АКАДЕМИЈА ТЕХНИЧКО-УМЕТНИЧКИХ СТРУКОВНИХ СТУДИЈА БЕОГРАД

ОДСЕК ВИСОКА ШКОЛА ЕЛЕКТРОТЕХНИКЕ И РАЧУНАРСТВА

**Лазар Гајић**

**Упоредна анализа реализација множача у програмабилној логици**

**- завршни рад -**



Београд, новембар 2023.

Кандидат: **Лазар Гајић**

Број индекса: **ЕЛИТЕ-7/19**

Студијски програм: **Електроника и телекомуникације**

Тема: **Упоредна анализа реализација множача у програмабилној логици**

Основни задаци:

**1. Анализа различитих решења бинарних множача погодних за имплементацију у програмабилној логици**

**2. Имплементација изабраних решења у FPGA колу**

**3. Тестирање дизајна симулацијом и у реалном времену на развојном систему**

Ментор:

Београд, новембар 2023 годинe.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Др Драгана Прокин, проф. ВИШЕР

Резиме:

Циљ овог завршног рада је био анализа и имплементација брзих множача. У првом делу обрађене су основне компоненте потребне за имплементацију множача. У другом делу представљен је начин реализације датих решења у развојном окружењу Quartus II као и анализа самих перформанси имплементираних решења. Рад имплементираних решења је тестиран симулацијом и на развојном систему.

**Кључне речи:** множач, регистар, померачки регистар, сабирач

ABSTRACT:

The aim of this final paper was the analysis and implementation of fast multipliers. The first part covered the basic components necessary for the implementation of the multipliers. The second part presents the actual implementation process of the given solutions in the development environment Quartus II, as well as the analysis of the performance of the implemented multiplier solutions. The functionality of implemented solutions was tested through simulation and on the development system.

**Key words**: multiplier, register, shift register, adder

САДРЖАЈ:

[1. УВОД 1](#_Toc151068552)

[2. АНАЛИЗА РЕШЕЊА БИНАРНИХ МНОЖАЧА ПОГОДНИХ ЗА ИМПЛЕМЕНТАЦИЈУ У ПРОГРАМАБИЛНОЈ ЛОГИЦИ 2](#_Toc151068553)

[2.1. Множач са померачким регистрима и једним сабирачем 2](#_Toc151068554)

[2.2. Множач са прихватним регистрима и више сабирача 3](#_Toc151068555)

[2.3. Основне Компоненте бинарних множача 6](#_Toc151068556)

[2.3.1. Регистри 6](#_Toc151068557)

[2.3.2. Сабирачи 8](#_Toc151068558)

[3. ИМПЛЕМЕНТАЦИЈА ИЗАБРАНИХ РЕШЕЊА МНОЖАЧА У FPGA КОЛУ 10](#_Toc151068559)

[3.1. Имплементација Потребних модула за реализацију множача 10](#_Toc151068560)

[3.1.1. Имплементација прихватног регистра са паралелним уписом и читањем 10](#_Toc151068561)

[3.1.2. Имплементација померачког регистра 14](#_Toc151068562)

[3.1.3. Имплементација потпуног сабирача 19](#_Toc151068563)

[3.1.4. Имплементација линијског разделника 20](#_Toc151068564)

[3.2. Имплементација множача са померачким регистрима и једним сабирачем 21](#_Toc151068565)

[3.3. Иплементација множача са прихватним регистрима и три сабирача 22](#_Toc151068566)

[3.4. Упоредна Анализа заузећа ресурса и брзине рада имплементираних множача 22](#_Toc151068567)

[3.4.1. Имплементација у програмабилном логичком колу CPLD типа 23](#_Toc151068568)

[3.4.2. Имплементација у програмабилном логичком колу FPGA типа 24](#_Toc151068569)

[4. ТЕСТИРАЊЕ ДИЗАЈНА СИМУЛАЦИЈОМ 27](#_Toc151068570)

[4.1. Тестирање прихватних регистара са паралелним уписом и читањем симулацијом 27](#_Toc151068571)

[4.2. Тестирање померачких регистара симулацијом 28](#_Toc151068572)

[4.3. Тестирање потпуних сабирача симулацијом 31](#_Toc151068573)

[4.4. Тестирање множача симулацијом 32](#_Toc151068574)

[4.4.1. Тестирање множача са померачким регистрима и једним сабирачем 32](#_Toc151068575)

[4.4.2. Тестирање множача са прихватним регистрима и више сабирача 32](#_Toc151068576)

[5. ТЕСТИРАЊЕ МНОЖАЧА У РЕАЛНОМ ВРЕМЕНУ 33](#_Toc151068577)

[6. ЗАКЉУЧАК 37](#_Toc151068579)

[7. ИНДЕКС ПОЈМОВА 38](#_Toc151068580)

[8. ЛИТЕРАТУРА 39](#_Toc151068581)

[9. ИЗЈАВА О АКАДЕМСКОЈ ЧЕСТИТОСТИ 40](#_Toc151068582)

# УВОД

Главни циљ овог дипломског рада је анализа и имплементација различитих варијанти бинарних множача. У овом раду су анализирана два типа множача, која се разликују по хардверској сложености и брзини рада.

Прво решење, са померачким регистрима и једним сабирачем, захтева већу хардверску сложеност и спорије се долази до производа. Логика овог решења се заснива на имплементацији множења "са потписивањем" у бинарном бројном систему, које се извршава у неколико корака (тактова). За сампри чему је за извршење операције множења потребно између три и седам периода сигнала такта.

За разлику од првог решења, у другом решењу које се састоји из прихватних регистара и већег броја сабирача, до резултата множења се долази у краћем временском интервалу. Логика овог решења заснована је на сличном принципу као код првог решења, али се до производа долази у једном такту истовременим обављањем већег броја сабирања.

У првом делу рада, спроведена је анализа основних модула потребних за имплементирање множача, као што су: сабирачи, логичка кола, регистри, итд.

У другом делу извршена је имплементација датих решења, као и потребних модула за рад самих множача. Оба решења множача су описана графичким начином уношења дизајна у софтверском развојном окружењу Quartus II. Такође је анализирано заузеће ресурса и брзина рада у изабраним програмабилним логичким колима CPLD и FPGA типа.

У трећем делу су тестиране компоненте потребне за рад множача, креирана је тест шема и изведена симулација рада предложених решења у софтверском развојном окружењу Quartus II. Такође након извршене симулације предложена решења множача су тестирана и на развојном систему.

# АНАЛИЗА РЕШЕЊА БИНАРНИХ МНОЖАЧА ПОГОДНИХ ЗА ИМПЛЕМЕНТАЦИЈУ У ПРОГРАМАБИЛНОЈ ЛОГИЦИ

Када се пројектују бинарни множачи, увек се прави компромис између брзине и сложености хардвера за имплементацију. У овом дипломском раду анализиране су имплементације две типа множача са различитим брзинама и различитим сложеностима хардверске реализације.

## Множач са померачким регистрима и једним сабирачем

Први тип множача се заснива на коришћењу једног сабирача и два померачка регистара, чији садржај може да се помера улево и удесно (слика 2.1).

*A diagram of a wire connected to a couple of square boxes

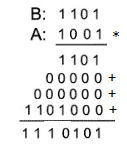
Description automatically generated*

*Слика 2.1 – Множач са једним сабирачем и померачким регистрима*

Ово решење се заснива на множењу "са потписивањем" у бинарном бројном систему. Најпре се у два регистра учитају подаци које треба помножити, а затим се гледа бит најмање тежине другог чиниоца:

* уколико је бит једнак јединици ("1") вредност првог чиниоца се додаје у коначну суму
* уколико је бит једнак нули ("0") коначна сума остаје непромењена.

Након тога се први чинилац логички помера за једно место улево, а други чинилац за једно место удесно и понавља се претходни корак. Кораци 2 и 3 се понављају све док вредност другог чиниоца не постане једнака нули, након чега се добија производ множења. У наставку се налази пример реализације операције бинарног множења бројева 1310 (11012) и 910 (10012), који је приказан на слици 2.2.

**

Slika na kojoj se nalazi tekst, snimak ekrana, Font, broj

Opis je automatski generisan

*Слика 2.2 – Пример множења бројева 11 и 9 у бинарној аритметици применом једног сабирача и померачких регистара*

Брзина извршавања целокупне операције је минимално два такта (иницијализација података и чување резултата), односно зависи од броја бита са којим је представљен други операнд. У приказаном примеру, операнди су били четворобитни, тако да је за извршавање саме операције множења било потребно четири такта.

## Множач са прихватним регистрима и више сабирача

Други тип множача се заснива на коришћењу више сабирача у циљу добијања бржег резултата у односу на множач са једним сабирачем. Ово решење не захтева употребу померачких регистара, већ се користе прихватни регистри са паралелним уписом и читањем (слика 2.3).

*A diagram of a circuit

Description automatically generated*

*Слика 2.3 – Множач са више сабирача и прихватним регистрима*

Решење са прихватним регистрима и више сабирача, приказано на слици 2.3, операцију множења обавља на сличан начин као и решење приказано на слици 2.1. Главна разлика између ова два решења је у томе што решење са слике 2.3 операцију множења извршава у једном такту применом више сабирача. У наставку је објашњен принцип реализације операције множења на два примера који су приказани на слици 2.4:

* 1510 (11112) и 1510 (11112)
* 1310 (11012) и 910 (10012)

Ознаке за вредности у примерима на слици 2.4 су следеће:

* (X) – Вредност која се налази у регистру X
* X[N] – Вредност бита на позицији N регистра X
* SNout – Вредност на излазу N-тог сабирача
* SNin – Вредност на улазу N-тог сабирача
* X[M..N] – Вредност бита на позицијама од M до N

Потребно је приметити да се након сваке извршене операција сабирања, бит најмање тежине излаза из сабирача уписује у одговарајући бит регистра P. Након завршетка последње операције сабирања и уписа бита најмање тежине у одговарајући бит регистра P, уписује се и сам резултат сабирања у регистар P на позицију бита највеће тежине, што представља и завршетак операције множења.

Графички приказ релизације множења бројева 11 и 9, у коме се операнди налазе у регистрима А и B, док се резултат чува у регистру P, налази се на слици 2.5.

Slika na kojoj se nalazi tekst, snimak ekrana, Font, broj

Opis je automatski generisan

Слика 2.4 – Примери множења бројева у бинарној аритметици са четири сабирача и прихватним регистрима

Slika na kojoj se nalazi tekst, snimak ekrana, Font, broj

Opis je automatski generisan

Слика 2.5 – Графички приказ множења бројева 11 и 9 у бинарној аритметици са четири сабирача и прихватним регистрима

## основне Компоненте бинарних множача

Да би се извршила имплементација множача у програмабилној логици графичким описом дизајна, неопходно је имплементирати основне хардверске компоненте за реализацију множача, које обухватају регистре и бројаче. Принцип рада ових компонената је описан у наставку.

### Регистри

Регистри су секвенцијалне мреже, чија је улога привремено чување података добијених у току извршавања обраде или преноса података. Сваки регистар има могућност чувања n бита података, који могу да имају вредности логичка ’0’ или ’1’. У зависности од реализације регистра, бити могу да се уписују серијски (један по један) или паралелно (сви истовремено), а исто важи и за начин очитавања. Тренуци у којима се врше упис и читање података код синхроних регистара дефинисани су узлазном или силазном ивицом такта.

#### Регистри са паралелним уписом и читањем

Прихватни регистар са паралелним уписом и читањем података, који је приказан на слици 2.6, осим прикључака за паралелни упис података (Din), паралелно читање излазних података (Dout) и сигнал такта (Clk), садржи и додатне контролне сигнале: за дозволу учитавања (Load), аутоматско постављање излаза регистра на вредност логичка ’1’ или "сетовање" (Set), аутоматско постављање излаза регистра на вредност логичка ’0’ или "ресетовање" (Reset) и сигнал дозволе промене садржаја регистра под дејством такта (CE). Излазни подаци (Dout) дају тренутни садржај регистра.

A diagram of a rectangular object with text

Description automatically generated

Слика 2.6 – Логичи симбол прихватног регистра са паралелним уписом и читањем

Када су улазни подаци присутни на улазу за паралелни упис података, тада је присутан и сигнал за дозволу учитавања којим се активира учитавање новог садржаја (који је присутан на улазу) у регистар. Сигнали сета и ресета могу бити синхрони или асинхрони (при томе сет и ресет могу бити различити по том питању, тј. један може бити синхрон, а један асинхрон). У случају активирања сигнала синхроног сета или ресета, садржај регистра се на прву наредну ивицу такта сетује на све ’1’ или ресетује на све ’0’. Асинхрони сет и ресет раде по идентичном принципу као и у случају синхроних флип-флопова. Наравно, сигнали сета и ресета су опциони, у смислу може да постоји само један од њих, оба или ниједан. Такође, могу постојати истовремено и асинхрони и синхрони сет сигнали (исто важи и за ресет сигнале), али то се углавном не користи. Као код синхроних флип-флопова, и код регистара не би смели да буду истовремено активни и сет и ресет сигнал.

Сигнал дозволе за такт је такође опцион, а ако постоји онда је његова улога идентична његовој функцији у случају синхроних флип-флопова. То значи да кад је сигнал дозволе неактиван, регистар ’замрзава’ своје стање и тада може да реагује само на асинхрони сет или ресет сигнал. На слици 2.1 је приказана варијанта са синхроним ресетом и асинхроним сетом, а сви контролни сигнали (Load, CE, Set, Reset) су активни на логичкој ’1’.

#### Померачки регистри

Померачки (shift) регистри су посебна класа регистара. Ови регистри могу да имају све опције као и обични, односно стационарни регистри, са додатном могућношћу померања или ротирања садржаја регистра. Померање/ротирање може да се ради у само једну страну (улево или у десно) или постоји могућност избора смера померања/ротирања. У случају померања садржаја упражњена места се типично попуњавају нулама, али могу да се имплементирају и друге опције за попуњавање садржаја упражњених места. Принципи померања и ротирања садржаја регистра су приказани на слици 2.7. У примерима са слике се врши померање/ротирање за једно место у сваком тактном интервалу.

Додатни контролни сигнали за померачке регистре у односу на обичне регистре су: сигнал за померање/ротирање улево (Shl) и сигнал за померање/ротирање удесно (Shr). Уколико је померање/ротирање могуће у само једну страну, биће присутан само један од два наведена сигнала. Уколико су могућа померања/ротирања у обе стране постоји и варијанта која користи сигнал за померање/ротирање (Shift) и сигнал за избор смера за померање/ротирање (Dir). У случају када је сигнал за померање/ротирање активан врши се померање садржаја регистра, док у супротном регистар задржава тренутно стање.

A diagram of a diagram

Description automatically generated

Слика 2.7 – Принцип померања и ротирања садржаја померачких регистара

Када постоји и сигнал за учитавање (Load), тај сигнал има приоритет над сигналима за померање/ротирање. Уколико се код бидирекционих померачких регистара (који могу да померају садржај у обе стране) користи варијанта са сигналима Shl и Shr, тада не смеју та два сигнала бити истовремено активна. Величина помераја (за колико места ће садржај регистра бити померен у једном тактном интервалу) је углавном статичка тј. не може да се мења и типично износи један. Неки померачки регистри омогућавају и да се динамички дефинише величина помераја, али с обзиром да такво својство повећава хардверске ресурсе оваква опција се користи само када је заиста неопходна. Логички симбола за померачки регистар са описаним прикључцима је дат на слици 2.8.

A diagram of a shift register

Description automatically generated

Слика 2.8 – Логички симбол померачког регистра

Померачки регистри се могу користити за имплементирање функција множења и дељења садржаја померачког регистра са бројевима који су степена двојке (2n), где је n величина помераја. Померање удесно одговара операцији дељења, а померање улево операцији множења. У оба случаја се подразумева да се упражњена места попуњавају са нулама. На пример, ако је n=1, тада је померање удесно за једно место еквивалентно дељењу садржаја померачког регистра са два, а померање улево за једно место је еквивалентно множењу садржаја померачког регистра са два.

### Сабирачи

Основна кола у бинарној аритметици су сабирачи. Операција одузимања се реализује сабирањем у другом комплементу. Операција множења се често изводи коришћењем сабирачких кола. Основну јединицу потпуног сабирача, који чини основни логички блок за реализацију вишебитних бинарних сабирача чини полусабирач, који сабирају два једнобитна бинарна броја према табели која је приказана на слици 2.9. Резултат сабирања у децималном бројном систему може бити 0, 1 или 2. Најмање значајан бит у резултату представља збир сабирања (sf) док је најзначајнији бит (c - енгл.: carry) представља пренос и формира улаз за следећи степен. На основу комбинационе табеле могу да се добију логички изрази који описују рад кола полусабирач, чија је шема приказана на слици 2.9.

A diagram of a machine

Description automatically generated

A math equation with black text

Description automatically generated

Слика 2.9 – Комбинациона табела, логички изрази и логичка шема полусабирача

Недостатак полусабирача је да се не могу повезати у каскаду како би сабирали вишебитне бројеве. За ово је потребно решити сабирање бита преноса из претходног полусабирача са битовима следећег кола за сабирање, односно неопоходно је коришћење потпуног сабирача. Коло потпуног сабирача се састоји од три улаза, од којих су два за једнобитне податке које треба сабрати, док је трећи бит намењен за пренос који је настао сабирањем бита мање тежине (два бита из нижег разреда бинарног записа сабирака) Рад потпуног сабирача може да се опише комбинационом табелом на слици 2.10, на основу које су добијени логички изрази.

Slika na kojoj se nalazi tekst, snimak ekrana, Font, broj

Opis je automatski generisan

Слика 2.10. – Комбинациона табела и логички изрази потпуног сабирача

Логичка шема потпуног сабирача, добијена на основу логичких израза и симбол потпуног сабирача приказани су на слици 2.11.

A diagram of a circuit

Description automatically generatedA white square with black text

Description automatically generated

Слика 2.11 – Логичка шема и симбол потпуног сабирача

Каскадном везом потпуних сабирача на начин који је приказан на слици 2.12, могуће је сабирање вишебитних бројева. Формирање збира код дате мреже се врши паралелно, с тим да се битови преноса простиру серијски кроз поједине модуле сабирача, што значајно успорава рад кола. Постоје кола која помоћу двостепене логичке мреже одређују да ли се може очекивати бит преноса при сабирању неких бита. Такво решење значајно убрзава рад сабирача.

A diagram of a circuit

Description automatically generated

Слика 2.12 – Сабирање два четворобитна броја каскадном везом потпуних сабирача

# ИМПЛЕМЕНТАЦИЈА ИЗАБРАНИХ РЕШЕЊА МНОЖАЧА У FPGA КОЛУ

Имплементација изабраних решење множача урађена је под претпоставком да су чиниоци дужине четири бита. За опис дизајна множача коришећен је графички едитор софтвера Quartus II, у коме је примењен модуларан и хијерархијски принцип пројектовања хардвера. У графичком едитору су најпре описани основни логички елементи регистара и сабирача који се налазе у архитектури множача, као једнобитне компоненте. Основни логички елементи су дефинисани као компоненте дизајна са одговарајућим улазним и излазним сигналима, преко којих су затим повезани хијерархијским принципом пројектовања у сложеније хардверске целине, као што су вишебитни регистри и сабирачи, чиме су добијени основни модули за реализацију множача. Графички опис множача представља највиши ниво хијерархије, на коме су повезани реализовани модули у зависности од изабраног решења множача.

## Имплементација Потребних модула за реализацију множача

### Имплементација прихватног регистра са паралелним уписом и читањем

Основни логички елемент прихватног регистра са паралелним уписом и читањем је једнобитни прихватни регистар који се састоји од једног флип-флопа и комбинационе мреже која дефинише промену стања на излазу флип флопа (комбинационе мреже побуде). Комбинациона мрежа побуде сачињена је од неколико логичких кола (OR, AND, NOT). У наставку су дати сви улазни и излазни сигнали једнобитног прихватног регистра и њихова објашњења.

* Li – једнобитни улазни сигнал који се складишти
* LD – контролни сигнал уписа
* CL – контролни сигнал брисања садржаја
* CLK – сигнал такта
* Output – излазни бит регистра

За генерисање комбинационе мреже побуде D флип-флопа, односно одређивање побудног сигнала Di, потребно је попунити табеле прелаза стања из којих може да се види промена на излазу флип-флопа у зависности од промене контролних сигнала LD и CL. Добијене изразе за побудне контролне сигнале на основу табела промене стања треба објединити у један побудни сигнал Di. Табела прелаза стања за контролни сигнал уписа (LD) дата је на слици 3.1. Објашњење сигнала у табели прелаза стања дати су у наставку:

* LD – контролни сигнал уписа
* li – податак који је потребно уписати у флип-флоп
* Ai – податак који се налази у флип-флопу
* Ai(t+1) – податак у који ће се налазити у флип-флопу у следећем такту
* D – Излаз флип-флопа који је еквивалентан сигналу Ai(t+1)

Slika na kojoj se nalazi tekst, broj, snimak ekrana, kvadrat

Opis je automatski generisan

*Слика 3.1 – Табела прелаза стања на основу промене контролног сигнала уписа (LD)*

На основу комбинационе табеле са слике 3.1 могуће је попунити Карноову мапу за сигнал побуде Dld. Карноова мапа за побудни сигнал Dld као и добијени израз за побудни сигнал приказани су на слици 3.2. Израз за побудни сигнал Dld добијен је одређивањем минималне дисјунктивне форме уз помоћ Карноове мапе.

Slika na kojoj se nalazi tekst, snimak ekrana, dijagram, broj

Opis je automatski generisan

*Слика 3.2 – Карноова мапа и израз за побудног сигнал Dld*

Применом сличног принципа попуњена је и табела прелаза стања за флипо-флопо у зависности од промене контролног сигнала за брисање (CL), која је приказана на слици 3.3. У овој табели сигнали Ai, Ai(t+1) и D имају исту функцију као у табели прелаза стања приказаној на слици 3.1.

Slika na kojoj se nalazi broj, linija, kvadrat, Četvorougao

Opis je automatski generisan

*Слика 3.3 – Табела прелаза стања на основу промене контролног сигнала брисања CL*

Карноова мапа и израз за побудни сигнал Dcl приказани су на слици 3.4. Израз за побудни сигнал Dcl добијен је одређивањем минималне дисјунктивне форме као и у претходном примеру.

Slika na kojoj se nalazi dijagram, tekst, Plan, dizajn

Opis je automatski generisan

*Слика 3.4 – Карноова мапа и израз за побудни сигнал Dcl и Di*

Обједињавањем израза побудних сигнала Dld, Dcl и применом основних правила Булове алгебре, добијен је коначан израз побуде (Di) једнобитног прихватног регистра, који је приказан на слици 3.5.

Slika na kojoj se nalazi tekst, Font, logotip, belo

Opis je automatski generisan

*Слика 3.5 – Израз за побудни сигнал Di*

На основу израза пза обудни сигнал D,i генерисана је комбинациона мрежа побуде једнобитног прихватног регистра. Комплетна шема једнобитног прихватног регистара приказана је слици 3.6.

*A diagram of a circuit

Description automatically generated*

*Слика 3.6 – Имплементација једнобитног прихватног регистра*

Графички опис једнобитног прихватног регистра је сачуван као симбол компоненте, која представља основни логички елемент у дизајну двобитног прихватног регистра са паралелним уписом и читањем.

Двобитни прихватни регистар са паралелним уписом и читањем се састоји од два једнобитна прихватна регистара, истих улазних и излазних сигнала као једнобитни регистар с тим што су улазни и излазни сигнали података ширине два бита (слика 3.7).

*A diagram of a computer

Description automatically generated*

*Слика 3.7 – Имплементација двобитног регистра са паралелним уписом и читањем*

Графички опис двобитног прихватног регистра са паралелним уписом и читањем је сачуван као симбол компоненте, која представља основни логички елемент у дизајну четворобитног прихватног регистра са паралелним уписом и читањем, који је приказан на слици 3.8.

*A diagram of a computer

Description automatically generated*

*Слика 3.8 – Имплементација четворобитног прихватног регистра са паралелним уписом и читањем*

Графички опис четворобитног прихватног регистра са паралелним уписом и читањем је сачуван као симбол компоненте, која представља основни логички елемент у дизајну осмобитног прихватног регистра, са паралелним уписом и читањем који је приказан на слици 3.9.

*A diagram of a computer

Description automatically generated*

*Слика 3.9 – Имплементација осмобитног прихватног регистра*

### Имплементација померачког регистра

Основни логички елемент померачког регистра је једнобитни померачки регистар који се састоји од једног флип-флопа и комбинационе мреже побуде. Комбинациона мрежа побуде сачињена је од неколико логичких кола (AND, OR, NOT).

Померачки регистар има следеће улазне и излазне сигнале:

* Li – једнобитни улазни сигнал који се складишти
* LD – контролни сигнал уписа
* IR – бит који се уписује на крајњу леву позицију приликом логичке операције померања у десно (у овом случају то је и једина позиција)
* SR – контролни сигнал операције померања у десно
* IL – бит који се уписује на крајњу десну позицију приликом логичке операције померања улево (у овом случају то је и једина позиција)
* SL – контролни сигнал операције померања улево
* Output – излазни бит померачког регистра
* CLK – сигнал такта

Слично као и код једнобитног прихватног регистра, генерисање комбинационе мреже побуде D флип-флопа, односно одређивање побудног сигнала Di, врши се попуњавањем табеле прелаза стања за флип-флоп у зависности од сваког контролног сигнала (LD, SR, SL), затим генерисањем побудних израза у зависности од контролних сигнала и обједињавањем тих сигнала у један побудни сигнал Di. Табела промене стања на основу сигнала уписа (LD) као и израз побуде не разликују се у односу на једнобитни прихватни регистар. Табела прелаза стања у зависности од сигнала уписа једнобитног прихватног регистра дата је на слици 3.10.

Slika na kojoj se nalazi broj, tekst, kvadrat, snimak ekrana

Opis je automatski generisan

Слика 3.10 – Табела прелаза стања на основу промене контролног сигнала операције померања улево SL

Табела прелаза стања добијена је на сличан начин као и код једнобитног прихватног регистра. Сигнал SL представља контролни сигнал операције померања улево, сигнал IL представља податак који треба да се нађе у флип-флопу након извршене операције померања у лево. Остали сигнали имају исту функционалност као и код једнобитног прихватног регистра.

Карноова мапа и израз за побудни сигнал Dsl приказани су на слици 3.11. Израз побудног сигнала Dsl добијен је одређивањем минималне дисјунктивне форме.

Slika na kojoj se nalazi tekst, dijagram, linija, Font

Opis je automatski generisan

Слика 3.11 – Карноова карта и израз за побудни сигнал Dsl

Табела прелаза стања у зависности од контролног сигнала операције померања удесно (SR) приказана је на слици 3.12. Сигнали у табели имају сличну функционалност као и сигнали операције померања улево.

Slika na kojoj se nalazi tekst, broj, snimak ekrana, kvadrat

Opis je automatski generisan

Слика 3.12 – Табела прелаза стања на основу промене контролног сигнала операције померања улево SR

На исти начин као и у претходним примерима попуњена је Карноова мапа и одрећен побудни сигнал Dsr. Карноова мапа и побудни сигнал Dsr приказани су на слици 3.13.

Slika na kojoj se nalazi tekst, dijagram, snimak ekrana, broj

Opis je automatski generisan

Слика 3.13 – Карноова мапа и израз за побуднисигнал Dsr

Обједињавањем израза побудних сигнала (Dld, Dsl и Dsr) и применом основних теорема Булове алгебре добијен је коначан израз за побудни сигнал Di једнобитног померачког регистрао, који је приказан на слици 3.14.

Slika na kojoj se nalazi tekst, Font, belo

Opis je automatski generisan

Слика 3.14 – Израз побудног сигнала Di

На основу израза побудног сигнала Di генерисана је комбинациона мрежа побуде једнобитног померачког регистра. Комплетна шема једнобитног померачког регистара приказана је слици 3.15.

A diagram of a circuit

Description automatically generated

Слика 3.15 – Имплементација једнобитног померачког регистра

Графички опис једнобитног померачког регистра је сачуван као симбол компоненте, која представља основни логички елемент у дизајну двобитног померачког регистра.

Двобитни померачки регистар се састоји од два једнобитна померачка регистара, који има исте улазне и излазне сигнале као једнобитни регистар, с тим што су улазни и излазни сигнали података ширине два бита (слика 3.16).

A diagram of a circuit

Description automatically generated

Слика 3.16 – Имплементација двобитног поморачког регистра

За разлику од двобитног прихватног регистра, шема двобитног померачког регистра не може се реализовати једноставним спајањем улазних сигнала са одговарајућим пиновима једнобитног померачког регистра. Проблем настаје у извршавању операција логичког померања улево и удесно. Приликом извршавања операције померања улево потребно је пребацити садржај једнобитног померачког регистра који одговара биту мање тежине у једнобитни померачки регистар који одговара биту веће тежине. Такође, потребно је учитати нову вредност у једнобитни померачки регистар који представља бит мање тежине. То је остварено повезивањем излаза једнобитног померачког регистра који одговара биту мање тежине са улазним пином IL једнобитног померачког регистра који одговара биту веће тежине и повезивањем улазног сигнала IL са улазним пином IL једнобитног померачког регистра који одговара биту мање тежине. Коректно извршавање операције померања удесно реализовано је обрнутим поступком у односу на операцију померања улево.

Графички опис двобитног померачког регистра је сачуван као симбол компоненте, која представља основни логички елемента у дизајну четворобитног померачког регистра, који је приказан на слици 3.17.

Реализација четворобитног померачког регистра у графичком едитору остварена је сличним поступком као и код двобитног померачког регистра, одговарајућим повезивањем улазних прикључака, чиме је обезбеђена исправност рада операција померања улево и удесно (слика 3.17).

A diagram of a circuit

Description automatically generated

Слика 3.17. – Имплементација четворобитног поморачког регистра

Графички опис четворобитног померачког регистра је сачуван као симбол компоненте, која представља основни логички елемент у дизајну осмобитног померачког регистра, који је приказан на слици 3.18.

A diagram of a circuit

Description automatically generated

Слика 3.18 – Имплементација осмобитног поморачког регистра

### Имплементација потпуног сабирача

Једнобитни потпуни сабирач је основни елемент вишебитних потпуних сабирача. Једнобитни сабирач, чији је графички опис приказан на слици 3.19 састоји се од основних логичких кола (XOR, AND, OR), и следећих улазних и излазних сигнала:

* А – Улазни бит првог броја
* B – Улазни бит другог броја
* Cin – Улазни бит преноса из претходног разреда
* S – Пренос у следећи разред
* Cout – Резултат сабирања

A diagram of a circuit

Description automatically generated

Слика 3.19 – Имплементација једнобитног потпуног сабирача

Графички опис једнобитног сабирача је сачуван као симбол компоненте, која представља основни логички елемента у дизајну четворобитног бинарног сабирача, чији је графички опис приказан на слици 3.20.

A diagram of a computer

Description automatically generated

Слика 3. 20 – Имплементација четворобитног потпуног сабирача

Четворобитни потпуни сабирач је имплементиран повезивањем четири једнобитна потпуна сабирача преко бита преноса Cout. Графички опис четворобитног сабирача је сачуван као симбол компоненте, која представља основни логички елемент у дизајну осмобитног бинарног сабирача, чији је графички опис приказн на слици 3.21.

A computer screen shot of a computer

Description automatically generated

Слика 3.21 – Имплементација осмобитног потпуног сабирача

### Имплементација линијског разделника

Линијски разделник је компонента која се користи за раздвајање вишебитних података на појединачне бите или групе бита како би се подаци могли обрадити, анализирати или упутити другим деловима система. Линијски разделник који је имплементиран у овом раду је разделник који раздваја линију ширине четири бита на појединачне битове. Графички опис линијског разделника приказан је на слици 3.22.

Slika na kojoj se nalazi tekst, snimak ekrana, linija, dijagram

Opis je automatski generisan

Слика 3.22 – Имплементација линијског разделника

## имплементација множача са померачким регистрима и једним сабирачем

За имплементацију наведеног решења, које је приказано на слици 3.23 потребне су следеће компоненте:

* Један осмобитни померачки регистар
* Један четворобитни померачки регистар
* Један осмобитни прихватни регистар
* Један осмобитни сабирач
* Један D флип-флоп
* Основна логичких кола
* Детектор успонске ивице

A diagram of a circuit board

Description automatically generated

Слика 3.23 – Имплементација множача са померачким регистрима и једним сабирачем

У графичком опису множача (слика 3.23), D флип-флоп је искоришћен да раздвоји стање учитавања података од стања множења самих података.

## Иплементација множача са прихватним регистрима и три сабирача

Ради оптимизације множача, уклоњен је један сабирач у односу на модел дат у анализи рада множача приказан на слици 2.3. У имплементацији је уклоњен први сабирач који врши сабирање вредности из регистра B (у зависности од бита најмање тежине регистра A) и нуле. Сабирање било ког броја са нулом даје управо тај број па је из тог разлог сабирач могао бити уклоњен.

За имплементацију наведеног решења потребне су следеће компоненте:

* Један осмобитни прихватни регистар
* Три четворобитна сабирача
* Основна логичких кола
* Детектор успонске ивице
* Три линијска разделника

Ово решење, чији је графички опис приказан на слици 3.24 се заснива на сличном принципу рада као и претходно. Главна разлика је у томе што је у овом решењу коришћено више сабирача те се сама операција множења извршава у једном такту. Такође нису потребни померачки регистри.

Slika na kojoj se nalazi tekst, dijagram

Opis je automatski generisan

Слика 3.24 – Имплементација множача са прихватним регистрима и више сабирача

## упоредна Анализа заузећа ресурса и брзине рада имплементираних множача

Након уноса дизајна изабраних типова множача применом графичког едитора, извршено је превођење дизајна преводиоцем софтвера Quartus II. Након превођења дизајна генерисани су извештаји из којих може да се види заузеће ресурса у изабраном програмабилном логичком колу.

### Имплементација у програмабилном логичком колу CPLD типа

За имплементацију описаних множача у програмабилном логичком колу, приликом креирања пројекта изабран је CPLD чип EPM7128SCL84-6 из фамилије MAX7000S.

На основу слике 3.25. на којој је приказан резултат превођења дизајна множача са слике 3.23, види се да ово решење заузима 44 логичке ћелије од 128, што износи 34%. Такође се користе 23 регистарске компоненте од могућих 128 или 18%.

A screenshot of a computer

Description automatically generated

Слика 3.25 – Заузеће ресурса множача са померачким регистрима и једним сабирачем у CPLD чипу EPM7128SCL84-6

На основу слике 3.26. на којој је приказан резултат превођења дизајна множача са слике 3.24, види се да ово решење заузима 27 логичке ћелије од 128, што износи 21%. Такође се користи 10 регистарских компонената од могућих 128 или 8%.

Slika na kojoj se nalazi tekst, snimak ekrana, broj, Font

Opis je automatski generisan

Слика 3.26 – Заузеће ресурса множача са прихватним регистрима и више сабирача у CPLD чипу EPM7128SCL84-6

Из наведене анализе заузећа ресурса, очигледно је да решење са слике 3.23 заузима 1,6 пута мање логичких ресурса и 2,3 пута мање регистарских ресурса у односу на решење са слике 3.24.

На основу извештаја са временском анализом, приказаног на слици 3.27, види се да максимална фреквенца сигнала такта решења на слици 3.23 износи 62,11MHz.

Slika na kojoj se nalazi tekst, snimak ekrana, Font, broj

Opis je automatski generisan

Слика 3.27 – Временски извештај множача са померачким регистрима и једним сабирачем у CPLD чипу EPM7128SCL84-6

На основу извештаја са временском анализом приказаном на слици 3.28, види се да максимална фреквенција сигнала такта решења на слици 3.24 износи 144,93MHz.

Slika na kojoj se nalazi tekst, snimak ekrana, Font, broj

Opis je automatski generisan

Слика 3.28 – Временски извештај множача са са прихватним регистрима и више сабирача у CPLD чипу EPM7128SCL84-6

Из наведених временских извештаја, очигледно је да решење са прихватним регистрима и сабирачима има 2,3 пута већу брзину рада у односу на решење приказано на слици 3.23, што је очекивани резултат, обзиром на принцип рада и могућност истовременог добијања свих бита резултата множења.

### Имплементација у програмабилном логичком колу FPGA типа

Заузеће ресурса и брзина рада описаних решења множача анализирани су и у случају имплементације дизајна у FPGA програмабилном логичком колу. У овом случају приликом креирања пројекта изабран је FPGA чип EPF10K70RC240-4 из фамилије FLEX10K.

На основу слике 3.29.на којој је приказан резултат превођења дизајна множача са слике 3.23, види се да ово решење заузима 53 логичка елемента од расположивих 3744 у чипу, што износи 1%. Такође се користе 23 регистарске компоненте од расположивих 3744 или <1%.

Slika na kojoj se nalazi tekst, snimak ekrana, broj, Font

Opis je automatski generisan

Слика 3.29 – Заузеће ресурса множача са померачким регистрима и једним сабирачем у FPGA чипу EPF10K70RC240-4

На основу слике 3.30. на којој је приказан резултат превођења дизајна множача на FPGA чипу са слике 3.24, види се да ово решење заузима 38 логичке ћелије од расположивих 3744 у чипу, што износи 1% и представља 1,4 пута мање заузеће ресурса од решења на слици 3.23. Такође се користи 10 регистарских компоненат од расположивих 3744 или <1%, што је 2,3 пута мање од решења на слици 3.23.

Slika na kojoj se nalazi tekst, snimak ekrana, broj, Font

Opis je automatski generisan

Слика 3.30 – Заузеће ресурса множача са прихватним регистрима и више сабирача у FPGA чипу EPF10K70RC240-4

Из наведене анализе заузећа ресурса следи да решење са слике 3.24 има значајно мање заузеће ресурса, без обзира на програмабилно логичко коло у коме се врши имплементација.

На основу извештаја са временском анализом приказаном на слици 3.31, види се да максимална фреквенција сигнала такта на FPGA чипу са слике 3.23 износи 31,06MHz. Ова учестаност је дупло нижа од учестаности, која је добијена у временском извештају за имплементацију дизајна у CPLD чипу.

Slika na kojoj se nalazi tekst, snimak ekrana, Font, broj

Opis je automatski generisan

Слика 3.31 – Временски извештај множача са померачким регистрима и једним сабирачем у FPGA чипу EPF10K70RC240-4

На основу извештаја са временском анализом приказаном на слици 3.32, види се да максимална фреквенција сигнала такта на FPGA чипу са слике 3.24 износи 86,96MHz. Ова учестаност је 2,8 пута виша него у случају дизајна са слике 3.23, али је такође скоро дупло нижа од учестаности, која је добијена у временском извештају за имплементацију истог дизајна у CPLD чипу.

Slika na kojoj se nalazi tekst, snimak ekrana, Font, broj

Opis je automatski generisan

Слика 3.32 – Временски извештај са прихватним регистрима и више сабирача у FPGA чипу EPF10K70RC240-4

Иако изабрани FPGA чип има много више хардверских ресурса од CPLD чипа, архитектуре проводних веза у CPLD чипу, заснована на програмабилној матрици преко које се повезују логички блокови, у конкретном случају омогућава мање кашњење сигнала, па самим тим и већу брзину рада, у односу на имплементацију у FPGA чипу. С друге стране, заузеће ресурса у CPLD чипу оставља мало простора за имплементацију додатних хардвера у истом чипу. Због тога избор програмабилног логичког кола треба да одговара захтевима у погледу брзине и заузећа ресурса дизајна дигиталног хардвера, у оквиру кога би се користио множач.

# **Тестирање дизајна симулацијом**

Пре тестирања имплементираних решења множача симулацијом, неопходно је прво извршити тестирање компоненти неопходних за њихову реализацију, а то укључује тестирање:

* Прихватних регистара
* Померачких регистара
* Потпуних сабирача

## Тестирање прихватних регистара са паралелним уписом и читањем симулацијом

Да би се проверили исправност рада имплементираних прихватних регистара прво се тестира једнобитни прихватни регистар, који је уједно и најнижи део хијерархије. Симулациони дијаграм једнобитног прихватног регистра са паралелним уписом и читањем је приказан на слици 4.1.

Slika na kojoj se nalazi linija, tekst, broj, Font

Opis je automatski generisan

Слика 4.1 – Симулациони дијаграм једнобитног прихватног регистра са паралелним уписом и читањем

На основу симулационог дијаграма, приказаног на слици 4.1, може се приметити да се вредност улазног сигнала (li), који представља улазни сигнал података, уписује у регистар (сигнал: OUT), када је активан контролни сигнал уписа (LD). Такође када је активан контролни сигнал брисања садржаја (CL), успешно се поставља вредност излазног сигнала на логичку нулу.

Потребно је напоменути да оба контролна сигнала, било учитавања или брисања садржаја, обављају своју функцију на узлазној ивици сигнала такта (CLK), као и да операција учитавања садржаја има приоритет у односу на брисање садржаја регистра.

Након извршеног тестирања једнобитног прихватног регистра (који је саставни део двобитног прихватног регистра) и уверавања у исправност његовог рада могуће је тестирати двобитни прихватни регистар. Резултат тестирања је приказан на слици 4.2. Као и код једнобитног прихватног регистра, контролни сигнали уписа и брисања имају исту функцију. Једина разлика је у ширини улазних и излазних сигнала података, који сада имају два бита.

Slika na kojoj se nalazi tekst, snimak ekrana, linija, Paralela

Opis je automatski generisan

Слика 4.2 – Симулациони дијаграм двобитног прихватног регистра са паралелним уписом и читањем

Након извршеног тестирања двобитног прихватног регистра (који је саставни део четворобитног прихватног регистра ) и потврде исправности његовог рада, могуће је тестирати четворобитни прихватни регистар. Симулациони дијаграм четворобитног прихватног регистра је приказан на слици 4.3. Као и код претходних реализација прихватних регистара, контролни сигнали уписа и брисања имају исту функцију. Једина разлика је у ширини улазних и излазних сигнала података четири бита

Slika na kojoj se nalazi tekst, linija, Font, broj

Opis je automatski generisan

Слика 4.3 – Симулациони дијаграм четворобитног прихватног регистра са паралелним уписом и читањем

Након успешног тестирања четворобитног прихватног регистра (који је саставни део осмобитног прихватног регистра), извршена је симулација и потврда исправности рада осмобитног прихватног регистра, као што се види из резултата тестирања приказаних на слици 4.4.

Slika na kojoj se nalazi snimak ekrana, linija, tekst, broj

Opis je automatski generisan

Слика 4.4 – Симулацјони дијаграм осмобитног прихватног регистра са паралелним уписом и читањем

## Тестирање померачких регистара симулацијом

Да би се проверили исправност рада имплементираних померачких регистара, прво се тестира једнобитни померачки регистар (као и код прихватних регистара). Симулациои дијаграм једнобитног померачког регистра је приказан на слици 4.5.

Slika na kojoj se nalazi tekst, linija, broj, Font

Opis je automatski generisan

Слика 4.5 – Симулациони дијаграм једнобитног померачког регистра

На основу симулационог дијаграма, приказаног на слици 4.5, може се приметити да када је активан контролни сигнал уписа (LD), вредност сигнала (li), који представља улазни сигнал података, уписује у регистар (сигнал: OUT). Контролни сигнали логичког померања улево (SL) и логичког померања удесно (SR), када су активни, омогућавају упису у регистар вредности улазних сигнала IL и IR респективно (у зависности од тога који контролни сигнал је активан). Пошто је у питању једнобитни померачки регистар, не може у потпуности да се сагледа ефекат контролних сигнала на померање садржај, већ се њима постиже брисање постојеће једнобитне вредности у регистру и замена новом.

Као и код прихватних регистара, контролни сигнали обављају своју функцију у складу на узлазној ивици сигнала такта (CLK), при чему при пројектовању регистара, као и треба водити рачуна да не буде активно више контролних сигнала у исто време. Уколико су два или више контролних сигнала активни у исто време, не може са сигурношћу да се зна који ће податак бити уписан у регистар је непредвидив (не саветује се активација више од једног контролног сигнала у истом такту).

Након успешног тестирања једнобитног померачког регистра (који је саставни део двобитног померачког регистра) и потврде исправност његовог рада, тестиран је двобитни померачки регистар. Резултат тестирања је приказан са слици 4.6

Slika na kojoj se nalazi tekst, snimak ekrana, broj, linija

Opis je automatski generisan

Слика 4.6 – Симулациони дијаграм двобитног померачког регистра

Сигнал за дозволу уписа података (LD) има исту функционалност као и код једнобитног померачког регистра, са тим што се сада у регистар уписују два бита података. На симулационом дијаграму, приказаном на слици 4.6, може се приметити да операције логичког померања исправно раде. Такође улазни сигнали IL и IR респективно одговарају биту најмање тежине приликом операције логичког померања улево, односно биту највеће тежине приликом логичке операције померања у десно.

Након успешног тестирања двобитног померачког регистра и потврде исправност његовог рада, тестиран је четворобитни померачки регистар. Резултат тестирања је приказан на слици 4.7.

Slika na kojoj se nalazi tekst, snimak ekrana, broj, linija

Opis je automatski generisan

Слика 4.7 – Симулациони дијаграм четворобитног померачког регистра

На основу приложеног симулационог дијаграма на слици 4.7, може се закључити да се операције уписа података и логичког померања исправно извршавају. Једина разлика у односу на двобитни померачки регистар је што су подаци ширине четири бита.

Након успешног тестирања четворобитног померачког регистра и потврде исправности рада, тестиран је осмобитног померачки регистар. Симулациони дијаграм осмобитног померачког регистра приказан је на слици 4.8.

Slika na kojoj se nalazi tekst, snimak ekrana, broj, Paralela

Opis je automatski generisan

Слика 4.8 – Симулациони дијаграм осмобитног померачког регистра

Као и у случају претходних регистара, са мањим бројем бита, из симулационог дијаграма се може закључити да се све операције у осмобитном померачком регистру извршавају коректно.

Такође, са симулационог дијаграма, приказаног на слици 4.8, се може приметити да улазни сигнал IL никако не утиче на исход операције логичког померања удесно, као и да улазни сигнал IR не утиче на извршавање операције логичког померања улево. Ово такође важи за двобитне и четворобитне померачке регистре.

## Тестирање потпуних сабирача симулацијом

Да би се проверили исправност рада потпуних сабирача, најпре је неопходно тестирати једнобитни потпуни сабирач, који је уједно и основна компонента потребна за израду вишебитних сабирача. Симулациони дијаграм једнобитног потпуног сабирача приказан је на слици 4.9. Улазни сигнали A и B представљају први и други операнд операције сабирања. Улазни сигнал Cin представља пренос из претходног разреда сабирача. Излазни сигнал Cout представља пренос у наредни разред сабирача. Излазни сигнал S представља резултат операције сабирања.

Slika na kojoj se nalazi linija, dijagram, broj, Plot

Opis je automatski generisan

Слика 4.9 – Симулациони дијаграм једнобитног потпуног сабирача

На основу симулационог дијаграма приказаног на слици 4.8 се може закључити да су резултати исправни и да се поклапају са комбинационом табелом потпуног сабирача са слике 2.11 из другог поглавља овог рада.

Након успешног тестирања једнобитног потпуног сабирача и уверавања у потврде исправност његовог рада, тестиран је четворобитни потпуни сабирач. Резултат тестирања четворобитног потпуног сабирача је приказан на слици 4.10.

Slika na kojoj se nalazi tekst, snimak ekrana, linija, dijagram

Opis je automatski generisan

Слика 4.10 – Симулациони дијаграм четворобитног потпуног сабирача

На основу симулационог дијаграма приказаног на слици 4.10, може се закључити да четворобитни потпуни сабирач исправно ради. Кључна разлика између једнобитног сабирача и четворобитног сабирача је у броју бита резултата и операнада.

На крају, након успешног тестирања четворобитног потпуног сабирача, тестиран је и осмобитни потпуни сабирач. Симулациони дијаграм осмобитног потпуног сабирача приказан, који је приказан на слици 4.11, потврђује исправност рада осмобитног сабирача за све задате вредности улазних података.

Slika na kojoj se nalazi linija, tekst, Paralela, broj

Opis je automatski generisan

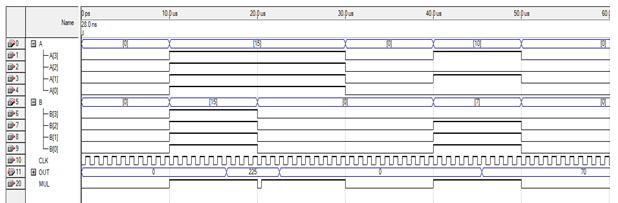
Слика 4.11 – Симулациони дијаграм осмобитног потпуног сабирача

## Тестирање множача симулацијом

Након успешног тестирања свих неопходних комопонената за имплементацију оба типа множача и потврде исправности рада, извршено је њихово повезивање у графичком едитору, као што је приказано на сликама 3.23 и 3.24, након чега је извршено тестирање множача симулацијом.

### Тестирање множача са померачким регистрима и једним сабирачем

Након успешно превођења дизајна са померачким регистрима и једним сабирачем, направљен је симулациони фајл за тестирање рада множача (слика 4.12).



Слика 4.12 - Симулација рада множача са померачким регистрима и једним сабирачем

Симулацијом је потврђено да се операција множења исправно обавља, као и да трајање саме операције зависи од ширине другог операнда.

### Тестирање множача са прихватним регистрима и више сабирача

Након успешно превођења дизајна са прихватним регистрима и више сабирача, направљен је симулациони фајл за тестирање рада множача (слика 4.13).

Slika na kojoj se nalazi linija, broj, Paralela, tekst

Opis je automatski generisan

Слика 4.13 Симулација решења са прихватним регистрима и више сабирача

Симулацијом је потврђено да се операција множења исправно обавља, као и да трајање саме операције зависи од ширине другог операнда.

# тестирање множача у реалном времену

За успешно тестирање имплементираних решења множача у реалном времену, на развојном систему, прво је потребно направити тест шему и доделити одговарајуће пинове програмабилног логичког кола улазним и излазним прикључцима дизајна. Тест шема, која је приказана на слици 4.14 је идентична за оба типа множача, с тим што је додела прикључака извршена под претпоставком да се дизајн имплементира у FPGA чипу EPF10K70RC240-4.

На тест шеми приказаној на слици 4.14 цео дизајн множача је представљен као једна компонента, на чије улазне прикључке се доводе вредности сигнала, док се резултат симулације прати на излазним прикључцима.

Slika na kojoj se nalazi tekst, linija, snimak ekrana, broj

Opis je automatski generisan

Слика 4.14 - Тест шема множача

Тестирање дизајна на развојном систему спроведено је на FPGA чипу EPF10K70RC240-4 из серије FLEX10K.

Да би се дизајн тестирао на развојном систему, тест шему је потребно допунити компонентама које су такође имплементиране у програмабилном логичком колу и у које је уписан дизајн множача. Те компоненте омоћавају и интерфејс улазних и излазних прикључака множача са расположивим тастерима и светлосним индикаторима на развојном систему. Генератор сигнала такта је такође имплементиран у програмабилном логичком колу. Допуњена тест шема, која је идентична за оба типа множача, приказана је на слици 4.15.

На тест шеми приказаној на слици 4.15 приказан је начин на који су додељени прикључци улазним и излазним сигналима, тако да одговарају прикључцима на развојном систему.

Slika na kojoj se nalazi tekst, linija, snimak ekrana, dijagram

Opis je automatski generisan

Слика 4.15 – Допуњена тест шема множача са померачким регистрима и једним сабирачем за покретање на развојном окружењу

Претходни дизајн је најпре преведен и уз помоћ алата „Programmer”, из софтверског развојног окружења Quartus II, пренет је на развојно окружење. Пренос на развојно окружење започиње се притиском на дугме старт, у горњем десном углу налази се „Progress bar“ помоћу ког се може пратити процес преноса дизајна.

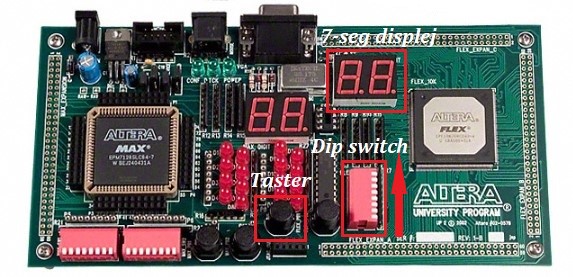
Slika na kojoj se nalazi tekst, snimak ekrana, softver

Opis je automatski generisan

Слика 4.16 – Пренос преведеног дизајна на развојно окружење

Након преноса дизајна на развојно окружење, потребно је тестирати имплементирани дизајн на развојном окружењу.

На слици 4.17 елементи уоквирени црвеном бојом представљају кључне компоненте за тестирање рада имплементираног дизајна. Тастер и „Dip switch“ прекидачи представљају улазне сигнале у односу на тест шему. Док седмо-сегментни дисплеј представља излазни сигнал у односу на тест шему.



Слика 4.17 – UP2 развојно окружење

Вредности улазних операнада се постављају помоћу „Dip switch“ прекидача. Гледано у правцу стрелице прва четири „Dip switch“ прекидача представљају вредности првог операнда, при чему први „Dip switch“ прекидач одговара највишем биту првог операнда. Осталих четири прекидача одговарају другом операнду, при чему је највиши бит додељен као у претходном случају. Подигнут Dip switch“ прекидач означава логичку јединицу, док спуштен прекидач означава логичку нулу.

Када се доделе вредности операндима, да би се започео процес множења потребно је притиснути тастер (контролни сигнал операције множења). Након извршења операције множења резултат се приказује на седмо-сегментом дисплеју.

У наставку су приказане слике тестирања дизајна различитим улазним вредностима (слика 4.18, слика 4.19, слика 4.20 и 4.21).

На слици 4.18 види се резултат множења када су на развојном систему задате вредности 1110 (10112) и 910 (1001), при чему се као резултат добија 6316 (9910).

Slika na kojoj se nalazi elektronika, Elektrotehnika, kolo, Komponenta kola

Opis je automatski generisan

Слика 4.18 – Тестирање на развојном окружењу са улазним вредностима 11 и 9

На слици 4.19 види се резултат множења када су на развојном систему задате вредности 710 (01112) и 610 (0110), при чему се као резултат добија 2А16 (4210).

Slika na kojoj se nalazi elektronika, tekst, kolo, Elektrotehnika

Opis je automatski generisan

Слика 4.19 – Тестирање на развојном окружењу са улазним вредностима 7 и 6

На слици 4.20 види се резултат множења када су оба податка једнака нули (сви „Dip switch“ прекидачи су спуштени), те је резултат множења једнак нули.

Slika na kojoj se nalazi tekst, elektronika, Elektrotehnika, Komponenta kola

Opis je automatski generisan

Слика 4.20 – Тестирање на развојном окружењу са улазним вредностима 0 и 0

На слици 4.21 види се резултат множења када су на развојном систему задате вредности 1510 (11112) и 1510 (1111) улазним вредностима (сви „Dip switch“ прекидачи су подигнути), те се као резултат множења добија Е116 (22510).

Slika na kojoj se nalazi elektronika, Elektrotehnika, kolo, Komponenta kola

Opis je automatski generisan

Слика 4.21 – Тестирање на развојном окружењу са улазним вредностима 15 и 15

Приказани резултати тестирања су идентични за обе реализације множача описане у овом раду.

# ЗАКЉУЧАК

У овом завршном раду су анализирана и имплементирана два могућа решења бинарних множача применом регистара и сабирача у софтверском развојном окружењу Quartus II.

Наведена решења су имплементирана као независне целине и могу бити интегрисана у различите врсте аритметичко-логичких јединица у којима се извршавају операције множења произвољних бинарних вредности.

На основу извештаја о заузећу ресурса у изабраним програмабилним логичким колима, који су генерисани након превођења дизајна у софтверском развојном окружењу Quartus II, може да се закључи да би решење са померачким регистрима и једним сабирачем захтевало више хардверских ресурса при синтези и у CPLD и у FPGA колу, иако је једноставније са аспекта архитектуре, односно броја логичких мрежа које се користе за имплементацију функције множача, у односу на решење са прихватним регистрима и више сабирача. Значајан утицај на веће заузећу ресурса првог решења има сложенија структура померачких регистара у односу на прихватне регистре који се користе у другом решењу.

На основу извештаја о брзини рада, може се закључити да би решење са прихватним регистрима и више сабирача могло да ради на вишој учестаности и у CPLD и у FPGA колу.

Тестирањем симулацијом и на развојном систему потврђена је исправност рада за различите улазне податке оба описана решења, што значи да би она могла да имају практичну примену.

Захваљујући модуларном и хијерархијском принципу пројектовања, омогућена је флексибилност дизајна, јер обрађена решења множача са четворобитним чиниоцима могу лако да се прошире на множаче са чиниоцима који имају већи број бита.

# ИНДЕКС ПОЈМОВА

**А**

Анализа

**Б**

Бинарни запис

Бинарни множач

Булова алгебра

**Д**

Двобитни

Децимални запис

**И**

Излазни сигнал

Имплементација

**Ј**

Једнобитни

**К**

Комбинациона кола

Компонента

Контролни сигнал

**Л**

Логичка кола

**М**

Милијева секвенцијална мрежа

Множач

Модул

Мурова секвенцијална мрежа

**О**

Осмобитни

**П**

Померачки регистар

Програмабилна логика

**Р**

Регистар

**С**

Сабирач

Секвенцијална кола

Симулација

**Т**

Табела истинитости

Такт

Тренутно стање

**У**

Улазни сигнал

**Ф**

Флип-флоп

**П**

Померачки регистар

**Х**

Хардверска ефикасност

Хардверска сложеност

**Ч**

Четворобитни

**C**

CPLD коло

**F**

FPGA коло

# ЛИТЕРАТУРА

[1] Z. Navabi, *Digital Design and Implementation with Field Programmable Devices*, Kluwer Academic Publishers, Norwell, USA, 2005.

[2] <https://www.southampton.ac.uk/~bim/notes/ice/spec.html>, август 2023

[3] <https://www.etf.bg.ac.rs/uploads/files/udzbenici/ElementiElektronike_DigitalnaKola.pdf>, август 2023

# ИЗЈАВА О АКАДЕМСКОЈ ЧЕСТИТОСТИ

**ИЗЈАВА О АКАДЕМСКОЈ ЧЕСТИТОСТИ**

|  |  |
| --- | --- |
| **Студент (име, име једног родитеља и презиме):** |  |
| **Број индекса:** |  |

Под пуном моралном, материјалном, дисциплинском и кривичном одговорношћу изјављујем да је завршни рад, под насловом:

1. резултат сопственог истраживачког рада;
2. да овaj рад, ни у целини, нити у деловима, нисам пријављиво/ла на другим високошколским установама;
3. да нисам повредио/ла ауторска права, нити злоупотребио/ла интелектуалну својину других лица;
4. да сам рад и мишљења других аутора које сам користио/ла у овом раду назначио/ла или цитирао/ла у складу са Упутством;
5. да су сви радови и мишљења других аутора наведени у списку литературе/референци који је саставни део овог рада, пописани у складу са Упутством;
6. да сам свестан/свесна да је плагијат коришћење туђих радова у било ком облику (као цитата, прафраза, слика, табела, дијаграма, дизајна, планова, фотографија, филма, музике, формула, вебсајтова, компјутерских програма и сл.) без навођења аутора или представљање туђих ауторских дела као мојих, кажњиво по закону (Закон о ауторском и сродним правима), као и других закона и одговарајућих аката Високе школе електротехнике и рачунарства струковних студија у Београду;
7. да је електронска верзија овог рада идентична штампаном примерку овог рада и да пристајем на његово објављивање под условима прописаним актима Високе школе електротехнике и рачунарства струковних студија у Београду;
8. да сам свестан/свесна последица уколико се докаже да је овај рад плагијат.

У Београду, \_\_. \_\_. 201\_. године

Својеручни потпис студента

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_