IEEE Journal of Solid-State Circuits, 1992, pp. 281-287.
- [18] W. E. Donath. Placement and Average Interconnection lengths of Computer Logic. IEEE Trans. on Circuits and Systems, CAS-26(4):272-277, 1979.
- [19] Yang S., "Logic Synthesis and Optimization Benchmarks, Version 3.0", Tech. Report, Microelectronics Centre of North Carolina, 1991.