Факултет инжењерских наука

Универзитета у Крагујевцу

Ђорђе С. Гачић

Пројектовање синтетизабилне сведигиталне фреквенцијски затворене петље са широким опсегом подешавања до учестаности од 640 MHz

Мастер рад

Крагујевац, 2024.

Факултет инжењерских наука Универзитета у Крагујевцу

Назив студијског програма: Електротехника и рачунарство

Ниво студија: Мастер академске студије

Предмет: Анализа и пројектовање алгоритама

Број индекса: 408/2022

Ђорђе С. Гачић

Пројектовање синтетизабилне сведигиталне фреквенцијски затворене петље са широким опсегом подешавања до учестаности од 640 MHz

Мастер рад

Комисија за преглед и одбрану:

1. **проф. др Владимир М. Миловановић - ментор**
2. **др Шћепан Шћекић**
3. **доц. др Жарко Попара**

Датум одбране: _____

Оцена: _____

У оквиру овог мастер рада кандидат треба да реализује дигиталну фреквенцијски затворену петљу у стандардној 130 nm CMOS технологији, за циљану учестаност рада до 640 MHz. Систем треба да се састоји од дигитално контролисаног осцилатора у облику матрице тростатичких инвертора коме претходи управљачка логика која омогућава брзо достизање и стабилно одржавање жељене радне учестаности која је задата целобројним умношком у односу на референтну фреквенцију од 16 MHz. Приказати резултате симулација у најгорим, типичним и најбољим условима рада узимајући у обзир напон напајања, температуру и процесни угао. Потребно је теоријски обрадити структуру система као и појаве које се запажају у резултатима симулација. У сврху додатне провере исправности рада система, потребно је направити модел система у програмском језику Пајтон који треба веродостојно симулирати рад свих компоненти система и упоредити пар различитих контролера укључујући и стандардни ПИД (пропорционално, интегрални, диференцијални) контролер.

Препоручена литература:

[1] Robert Bogdan Staszewski и Poras T. Balsara, *All-Digital Frequency Synthesizer in Deep-Submicron CMOS*, John Wiley & Sons, Inc., Hoboken, New Jersey, 2005.

[2] Behzad Razavi, *Design of CMOS Phase-Locked Loops: From Circuit Level to Architecture Level*, Cambridge University Press, 2020.

Крагујевац, 10. 9. 2024.

Ментор:

Др Владимир М. Миловановић,
ванредни професор

ПРИЛОГ 6: Пријава мастер рада

	Универзитет у Крагујевцу **Факултет инжењерских наука Универзитета у Крагујевцу**	

Мастер академске студије: **Електротехника и рачунарство**

Назив предмета: **Анализа и пројектовање алгоритама**

Име и презиме: **Ђорђе Гачић**

Број индекса: **408/2022**

ПРИЈАВА МАСТЕР РАДА

Тема рада: Пројектовање синтетизабилне сведигиталне фреквенцијски затворене петље са широким опсегом подешавања до учестаности од 640MHz

Задатак: Потребно је реализовати дигиталну фреквенцијски затворену петљу у стандардној 130nm CMOS технологији, за циљану учестаност рада до 640MHz. Систем треба да се састоји од дигитално контролисаног осцилатора у облику матрице тростатичких инвертора коме претходи управљачка логика која омогућава брзо достизање и стабилно одржавање жељене радне учестаности која је задата целобројним умношком у односу на референтну фреквенцију од 16MHz. Приказати резултате симулација у најгорим, типичним и најбољим условима рада узимајући у обзир напон напајања, температуру и процесни угао. Потребно је теоријски обрадити структуру система као и појаве које се запажају у резултатима симулација. У сврху додатне провере исправности рада система, потребно је направити модел система у програмском језику Пајтон који треба веродостојно симулирати рад свих компоненти система и упоређује пар различитих контролера укључујући и стандардни ПИД (пропорционално, интегрални, диференцијални) контролер.

Крагујевац, 20. јун 2024. године

Ментор:

Др Владимир Миловановић, ванр. проф.

Садржај

1 Увод **3**

2 Опис и структура фреквенцијски затворене петље **4**

3 Управљачка логика дигиталне фреквенцијски затворене петље **6**
3.1 Управљачка предобрада . . . 6
3.2 Двостепени контролер . . . 7
3.3 PID контролер . . . 8
3.4 Управљачки декодер . . . 10

4 Теоријско разматрање осцилатора **11**
4.1 Основни принципи осциловања . . . 11
4.2 Систем осцилаторне повратне спреге . . . 12
4.3 Основе прстенастог осцилатора . . . 14
4.3.1 Инверторски прстен . . . 19
4.4 Шум у осцилаторима . . . 21

5 Дигитално контролисани осцилатор **23**
5.1 Архитектура предложеног DCO-а . . . 24
5.2 Блок дијаграм ћелије DCO-а . . . 25
5.3 Претварачи напонског нивоа DCO-а . . . 27

6 Имплементација и резултати симулација **29**
6.1 SystemVerilog код . . . 29
6.2 Лејаут . . . 37
6.3 Симулација рада фреквенцијски затворене петље . . . 39
6.4 PVT зависност дигитално контролисаног осцилатора . . . 39
6.5 Анализа потрошње система . . . 42
6.6 Временски одзив дигитално контролисаног осцилатора . . . 43
6.7 Спектар снаге фреквенцијски затворене петље . . . 44
6.8 Фазни шум дигитално контролисаног осцилатора . . . 45

7 Пајтон модел FLL-а **46**
7.1 Пајтон код . . . 46
7.2 Поређење рада различитих управљачких режима . . . 51
7.3 Утицај константи PID контролера на одзив система . . . 51
7.4 Утицај резолуције DCO-а на одзив система . . . 53

8 Поређење DCO-а пројектованих у 130 nm и 180 nm технологији **54**
8.1 Начин и утицај коришћења дубоке јаме N типа у аналогном пројектовању DCO-а . . . 56

9 Закључак **58**

Литература **59**

Објављени радови **60**

Abstract

Frequency-locked loop (FLL) represents a viable way of generating a range of frequencies from a single reference frequency by using a negative feedback electronic control system that compares the frequency of a controlled oscillator to the reference one. A digital synthesizable FLL is designed in 130 nm CMOS technology for a target frequency of up to 640 MHz. It employs a wide-tuning range digitally controlled oscillator (DCO) assembled from tri-state inverters in the form of a matrix. The FLL can optionally use a bang-bang or a soft-programmable standard proportional-integral-derivative (PID) controller to regulate the feedback loop. Its design practically minimizes metastability occurrence. The proposed digital FLL occupies 100 µm × 330 µm and consumes 3.5 mW in typical operating conditions. The reference clock is 16 MHz, and the output oscillation frequency is set to 640 MHz, while the achieved frequency resolution is 2.8 MHz. The system is additionally modeled in software using the Python programming language. A comparison of the characteristics of DCOs designed in 130 nm and 180 nm TSMC technologies is also provided.

Keywords: Frequency-locked loop, digitally controlled oscillator, clock generator, synthesizable, CMOS technology, PID controller, metastability, Python model.

Резиме

Фреквенцијски затворена петља (енгл. *Frequency-Locked Loop*, FLL) представља одржив начин генерисања опсега фреквенција из једне референтне фреквенције коришћењем електронског система управљања са негативном повратном спрегом, који пореди фреквенцију контролисаног осцилатора са поменутом референтном фреквенцијом. Дигитални синтетизабилни FLL је дизајниран у 130 nm технологији за циљану фреквенцију до 640 MHz. Он садржи дигитално контролисани осцилатор (енгл. *Digitally Controlled Oscillator*, DCO) са широким подешавањем опсега који се састоји од тростатичких инвертора у облику матрице. FLL може произвољно користити двостепени (енгл. *Bang-bang*) контролер или софтверски програмабилни стандардни пропорционално, интегрални, диференцијални (енгл. *Proportional-Integral-Derivative*, PID) контролер за управљање повратном спрегом. Такав дизајн у пракси минимизује појаву метастабилности. Предложени дигитални FLL заузима површину од 100 µm × 330 µm и троши 3,5 mW у типичним условима рада. Референтни такт је 16 MHz, а излазна фреквенција осциловања је подешена на 640 MHz, док постигнута резолуција фреквенције износи 2,8 MHz. Систем је додатно софтверски моделиран у Пајтон програмском језику. Такође је дато поређење карактеристика DCO-а пројектованих у 130 nm и 180 nm TSMC технологијама.

Кључне ријечи: Фреквенцијски затворена петља, дигитално контролисани осцилатор, генератор такта, синтетизабилност, CMOS технологија, PID контролер, метастабилност, Пајтон модел.

1 Увод

У данашње вријеме, фазно затворена петља (енгл. *Phase-Locked Loop*, PLL) и петља са затвореним кашњењем (енгл. *Delay-Locked Loop*, DLL) представљају свеприсутне блокове у дизајну чипова. Безброј примјена самих чипова захтијевају или генератор такта или синтетизатор фреквенције, што подразумијева уградњу неког од поменутих блокова унутар система који се пројектује. Главна улога таквог блока у дизајну је да генерише стабилан и прецизан излазни сигнал чија је фаза подесива у односу на фазу улазног сигнала, самим тим одржавајући везу између улазне и излазне фреквенције. Међутим, чак и веома сложени системи често захтијевају генератор такта, који само множи улазну фреквенцију без да посебно води рачуна о фази такта или апсолутном џитеру (енгл. *Jitter*). У таквим примјенама, потребна и довољна је само фреквенцијски затворена петља (енгл. *Frequency-Locked Loop*, FLL) да би се испунили тражени захтјеви.

Постоје многи начини имплементације FLL-а [1]. Штавише, FLL као интегрисано коло може се сврстати у двије групе: дигитални и аналогни FLL. Иако је очигледан недостатак првог максимална фреквенција и њена резолуција, они посједују многе друге предности наспрам њихових аналогних парњака. Заузимају мање простора, истичу се већом отпорношћу на промјене процесних углова, напона и температуре (енгл. *Process-Voltage-Temperature*, PVT), лако су преносиви у различите технологије, и стога омогућавају поновну употребу, скалирање, једноставнију методологију, као и брже циклусе пројектовања. Узимајући у обзир све претходно поменуто, испоставља се да је у општем случају боље ићи ка развоју дигиталног FLL-а кад год спецификација архитектуре система то дозвољава.

Овај рад конкретно предлаже синтетизабилни дигитални FLL сличан предложеном у литератури [2], са побољшаном брзином закључавања FLL-а [3] и смањеним ризиком од метастабилности. Осцилатор је састављен од тростатичких инвертора и заснован на прстенастом DCO-у из литературе [4], измијењен додавањем независног напона напајања DCO-а са претварачима напонских нивоа (енгл. *Level Shifters*) и употребом петостепене [5] умјесто тростепене толологије прстена осцилатора.

У поглављу 2 се даје основни опис рада и структуре предложеног дигиталног FLL-а. Управљачка логика FLL-а је детаљно описана у поглављу 3. Теоријска разматрања осцилатора, као главне компоненте у систему дата су у поглављу 4, док је коришћени дигитално контролисан осцилатор описан у поглављу 5. Поглавље 6 садржи анализиран код на нивоу регистара (енгл. *Register-Transfer Level*, RTL) и лејаут FLL-а и његових појединачних компоненти, а дат је и увид у имплементацију и добијене резултате симулација, такође уз теоријска разматрања појава које су од значаја за рад читавог система. У поглављу 7 описан је значај софтверског моделирања система и приказана је имплементација модела понашања FLL-а у Пајтон програмском језику. У сврху поређења карактеристика осцилатора у различитим технологијама, рад садржи поглавље 8. Коначно, поглавље 9 закључује рад и наговјештава могућности даљег рада на побољшању и проширењу система.

2 Опис и структура фреквенцијски затворене петље

По дефиницији, FLL је управљачки систем са негативном повратном спрегом који закључава фреквенцију излазног сигнала на предвиђену циљану фреквенцију. У принципу, непрестано управља фреквенцијом осцилатора на аутоматски начин све док излазна фреквенција не достигне циљану вриједност, након чега се та вриједност фреквенције одржава на излазу. То значи да FLL спада у синтетизаторе фреквенције који се уопштено описују као активна електронска кола која имају неки сигнал са референтном и веома стабилном фреквенцијом (f_{REF}) на улазу и затим генеришу излазни сигнал са фреквенцијом (f_{OUT}) која представља референтну фреквенцију помножену са одговарајућим чиниоцем тј. умношком фреквенције (FMUL) као у једначини 1.

$$f_{\mathrm{OUT}} = \mathrm{FMUL} \cdot f_{\mathrm{REF}} \tag{1}$$

Треба напоменути да је стабилност, прецизност и спектрална чистоћа излазног сигнала у константној вези са квалитетом сигнала на улазу [6]. Слика 1 приказује поједностављен дигитални сигнал на улазу (REF) и излазу (OUT) система са одређеном вриједношћу умношка фреквенције.

Слика 1: Демонстрација рада синтетизатора фреквенције за FMUL=4.

$$f_{\mathrm{REF}} = \frac{1}{T_{\mathrm{REF}}} \tag{2}$$

$$f_{\mathrm{OUT}} = \frac{1}{T_{\mathrm{OUT}}} \tag{3}$$

Дефиниција изнад не предвиђа облик синтетисаног излазног сигнала. Он може бити синусоида или правоугаони (дигитални) сигнал, који нам је у CMOS технологији дефинитивно кориснији за разматрање.

У овом раду описана је релативно једноставна али ефикасна дигитална фреквенцијски затворена петља чију системску архитектуру на нивоу блокова приказује Слика 2. Описани FLL се практично састоји из два блока: дигитално контролисаног осцилатора и блока управљачке логике, који генерише улазне сигнале осцилатора на основу тренутне фреквенције DCO-а. У сврху поједностављења, са слике су изостављени неки конфигурациони улази FLL-а, као што су коефицијенти PID контролера или улаз за одабир режима рада, чија улога је детаљније описана у поглављу 6.

Слика 2: Блок дијаграм дигиталне фреквенцијски затворене петље састављене од: блока управљачке логике (лијево) и дигитално контролисаног осцилатора (десно).

Основни улази у систем су:

- RSTN - Ресет сигнал који је активан на нулу,
- FREF - Референтни такт,
- FMUL - Улазни умножак фреквенције (множењем са фреквенцијом референтног такта добија се вриједност фреквенције коју треба постићи и одржавати на излазу),
- CTRL - Сигнал који одређује који од два управљачка режима се користи.

Излази из система су:

- FOUT - Излазни такт,
- LOCK - Сигнал који говори да ли је систем ушао у стабилно стање тј. да ли ради на жељеној фреквенцији.

3 Управљачка логика дигиталне фреквенцијски затворене петље

Управљачка логика FLL-а састоји се од двије независне процесне гране, које представљају два међусобно искључива режима управљања FLL-а. Оба режима на улазу примају бинарну вриједност која представља број периода такта DCO-а унутар периода референтног такта. Такође, оба управљачка режима генеришу бинарну вриједност на излазу, која представља управљачку бинарну ријеч осцилатора директно пропорционалну излазној фреквенцији. Главне разлике између два поменута режима су брзина затварања (закључавања) FLL-а и једноставност подешавања. Циљ управљачке логике FLL-а је изједначити вриједност улазног множача фреквенције са бројем периода такта DCO-а унутар периода референтног такта што је брже и прецизније могуће, чиме се долази до постизања жељене фреквенције на излазу DCO-а. Поред самог постизања жељене фреквенције, задатак управљачке логике је и њено одржавање током времена јер, због утицаја разних фактора, као што је нпр. температура, константно долази до нежељеног смањења или повећања вриједности фреквенције. То значи да управљачка логика константно прати рад система и по потреби реагује на исход непожељних утицаја на излазну фреквенцију. Комплетна управљачка логика FLL-а је подијељена на неколико фаза, које су детаљно описане у наредним поглављима, а то су:

- Управљачка предобрада
- Двостепени контролер / PID контролер
- Управљачки декодер

3.1 Управљачка предобрада

Фаза управљачке предобраде (енгл. *Control Preprocessing*) укључује неколико блокова чија је улога претворити информацију о фреквенцији излазног такта DCO-а у бинарну вриједност која ће бити прослијеђена као улаз наредној фази управљачке логике FLL-а. Као прво, да би се одредила брзина осциловања DCO-а, потребан је бројач. У имплементацији описаној у овом раду коришћен је Грејев бројач. У Грејевом коду свака узастопна вриједност се разликује за само један бит. То му даје предност у односу на природни бинарни бројач из разлога што значајно умањује метастабилност бројача изазвану узорковањем (енгл. *Sampling*) вриједности бројача. Табела 1 приказује првих осам цифара у Грејевом коду, у поређењу са природним бинарним цифрама.

Као примјер појаве метастабилности може се навести проблем који се може јавити у употреби природних бинарних кодова, а то је да се, при преласку у наредно стање бројача, стање свих битова који се мијењају не мора промијенити тачно синхронизовано. Рецимо, при преласку из стања 3 у стање 4 (бинарно 011 у 100), код природног бинарног бројача сви битови мијењају стање, тако да та транзиција може да иде редом $011 \rightarrow 001 \rightarrow 101 \rightarrow 100$, што значи да могу да постоје два прелазна стања, која могу бити прочитана чиме би коначно стање бројача било погрешно. У овом конкретном случају, умјесто у стање 4 (бинарно 100), бројач би завршио у стању 1 (бинарно 001) или 5 (бинарно 101), за шта не постоји могућност при употреби Грејевог кода јер се у њему из стања 011 прелази у стање 010, при чему се мијења само

Табела 1: Првих осам цифара у Грејевом коду

Децимални запис	Бинарни запис	Грејев код	Грејев код у децималном запису
0	000	000	0
1	001	001	1
2	010	011	3
3	011	010	2
4	100	110	6
5	101	111	7
6	110	101	5
7	111	100	4

један бит. Треба напоменути да могућност појаве метастабилности и даље постоји јер је то много општија појава иза проблема погрешног узорковања у случају када се у исто вријеме мијењају такт и подаци, али Грејев код омогућује да се потенцијална грешка сведе на максимално један бит.

Недостатак овог приступа је то што имплементација Грејевог бројача из овог рада има мању максималну радну фреквенцију од природног бинарног бројача. Да би се значајно више смањио ризик од метастабилности, сваки бит са излаза Грејевог бројача се пропушта кроз синхронизатор са два флип-флопа да би безбједно прешао у подручје референтног такта. Затим, синхронизована вриједност Грејевог бројача се претвара у природан бинарни формат и узоркује се за даљу обраду.

3.2 Двостепени контролер

У теорији управљања, двостепени контролер (двофазни, bang-bang или on-off контролер) је контролер повратне спреге које се нагло пребацује између два стања. Математички модел двостепеног контролера може се представити преко Хевисајдове функције (или јединичне одскочне функције) која се иначе користи у математици система управљања и обради сигнала да би се представио сигнал који мијења стање у одређено вријеме. Она има вриједност 0 за негативне вриједности аргумента и 1 за позитивне вриједности аргумента:

Слика 3: График Хевисајдове функције.

$$u(x) = \begin{cases} 0, & x < 0 \\ 1, & x > 0 \end{cases} \tag{4}$$

Главне предности примјене двостепеног контролера су једноставна имплементација и брз одговор на промјене, док су недостаци могуће осцилације око жељене вриједности након што се она постигне, као и непогодност за системе гдје је потребно веома прецизно управљање.

Као што је већ наведено, FLL из овог рада има два међусобно искључива управљачка блока. Један управљачки блок FLL-а је веома сличан двостепеном контролеру описаном изнад. Он пореди улазни умножак фреквенције са узоркованом вриједношћу бројача и одлучује да ли инкрементирати, декрементирати или онемогућити предстојећи блок тј. двосмјерни бројач (енгл. *Up-Down Counter*). Ако је улазни умножак фреквенције већи од вриједности бројача узорковане у блоку управљачке предобраде, тада се двосмјерни бројач инкрементира, ако је мањи тада се декрементира, а у случају да је једнак, двосмјерни бројач задржава претходно стање. То осигурава постепено управљање и закључавање све до постизања жељене фреквенције тј. преласка у стабилно стање. Како би се спријечиле (или барем прориједиле) могуће осцилације око жељене вриједности, у овом блоку је доведен додатни сигнал који омогућава да се, након што систем једном уђе у стабилно стање, бројач двостепеног контролера не инкрементира или декрементира баш након сваког одступања од жељене вриједности јер се очекује да у стабилном стању та одступања буду незнатна. Тај додатни сигнал се користи и у PID контролеру који представља други управљачки режим о коме ће више ријечи бити у наредном поглављу.

3.3 PID контролер

У другом управљачком режиму, управљачка бинарна ријеч за DCO се генерише подесивим PID контролером. PID контролер је далеко најчешће коришћен алгоритам за управљање у инжењерству. Највећим бројем повратних спрега се управља преко PID контролера или његових дјелимичних варијација [7]. PID контролер је управљачки механизам заснован на повратној спрези, који ради тако што непрекидно исправља и скалира сигнал грешке. Грешка у овом случају представља разлику између измјерене вриједности у управљачкој предобради (узоркована вриједност бројача) и жељене референтне задате вриједности (вриједност улазног умношка фреквенције). Исправљање и скалирање се распоређује у три компоненте које су имплементиране помоћу подесивих коефицијената, цијелих бројева у бинарном формату са фиксном тачком, добијених из регистарске банке:

- Пропорционална (P) - управља брзином одзива управљачког системам непосредно множећи сигнал грешке константним чиниоцем који се назива и константа пропорционалног појачања. Ако је она превелика, систем може постати нестабилан. Насупрот томе, њена премала вриједност доводи до малог излазног одзива на велику улазну грешку чиме контролер постаје мање осјетљив.

- Интегрална (I) - користи се за смањење грешке стабилног стања (енгл. *Steady State*) скалирањем грешке константним чиниоцем и сумирањем резултата током времена. Грешка стабилног стања је разлика између жељене и стварне вриједности коначног излаза [8]. Интегрално дејство убрзава кретање излаза ка жељеној вриједности.

- Диференцијална (D) - пропорционална брзини промјене грешке, и њен циљ је ограничити излаз да би се смањила могућа прекорачења или осцилације узро-

ковane P и I компонентама, без смањења брзине контролера. Како је у овом систему референтна задата вриједност константна и нема брзих промјена на улазу које могу изазвати такав исход, P и I компоненте су довољне за гладак и стабилан одзив система.

Математички, цјелокупна управљачка функција PID контролера представља суму пропорционалног, интегралног и диференцијалног дејства и записује се на следећи начин:

$$y(t) = K_\mathrm{p} e(t) + K_\mathrm{i} \int_0^t e(\tau) d\tau + K_\mathrm{d} \frac{de(t)}{dt}, \tag{5}$$

гдје су K_p, K_i и K_d коефицијенти за пропорционални, интегрални и диференцијални дио, респективно. У стандардном облику, једначина 5 се записује као

$$y(t) = K_\mathrm{p} \left(e(t) + \frac{1}{T_\mathrm{i}} \int_0^t e(\tau) d\tau + T_\mathrm{d} \frac{de(t)}{dt} \right). \tag{6}$$

Као што се може видјети, коефицијенти K_i и K_d су редом замјењени са $K_\mathrm{p}/T_\mathrm{i}$ и $K_\mathrm{p} T_\mathrm{d}$, а предност таквог записа јесте што T_i и T_d имају одређено разумљиво физичко значење, јер представљају интегрално и диференцијално вријеме, респективно. $K_\mathrm{p}/T_\mathrm{i}$ одређује колико временски дуго ће контролер толерисати излаз који се налази изнад или испод жељене вриједности. $K_\mathrm{p} T_\mathrm{d}$ је временска константа којом се контролер покушава приближити жељеној вриједности.

Слика 4: Блок дијаграм PID контролера у повратној спрези.

Слика 4 представља уопштен блок дијаграм PID контролера у повратној спрези који приказује структуру компоненти контролера и њихов принцип рада. PID контролер константно рачуна вриједност грешке $e(t)$ као разлику између жељене вриједности (енгл. *Setpoint*) $u(t)$, која у овом конкретном случају представља улазни умножак фреквенције, и измјерене процесне вриједности $y(t)$, односно вриједности измјерене у блоку управљачке предобраде. У том случају, једначина грешке може се записати као

$$e(t) = u(t) - y(t). \tag{7}$$

Одзив система тј. брзина и начин достизања жељене фреквенције у PID режиму рада највише зависи од одабира константи контролера, а у овом раду је остављена могућност софтверског уписа константи у регистре, што даље омогућава њихово накнадно мијењање и прилагођавање. У поглављу 7, гдје је приказан софтверски

модел понашања FLL-а имплементиран у Пајтон програмском језику, поред осталог је дат и сликовит приказ утицаја константи PID контролера на одзив система.

3.4 Управљачки декодер

Улога управљачког декодера је претворити управљачке податке из једне бинарне вриједности у скуп управљачких улаза DCO-а. Постоје три таква улаза: *Row On*, унарни вектор, који може да укључује само читаве редове тростатичких инвертора DCO-а; *Row Select*, један од n (енгл. *One-Hot*) вектор, који укључује један додатни ред тростатичких инвертора DCO-а; и *Column Select*, унарни вектор, који може да укључује колоне тростатичких инвертора DCO-а.

Један од n вектор представља групу битова гдје је једина исправна комбинација она са једном јединицом и свим осталим нулама [9]. При представљању неког природног броја као један од n вектора, вриједност броја је једнака позицији јединице у вектору. С друге стране, да би се природан број N представио као унарни вектор, јединица се понавља N пута узастопно [10]. Табела 2 приказује примјер представљања природних бројева у унарном и један од n запису.

Табела 2: Поређење унарног и један од n записа

Децимални	Бинарни	Унарни	Један од n
0	000	00000000	00000001
1	001	00000001	00000010
2	010	00000011	00000100
3	011	00000111	00001000
4	100	00001111	00010000
5	101	00011111	00100000
6	110	00111111	01000000
7	111	01111111	10000000

Да би се инвертор укључио, или *Row On*, или и *Row Select* и *Column Select* за одговарајући бит морају бити подешени на 1 (једначина 46). Ширина сваког вектора је једнака ширини улаза DCO-а. Структура DCO-а је детаљно описана у поглављу 5.

Управљачки податак који долази из претходног степена тј. контолера представља број тростатичких инвертора DCO-а који требају бити укључени, и што их је више укључено, то је излазна фреквенција већа. Рецимо да је тај број 50, и претпоставимо да је DCO састављен од 255 тростатичких инвертора које поједностављено можемо посматрати као прекидаче и нека су организовани у 17 редова и 15 колона. Како постоји 17 редова, ширине *Row On* и *Row Select* вектора биће по 17 битова, док ће због 15 колона, ширина *Column Select* вектора бити 15 битова. То значи, да би се укључило 50 прекидача, потребно је укључити 3 читава реда прекидача и још 5 прекидача из четвртог реда. Да би се то постигло, управљчки улази DCO-а морају имати следеће вриједности:

- *Row On* = 00000000000000111 → Сви прекидачи у три прва реда су укључени.
- *Row Select* = 00000000000001000 → У четвртом реду је укључено још инвертора.
- *Column Select* = 000000000011111 → Пет инвертора је укључено у реду који је одређен са *Row Select*.

4 Теоријско разматрање осцилатора

У срцу сваке фазно затворене петље се налази осцилатор који игра кључну улогу у учинку који може бити постигнут [11]. Имајући то у виду, у наставку ће кроз теорију и примјере бити описани основни концепти осциловања, а након тога се прелази на разматрање концепта дигитално контролисаног осцилатора.

4.1 Основни принципи осциловања

У сврху теоријског разматрања осцилатора, као примјер биће узето клатно, као на Слици 5.

Слика 5: Клатно које дјелује као осцилаторни систем [11].

Ако пустимо клатно под одређеним углом, оно се неко вријеме њише и постепено се зауставља. Осцилација почиње јер се првобитна потенцијална енергија претвара у кинетичку енергију како клатно достиже свој вертикални положај, дозвољавајући му да настави своју путању до другог екстремног угла (положај 3), под којим је енергија поново у потенцијалном облику. У реалним условима осцилација престаје јер трење на оси око које се окреће клатно и отпор ваздуха претварају дио енергије клатна у топлоту у сваком периоду осциловања.

Да би се одржала осцилација, потребно је обезбиједити спољну енергију клатну, како би се надомјестио губитак изазван трењем на оси и отпором ваздуха. Рецимо ако се лагано гурне клатно сваки пут када се врати у положај 1, оно ће настављати да се њише. Ако је притисак сувише слаб, врши се недовољна компензација, чиме се дозвољава осцилацији да ослаби и временом нестане. С друге стране, ако је притисак прејак, врши се прекомјерна компензација, приморавајући амплитуду замаха да се повећава из једног циклуса у други. Такође, треба напоменути да је период осциловања независан од амплитуде осциловања [11].

Механички примјер изнад указује на неколико саставних дијелова осцилаторног система:

1. Почетна „неравнотежа“, тј. почетно стање или количина енергије (обезбијеђена довођењем клатна у положај 1).
2. Склоност да се једна врста енергије претвара у другу и обрнуто.
3. Механизам за одржавање који допуњује енергију изгубљену усљед неизбјежних несавршености.

Не садрже сви осцилаторни системи све поменуте саставне дијелове, али је корисно имати ове концепте на уму при разматрању неког система који осцилује.

Ако се идеално клатно (клатно без губитака) пусти под одређеним углом, такво клатно осцилује бесконачно. Ако се дода одређени притисак сваки пут када клатно дође до лијевог краја, тада замах наставља да расте због додатне енергије која се уноси у систем у сваком циклусу. Треба имати на уму да се тај неограничени раст не дешава ако се додатни притисак унесе на фреквенцији различитој од природне фреквенције осциловања клатна. Ако систем има склоност да осцилује на фреквенцији ω_0, тада се ствара растући осцилаторни излаз као одговор на спољашњу побуду на фреквенцији ω_0. Из друге перспективе, такав систем неограничено појачава периодични улаз на поменутој фреквенцији.

4.2 Систем осцилаторне повратне спреге

Из основа аутоматског управљања се зна да систем са негативном повратном спрегом може постати нестабилан. Ово својство се користи за конструисање осцилатора.

У сврху проучавања осцилација у фреквенцијском домену, разматрамо систем повратне спреге приказан на Слици 6, гдје негативан предзнак на једном улазу сабирача означава негативну повратну спрегу на ниским фреквенцијама. На Слици 7 је приказан примјер имплементације такве повратне спреге, који, на нискофреквентни синусоидни улаз дјелује као јединични бафер. Треба имати на уму да операциони појачавач показује занемарљив фазни помак на ниским фреквенцијама.

Слика 6: Једноставан систем повратне спреге.

Слика 7: Реализација повратне спреге коришћењем операционог појачавача.

Међутим, поставља се питање како могу системи са Слика 6 и 7 да осцилују? Записивањем преносне функције затворене петље као

$$\frac{Y(s)}{X(s)} = \frac{H(s)}{1 + H(s)}, \tag{8}$$

примјећује се да именилац пада на нулу ако је $H(s) = -1$ за неку вриједност промјенљиве s. Ако је $X(s)$ синусоида, онда је $s = j\omega_0$ и мора важити $H(j\omega_0) = -1$. С

обзиром да је у питању синусоида, то ће на ω_0 важити за фазни помјерај од $-180°$. То даље значи да важи $(Y/X)(j\omega_0) \to \pm\infty$, одакле се може закључити да систем обезбјеђује неограничен добитак за такву синусоиду. Као што је претпостављено у претходним поглављу, такво понашање наговјештава осцилаторну петљу.

Једнакост $H(j\omega_0) = -1$ значи да $H(s)$ инвертује улаз на фреквенцији ω_0, што је илустровано на Слици 8. Другим ријечима, $H(s)$ има толико помјерања фазе (или кашњења) на ω_0 да укупна повратна спрега постаје позитивна. То се може видјети постављањем главног улаза, $X(s)$, на нулу, прекидом петље и примјеном стимулуса на истој фреквенцији (Слика 9). Враћени сигнал је у фази са испитним напоном, V_t. У овом случају, каже се да петља садржи фазни помјерај од 180° због номинално негативне повратне спреге и други, фреквенцијски зависан, фазни помјерај од 180° који произилази из $H(s)$. Та два фазна помјераја се не смију мијешати један с другим.

Слика 8: Инверзија сигнала на фреквенцији ω_0.

Слика 9: Пропагација синусоиде кроз петљу на фреквенцији ω_0.

Укупан фазни помјерај од 360° на ω_0 имплицира да се сигнал враћа да би се појачао док кружи у петљи. Ова појава доводи до пораста амплитуде јер је враћени сигнал бар толико велик као и почетни сигнал, односно зато што је $|H(j\omega_0)| = 1$. Стога се услови за осциловање сумирају као:

$$|H(j\omega_0)| = 1 \tag{9}$$

$$\angle H(j\omega_0) = -180°, \tag{10}$$

и називају се Баркхаузеновим критеријумима [11]. Услов $H(j\omega_0) = -1$ се назива и услов покретања или почетни услов.

Нагомилавање осцилација се такође може посматрати и у временском домену. Са Слике 10а се примјећује да је, са $H(j\omega_0) = -1$, излаз једнак улазу, али помјерен за 180°. Ако је петља затворена (Слика 10б), излаз се одузима од улаза, што даје већу

Слика 10: Раст амплитуде улазне синусоиде кроз петљу у току времена.

амплитуду (замах) на A. Тај сигнал се затим поново инвертује и одузима од улаза, и тако у круг, што доводи до неограниченог раста амплитуде (Слика 10в).

Укратко, систем са негативном повратном спрегом може да генерише растући периодични излаз као одговор на синусоидни улаз ако његово појачање у петљи достигне -1 на коначној фреквенцији, ω_0. Али, да ли такав систем осцилује ако не примјенимо никакав улаз? Одговор је да, јер широкопојасни шум уређаја унутар петље има коначну енергију у близини ω_0, стварајући малу компоненту која кружи у петљи и изазива осцилације. Нпр. као што је приказано на Слици 11, извор шума, V_n, на улазу $H(s)$ даје излаз који је једнак:

$$Y = V_n \frac{H(s)}{1 + H(s)}, \tag{11}$$

чиме се доводи до неограниченог појачања за $s = j\omega_0$. То такође значи да, чак иако је V_n бесконачно мало на ω_0, на Y се може претпоставити појављивање неке коначне амплитуде осциловања тј. замаха (енгл. *Swing*).

Слика 11: Извор шума у систему затворене петље.

4.3 Основе прстенастог осцилатора

Прстенасти осцилатори су у данашње вријеме веома популарни у PLL системима захваљујући својој прилагодљивости дизајна и широког опсега подешавања фреквенције. Ово поглавље уопштено разматра основе прстенастих осцилатора, да би се стекло дубље разумијевање рада коришћеног дигитално контролисаног осцилатора, чији конкретан дизајн је описан у поглављу 5.

Слика 12 приказује основни концепт прстенастог осцилатора са три степена. Ако се користи само један степен, тада се не обезбјеђује довољан фазни помјерај да би важио услов из једначине 10. Чак и примјеном два степена, систем не може да осцилује јер $\angle H(j\omega_0)$ достиже $-180°$ тек за $\omega = \infty$ [11]. Због тога се претпоставља да коришћењем

три степена могу бити задовољени услови из једначина 9 и 10, па се стога врши даља анализа како би се то и показало.

На Слици 12, коло означено испрекиданом линијом је означено са $-H(s)$ да би било у складу са системом негативне повратне спреге приказаним на Слици 7.

Слика 12: Три степена са заједничким извором у повратној спрези.

Слика 13: Еквивалентно коло једног степена у петљи.

На Слици 13 приказано је еквивалентно коло на нивоу малих сигнала за један степен прстенастог осцилатора приказаног на Слици 12. У случају да за тренутак занемаримо капацитивно оптерећење тј. да важи $C_L = 0$, добија се уобичајена функција одзива појачавача

$$v_o = -K(V_I - V_T)R_L v_i = -g_m R_D v_i. \tag{12}$$

Када су $v_i = V_i \cos(\omega t)$ и $C_L = 0$, тада је напон на излазу једнак

$$v_o = -g_m R_D V_i \cos(\omega t). \tag{13}$$

У једначинама 12 и 13, са g_m је означена транскондуктанса NMOS транзистора која представља фактор појачања транзистора. Она је обрнуто сразмјерна отпорности и уопштено за MOSFET је једнака

$$g_m = K(V_I - V_T), \tag{14}$$

гдје је $V_I = V_{GS}$ тј. напон од гејта до сорса, V_T је напон прага, а K је параметар повезан са физичком структуром MOSFET-а и једнак је

$$K = K_n \frac{W}{L}. \tag{15}$$

Са W је означена ширина гејта, са L његова дужина, док је K_n константа повезана са MOSFET карактеристикама као што је дебљина оксида [12].

Ако се сада и кондензатор узме у разматрање, да би се пронашла веза између v_i и v_o за коло са Слике 13, потребно је направити импедансни модел кола, а он је приказан на Слици 14а.

Слика 14: Импедансни модел еквивалентног кола једног степена у петљи.

Ефективна вриједност импедансе оптерећења, Z_L (Слика 14б) дата је са

$$Z_L = R_D || \frac{1}{sC_L}. \tag{16}$$

Тада је израз за амплитуду излаза (V_o) једнак

$$V_o = -g_m Z_L V_i = -g_m \left(R_D \,||\, \frac{1}{sC_L} \right) V_i. \tag{17}$$

То је еквивалентно са

$$\frac{V_o}{V_i} = -g_m \left(R_D \,||\, \frac{1}{sC_L} \right). \tag{18}$$

Како је једначина 18 изведена за један степен, тада за преносну функцију тростепеног прстена са Слике 12 важи [11]

$$-H(s) = \left[-g_m \left(R_D \,||\, \frac{1}{sC_L} \right) \right]^3. \tag{19}$$

Ако је $s = j\omega_0$, тада једначину 19 пишемо као

$$H(j\omega_0) = \left[g_m \left(R_D \| \frac{1}{C_L j\omega_0} \right) \right]^3 = \left(\frac{g_m R_D \dfrac{1}{C_L j\omega_0}}{R_D + \dfrac{1}{C_L j\omega_0}} \right)^3 = \left(\frac{g_m R_D}{1 + R_D C_L j\omega_0} \right)^3. \tag{20}$$

Добијени резултат представља комплексни разломак који даље желимо представити у стандардном комплексном облику $z = x + jy$:

$$\begin{aligned} H(j\omega_0) &= \left(\frac{g_m R_D}{1 + R_D C_L j\omega_0} \cdot \frac{1 - R_D C_L j\omega_0}{1 - R_D C_L j\omega_0} \right)^3 = \\ &= \left(\frac{g_m R_D - g_m R_D^2 C_L j\omega_0}{1 + R_D^2 C_L^2 \omega_0^2} \right)^3 = \\ &= \left(\frac{g_m R_D}{1 + R_D^2 C_L^2 \omega_0^2} - j \frac{g_m R_D^2 C_L \omega_0}{1 + R_D^2 C_L^2 \omega_0^2} \right)^3. \end{aligned} \tag{21}$$

Да би се поставио услов $|H(j\omega_0)| = 1$, потребно је наћи модул (интензитет) степеноване вриједности комплексног израза добијеног изнад. Рецимо да имамо неки комплексан број $z = x + jy$. Он се може представити и у поларном облику као

$$z = A(\cos\theta + j\sin\theta), \tag{22}$$

гдје су:

- $A = |z| \rightarrow$ модул (апсолутна вриједност) комплексног броја,
- $\theta = \arg(z) = \arctan\left(\dfrac{y}{x}\right) \rightarrow$ фазни угао (аргумент).

Коришћењем Ојлерове формуле, $e^{j\theta} = \cos\theta + j\sin\theta$, комплексни број можемо представити као

$$z = Ae^{j\theta}. \tag{23}$$

Ако тако представљен комплексни број степенујемо неким природним бројем n, добија се

$$z^n = \left(Ae^{j\theta}\right)^n = A^n \left(e^{j\theta}\right)^n = A^n e^{jn\theta}. \tag{24}$$

То значи да степеновањем комплексног броја важи следеће:

1. Модул комплексног броја се подиже на исти степен.
2. Фазни угао се множи са вриједношћу степена.

Ако се први закључак примијени на наш случај, важи следеће:

$$
\begin{aligned}
|H(j\omega_0)| &= \left| \frac{g_m R_D}{1 + R_D^2 C_L^2 \omega_0^2} - j \frac{g_m R_D^2 C_L \omega_0}{1 + R_D^2 C_L^2 \omega_0^2} \right|^3 = \\
&= \left(\sqrt{\frac{g_m^2 R_D^2}{(1 + R_D^2 C_L^2 \omega_0^2)^2} + \frac{g_m^2 R_D^4 C_L^2 \omega_0^2}{(1 + R_D^2 C_L^2 \omega_0^2)^2}} \right)^3 = \\
&= \left(\sqrt{\frac{g_m^2 R_D^2 (1 + R_D^2 C_L^2 \omega_0^2)}{(1 + R_D^2 C_L^2 \omega_0^2)^2}} \right)^3 = \\
&= \left(\frac{g_m R_D}{\sqrt{1 + R_D^2 C_L^2 \omega_0^2}} \right)^3 .
\end{aligned}
\tag{25}
$$

На основу претходног израза, услов $|H(j\omega_0)| = 1$ се може представити као

$$
\left(\frac{g_m R_D}{\sqrt{1 + R_D^2 C_L^2 \omega_0^2}} \right)^3 = 1. \tag{26}
$$

Одатле се може изразити ω_0 на следећи начин:

$$
\omega_0 = \frac{\sqrt{g_m^2 R_D^2 - 1}}{R_D C_L}. \tag{27}
$$

Ако сада други закључак примјенимо на фазни угао преносне функције, тада ће $\angle H(j\omega_0)$ бити једнак

$$
\begin{aligned}
\angle H(j\omega_0) &= 3 \cdot \arctan \left(\frac{-\dfrac{g_m R_D^2 C_L \omega_0}{1 + R_D^2 C_L^2 \omega_0^2}}{\dfrac{g_m R_D}{1 + R_D^2 C_L^2 \omega_0^2}} \right) = \\
&= 3 \cdot \arctan(-R_D C_L \omega_0) = \\
&= -3 \cdot \arctan(R_D C_L \omega_0).
\end{aligned}
\tag{28}
$$

На основу претходног израза, услов $\angle H(j\omega_0) = -180^\circ$ може се представити као

$$
\arctan(R_D C_L \omega_0) = 60^\circ. \tag{29}
$$

Одатле се може представити ω_0 на следећи начин:

$$
\omega_0 = \frac{\sqrt{3}}{R_D C_L}. \tag{30}
$$

Ако сада изједначимо изразе 27 и 30, добија се:

$$
\frac{\sqrt{g_m^2 R_D^2 - 1}}{R_D C_L} = \frac{\sqrt{3}}{R_D C_L}. \tag{31}
$$

Одатле се даље добија:

$$
g_m R_D = 2. \tag{32}
$$

Другим ријечима, сваки степен мора обезбједити појачање напона ниске фреквенције од 2, да бе се гарантовало осциловање [11].

У општем случају, ако се уведе N идентичних степени у прстен, гдје је N непаран број, петља обезбјеђује негативну повратну спрегу на ниским фреквенцијама, што захтјева фазни помјерај од $-180°/N$ по једном степену да би систем осциловао. Тада ће ω_0 бити једнако

$$\omega_0 = \frac{1}{R_D C_L} \tan \frac{180°}{N}. \tag{33}$$

Како се N повећава, ω_0 опада јер свака фаза мора унијети мање фазног помјераја. Такође, у општем случају, како N расте, захтијевано нискофреквентно појачање се смањује што се може видјети из израза

$$g_m R_D = \sqrt{\tan^2 \frac{180°}{N} + 1}. \tag{34}$$

Видјели смо да су таласни облици прстенастог осцилаотра блиски синусоиди за $|H(j\omega_0)| = 1$. Међутим, они почињу да личе на квадратни талас ако је $|H(j\omega_0)|$ знатно изнад 1. Већина апликација тежи оштријим прелазима сигнала како би се осигурало брзо укључивање и искључивање уређаја. Стога су квадратни таласни облици пожељнији за примјену, и у те сврхе се уводи инверторски прстен обрађен у наредном поглављу.

4.3.1 Инверторски прстен

Најчешћа имплементација прстенастих осцилатора је коришћењем CMOS инвертора умјесто отпорнички оптерећених појачавача као фаза појачања у прстену. Слика 15 приказује тростепени инверторски прстен. Ако претпоставимо да је у почетку $V_X = V_Y = V_Z$, свака фаза уводи напонско појачање од $-(g_{mN} + g_{mP})(r_{ON}||r_{OP})$, које типично има вриједност преко 2 [11]. Након тога, шум унутар компоненти узрокује обнављање сигнала све док амплитуда осциловања не достигне пун напонски опсег (енгл. *Rail-to-Rail*).

Слика 15: Тростепени прстенасти осцилатор састављен од CMOS инвертора.

Шта ако, након укључења, прстен почиње са једним чвором на нули? Рецимо, ако је $V_X = 0$, тада је $V_Y = V_{DD}$, а $V_Z = 0$, што значи да трећи инвертор подиже V_X на

V_{DD}. Слика 16 приказује идеализоване облике сигнала на сва три степена у прстену, у току времена. Закључујемо да прелаз на једном чвору узрокује прелаз на наредном након одређеног кашњења, T_D. Укупан период осциловања је стога једнак $6T_D$, па је фреквенција осциловања једнака $f = 1/6T_D$.

Слика 16: Облици дигиталног сигнала на фазама инверторског прстена.

Стварни облици сигнала претходног осцилатора нису тако оштри као на Слици 16, али нису ни правилни као синусоида. Реалнији примјер је илустрован на Слици 17а, гдје V_X мијења правац чим достигне V_{DD} или 0, због кратког кашњења кроз петљу. Поређења ради, петостепени прстен омогућава осцилације са периодом $10T_D$, дозвољавајући сигналу на V_X да дуже времена остане на високом, односно ниском напонском нивоу (Слика 17б).

(а)

(б)

Слика 17: Реалан облик сигнала (а) тростепеног и (б) петостепеног прстенастог осцилатора.

Прстенасти осцилатори су уопштено прилично осјетљиви на вриједности напона напајања. Прво, максимална фреквенција осциловања расте како расте напон напајања. Друго, динамичка потрошња снаге прстенастог осцилатора има чак квадратну зависност од вриједности напона напајања, о чему ће више ријечи бити у поглављу 5.

4.4 Шум у осцилаторима

За идеалан осцилатор који ради на фреквенцији ω_c, сва његова снага је концентрисана на једну фреквенцију, ω_c, као што је приказано на Слици 18а. У реалном осцилатору, спектар се шири на околне фреквенције око ω_c (Слика 18б). У осцилаторима се такво ширење назива фазни шум.

Слика 18: Излазни спектар (а) идеалног и (б) реалног осцилатора.

Фазни шум се обично карактерише у фреквенцијском домену. За идеалан осцилатор који ради на фреквенцији ω_c, излазни напон може бити изражен као

$$v(t) = A\cos(\omega_c t + \phi), \tag{35}$$

гдје је A амплитуда, а ϕ је произвољна, али фиксна референтна фаза. Снага је концентрисана на једну фреквенцију ω_c. Еквивалентно, спектар снаге се може записати као

$$S_v(\omega) = \frac{A^2}{2}\delta(\omega - \omega_c), \tag{36}$$

гдје је δ јединични импулс или Диракова делта функција (Слика 18а). Међутим, у реалном осцилатору и амплитуда и фаза су временски промјенљиве, и спектар ће се зато ширити око фреквенције носиоца на околне фреквенције. У већини случајева, поремећај у амплитуди је занемарљив или неважан, а и може се лако уклонити додавањем кола која ограничавају амплитуду [6]. Стога се разматра само насумично одступање фазе:

$$v(t) = A\cos[\omega_c t + \phi(t)], \tag{37}$$

гдје је $\phi(t)$ мало насумично одступање фазе које представља варијације у периоду и најчешће се назива фазни шум. За мале вриједности варијација фазног шума, $|\phi(t)| \ll 1\,\mathrm{rad}$, једначина 37 се може поједноставити на

$$v(t) = A\cos\omega_c t + A\phi(t)\sin\omega_c(t), \tag{38}$$

што значи да се спектар $\phi(t)$ фреквенцијски преводи у $\pm\omega_c$.

Фазни шум се може квантификовати узимањем опсега од 1 Hz на растојању $\Delta\omega$ од носиоца, затим рачунањем снаге шума у том опсегу и дијељењем тог резултата са

снагом носиоца [6]. То се назива једнострани (енгл. *single-sided*) фазни шум и изражава се у јединици децибела по херцу [dBc/Hz]. По конвенцији, „c“ у „dBc“ значи „у односу на носилац“.

$$\mathcal{L}(\Delta\omega) = 10\log_{10}\left(\frac{\text{снага шума у опсегу од 1 Hz при фреквенцији } w_c + \Delta\omega}{\text{снага носиоца}}\right) \quad (39)$$

Једнострани фазни шум из једначине 39 је половина спектра фазног шума, тј. може се записати као

$$\mathcal{L}(\Delta\omega) = 10\log_{10}\left(\frac{S_\phi(\Delta\omega)}{2}\right), \quad (40)$$

гдје је $S_\phi(\Delta\omega)$ дато са

$$S_\phi(\Delta\omega) = \frac{S_v(\Delta\omega)}{\text{снага носиоца}}. \quad (41)$$

Типични спектар фазног шума осцилатора приказан је на Слици 19. У овом логаритамском дијаграму, фазни шум нормализован на dBc/Hz, приказан је у односу на офсет фреквенцију $\Delta\omega$ од носиоца ω_c.

Слика 19: Спектар фазног шума реалног осцилатора.

Профил фазног шума прати криву приказану на слици, гдје пролази кроз области са нагибима од $1/\omega^3$, $1/\omega^2$ и $1/\omega^0$. Област $1/\omega^2$ се уопштено назива област топлотног шума (енгл. *Thermal Noise Region*) зато што је изазвана тзв. бијелим или неповезаним временским флуктуацијама унутар периода осциловања. Шум треперења (енгл. *Flicker Noise*) $1/f$ електронских уређаја је такође значајан за ниже офсетне фреквенције. Он се конвертује навише у $1/\omega^3$ област. Коначно, $1/\omega^0$ је топлотни електронски шум који се додаје изван осцилатора, као нпр. у излазном баферу, и не утиче на временски одзив осцилатора.

5 Дигитално контролисани осцилатор

Дигитално контролисани осцилатор описан у овом раду је прстенасти осцилатор, погодан за систем генерисања такта. Прстенасти осцилатор је каскадна комбинација фаза кашњења повезаних у ланац затворене петље [13]. Прстенасте архитектуре су компактније од LC осцилатора и имају доста предности захваљујући својој правилној и периодичној просторној структури. Уопштена структура DCO-а коришћеног унутар описаног FLL-а заснована је на матрици тростатичких CMOS инвертора [14]. Ова матрица је састављена од N-фазних прстена тростатичких инвертора повезаних паралелно. N представља број DCO фаза (степени) и мора бити непаран број већи или једнак 3. Слика 20 приказује примјер N-степеног дигитално контролисаног осцилатора који има један прстен стално укључених инвертора (најдоњи прстен без стрелице управљачког сигнала са доње стране инвертора).

Слика 20: N-степени дигитално контролисани осцилатор.

У физичком смислу, матрица може бити преобликована у квадрат, што омогућава једноставнију управљачку логику. На Слици 20, основну фреквенцију DCO-а дефинише само један прстен који је стално укључен, а у општем случају их може бити више. Остали тростатички инвертори се укључују и искључују у зависности од управљачке логике.

Формула за фреквенцију осциловања DCO-а гласи:

$$f_{\mathrm{osc}} = \frac{1}{2Nt_{\mathrm{d}}} \approx \frac{I_{\mathrm{d}}}{2NC_{\mathrm{load}}V_{\mathrm{DDL}}}, \tag{42}$$

гдје је N број тростатичких инвертора унутар прстена, t_{d} представља кашњење једне ћелије DCO-а, у чијем саставу је тростатички инвертор (у наставку ће бити детаљније објашњена структура саме ћелије DCO-а), I_{d} је струја која протиче кроз инвертор, C_{load} је капацитивно оптерећење истог инвертора, и V_{DDL} је напон напајања DCO-а. Производ Nt_{d} је помножен са 2 да би се добила читава периода такта, а не полупериода.

Када је ријеч о топологији, са повећањем броја DCO фаза (степени), фреквенцијски корак (K_{DCO}) опада, чиме се повећава прецизност DCO-а. Максимална фреквенција осцилатора се се такође смањује, и да би се то надомјестило, напон напајања се може повећати, што с друге стране доводи до веће потрошње снаге. Ако претпоставимо да напон напајања и капацитивно оптерећење по једној фази остану непромјењени, повећање броја фаза не утиче на потрошњу снаге. Међутим, ако укупан број тростатичких инвертора остане непромјењен и подијели се на већи број фаза, то ће довести до смањења капацитивног оптерећења по фазама појединачно, што даље доводи до смањења потрошње снаге. Математичком анализом се то може објаснити на следећи начин: N-фазни прстенасти осцилатор који ради на фреквенцији f_{osc} има динамичку потрошњу снаге која се може представити једначином

$$P = N f_{\mathrm{osc}} C_{\mathrm{tot}} V_{\mathrm{DDL}}^2, \tag{43}$$

гдје C_{tot} представља укупно капацитивно оптерећење на једној фази. Пошто је фреквенција осциловања једнака $f_{\mathrm{osc}} = 1/(2Nt_{\mathrm{d}})$, једначину за динамичку снагу можемо написати на следећи начин:

$$P = \frac{C_{\mathrm{tot}} V_{\mathrm{DDL}}^2}{2t_{\mathrm{d}}}, \tag{44}$$

одакле се види да је добијена динамичка снага независна од N [11].

С обзиром да је одређени број тростатичких инвертора увијек укључен, DCO има одређену почетну фреквенцију која не зависи од управљачке ријечи DCO-а. Тренутна излазна фреквенција ће стога у основи зависити од почетне фреквенције, фреквенцијског корака тј. резолуције и управљачке ријечи. Стога се може написати следећа линеарна зависност фреквенције осциловања, која може послужити за софтверску имплементацију рада DCO-а [15]:

$$f_{\mathrm{DCO}} = f_0 + K_{\mathrm{DCO}} \cdot d, \tag{45}$$

гдје је f_0 почетна фреквенција, K_{DCO} резолуција фреквенције, а d је управљачка дигитална ријеч.

5.1 Архитектура предложеног DCO-а

У овом раду описана је топологија DCO-а са пет фаза, због тога што је таквом топологијом остварен задовољавајући компромис између учинка и потрошње снаге тј. под задовољавајућим условима се достиже жељена фреквенција са довољном прецизношћу. Слика 21 приказује структуру DCO-а коришћеног у описаној FLL имплементацији.

Свака фаза DCO-а састоји се од 54 тростатичка инвертора, што ако помножимо са бројем фаза даје укупно 270 инвертора. Тростатички инвертори су распоређени у 18 редова и 15 колона. Управљачка логика FLL-а управља са 17 редова и свих 15 колона, што значи да постоји 255 фреквенцијских корака. Преостали ред са 3×5 тростатичких инвертора је стално укључен и на њега не утиче управљачка логика FLL-а. Што се више тростатичких инвертора у свакој фази укључи под утицајем управљачке логике FLL-а, повећавају се струја и тренутна снага покретања (енгл. *Driving Strength*) једне фазе, док њено капацитивно оптерећење у суштини остаје константно, што резултује повећањем излазне фреквенције осциловања [4]. Поред

Слика 21: Петостепени прстенасти дигитално контролисани осцилатор (DCO) састављен од тростатичких инвертора, са додатим претварачима напонских нивоа (HLLS и LHLS) и уквиреним редом стално укључених тростатичких инвертора.

самих тростатичких инвертора, на Слици 21 се могу уочити и блокови претварача напонког нивоа на управљачким улазима и фазама DCO-а, чија је намјена детаљно објашњена у поглављу 5.3.

5.2 Блок дијаграм ћелије DCO-а

Да би се ефикасно подесила излазна фреквенција DCO-а, уведен је скуп управљачких улаза DCO-а, а то су сигнали *Row On*, *Row Select* и *Column Select*, поменути такође у поглављу 3.4. Према томе, сваки тростатички инвертор садржи сопствену управљачку јединицу у облику И-ИЛИ стандардне ћелије, и заједно граде већи блок назван ћелија DCO-а. Слика 22 приказује шему ћелије DCO-а на нивоу логичких кола и на нивоу CMOS транзистора. Уочава се да И-ИЛИ коло и тростатички инвертор имају заједнички инвертор како би се уштедило на броју транзистора.

У дигиталном домену сигнал Out може имати три стања. Једно је недефинисано, када је тростатички инвертор искључен. А друга два су када је тростатички инвертор укључен и представљају инвертован сигнал In. Укључивање и искључивање тростатичког инвертора контролише ctrl сигнал који представља излаз из контролног дијела DCO ћелије тј. И-ИЛИ блока стандардне ћелије. За вриједност ctrl сигнала 1, тростатички инвертор је укључен, а за вриједност 0 је искључен. Табела 3 је таблица истинитости управљачког сигнала ctrl. Логички израз за сигнал ctrl у зависности од управљачких сигнала редова и колона гласи:

$$\text{ctrl} = (\text{Row Sel.} \cdot \text{Col. Sel.}) + \text{Row On} \tag{46}$$

Табела 4 приказује табелу истинитости Out сигнала у зависности од управљачког сигнала ctrl и улазног In.

(а)

(б)

Слика 22: Ћелија DCO-а на нивоу (а) логичких кола и (б) CMOS транзистора.

Табела 3: Таблица истинитости контролног сигнала тростатичког инвертора.

Row On	Row Sel.	Col. Sel.	ctrl
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Табела 4: Таблица истинитости излазног сигнала тростатичког инвертора.

ctrl	In	Out
0	0	X
0	1	X
1	0	1
1	1	0

5.3 Претварачи напонског нивоа DCO-а

Напон напајања који користи дигитално контролисани осцилатор (V_{DDL}) се у овом раду разликује од напона напајања који користи остатак логике FLL-а (V_{DD}). Предност таквог дизајна је могућност подешавања напона напајања DCO-а независно након производње чипа, што даље омогућава накнадно постизање жељене резолуције фреквенције (фреквенцијског корака) и фреквенцијског опсега зависно од процесног угла у коме се одвијала производња чипа. Додатна предност јесте и то што могућност смањења напона напајања DCO-а аутоматски доводи и до значајног смањења потрошње због њене квадратне зависности од напона напајања. Такав независан домен напајања DCO-а постигнут је додавањем претварача напонског нивоа на улазе и излазе DCO-а, као што је и приказано на Слици 21.

(а)

(б)

Слика 23: Шема претварача (а) са високог на низак и (б) са ниског на висок напонски ниво [16].

Како је напон напајања DCO-а нижи од остатка система, на управљачке улазе DCO-а постављени су претварачи са високог на низак напонски ниво (енгл. *High-Low Level Shifter*, HLLS), док су на фазне излазе DCO-а постављени претварачи са ниског на

висок напонски ниво (енгл. *Low-High Level Shifter*, LHLS). Слика 23 приказује шеме конвенционалних претварача напонског нивоа који су коришћени у претходно описаном систему. Као што се може видјети са Слике 23а, претварач са високог на низак напонски ниво није ништа друго до обичан бафер састављен од два инвертора чије напајање ће у коначној реализацији долазити од линије ниског напонског нивоа (V_{DDL}). Слика 23б приказује шему претварача са ниског на висок напонски ниво, представља регенеративно логичко коло засновано на позитивној повратној спрези [16] и састоји се од два унакрсно спрегнута PMOS транзистора (MP1 и MP2) и од NMOS транзистора (MN1 и MN2), који се покрећу преко комплементарних улазних сигнала INL и $\overline{\mathrm{INL}}$, гдје је за инвертовање сигнала потребно додати инвертор.

С обзиром да је претварач са ниског на висок напонски ниво нешто комплекснији, његов рад ће бити детаљније објашњен. Када напони сигнала INL и $\overline{\mathrm{INL}}$ имају вриједност 0 и V_{DDL}, тада ће MN1 и MN2, редом, бити искључен и укључен. MN2 затим повлачи на нулу напон чвора OUTH, укључујући тако MP1. Тада се и напон чвора $\overline{\mathrm{OUTH}}$ подиже на V_{DD}, MP2 се искључује, а напон на чвору OUTH остаје на нули. С друге стране ако су вриједности напона на INL и $\overline{\mathrm{INL}}$ једнаки V_{DDL} и 0, тада ће MN1 и MN2 бити укључен и искључен, респективно. MN1 тада повлачи на нулу напон чвора $\overline{\mathrm{OUTH}}$, укључујући тако MP2, чиме се напон OUTH подиже на V_{DD}, а MP1 се искључује.

Треба запазити да је напон чвора OUTH одређен покретачком струјом (енгл. *Drive Current*) транзистора MP2 и MN2, тј. струјом која тече кроз канал транзистора када су укључени. Зато, ако је покретачка струја кроз транзистор MP2 много већа него кроз MN2, тада OUTH не може бити испражњен што доводи до тога да претварач не може исправно радити у таквим ситуацијама [16]. До њих иначе може доћи ако је разлика између V_{DD} и V_{DDL} превелика, што у примјени из овог рада није случај.

Претварачи напонских нивоа могу утицати на фактор испуњености импулса тј. периоде такта (енгл. *Duty Cycle*), поготово на излазу претварача са ниског на висок напонски ниво. Међутим, то се углавном може избјећи додатним баферовањем излаза DCO-а тј. физичким уметањем одговарајућих бафера између фазног излаза DCO-а и претварача напонског нивоа чиме се фактор испуњености импулса такта може приближити идеалној вредности од 50 %, како би такав дошао на улаз претварача са ниског на висок напонски ниво.

6 Имплементација и резултати симулација

У овом поглављу биће приказана имплементација система кроз анализу SystemVerilog кода и хијерархијским приказом лејаута комплетног FLL-а. Затим ће бити приказани резултати симулација у различитим условима након екстракције паразитних капацитивности. Такође ће бити приказани резултати софтверског модела рада FLL-а, а биће направљено и поређење два DCO-а имплементирана у 2 различите технологије. Дигитални FLL описан у овом раду, имплементиран је коришћењем SystemVerilog језика за опис хардвера у 130 nm CMOS технологији. Улазни референтни такт је 16 MHz, док се такт DCO-а подешава до 640 MHz, при чему је резолуција учестаности око 2,8 MHz у типичним условима рада.

6.1 SystemVerilog код

У овом поглављу биће приказани најбитнији дијелови имплементације FLL-а на нивоу SystemVerilog [17] језика за опис хардвера. Уз поједине дијелове кода ће бити објашњења у сврху лакшег разумијевања. Листинг 1 приказује улазе и излазе FLL-а, који су побројани и у поглављу 2, са додатим сигналима за подешавање управљачке логике.

Листинг 1: Улазни и излазни сигнали хијерархијски највишег FLL модула.

```
 // Frequency-locked loop
module fll #(
  parameter integer CW = 8  // frequency control word width
)(
  ////////////////////
  // Clock and reset //
  ////////////////////
  input            clk_ref_i,        // [0] reference clock @ 32MHz
  input            clk_ref_en_i,     // [1] reference clock enable signal
                                     //     0: reference clock propagation disabled
                                     //     1: reference clock propagation enabled
  input            rst_ni,           // [0] async reset
                                     //     0: active/asserted
                                     //     1: inactive/deasserted
  /////////
  // FLL //
  /////////
  input  [CW-1:0] clk_mult_i,        // [0d20] reference frequency multiplier (or frequency control word)
                                     //        20 * 32MHz = 640MHz
  /////////
  // DCO //
  /////////
  input            dco_pwr_sel_i,    // [1]     DCO clock output power supply selection
                                     //         0: FLL uses DCO clock output @ VDDL (~1.1V,     adjustable)
```

```
                                        //          1: FLL uses DCO clock output @ VDD  (~1.2V, non-adjustable)
  input      [1:0] dco_ctrl_mode_i, // [1]      DCO control mode
                                        //          0d0: FIXED     - DCO control word is internally set to 0d203
                                        //          0d1: MANUAL    - DCO control word is set via "dco_ctrl_man_i" register
                                        //          0d2: BANG-BANG - DCO control word is set by a comparator and a counter
                                        //          0d3: PID       - DCO control word is PID regulated
  input  [CW-1:0] dco_ctrl_man_i,  // [0d200] DCO control word value for MANUAL mode
                                        //          In simulation in typical conditions, a value around 0d200 achieved 640MHz,
                                        //          a value of 0 achieved ~42 MHz and a value of 0d255 achieved ~764MHz
  /////////
  // PID //
  /////////
  input  [CW-1:0] pid_kp_i,         // [0b01000000] PID controller proportional constant
                                        //               A signed number in Q6.2 format (0b010000_00 == 0d16)
  input  [CW-1:0] pid_ki_i,         // [0b01000000] PID controller integral constant
                                        //               A signed number in Q6.2 format (0b010000_00 == 0d16)
  input  [CW-1:0] pid_kd_i,         // [0]           PID controller derivative constant
                                        //               A signed number in Q6.2 format
  /////////
  // FLL //
  /////////
  input      [3:0] lock_cnt_i,       // [0b1000] FLL internal counter lock condition threshold
  output           lock_o,           // [0]      FLL lock signal - indicates if FLL achieved the target frequency
                                        //           0: the FLL did not its target frequency yet
                                        //           1: the FLL achieved its target frequency
  ///////////////
  // OSC output //
  ///////////////
  output         clk_dco_o           // DCO clock - one phase
);
...
endmodule
```

У поглављу 3.1 описана је улога и начин рада управљачке предобраде, а Листинг 2 приказује хардверску имплементацију Грејевог бројача и начин како се генеришу узастопне вриједности бројача. Као што се из кода може видјети, оне се генеришу из наредне природне бинарне вриједности тако што се најстарији бит задржава, док су остали битови резултат бит по бит ексклузивно ИЛИ (енгл. *XOR*) операције између свих битова осим најмлађег и свих битова осим најстаријег (линија 22 у Листингу 2). Рецимо ако је наредна природна вриједност бројача 5 (бинарно 00000101, ако је ширина бројача 8), да би се претворила у Грејев код, задржава се најстарији бит, тј. 0, а осталих седам се додаје као резултат поменуте ексклузивно ИЛИ операције, тј. 0000010 ˆ 0000101, што даје вриједност 0000111 коју треба додати на 0, чиме се добија вриједност 00000111, која, ако погледамо Табелу 1, одговара природној вриједности 5.

Листинг 2: Модул Грејевог бројача.

```
1  module cnt_gray #(
2    parameter int  W   = 8,
3    parameter byte Val = W'(0)
4  )(
5    input          clk_i,
6    input          rst_ni,
7    input          en_i,
8    output [W-1:0] cnt_gray_o
9  );
10
11   logic [W-1:0] cnt_bin;
12   logic [W-1:0] cnt_bin_next;
13   logic [W-1:0] cnt_gray;
14   assign cnt_bin_next = cnt_bin + W'(1);
15   always_ff @(posedge clk_i or negedge rst_ni) begin
16     if (!rst_ni) begin
17       cnt_bin <= Val;
18       cnt_gray <= {Val[W-1], Val[W-1:1] ^ Val[W-2:0]};
19     end else begin
20       if (en_i) begin
21         cnt_bin <= cnt_bin_next;
22         cnt_gray <= {cnt_bin_next[W-1], cnt_bin_next[W-1:1] ^
     ↪ cnt_bin_next[W-2:0]};
23       end else begin
24         cnt_bin <= Val;
25         cnt_gray <= {Val[W-1], Val[W-1:1] ^ Val[W-2:0]};
26       end
27     end
28   end
29   assign cnt_gray_o = cnt_gray;
30
31 endmodule
```

Добијена вриједност у Грејевом коду се синхронизује кроз два флип флопа и претвара се у природну бинарну вриједност, а имплементација тога је унутар `fll` модула и дата је у Листингу 3. Узоркована вриједност Грејевог бројача је на магистрали

cnt_dco_gray_q2, док је природна бинарна вриједност на магистрали cnt_dco и она суштински представља тренутни умножак фреквенције добијене на излазу DCO-а, и представља податак спреман за поређење са жељеним умношком са улаза, унутар блока управљачке логике. Сваки бит у тој магистали се добија као резултат логичке XOR операције над свим битовима Грејевог записа, помјереног онолико мјеста удесно колико износи тежина тренутног бита у cnt_dco магистрали (имплементација на линији 7 у Листингу 3).

Листинг 3: Претварање Грејевог кода у природну бинарну вриједност.

```
1  module fll #(
2    ...
3    // Gray to binary conversion
4    genvar i;
5    generate
6      for(i=0; i<CW; i++) begin: gen_gray2bin_conv
7         assign cnt_dco[i] = ^(cnt_dco_gray_q2 >> i);
8      end: gen_gray2bin_conv
9    endgenerate
10   ...
11 endmodule
```

Двостепени контролер, као један дио управљачке логике, је подробније објашњен у поглављу 3.2. Као што је тамо напоменуто, састоји се од компаратора и једног бројача. Компаратор служи за поређење добијеног и жељеног умношка фреквенције, а резултат се шаље у бројач који на основи њега одређује како подесити управљачку ријеч за DCO. Листинг 4 приказује компараторски дио који се налази унутар fll модула. Као што се може закључити, cnt_dco_q представља семпловану вриједност тренутног умношка, односно вриједност cnt_dco из Листинга 3. Сигнал cnt_ref_up_dn одређује да ли инкрементирати или декрементирати управљачку ријеч DCO-а, док сигнал cnt_ref_en одређује да ли уопште мијењати вриједност бројача (сигнал lock_o ће бити поменут при анализи имплементације PID контолера). Закључује се да се поменути сигнали прослијеђују даље у блок бројача, чија је имплементација, као посебног cnt модула, приказана у Листингу 5. Сигнали cnt_ref_up_dn и cnt_ref_en се при инстанцирању cnt модула прослијеђују на његове up_down_i и en_i улазе, респективно.

Листинг 4: Компараторски дио двостепеног контролера.

```
module fll #(
  ...
  // Comparator
  logic  cnt_ref_up_dn;
  logic  cnt_ref_en;
  assign cnt_ref_up_dn = (clk_mult_i >= cnt_dco_q) ?    1'b1 : 1'
   ↪ b0;
  assign cnt_ref_en    = (clk_mult_i != cnt_dco_q) ? ~lock_o : 1'
   ↪ b0;
  ...
endmodule
```

Листинг 5: Имплементација бројача из двостепеног контролера.

```
// Up-down counter with enable
module cnt #(
  parameter int  W   = 8,      // count width (number of bits)
  parameter byte Val = W'(0)   // counter default value
) (
  // Clock and reset
  input                clk_i,      // clock
  input                rst_ni,     // async reset
  // Control
  input                en_i,       // enables counting
  input                up_down_i, // count direction
  // Count data output
  output logic [W-1:0] cnt_o       // count output
);

always_ff @(posedge clk_i or negedge rst_ni) begin
  if (!rst_ni) begin
    cnt_o <= Val;
  end else begin
    if (en_i) begin
      if (up_down_i == 1'b1) begin
        cnt_o <= W'(cnt_o + W'(1));
      end else begin
        cnt_o <= W'(cnt_o - W'(1));
      end
    end else begin
      cnt_o <= cnt_o;
    end
  end
end

endmodule
```

Имплементација PID контролера, детаљније објашњеног у поглављу 3.3, приказана је у Листингу 6, гдје се види да је имплементиран као посебан `pid` модул. Поред улаза такта, ресета и коефицијената контролера, ту су још улази:

- `setpoint_i` који представља жељени умножак фреквенције (при инстанцирању модула прослијеђује му се вриједност `clk_mult_i` са улаза `fll` модула),
- `feedback_i` који представља семпловани тренутни умножак (при инстанцирању модула прослијеђује му се семплована вриједност `cnt_dco` из Листинга 3) и
- `lock_cnt_i` који означава колико узлазних ивица такта грешка мора бити једнака нули да би се достигло закључавање, односно да би се излазни сигнал `lock_o` подигао на 1.

Излаз `ctrl_o` представља податак који ће да служи као управљачка ријеч за DCO (аналогно излазу `cnt_o` из `cnt` модула двостепеног контролера). Он означава 8-битну цјелобројну вриједност и добија се издвајањем 8 бита од позиције 13 до позиције 6 из промјенљиве `ctrl_pid` (линија `64` у Листингу 6) у којој је смјештена сума про-

порционалног, интегралног и диференцијалног дејства. Такво издвајање битова је резултат коришћења бинарног бројевног система са фиксном тачком, о коме се више може наћи у литератури [9], а све то у сврху повећања резолуције подешавања PID контролера. Наиме, улазни коефицијенти `kp_i`, `ki_i` и `kd_i` су 8-битни у формату Q6.2 (6 бита за цјелобројни дио, 2 за децимални дио), док је формат `ctrl_pid` промјенљиве Q12.6 (12 битова за цјелобројни дио, 6 за децимални дио), па се поменутим одсијецањем врши заокруживање на 8-битни цијели број.

Листинг 6: Имплементација PID контролера.

```
// PID controller
module pid #(
  parameter int DW = 8,  // data input width
  parameter int CW = 8   // control output width
) (
  input                 clk_i,
  input                 rst_ni,
  input        [DW-1:0] kp_i,
  input        [DW-1:0] ki_i,
  input        [DW-1:0] kd_i,
  input        [DW-1:0] setpoint_i,
  input        [DW-1:0] feedback_i,
  input        [3:0]    lock_cnt_i,
  output logic [CW-1:0] ctrl_o,
  output logic          lock_o
);

  // Localparam
  localparam int ErrCntW = 4;

  // Error logic
  logic signed [DW-1:0] err;
  logic signed [DW-1:0] err_prev;
  logic signed [DW-1:0] err_diff;
  assign err = setpoint_i - feedback_i;
  assign err_diff = err - err_prev;
  logic [ErrCntW-1:0] err_eq0_cnt;
  logic [ErrCntW-1:0] err_cnt_lock;
  assign err_cnt_lock = ErrCntW'(lock_cnt_i);
  always_ff @(posedge clk_i or negedge rst_ni) begin
    if (!rst_ni) begin
      err_eq0_cnt   <= '0;
    end else begin
      if ((err == '0) && (err_eq0_cnt < err_cnt_lock)) begin
        err_eq0_cnt <= err_eq0_cnt + ErrCntW'(1);
      end
    end
  end

  // PID calculation & lock
  logic signed [CW+DW-1:0] prop;
```

```
  logic signed [CW+DW  :0] integ;
  logic signed [CW+DW-1:0] deriv;
  logic signed [CW+DW+1:0] ctrl_pid;
  always_ff @(posedge clk_i or negedge rst_ni) begin
    if (!rst_ni) begin
      prop     <= '0;
      integ    <= '0;
      deriv    <= '0;
      err_prev <= '0;
      ctrl_pid <= '0;
      lock_o   <= '0;
    end else begin
      prop     <= (kp_i * err);
      integ    <= integ + (ki_i * err);
      deriv    <= (kd_i * err_diff);
      err_prev <= err;
      if ((err == '0) && (err_eq0_cnt == err_cnt_lock) && (
    ↪ err_cnt_lock != '0)) begin
        lock_o <= 1'b1;
      end
      ctrl_pid <= lock_o ? ctrl_pid : 18'(prop) + integ + 18'(
    ↪ deriv);
    end
  end
  assign ctrl_o = ctrl_pid[13:6];

endmodule
```

Излази `cnt_o` и `ctrl_o` из двостепеног и PID контролера, редом, представљају суштински број тростатичких инвертора DCO-а који требају бити укључени, што утиче на његову излазну фреквенцију. Међутим, та вриједност се не прослијеђује директно у модул DCO-а, већ се преко управљачког декодера анализираног у поглављу 3.4 претвара у скуп одговарајућих векторских улаза за управљање DCO-ом тј. редовима и колонама тростатичких инвертора унутар њега. Тај поступак претварања је приказан у Листингу 7, а имплементиран је у оквиру `fll` модула.

Листинг 7: Имплементација управљачког декодера из поглавља 3.4.

```
module fll #(
  ...
  // Convert digital code to real DCO inputs (row/col sel and row on)
  localparam int NumRows = 17;
  localparam int NumCols = 15;
  logic [NumRows-1:0] row_on_d;
  logic [NumRows-1:0] row_sel_d;
  logic [NumCols-1:0] col_sel_d;
  always_comb begin
      row_on_d  = '0;
      row_sel_d = '0;
      col_sel_d = '0;
      for(int r = 1; r <= NumRows; r = r + 1) begin
```

```
14          // Row on logic
15          if (int'(dco_ctrl) > r*NumCols) begin
16            row_on_d[r-1] = 1'b1;
17          end else begin
18            row_on_d[r-1] = 1'b0;
19          end
20          // Row & column select logic
21          if ((int'(dco_ctrl) > (r-1)*NumCols) && (int'(dco_ctrl)
    ↪ <= r*NumCols)) begin
22            for(int c = 0; c < NumCols; c = c + 1) begin
23              if ((r-1)*NumCols + c < int'(dco_ctrl)) begin
24                col_sel_d[c] = 1'b1;
25              end else begin
26                col_sel_d[c] = 1'b0;
27              end
28            end
29            row_sel_d[r-1] = 1'b1;
30          end else begin
31            row_sel_d[r-1] = 1'b0;
32          end
33        end
34    end
35    ...
36  endmodule
```

Сигнали `row_on_d`, `row_sel_d` и `col_sel_d` се синхронизују на одговарајући такт и прослијеђују се као улази у DCO модул који је имплементиран као посебан блок у алатима за аналогно пројектовање, па стога овдје нема свој опис у SystemVerilog-у, већ се тај блок само инстанцира унутар `fll` модула, као што је приказано у Листингу 8. Као што се може видјети, `dco` модул садржи излазе за свих 5 фаза на ниском (V_{DDL}) и високом (V_{DD}) напонском нивоу, а који од та два напонска нивоа ће се наћи на `clk_dco_o` излазу `fll` модула одређује његов улазни сигнал `dco_pwr_sel_i` (погледати Листинг 1).

Листинг 8: Инстанцирање DCO блока.

```
1  module fll #(
2    ...
3    // DCO model
4    logic clk_dco_ph0;
5    logic clk_dco_ph0_l;
6    logic clk_dco_ph0_h;
7    logic clk_dco_ph1;
8    logic clk_dco_ph1_l;
9    logic clk_dco_ph1_h;
10   logic clk_dco_ph2;
11   logic clk_dco_ph2_l;
12   logic clk_dco_ph2_h;
13   dco u_dco(
14     .ROW_ON    (row_on),
15     .ROW_SEL   (row_sel),
```

```
16      .COLUMN_SEL(col_sel),
17      .PH0        (clk_dco_ph0_h),
18      .PH0_L      (clk_dco_ph0_l),
19      .PH1        (clk_dco_ph1_h),
20      .PH1_L      (clk_dco_ph1_l),
21      .PH2        (clk_dco_ph2_h),
22      .PH2_L      (clk_dco_ph2_l),
23      .PH3        (),
24      .PH3_L      (),
25      .PH4        (),
26      .PH4_L      ()
27    );
28    assign clk_dco_ph0 = dco_pwr_sel_i ? clk_dco_ph0_h :
       ↪ clk_dco_ph0_l;
29    assign clk_dco_ph1 = dco_pwr_sel_i ? clk_dco_ph1_h :
       ↪ clk_dco_ph1_l;
30    assign clk_dco_ph2 = dco_pwr_sel_i ? clk_dco_ph2_h :
       ↪ clk_dco_ph2_l;
31    assign clk_dco_o = clk_dco_ph2;
32    ...
33  endmodule
```

6.2 Лејаут

За израду лејаута намјенски је коришћена 7T (енгл. *7-Track*) библиотака стандардних ћелија, гдје 7T означава да висина свих ћелија тачно толике висине да може да се вертикално провуче 7 линија метала минималне ширине и са минималним размаком. То доприноси лакшем аутоматском (а и ручном) рутирању и слагању ћелија у компактну цјелину у сврху оптималне искористивости простора. Лејаут коначне верзије цјелокупног FLL-а приказан је на Слици 24. Читав дизајн FLL-а заузима 33000 µm^2. Компонента у доњем десном углу је дигитално контролисани осцилатор, док остатак лејаута углавном припада управљачкој логици FLL-а.

Слика 24: Лејаут дигиталног FLL-а, са DCO-ом у доњем десном углу.

DCO је имплементиран као независна компонента коришћењем библиотека стандардних ћелија. Од читавог лејаута FLL-а, око 13 % заузима DCO, тачније око 4300 µm^2 (вриједност добијена сабирањем појединачних површина коришћених ћелија). Слика 25 приказује лејаут DCO-а са претварачима напонских нивоа који се могу уочити

са лијеве и горње стране лејаута (претварачи са високог на низак напонски ниво на управљачким улазима DCO-а), као и са десне стране (претварачи са ниског на висок напонски ниво на свих 5 фаза DCO-а).

Слика 25: Лејаут DCO-а са претварачима напонских нивоа.

Слика 26а приказује лејаут једне ћелије DCO-а. Са десне стране ћелије је оријентисан тростатички инвертор, док је са лијеве стране управљачки И-ИЛИ дио. Површина једне ћелије је око 13,7 μm^2.

Лејаут претварача са високог на низак и са ниског на висок напонски ниво је приказан на Слици 26б и 26в, респективно. Као што је већ напоменуто у поглављу 5.3, претварач са високог на низак напонски ниво је у суштини бафер и овдје је његова површина око 6,8 μm^2, док је претварач са ниског на висок напонски ниво нешто сложенији и, ради бољег уклапања тј. заузимања мање површине у комбинацији са DCO-ом, имплементиран је као ћелија са дуплом висином, што омогућава мању ширину ћелије, а површина јој је око 38,3 μm^2.

Слика 26: Лејаут (а) ћелије DCO-а, (б) претварача са високог на низак и (в) претварача са ниског на висок напонски ниво.

6.3 Симулација рада фреквенцијски затворене петље

Симулација рада FLL-а у оба управљачка режима приказана је на Слици 27. Са приказаних графика се може видјети да се у оба управљачка режима достиже стабилно закључавање FLL-а са излазном фреквенцијом веома блиској траженој фреквенцији, при чему грешка може бити у опсегу просјечне вриједности резолуције DCO-а (испод 3 MHz). Међутим, иако се у оба режима исправно достиже жељена фреквенција, између њих постоји разлика у брзини достизања исте. Тако, у PID режиму (P=15, I=15, D=0), закључавање се одвија много брже, и то за 1,83 µs, док је, поређења ради, за закључавање у режиму са двостепеним контролером потребно 13,13 µs.

6.4 PVT зависност дигитално контролисаног осцилатора

Да би се одредило понашање DCO-а у реалним условима, потребно је провјерити понашање DCO-а при промјени процесних углова, напона напајања и температуре, односно провјерити различите PVT случајеве. У оквиру овог рада симулирано је понашање у три различита PVT случаја:

1. Најспорији случај (slow-slow процесни угао, V_{DD}=1,1 V, V_{DDL}=1 V, 125 °C)
2. Типични случај (typical-typical процесни угао, V_{DD}=1,2 V, V_{DDL}=1,1 V, 40 °C)
3. Најбржи случај (fast-fast процесни угао, V_{DD}=1,3 V, V_{DDL}=1,2 V, -40 °C)

Слика 28 приказује зависност фреквенције DCO-а и фреквенцијског корака тј. резолуције фреквенције (K_{DCO}) од броја укључених тростатичких инвертора (тј. од управљачке ријечи DCO-а), за најспорији, типичан и најбржи PVT случај.

(а)

(б)

Слика 27: Симулација FLL-а у (а) PID и (б) двостепеном режиму.

Табела 5 приказује фреквенцијски опсег DCO-а и просјечну вриједност фреквенцијског корака (K_{DCO}) у три претходно обрађена случаја. Под опсегом се подразумијевају минималне и максималне вриједности фреквенције која се добија на излазу DCO-а за укључен мининалан (у овом случају 15) и максималан број (у овом случају 255) тростатичких инвертора DCO-а, респективно. Иако у најспоријем случају фреквенција не достиже жељених 640 MHz, она ипак може бити достигнута повећањем

(а)

(б)

Слика 28: Зависност (а) фреквенције осциловања и (б) фреквенцијског корака од броја укључених тростатичких инвертора за најспорији, типични и најбржи случај.

Табела 5: Фреквенцијски опсег DCO-а и просјечна вриједност фреквенцијског корака (K_{DCO}).

Случај	$f_{\min}$	$f_{\max}$	Просјечан K_{DCO}
Најспорији	27,2 MHz	502 MHz	1,8 MHz
Типичнни	42 MHz	764 MHz	2,8 MHz
Најбржи	63,3 MHz	1,146 GHz	4,2 MHz

засебног напона напајања осцилатора (V_{DDL}). Такође је могуће анализирати и зависност фреквенције од свих PVT параметара независно, међутим те зависности нису приказане у раду због временске захтјевности извршавања потребних симулација, као и због већ приказаних резултата симулација у најспоријем, типичном и најбр-

жем случају, који су ипак најбитнији за анализу понашања имплементираног система јер се ослањају на понашање система у граничним условима рада. Ипак, што се тиче понашања DCO-а у зависности од свих PVT параметара појединачно, потребно је напоменути да се фреквенција повећава са бржим процесним углом, већим напоном напајања, као и мањом температуром.

Да би се значајно смањило вријеме потребно за добијање резултата из симулација након екстракције лејаута, и то са минималним губитком прецизности добијених резултата, симулације DCO-а су извршене на нивоу појединачно екстракованих ћелија DCO-а и екстракованих претварача напонског нивоа, са додатним паразитним кондензатором на свакој фази DCO-а. Улога тих кондензатора је да надомјесте паразитне капацитивности које се јављају у потпуно екстракованом лејауту DCO-а због међусобно повезаних ћелија, а које нису аутоматски урачунате ако се симулације врше на нивоу појединачно екстракованих компоненти DCO-а, као што је то овде случај. Вриједност капацитивности додатог кондензатора прорачуната је на основу извјештаја генерисаног из екстракције паразитних капацитивности на нивоу читавог дизајна.

6.5 Анализа потрошње система

Потрошња снаге сваког дизајна утиче на велики број критичних одлука при самом пројектовању, као што су одређивање капацитивности извора напајања, трајања батерије, величине линија напајања, начина паковања, захтјева за хлађењем система итд. Значи да је потрошња снаге важна особина дизајна јер значајно утиче на изводљивост примјене крајњег производа, као и на трошкове и поузданост при коришћењу [18].

Зависно од проблема који се анализира, разматрају се различите мјере потрошње снаге. Рецимо, када се разматра величина линија напајања, тада је битна максимална вриједност снаге (енгл. *Peak Power*), P_{peak}, док при разматрању захтјева за хлађењем или складиштењем енергије тј. батеријом, од главног интереса је просјечна потрошња снаге (енгл. *Average Power Dissipation*) [18]. Те мјере су дефинисане следећим формулама:

$$P_{peak} = i_{peak} V_{supply} = \max[p(t)] \tag{47}$$

$$P_{avg} = \frac{1}{T}\int_0^T p(t)dt = \frac{V_{supply}}{T}\int_0^T i_{supply}(t)dt \tag{48}$$

гдје је i_{supply} струја која долази од извора са напоном V_{supply} у интервалу $t \in [0, T]$, док је i_{peak} максимална вриједност i_{supply} унутар тог интервала.

Од горе поменутих, у овом раду је кључна просјечна потрошња снаге. Како је DCO кључни елемент FLL-а, он је и убједљиво највећи потрошач у систему. Самим тим, његовим изолованим симулацијама се може умногоме предвидјети и потрошња коначног система, што значи да се пројектовање убрзава тиме што није потребно баш увијек радити симулације дизајна читавог FLL-а како би се приближно одредила потрошња.

Све симулације потрошње које слиједе су извршаване за фреквенцију осциловања од 640 MHz. Након симулације дизајна DCO-а (са претварачима напонских нивоа) у типичним условима рада, при чему су V_{DD}=1,2 V и V_{DDL}=1,1 V, добија се вриједност

просјечне потрошње од 2,5 mW од извора напајања V_{DDL} одакле се напајају осцилатор, претварачи са високог на низак и једним дијелом претварачи са ниског на висок напонски ниво и 0,22 mW од извора напајања V_{DD} одакле се, у овој симулацији, већим дијелом напајају претварачи са ниског на висок напонски ниво као и излазни бафери (погледати поглавље 5.3).

Потрошња се може утврдити и преко средње квадратне (енгл. *Root Mean Square*, RMS) вриједности струје која се добија при истим условима и овде износи 2,185 mA и 0,18 mA од извора напајања V_{DDL} и V_{DD}, респективно. С обзиром да је средња квадратна вриједност струје величина директно сразмјерна просјечној потрошњи снаге, при анализи потрошње углавном је довољно навести резултате у једној од тих величина. Формула за RMS вриједност потрошње струје неког извора гласи:

$$\mathrm{rms}(i_{supply}) = \sqrt{\frac{1}{T}\int_0^T i_{supply}^2(t)dt} \tag{49}$$

Резултати симулације наведени у претходним параграфу, од читавог FLL-а, обухватају DCO са претварачима напонских нивоа. Да би се утврдила и потрошња управљачке логике (која се напаја са извора V_{DD}), под истим условима је покренута и симулација потрошње читавог FLL-а, гдје се на изворима напајања V_{DDL} и V_{DD} добијају RMS вриједности потрошње струје 2,185 mA и 0,9 mA, респективно. Ако се од укупне RMS вриједности струје са V_{DD} извора напајања од 0,9 mA одузме изнад поменута вриједност коју троше само претварачи са ниског на висок напонски ниво и излазни бафери DCO-а (0,18 mA), добије се RMS вриједност потрошње струје управљачке логике FLL-а и износи 0,72 mA.

6.6 Временски одзив дигитално контролисаног осцилатора

Временски одзив (енгл. *Timing*) односи се на облик сигнала на фазама дигитално контролисаног осцилатора на нивоу једне периоде излазног такта. У сврху приказа временског одзива симулиран је рад екстракованог DCO-а у типичним условима.

Слика 29 приказује генерисани сигнал такта на свакој од пет фаза DCO-а. Приказани сигнали су добијени са улаза (V_{DDL} домен напона напајања) и излаза (V_{DD} домен напона напајања) претварача са ниског на висок напонски ниво. Као што је приказано у једначини 42, фреквенција осциловања f_{osc} зависи од t_{d} тј. кашњења кроз ћелију DCO-а која садржи тростатички инвертор. Та вриједност је практично једнака аритметичкој средини кашњења узлазне и силазне ивице сигнала такта (t_{lh} и t_{hl}). Други важан параметар за разматрање је фактор испуњености импулса такта тј. однос времена које излазни сигнал проведе на високом, односно ниском, напону у једној периоди такта и самог периода такта. Изражава се у процентима тј. колико процентуално сигнал такта проведе на високом, а колико на ниском напону тј. нули и идеално је да тај однос буде $50\% - 50\%$.

Добијене вриједности кашњења узлазне и силазне ивице такта на баферованом излазу претварача са ниског на висок напонски ниво су $t_{\mathrm{lh}} = 26\,\mathrm{ps}$ и $t_{\mathrm{hl}} = 20\,\mathrm{ps}$, респективно. Добијени однос унутар радног циклуса такта је апроксимативно $48\% - 52\%$. Временски помјерај између двије сусједне фазе је око 940 ps.

Слика 29: Генерисани такт на свих 5 фаза DCO-а у V_{DD} и V_{DDL} доменима напајања.

6.7 Спектар снаге фреквенцијски затворене петље

Нормализовани спектар снаге предложеног FLL-а у PID режиму, током процеса одржавања вриједности фреквенције од 640 MHz на излазу, приказан је на Слици 30.

Слика 30: Спектар снаге FLL-а, са фреквенцијом носиоца 640 MHz, најјачим нивоом непожељног сигнала (у средини лијево) на -23,6 dBc, и највећим непожељним сигналом референтне фреквенције унутар 640 ± 16 MHz (горе десно, зумирано) на нивоу -43,8 dBc.

Подаци са графика су добијени из дигиталне симулације на нивоу логичких кола, трајања 15 µm, која укључује идеалан бешумни модел DCO-а написан у SystemVerilog-у. Излаз симулације је потпуно детерминистички квадратни талас састављен од јединица и нула, одабран са учестаношћу 10 GHz, и сачуван за даљу обраду.

Највећа вриједност на графику представља фреквенцију осциловања FLL-а, односно фреквенцију носиоца (енгл. *Carrier Frequency*). Ниво највећег непожељног тона је $-23,6$ dBc, што је више од два реда величине испод нивоа фреквенције носиоца (0 dBc). У зумираној области, може се видјети непожељан сигнал унутар референтне фреквенције и то лијево и десно од носиоца са помјерајем (енгл. *Offset*) од 16 MHz, чији је најјачи ниво на $-43,8$ dBc.

6.8 Фазни шум дигитално контролисаног осцилатора

Због насумичних фазних одступања, спектар снаге реалног осцилатора се такође шири на фреквенције око фреквенције носиоца, f_{osc}. Извори насумичних фазних поремећаја [15] у виду шума треперења и топлотног шума се манифестују као $1/f^3$ and $1/f^2$ области, респективно, у добијеном профилу фазног шума на Слици 31. Приказани резултати су добијени у типичним PVT условима, са вриједношћу напона напајања DCO-а (V_{DDL}) од 1,1 V, и вриједности фреквенције носиоца (f_{osc}) од 640 MHz.

Слика 31: Спектар фазног шума DCO-а: област шума треперења (лијево), област топлотног шума (десно), и угаона фреквенција од ≈ 800 kHz.

7 Пајтон модел FLL-а

У сврху практичнијег симулирања и бољег предвиђања понашања FLL-а при различитим условима рада, направљен је основни Пајтон модел цјелокупног FLL-а. Предност посједовања таквог модела јесте што се одређене карактеристике конкретног дизајна могу параметризовати и затим је, у зависности од тога колико услова модел покрива, могуће веома брзо добити мање или више вјеродостојан резултат понашања тог дизајна, умјесто да се, с друге стране, мијења дизајн и додатним симулацијама утврђују карактеристике под различитим условима, па тек затим пређе на симулације крајњег понашања, што доста успорава процес пројектовања. Конкретна имплементација Пајтон модела FLL-а из овог рада обухвата, поред параметара самог FLL-а, и параметризовану почетну фреквенцију DCO-а и његову резолуцију, што омогућава тренутно добијање резултата симулације. Треба напоменути да направљени модел предвиђа константне услове рада тј. параметри модела се не мијењају током времена, односно кроз итерације у петљи модела, и да модел не покрива све услове рада. То значи да резултати приказани у наставку не морају да одговарају резултатима из RTL симулација или симулација на нивоу логичких кола, већ су ту да пројектанта наведу на исправну употребу одређених константи (као што су PID константе) или да му помогну око одабира компоненти дизајна чије појединачно понашање му је већ познато и може да се параметризује (рецимо понашање DCO-а).

7.1 Пајтон код

У наставку ће бити приказан изворни Пајтон код модела, са логиком и називима промјенљивих који углавном одговарају онима из SystemVerilog хардверске имплементације. Биће наведени само листинзи, без детаљног објашњења кода, јер је већина принципа рада већ анализирана при опису хардверске имплементације. Прегледности ради, код је подијељен у класе и то:

- `fll` - Класа FLL компоненте (Листинг 9).
- `pid` - Класа PID контролера (Листинг 10).
- `cnt` - Класа бројача из двостепеног контролера (Листинг 11).
- `dco` - Класа дигитално контролисаног осцилатора (Листинг 12).

С обзиром да `dco` класа нема хардверску имплементацију у виду SystemVerilog модула, треба напоменути да се логика имплементације софтверског модела ослања на једначину 45 што се може видјети у Листингу 12.

Листинг 9: Имплементација модела FLL компоненте.

```
from dco import dco
from cnt import cnt
from pid import pid

class fll:
    def __init__(self,
                    # Inputs from design
                    CW,                 # control word width
                    clk_ref_i,          # reference clock
```

```
                clk_ref_en_i,      # reference clock enable
                rst_ni,            # async reset
                clk_mult_i,        # reference frequency multiplier [40]
                dco_pwr_sel_i,     # DCO clock output power supply selection: VDDL/VDD [0/1]
                dco_ctrl_man_i,    # manual DCO control word [200]
                dco_ctrl_mode_i,   # DCO control mode: FIXED/MANUAL/CNT/PID [PID=3/CNT=2]
                pid_kp_i,          # PID controller proportional constant [0b01000000]
                pid_ki_i,          # PID controller integral constant [0b01000000]
                pid_kd_i,          # PID controller derivative constant [0]
                lock_cnt_i,        # FLL lock counter condition [0b1000]
                # Additional input parameters
                k_dco,             # DCO frequency resolution
                clk_dco_init):     # DCO initial frequency (dco_ctrl=0)
        """
        Frequency-locked loop (FLL) is a negative feedback control system that locks the output frequency signal to the target frequency. Principally, it continuously controls the oscillator frequency in an automated way until its output frequency matches the target value, after which that value is maintained at the output.
        """
        # Inputs from design
        self.CW = CW
        self.clk_ref_i = clk_ref_i
        self.clk_ref_en_i = clk_ref_en_i
        self.rst_ni = rst_ni
        self.clk_mult_i = clk_mult_i
        self.dco_pwr_sel_i = dco_pwr_sel_i
        self.dco_ctrl_man_i = dco_ctrl_man_i
        self.dco_ctrl_mode_i = dco_ctrl_mode_i
        self.pid_kp_i = pid_kp_i
        self.pid_ki_i = pid_ki_i
        self.pid_kd_i = pid_kd_i
        self.lock_cnt_i = lock_cnt_i
        # Additional input parameters
        self.k_dco = k_dco
        self.clk_dco_init = clk_dco_init
        # Component instantiation
        self.dco_inst = dco(k_dco=self.k_dco,
                            init_freq=self.clk_dco_init)
        self.cnt_inst = cnt()
        self.pid_inst = pid(rst_ni=self.rst_ni,
                            kp_i=self.pid_kp_i,
                            ki_i=self.pid_ki_i,
                            kd_i=self.pid_kd_i,
```

```
                                setpoint_i=self.clk_mult_i,
                                feedback_i=self.clk_dco_init//self.clk_ref_i,
                                lock_cnt_i=self.lock_cnt_i)
        # Outputs from design
        self.lock_o = self.clk_dco_init//self.clk_ref_i == self.clk_mult_i
        self.clk_dco_o = self.clk_dco_init

    def run(self,
            rst_ni):
        self.rst_ni = rst_ni

        # Convert control mode integer value to the corresponding binary array (1/0 integers)
        dco_ctrl_mode_bin_arr = list(map(int, list(bin(self.dco_ctrl_mode_i)[2:][::-1])))

        # PID logic
        self.pid_inst.feedback_i = self.clk_dco_o//self.clk_ref_i
        self.pid_inst.run(rst_ni=self.rst_ni)
        dco_ctrl_pid = self.pid_inst.ctrl_o

        # Bang-bang logic
        #cnt_en_i = 1  # used if you want frequency oscillation in the stable state
        cnt_en_i = not self.pid_inst.lock_o if (self.clk_mult_i != self.clk_dco_o//self.clk_ref_i) else 0
        cnt_up_down = 1 if (self.clk_mult_i > self.clk_dco_o//self.clk_ref_i) else 0
        self.cnt_inst.run(rst_ni=self.rst_ni,
                          en_i=cnt_en_i and dco_ctrl_mode_bin_arr[1],
                          up_down_i=cnt_up_down)
        dco_ctrl_cnt = self.cnt_inst.cnt_o

        # Control preprocessing
        dco_ctrl_man = self.dco_ctrl_man_i if (dco_ctrl_mode_bin_arr[0]) else 203
        dco_ctrl_auto = dco_ctrl_pid if (dco_ctrl_mode_bin_arr[0]) else dco_ctrl_cnt
        dco_ctrl = dco_ctrl_auto if (dco_ctrl_mode_bin_arr[1]) else dco_ctrl_man

        # DCO logic
        self.dco_inst.run(ctrl=dco_ctrl)

        # FLL outputs setup
        self.clk_dco_o = int(self.dco_inst.freq_out)
        self.lock_o = self.pid_inst.lock_o
```

Листинг 10: Имплементација класе PID контролера.

```
import copy

INT_WIDTH = 8                       # width of output control signal
FX_INT_WIDTH = 12                   # integer part width of PID sum (Q12.6)
FX_DEC_WIDTH = 6                    # decimal part width of PID sum (Q12.6)
CTRL_WIDTH = FX_INT_WIDTH + FX_DEC_WIDTH  # total width of PID sum

class pid:
    def __init__(self,
                 rst_ni,
                 kp_i,
                 ki_i,
                 kd_i,
                 setpoint_i,
                 feedback_i,
                 lock_cnt_i):
        self.rst_ni = rst_ni
        self.kp_i = kp_i
        self.ki_i = ki_i
        self.kd_i = kd_i
        self.setpoint_i = setpoint_i
        self.feedback_i = feedback_i
        self.lock_cnt_i = lock_cnt_i
        self.ctrl_o = 0
        self.lock_o = 0
        self.err_eq0_cnt = 0
        self.integ = 0
        self.err = setpoint_i - feedback_i
        self.err_prev = 0
        self.ctrl_pid = 0

    def run(self,
            rst_ni):
        self.rst_ni = rst_ni

        if (not self.rst_ni):
            self.err_eq0_cnt = 0
        else:
            if ((self.err == 0) and (self.err_eq0_cnt < self.
    ↪ lock_cnt_i)):
                self.err_eq0_cnt += 1

        prop = 0
        deriv = 0
        self.err = self.setpoint_i - self.feedback_i
        err_diff = self.err - self.err_prev

        if (not self.rst_ni):
            prop = 0
            self.integ = 0
```

```
50            deriv = 0
51            self.err_prev = 0
52            self.ctrl_pid = 0
53            self.lock_o = 0
54        else:
55            prop = self.kp_i * self.err
56            self.integ += self.ki_i * self.err
57            deriv = self.kd_i * err_diff
58            self.err_prev = copy.deepcopy(self.err)
59            if ((self.err == 0) and (self.err_eq0_cnt == self.
    ↪ lock_cnt_i) and (self.lock_cnt_i != 0)):
60                self.lock_o = 1
61            if (not self.lock_o):
62                self.ctrl_pid = prop + self.integ + deriv
63
64        self.ctrl_o = int(format(int(self.ctrl_pid), '0'+str(
    ↪ CTRL_WIDTH)+'b')[FX_INT_WIDTH-INT_WIDTH:FX_INT_WIDTH], 2)
```

Листинг 11: Имплементација класе бројача двостепеног контролера.

```
1  class cnt:
2      def __init__(self):
3          self.cnt_o = 0
4          self.default_val = 0
5
6      def run(self,
7              rst_ni,
8              en_i,
9              up_down_i):
10         if (not rst_ni):
11             self.cnt_o = self.default_val
12         else:
13             if (en_i):
14                 if (up_down_i):
15                     self.cnt_o += 1
16                 else:
17                     self.cnt_o -= 1
18         return self.cnt_o
```

Листинг 12: Имплементација класе дигитално контролисаног осцилатора.

```
1  class dco:
2      def __init__(self,
3                   k_dco,
4                   init_freq):
5          self.k_dco = k_dco
6          self.init_freq = init_freq
7          self.freq_out = init_freq
8
9      def run(self,
10             ctrl):
11         self.freq_out = self.init_freq + self.k_dco * ctrl
```

7.2 Поређење рада различитих управљачких режима

Слика 32 приказује одзив софтверског модела FLL-а при раду у PID и двостепеном управљачком режиму рада. У конкретној симулацији коришћено је понашање DCO-а добијено симулацијом у типичним условима рада, па су при инстанцирању софтверског модела FLL-а коришћени подаци из Табеле 5, односно K_{DCO}=2,8 MHz и $f_{\min}$=42 MHz за почетну фреквенцију. Конкретна вриједност фреквенције која се овом симулацијом достиже у оба режима је 641,2 MHz. У PID режиму модел до те фреквенције стиже након 28 итерација, док му је са двостепеним управљачким режимом потребно 214 итерација.

Слика 32: Одзив модела у PID и двостепеном режиму рада.

У поглављу 3.2 поменута је појава осциловања фреквенције у стабилном стању када FLL ради у двостепеном режиму. На Слици 32 те осцилације не постоје, а проблем је ријешен условним довођењем `pid.lock_o` сигнала на `cnt.en_i` улаз бројача двостепеног контролера. Логика имплементације се налази на линији `75` Листинга 9. Да би се ипак показало како те осцилације изгледају, довољно је само закоментарисати поменуту линију, а одкоментарисати линију `74`. Тиме се добија одзив као на Слици 33, гдје фреквенција у стабилном стању, при коришћењу двостепеног контролера, варира између 638,4 MHz и 641,2 MHz (примјећујемо да разлика између поменутих вриједности представља вриједност резолуције, K_{DCO}=2,8 MHz).

7.3 Утицај константи PID контролера на одзив система

У овом поглављу биће софтверски симулиран утицај различитих вриједности константи PID контролера на одзив система. Треба напоменути да одзив не мора у потпуности одговарати стварном одзиву система већ му је циљ помоћи при исправном одабиру константи PID контролера, а све у циљу добијања најбржег и најправилнијег одзива система у крајњој имплементацији. Као што је објашњено у поглављу 3.3,

Слика 33: Одзив модела у PID и двостепеном режиму рада.

диференцијална константа није коришћена у имплементацији, па стога неће бити приказан утицај њених различитих вриједности на одзив система. Слике 34 и 35 редом приказују утицај различитих вриједности пропорционалне (K_p) и интегралне (K_i) константе на брзину и начин достизања жељене фреквенције.

Слика 34: Одзив модела за различите вриједности пропорционалне константе PID контролера.

Уочава се да пропорционална константа утиче на осјетљивост и стабилност система, а у општем случају систем може постати нестабилан ако је њена вриједност превелика. С друге стране, повећањем интегралне константе убрзава се кретање излаза ка жељеној вриједности. Треба напоменути да је децимални опсег вриједности константи од 0 до 63,75 јер представљају 8-битне вриједности у формату Q6.2, што је поменуто и у поглављу 6.1.

Слика 35: Одзив модела за различите вриједности интегралне константе PID контролера.

7.4 Утицај резолуције DCO-а на одзив система

Оно што такође знатно може убрзати, а самим тим и олакшати, пројектовање јесте предвиђање одзива система за различите карактеристике DCO-а. Тако ће у наставку бити приказан одзив за различите вриједности резолуције и то само у двостепеном режиму рада због поједностављења приказа. Слика 36 приказује одзив за три различите K_{DCO} вриједности при почетној фреквенцији од 42 MHz. У приказу је намјерно подешена верзија модела са осцилацијама у стабилном стању како би се што боље дочарао утицај резолуције на прецизност система. Такође се са графика може примјетити да повећањем резолуције тј. прецизности опада брзина одзива система.

Слика 36: Моделован одзив система за различите вриједности резолуције, K_{DCO}.

8 Поређење DCO-а пројектованих у 130 nm и 180 nm технологији

С обзиром да скалирање технологије у којој се врши фабрикација значајно утиче на перформансе система, ово поглавље ће бити посвећено поређењу особина два дигитално контролисана осцилатора, једног пројектованог у TSMC 130 nm технологији (у којој је пројектован читав FLL из овог рада), а другог у TSMC 180 nm технологији.

Како скалирање уређаја значајно смањује минималну величину компоненти CMOS уређаја, тако се и други CMOS параметри, сходно томе, повећавају или смањују, а све у циљу постизања бољих перформанси. Главни параметри који утичу на перформансе CMOS-а су: дебљина оксида t_{ox}, укупна капацитивност гејта C_{g} и укупна отпорност гејта R_{g}, а утицај скалирања на њихове вриједности и даље на перформансе је детаљно описан у литератури [19].

Уопштено, ознаке 180 nm и 130 nm односе се на дужину полисилицијумског гејта MOSFET-а, што значи да се скалирањем технологије дужина гејта смањује и тиме транзистори постају мањи. То омогућава већу густину транзистора на истом чипу и тиме долази до смањења укупне површине интегрисаног кола. Мањи транзистори омогућавају интеграцију више функција на истом чипу, чиме се смањује укупна површина потребна за реализацију исте функционалности. Слика 37 приказује конкретно утицај скалирања технологије на површину потребну за реализацију петостепеног DCO-а са укупно 255 ћелија. Мјере са слике се односе на дужине и ширине без претварача напонских нивоа (који се налазе са лијеве, десне и горње стране у оба лејаута), како би се добило мјеродавније поређење. Површина која је била потребна за реализацију лејаута DCO-а у 130 nm и 180 nm технологији износи око 3700 µm^2 и 6000 µm^2, респективно. То значи да је скоро 40 % мања површина потребна за реализацију у 130 nm.

Слика 37: Лејаут DCO-а пројектованог у TSMC (а) 130 nm и (б) 180 nm технологији.

Скалирање транзистора обично долази са смањењем напона напајања, јер мањи транзистори могу радити на нижим напонима. То смањење напона смањује динамичку

потрошњу, која је пропорционална квадрату напона напајања:

$$P_{dyn} = C_L V_{supply}^2 f. \tag{50}$$

Такође, смањење капацитивног оптерећења смањује динамичку потрошњу. У нашем случају извршене су симулације при раду на фреквенцији 640 MHz у типичним условима рада са вриједностима V_{DD}=1,2 V и V_{DDL}=1,1 V за DCO у 130 nm и V_{DD}=1,8 V и V_{DDL}=1,6 V за DCO у 180 nm. Средње квадратне (RMS) вриједности потрошње струје и просјечна потрошња снаге на тим линијама напајања, за оба осцилатора, дати су у Табели 6. Из резултата се види да је просјечна потрошња снаге мања за чак око 50 % (долази до изражаја квадратна зависност од напона напајања), а RMS вриједности струје за око 30 % за DCO пројектован у 130 nm.

Табела 6: Поређење RMS вриједности потрошње струје и просјечне снаге DCO-а пројектованих у 130 nm и 180 nm.

	130 nm	180 nm
rms(i_{DDL})	2,185 mA	3,105 mA
rms(i_{DD})	0,18 mA	0,27 mA
Укупно rms(i)	2,365 mA	3,375 mA
$P_{avg}(V_{\mathrm{DDL}})$	2,5 mW	5,0 mW
$P_{avg}(V_{\mathrm{DD}})$	0,22 mW	0,5 mW
Укупно P_{avg}	2,72 mW	5,5 mW

Додатна предност скалирања је што мањи транзистори могу радити на вишим фреквенцијама, што повећава укупне перформансе чипа. Разлог је то што скалирање транзистора доводи до смањења кашњења при пребацивању са 0 на 1 и обрнуто. Кашњење је сразмјерно отпорности и капацитивности гејта, а обје поменуте величине су директно пропорционалне димензијама гејта које се смањују скалирањем технологије. То значи да ће осцилатор у напреднијој технологији (у овом случају 130 nm) бити у могућности да синтетише сигнал веће фреквенције, за исти напон напајања. Табела 8 приказује резултате симулација при различитим условима рада и вриједностима напона напајања. Подешавање параметара који одређују услове рада приказани су у Табели 7.

Табела 7: Подешавање услова рада за симулације опсега фреквенције.

Случај	Технологија	Процесни угао	Температура	V_{DD}	V_{DDL}
Најспорији	130 nm	slow-slow	125° C	1,1 V	1,0 V
Типични	130 nm	typical-typical	40° C	1,2 V	1,1 V
Најбржи	130 nm	fast-fast	-40° C	1,3 V	1,2 V
Најспорији	180 nm	slow-slow	125° C	1,6 V	1,4 V
Типични	180 nm	typical-typical	40° C	1,8 V	1,6 V
Најбржи	180 nm	fast-fast	-40° C	2,0 V	1,8 V

Из резултата се види да се скалирањем технологије значајно може смањити напон напајања да би се добио жељени опсег фреквенције, што наравно значи да се уз много мању потрошњу постижу сличне, или чак боље, перформансе у раду DCO-а.

Табела 8: Поређење опсега фреквенција при различитим условима рада за DCO пројектован у 130 nm и 180 nm.

Случај	Технологија	$f_{\min}$	$f_{\max}$
Најспорији	130 nm	27,2 MHz	502 MHz
Типични	130 nm	42 MHz	764 MHz
Најбржи	130 nm	63,3 MHz	1,146 GHz
Најспорији	180 nm	23,06 MHz	421,4 MHz
Типични	180 nm	40 MHz	726 MHz
Најбржи	180 nm	64,43 MHz	1,168 GHz

8.1 Начин и утицај коришћења дубоке јаме N типа у аналогном пројектовању DCO-а

Дитално контролисани осцилатор пројектован у 180 nm TSMC технологији, поред стандардних јама (енгл. *Well*) P и N типа, посједује додатну тзв. дубоку јаму N типа (енгл. *Deep N-Well*, DNW). Да би се показало чему она служи, потребно је анализирати појаве које се дешавају у слојевима у које су сами транзистори урођени. Приликом рада транзистора појављује се шум у супстрату који се дешава због убризгавања наелектрисања у супстрат и јаме. Тај проблем се рјешава употребом n+ и p+ спојева (енгл. *Tap*) повезаних редом на V_{DD} и V_{SS}, за PMOS и NMOS транзисторе, респективно (Слика 38). Додатни проблем је што постоји капацитивно спрезање између N јаме и P супстрата, што значи да више шума долази до извора. У дигиталним колима то обично није проблем због прилично високе отпорности логичких кола на шум. Међутим, у аналогном пројектовању, шум може бити озбиљан проблем па се користе различите технике за његово сузбијање као што су додавање заштитних прстенова (енгл. *Guard Ring*) или коришћење одвојеног напајања за спојеве у супстрату и јамама. Треба поменути да заштитни прстенови не могу спријечити појаву шума дубоко у супстрату [20]. Други проблем је што NMOS транзистори обично нису изоловани од супстрата. У основној CMOS изради, NMOS транзистор је урођен у P супстрат повезан на масу, док је PMOS урођен у N јаму (енгл. *N-Well*) повезану на напајање, као што је приказано на Слици 38.

Слика 38: Попречни пресјек PMOS и NMOS транзистора.

Изолација NMOS транзистора се може постићи коришћењем поменуте дубоке јаме N типа. Она се формира уметањем високоенергетске јонске имплантације непосредно прије формирања стандардне јаме N типа [21]. Повезивање дубоке јаме N типа

врши се преко стандардне јаме N типа која је окружује и повезана је на напајање, V_{DD}. Тако се, стварањем изоловане јаме P типа (енгл. *P-Well*), изолује NMOS транзистор. Како додатном изолацијом глобални P супстрат остаје без p+ споја који је неопходан да се појавом шума у супстрату не наруше перформансе уређаја, додаје се заштитни прстен који се повезује на V_{SS}. Слика 39 приказује начин изолације NMOS транзистора преко дубоке јаме N типа уз додати заштитни прстен.

Слика 39: Попречни пресјек PMOS и NMOS транзистора, гдје је NMOS транзистор изолован дубоком јамом N типа (енгл. *Deep N-Well*) са додатим заштитним прстеном повезаним на V_{SS}.

Како NMOS и PMOS транзистори обично иду у пару приликом CMOS пројектовања, да би се боље искористио простор тј. смањила површина крајњег лејаута, за повезивање дубоке јаме N типа се може искористити стандардна јама N типа коју користи PMOS транзистор, а заштитни прстен се помјера да обухвати читав дизајн, што је приказано на Слици 40. У конкретном дизајну DCO-а, са Слике 37б се може уочити заштитни прстен који обухвата читав DCO.

Слика 40: Попречни пресјек просторно ефикаснијег лејаута PMOS и изолованог NMOS транзистора.

Дакле, дубока јама N типа има улогу у смањењу шума у супстрату и омогућава потпуно изоловање NMOS транзистора, који, теоретски, након тога могу бити на потенцијалу различитом од масе [20]. Како додавање дубоке јаме N типа изискује и додавање заштитног прстена, површина коју заузима дизајн се повећава, тако да то може бити недостатак овог приступа ако су захтјеви за површином строги. Углавном то није случај јер су захтјеви за перформансама обично битнији тако да је цијена која се плаћа у површини занемарљива у односу на предности које има изолација NMOS транзистора у погледу смањења шума у супстрату.

9 Закључак

У овом раду представљена је синтетизабилна дигитална фреквенцијски затворена петља са једноставном али ефикасном архитектуром. Дигитално контролисан осцилатор са прстенастом структуром, састављен од тростатичких инвертора омогућава достизање пристојне резолуције фреквенције и одговарајућих радних перформанси. Додатне предности су му и једноставно управљање, прилагодљивост и једноставна поновна употреба. Представљена су два управљачка режима FLL-а: двостепени контролер и PID контролер. За сваки од њих је детаљно објашњен начин рада и вјеродостојно су приказане разлике у одзиву система у различитим управљачким режимима. Детаљно је објашњена размјена података између различитих домена такта, као и методе заштите од метастабилности.

FLL из овог рада је имплементиран у SystemVerilog језику за опис хардвера. У зависности од технолошке библиотеке може бити направљен коришћењем само библиотеке стандардних ћелија без преуређивања у току дигиталног пројектовања. Обрађени FLL је послат на фабрикацију у 130 nm CMOS технологији.

Ради додатне верификације направљен је и демонстриран софтверски модел рада читавог FLL-а написан у Пајтон програмском језику. Поред саме демонстрације рада, захваљујући параметрима које прима, помоћу модела се на бржи и једноставнији начин могу предвидјети исправне вриједности константи које се задају систему (нпр. константе PID контролера), што значајно убрзава пројектовање.

У сврху приказа утицаја скалирања технологије на рад система тј. дигитално контролисаног осцилатора као његовог главног дијела, направљено је поређење DCO-а пројектованог у 130 nm технологији и DCO-а са идентичном архитектуром, али у 180 nm технологији. Упоређени су главни параметри као што су површина, потрошња, као и опсег и резолуција фреквенције при различитим условима рада.

Даљи рад на описаном и имплементираном FLL-у ће укључити тестирање чипа након производње и паковања, као и пројектовање додатних синтетизабилних дигиталних блокова као што је фазно спрегнута петља (енгл. *Phase-Locked Loop*, PLL), временско-дигитални претварач (енгл. *Time-to-Digital Converter*, TDC), регулатор с ниским падом напона (енгл. *Low-Dropout Regulator*, LDO) итд.

Литература

[1] Imran Ali, Hamed Abbasizadeh, Muhammad Riaz Ur Rehman, Muhammad Asif, Seong Jin Oh, Young Gun Pu, Minjae Lee, Keum Cheol Hwang, Youngoo Yang и Kang-Yoon Lee, „An Ultra-Low Power, Adaptive All-Digital Frequency-Locked Loop With Gain Estimation and Constant Current DCO", у: *IEEE Access* 8 (2020.), стр. 97215–97230, DOI: 10.1109/ACCESS.2020.2995853.

[2] Ahmed Musa, Wei Deng, Teerachot Siriburanon, Masaya Miyahara, Kenichi Okada и Akira Matsuzawa, „A Compact, Low-Power and Low-Jitter Dual-Loop Injection Locked PLL Using All-Digital PVT Calibration", у: *IEEE Journal of Solid-State Circuits* 49.1 (2014.), стр. 50–60, DOI: 10.1109/JSSC.2013.2284651.

[3] Wei Deng, Dongsheng Yang, Tomohiro Ueno, Teerachot Siriburanon, Satoshi Kondo, Kenichi Okada и Akira Matsuzawa, „A Fully Synthesizable All-Digital PLL With Interpolative Phase Coupled Oscillator, Current-Output DAC, and Fine-Resolution Digital Varactor Using Gated Edge Injection Technique", у: *IEEE Journal of Solid-State Circuits* 50.1 (2015.), стр. 68–80, DOI: 10.1109/JSSC.2014.2348311.

[4] JosÉ A. Tierno, Alexander V. Rylyakov и Daniel J. Friedman, „A Wide Power Supply Range, Wide Tuning Range, All Static CMOS All Digital PLL in 65 nm SOI", у: *IEEE Journal of Solid-State Circuits* 43.1 (2008.), стр. 42–51, DOI: 10.1109/JSSC.2007.910966.

[5] A. V. Rylyakov, J. A. Tierno, D. Z. Turker, J.-O. Plouchart, H. A. Ainspan и D. Friedman, „A Modular All-Digital PLL Architecture Enabling Both 1-to-2GHz and 24-to-32GHz Operation in 65nm CMOS", у: *2008 IEEE International Solid-State Circuits Conference - Digest of Technical Papers* (2008.), стр. 516–632, DOI: 10.1109/ISSCC.2008.4523284.

[6] Robert Bogdan Staszewski и Poras T. Balsara, *All-Digital Frequency Synthesizer in Deep-Submicron CMOS*, John Wiley & Sons, Inc., Hoboken, New Jersey, 2005.

[7] Karl Johan Astrom и Tore Hagglund, *PID Controllers: Theory, Design, and Tuning*, 2. издање, Instrument Society of America, 1995.

[8] Bela G. Liptak, *Instrument Engineers' Handbook: Process Control and Optimization*, 4. издање, св. 2, CRC Press, Taylor & Francis Group, 2006.

[9] David Money Harris и Sarah L. Harris, *Digital Design and Computer Architecture*, 2. издање, Morgan Kaufmann, 2012.

[10] Martin Davis, Ron Sigal и Elaine J. Weyuker, *Computability, Complexity, and Languages: Fundamentals of Theoretical Computer Science (Computer Science and Scientific Computing)*, 2. издање, Academic Press, Inc., 1994., стр. 117.

[11] Behzad Razavi, *Design of CMOS Phase-Locked Loops: From Circuit Level to Architecture Level*, Cambridge University Press, 2020.

[12] Anant Agarwal и Jeffrey H. Lang, *Foundations of Analog and Digital Electronic Circuits*, ISSN, Elsevier Science, 2005., ISBN: 9780080506814.

[13] Reddy B. Madhusudhana, P.P. Jose, P.L. Babu и V. Logeshwaran, „CMOS Based Digital Controlled Oscillators (DCO) – A Review", у: *International Journal of Applied Engineering Research* 10.20 (2015.), стр. 18626–18630.

[14] M. Terosiet, E. Zianbetov, F. Vallette, M. M. Louerat, P. Garda, D. Galayko и S. Feruglio, „A Comprehensive In-Depth Study of Tri-State Inverter Based DCO”, у: *Microelectronics Journal* (2020.), DOI: 10.1016/j.mejo.2020.104760.

[15] V. Milovanović и B. Nikolić, „An HDL model of a digitally controlled oscillator for rapid digital PLL prototyping”, у: *2017 IEEE 30th International Conference on Microelectronics (MIEL)*, 2017., стр. 205–208, DOI: 10.1109/MIEL.2017.8190103.

[16] Yuji Osaki, Tetsuya Hirose, Nobutaka Kuroki и Masahiro Numa, „A Low-Power level-shifter With Logic Error Correction for Extremely Low-Voltage Digital CMOS LSIs”, у: *IEEE Journal of Solid-State Circuits* 47.7 (2012.), стр. 1776–1783, DOI: 10.1109/JSSC.2012.2191320.

[17] „IEEE Standard for SystemVerilog–Unified Hardware Design, Specification, and Verification Language”, у: *IEEE Std 1800-2023 (Revision of IEEE Std 1800-2017)* (2024.), стр. 1–1354, DOI: 10.1109/IEEESTD.2024.10458102.

[18] Jan M. Rabaey, Anantha P. Chandrakasan и Borivoje Nikolić, *Digital Integrated Circuits: A Design Perspective*, 2. издање, Prentice Hall electronics and VLSI series, Pearson Education, 2003., ISBN: 9780131207646.

[19] Hassan Hassan, Mohab Anis и Mohamed Elmasry, „Impact of technology scaling and process variations on RF CMOS devices”, у: *Microelectronics Journal* 37.4 (2006.), стр. 275–282, ISSN: 1879-2391, DOI: 10.1016/j.mejo.2005.07.013.

[20] *Using Deep N Wells in Analog Design*, AspenCore, Electronics Industry Media & SaaS Group, https://www.planetanalog.com/using-deep-n-wells-in-analog-design/ (посећено 10. 9. 2024.).

[21] K. Wai Chew, J. Zhang, K. Shao, W. Boon Loh и S.-F. Chu, „Impact of Deep N-well Implantation on Substrate Noise Coupling and RF Transistor Performance for Systems-on-a-Chip Integration”, у: *32nd European Solid-State Device Research Conference*, 2002., стр. 251–254, DOI: 10.1109/ESSDERC.2002.194917.

Објављени радови

[1] Dejan D. Petković, Đorđe S. Gačić, Marijana R. Gavrilović Božović и Vladimir M. Milovanović, „A Synthesizable Digital Frequency-Locked Loop Widely Tunable up to 640 MHz in 130 nm CMOS”, у: *2024 11th International Conference on Electrical, Electronics and Computer Engineering (IcETRAN), Niš, Serbia*, јун 2024., стр. 1–6, DOI: 10.1109/IcETRAN62308.2024.10645145, https://ieeexplore.ieee.org/document/10645145 (посећено 10. 9. 2024.).