

BRNO UNIVERSITY OF TECHNOLOGY

Faculty of Electrical Engineering
and Communication

MASTER'S THESIS

Brno, 2021

Bc. JAKUB FRÁNEK



BRNO UNIVERSITY OF TECHNOLOGY

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

DEPARTMENT OF MICROELECTRONICS

ÚSTAV MIKROELEKTRONIKY

INJECTION LOCKED RING OSCILLATOR DESIGN FOR APPLICATION IN DIRECT TIME OF FLIGHT LIDAR

NÁVRH INJEKČÍ ZAVĚŠENÉHO KRUHOVÉHO OSCILÁTORU PRO APLIKACI V SYSTÉMECH
LIDAR PŘÍMO MĚŘÍCÍCH ČAS PRŮLETU

MASTER'S THESIS

DIPLOMOVÁ PRÁCE

AUTHOR

AUTOR PRÁCE

Bc. Jakub Fránek

ADVISOR

VEDOUCÍ PRÁCE

Ing. Vilém Kledrowetz, Ph.D.

BRNO 2021

Master's Thesis

Master's study program **Microelectronics**

Department of Microelectronics

Student: Bc. Jakub Fránek

ID: 186059

**Year of
study:** 2

Academic year: 2020/21

TITLE OF THESIS:

Injection locked ring oscillator design for application in Direct Time of Flight LIDAR

INSTRUCTION:

Provide an introduction to Direct Time of Flight (DToF) LIDAR technology, describe the operation of the single-photon avalanche diode, identify the main challenges of implementing DToF systems with sub-centimeter resolution. Compare different types of time to digital converters (TDC) in CMOS technologies. Analyze the operation of injection locked oscillators and describe the advantages of their application in DToF measuring TDCs. Create a MATLAB model of an injection locked ring oscillator, which takes the main non-idealities of their practical implementation in CMOS technologies into account. In ONK65 processing technology, design an injection locked ring oscillator capable of achieving 50 ps temporal resolution along with its biasing circuits, including the Delay Locked Loop (DLL), and evaluate its performance with computer simulations.

RECOMMENDED LITERATURE:

According to recommendations of supervisor

**Date of project
specification:** 8.2.2021

Deadline for submission: 25.5.2021

Supervisor: Ing. Vilém Kledrowetz, Ph.D.

Consultant: Ing. Ivan Koudar, Ph.D., ON Design Czech s.r.o.

doc. Ing. Lukáš Fucik, Ph.D.
Chair of study program board

WARNING:

The author of the Master's Thesis claims that by creating this thesis he/she did not infringe the rights of third persons and the personal and/or property rights of third persons were not subjected to derogatory treatment. The author is fully aware of the legal consequences of an infringement of provisions as per Section 11 and following of Act No 121/2000 Coll. on copyright and rights related to copyright and on amendments to some other laws (the Copyright Act) in the wording of subsequent directives including the possible criminal consequences as resulting from provisions of Part 2, Chapter VI, Article 4 of Criminal Code 40/2009 Coll.



Diplomová práce

magisterský navazující studijní program **Mikroelektronika**

Ústav mikroelektroniky

Student: Bc. Jakub Fránek

ID: 186059

Ročník: 2

Akademický rok: 2020/21

NÁZEV TÉMATU:

Návrh injekcí zavěšeného kruhového oscilátoru pro aplikaci v systémech LIDAR přímo měřících čas průletu

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s technologií LIDAR využívající přímého měření času průletu fotonu, popište operaci jednofotonové lavinové diody, identifikujte hlavní výzvy systémů LIDAR s lepším než centimetrovým rozlišením. Srovnajte způsoby realizace časově digitálních převodníků v technologiích CMOS. Popište funkci injekcí zavěšených oscilátorů a výhody jejich použití pro implementaci časově digitálních převodníků pro přímé měření času průletu. Ve výpočetním programu MATLAB vytvořte model injekcí zavěšeného kruhového oscilátoru, který zohledňuje hlavní neidealitu jejich praktické realizace v technologiích CMOS. Ve výrobní technologii ONK65 navrhnete injekcí zavěšený kruhový oscilátor poskytující časové rozlišení 50 ps společně s jeho biasovacími obvody, včetně závěsu zpoždění (DLL). Funkci navržených obvodů vyhodnoťte počítačovými simulacemi.

DOPORUČENÁ LITERATURA:

Podle pokynů vedoucího práce

Termín zadání: 8.2.2021

Termín odevzdání: 25.5.2021

Vedoucí práce: Ing. Vilém Kledrowetz, Ph.D.

Konzultant: Ing. Ivan Koudar, Ph.D., ON Design Czech s.r.o.

doc. Ing. Lukáš Fucík, Ph.D.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRACT

The diploma thesis provides an introduction to Direct Time of Flight LIDAR systems and Time to Digital Converters used in these systems. It discusses the problem of clock distribution in LIDAR Time to Digital Converter arrays, and examines one of the possible solutions to this problem based on injection locked oscillators. The injection locking phenomenon is thoroughly mathematically described and a Matlab model of an injection locked ring oscillator is presented, confirming the analytic predictions. In ONK65 processing technology, an injection locked ring oscillator biased by a delay locked loop meant specifically for application in Time to Digital Converters for LIDAR systems has been designed. The designed oscillator has been verified by computer simulations taking process, voltage and temperature variations into account and offers specified time resolution of 50 picosecond as well as two times less clock jitter than an equivalent free-running oscillator in the given processing technology.

KEYWORDS

LIDAR, time of flight, time to digital converter, injection locked oscillator, CMOS

ABSTRAKT

Diplomová práce přibližuje systémy LIDAR přímo měřící čas průletu a časově digitální převodníky určené k použití v těchto systémech. Představuje problematiku distribuce hodinových signálů napříč soubory časově digitálních převodníků v LIDAR systémech a věnuje se jednomu z nových řešení této problematiky, které je založené na injekci zavěšených oscilátorech. Technika injekčního zavěšení oscilátorů je důkladně matematicky popsána. V programu Matlab byl vytvořen simulační model injekci zavěšeného kruhového oscilátoru, který potvrzuje správnost uvedených analytických predikcí. Ve výrobní technologii ONK65 byl navržen injekci zavěšený kruhový oscilátor stabilizovaný pomocí smyčky závěsu zpoždění, určený pro implementaci časově digitálního převodníku pro systém LIDAR. Navržený injekci zavěšený kruhový oscilátor byl verifikován počítačovými simulacemi zohledňujícími vliv procesních, napěťových i teplotních variací. Oscilátor poskytuje specifikované časové rozlišení 50 pikosekund a dosahuje dvakrát nižší hodnoty fázového neklidu než ekvivalentní volnoběžný oscilátor v dané technologii.

KLÍČOVÁ SLOVA

LIDAR, čas průletu, časově digitální převodník, injekci zavěšený oscilátor, CMOS

ROZŠÍŘENÝ ABSTRAKT

Vynález jednofotonové lavinové diody a možnost její integrace do řady standardních CMOS výrobních procesů umožnily rozvoj nových snímacích a zobrazovacích technologií. Citlivost tohoto fotodetektoru na jednotlivé fotony a schopnost detekovat čas dopadu s pikosekundovým rozlišením využívají pozitronové emisní tomografy, různé druhy spektroskopie a mikroskopie nebo právě systémy LIDAR s přímým měřením doby průletu.

Základním principem těchto systémů je měření časového intervalu od okamžiku, kdy je vyslán laserový paprsek, do okamžiku, kdy je opětovně detekován detektorem, který se skládá právě z jednofotonových lavinových diod. Jelikož je rychlost světla v prostředí známá konstanta, lze s pomocí změřeného časového intervalu vypočítat vzdálenost k překážce, od které se paprsek odrazil. Systémy LIDAR proto fungují jako zobrazovací snímače, které místo vlnové délky nebo intenzity dopadajícího elektromagnetického záření snímají čas průletu, a tedy vzdálenost k objektům v jejich zorném poli. Ve srovnání s ostatními metodami měření vzdáleností v 3D prostoru, jako jsou například ultrazvukové senzory, stereoskopické kamerové systémy nebo mikrovlnné radary, mohou systémy LIDAR nabídnout lepší rozlišení, větší dynamický rozsah, nižší náročnost na výpočetní výkon a schopnost pracovat v prostředích s denním světlem i ve tmě.

Systémy LIDAR často nachází uplatnění v automobilovém průmyslu, kde tvoří jednu z klíčových komponent moderních pokročilých asistenčních systémů, protože umožňují přesné měření vzdáleností k překážkám v prostoru. Už nyní také existují prototypy samořídících automobilů, které se při orientaci v prostoru spoléhají právě na systémy LIDAR [1].

První kapitola semestrální práce se věnuje systémům LIDAR a jejich hlavním komponentám. Představuje dva základní přístupy snímání vzdáleností v zorném poli, tedy metodu skenování, která zorné pole ozařuje postupně po jednotlivých segmentech, a metodu záblesku, která zorné pole ozáří rozptýleným světlem v jediný okamžik a snímá jej celé současně. Další sekce první kapitoly se pak zabývá laserovými zdroji a požadavky, které jsou na ně kladeny. Výběr vlnové délky nebo optického výkonu laseru určuje do velké míry odstup signálu od šumu celého systému.

Velká část první kapitoly je věnována jednofotonovým lavinovým diodám, které jsou optimálními fotodetektory pro systémy LIDAR. Je popsána jejich struktura, jednotlivé fáze průrazu, šum a přeslechy, účinnost detekce nebo tzv. "hasič" obvody. Protože se od systémů LIDAR očekává vysoké prostorové rozlišení, jsou diskutovány i možnosti zapojení těchto detektorů do větších souborů.

V poslední sekci první kapitoly je pak osvětlena důležitá technika, která umožňuje přesnou rekonstrukci času průletu fotonů i v zašuměných venkovních prostředích. Tato technika se nazývá časově korelované čítání fotonů. Principem této metody je měření času průletu v několika cyklech, kdy je každý výsledek měření uložen do histogramu. Na

konci měření je pak výsledný čas průletu určen jako modus jednotlivých konverzí. Jsou diskutovány i další možnosti zlepšení tohoto algoritmu.

Druhá kapitola se věnuje časově digitálním převodníkům, které měří čas průletu. Jejich parametry, jako dynamický rozsah, rozlišení, šum apod., jsou tedy určující pro přesnost celého systému. V této kapitole jsou přiblíženy techniky používané při jejich realizaci pro LIDAR systémy, sloužící pro zvýšení linearity převodní charakteristiky nebo snížení výkonové spotřeby. Výkonová spotřeba těchto převodníků je obzvláště důležitá, protože LIDAR systémy obsahují velké soubory fotodetektorů, a tedy i velké soubory časově digitálních převodníků.

Dále jsou představeny jednotlivé typy převodníků – hrubé převodníky s velkým dynamickým rozsahem založené na jednoduchých čítačích, dále tzv. “jemné” převodníky spoléhající na propagační zpoždění digitálních signálů a umožňující dosažení rozlišení v řádu desítek pikosekund, i skutečně pokročilá řešení umožňující dosažení rozlišení v řádu jednotek pikosekund. Protože nastavení operačního bodu zpožďovacích buněk jemných časově digitálních převodníků se v praktických řešeních nastavuje pomocí tzv. závěsů zpoždění (DLL), část druhé kapitoly se věnuje i jim a způsobům implementace řízených zpožďovacích buněk v CMOS výrobních procesech.

Na závěr druhé kapitoly je představena problematika distribuce hodinových signálů napříč soubory časově digitálních převodníků. Jelikož je v LIDAR systémech velký počet převodníků, vzdálenosti mezi nimi jsou na poměry integrovaných obvodů vysoké a při distribuci hodinových signálů na dlouhé vzdálenosti může vlivem parazitních vlastností kovových propojovacích vrstev docházet k velkým dynamickým ztrátám. Alternativním přístupem je generování hodinových signálů lokálně pomocí kruhových oscilátorů, což je z pohledu výkonových ztrát optimální řešení, avšak vzájemný nesouběh těchto oscilátorů se projeví náhodnou chybou zisku a vzorovým šumem jednotlivých “pixelů” LIDAR systému.

Nový přístup řešení tohoto problému reprezentují tzv. injekcí zavěšené oscilátory. To jsou oscilátory, které je možné pomocí periodických injekcí relativně malé energie (nejčastěji v podobě proudových pulzů, resp. dávek náboje) zavěsit na frekvenci injekcí. Třetí kapitola je věnována tomuto fenoménu, který je vysvětlen jak pomocí modelu harmonického LC oscilátoru, tak pomocí modelu kruhového oscilátoru. Třetí kapitola obsahuje kompletní matematické postupy vedoucí k odvození rovnic pro tzv. rozsah závěsu, což je rozsah frekvencí injekcí, na které je oscilátor schopen se zavěsit. Je také představeno konkrétní řešení problému distribuce hodinových signálů napříč soubory časově digitálních převodníků, které díky injekčnímu zavěšení dokáže garantovat synchronizaci frekvencí jednotlivých kruhových oscilátorů, a to za pomoci slabých, a tudíž relativně energeticky nenáročných periodických injekcí.

Dosavadní literatura o injekcí zavěšených kruhových oscilátorech se však nevěnuje některým neidealitám jejich reálné implementace. Jednou z nich je například vliv středy hodinového signálu, od kterého se odvozují injekce, na rozsah závěsu injekcí zavěšeného

oscilátoru.

Ve čtvrté kapitole je tudíž představen makromodel injekcí zavěšeného kruhového oscilátoru vytvořený v programu Matlab Simulink, s jehož pomocí byla prostřednictvím řady analýz ověřena správnost analytických odvození představených ve třetí kapitole, a díky němuž byla prozkoumána citlivost oscilátoru na variaci střídy injektovaného signálu. Bylo zjištěno, že asymetrie střídy injektovaného signálu snižuje rozsah závěsu, avšak oscilátor je na tento jev méně citlivý, pokud jsou injektované pulzy delší v čase. To se shoduje s hypotézou tzv. "citlivého okna", která je v této práci představena a intuitivně vysvětluje princip zavěšení injekcí zavěšených kruhových oscilátorů na jednoduchém modelu invertoru.

Pátá kapitola se zabývá návrhem injekcí zavěšeného kruhového oscilátoru řízeného pomocí závěsu zpoždění, který je určen pro konstrukci časově digitálního převodníku pro systém LIDAR s rozlišením 50 pikosekund. Návrh proběhl ve výrobní technologii ONK65, což je standardní 65nm CMOS technologie. Pátá kapitola vysvětluje důležitá návrhová rozhodnutí jako volbu frekvence kruhových oscilátorů (625 MHz) nebo počet jejich stupňů (16). Velká část kapitoly je také věnována rozboru vzorové architektury celého časově digitálního převodníku, protože ač je v této diplomové práci navrhována jen jeho část, pro návrháře dílčích obvodů je důležité rozumět tomu, jakou roli navrhované bloky v rámci systému splňují, aby mohl optimalizovat jejich klíčové parametry.

Samotný návrh na tranzistorové úrovni začal návrhem a optimalizací jednotlivých zpožďovacích buněk, z kterých se kruhový oscilátor skládá. Poté následoval návrh injekčního obvodu, jehož hlavním cílem bylo zachovat co nejvyšší rozsah zavěšení, ale zároveň neplýtvat energií, k čemuž byly využity poznatky ze čtvrté kapitoly, ve které byl injekcí zavěšený oscilátor modelován v programu Matlab Simulink. Dále se pátá kapitola věnuje návrhu závěsu zpoždění, který slouží k stabilizaci injekcí zavěšených oscilátorů vůči procesním, napěťovým nebo teplotním výchytkám. Obvody závěsu zpoždění byly rozšířeny o trimovací obvody, které umožňují kompenzovat chyby způsobené výrobním rozptylem.

Funkce navržených obvodů byla ověřena počítačovými simulacemi zohledňujícími procesní, napěťové a teplotní (PVT) extrémů. Až na jednu statisticky nepravděpodobnou a z hlediska celkové výtěžnosti zanedbatelnou kombinaci těchto extrémů bylo prokázáno, že jsou navržené injekcí zavěšené kruhové oscilátory schopny záchyty na cílovou frekvenci 625 MHz i v nepříznivých podmínkách. Výše zmíněná problematická kombinace PVT extrémů byla analyzována a potenciální řešení společně s jejich případnými nevýhodami byla diskutována. Simulace zohledňující šum navržených obvodů dále potvrzují, že fázeový neklid zavěšeného oscilátoru je až o polovinu nižší oproti volnoběžnému stavu, což je v souladu s teoretickými poznatky. Počítačové simulace také dokládají, že střední hodnota časového rozlišení, které navržené oscilátory poskytují, je skutečně specifikovaných 50 pikosekund, přičemž směrodatná odchylka tohoto časového rozlišení je cca 17 %,

což je dáno zejména vlivem nesouběhu jednotlivých stupňů oscilátoru. Dosažení lepšího souběhu a přesnosti časového rozlišení by bylo možné pouze za cenu zvýšení velikosti jednotlivých stupňů, a tudíž i zvýšení jejich parazitních kapacit, navazujícího snížení rychlosti oscilátorů, nutnému navýšení proudové spotřeby pro kompenzaci snížené rychlosti a v konečném důsledku nezanedbatelného zhoršení výtěžnosti v případě “pomalých” procesních odchylek. Toto byl jen jeden z mnoha kompromisů učiněných během návrhu, jež dokládají, že specifikace rozlišení 50 pikosekund je na hranici praktické dosažitelnosti pro danou výrobní technologii. Pátá kapitola je zakončena krátkou diskuzí několika alternativních řešení, které byly objeveny během návrhu a které by mohly přispět k řešení některých výše zmíněných problémů nebo k odstranění jednoho z trimovacích kroků.

FRÁNEK, Jakub. *Injection locked ring oscillator design for application in Direct Time of Flight LIDAR*. Brno: Brno University of Technology, Faculty of Electrical Engineering and Communication, Department of Microelectronics, 2021, 194 p. Master's Thesis. Advised by Ing. Vilém Kledrowetz, Ph.D.

Author's Declaration

Author: Bc. Jakub Fránek
Author's ID: 186059
Paper type: Master's Thesis
Academic year: 2020/21
Topic: Injection locked ring oscillator design for application in Direct Time of Flight LIDAR

I declare that I have written this paper independently, under the guidance of the advisor and using exclusively the technical references and other sources of information cited in the paper and listed in the comprehensive bibliography at the end of the paper.

As the author, I furthermore declare that, with respect to the creation of this paper, I have not infringed any copyright or violated anyone's personal and/or ownership rights. In this context, I am fully aware of the consequences of breaking Regulation § 11 of the Copyright Act No. 121/2000 Coll. of the Czech Republic, as amended, and of any breach of rights related to intellectual property or introduced within amendments to relevant Acts such as the Intellectual Property Act or the Criminal Code, Act No. 40/2009 Coll. of the Czech Republic, Section 2, Head VI, Part 4.

Brno

.....

author's signature*

*The author signs only in the printed version.

ACKNOWLEDGEMENT

I wish to express my gratitude to Ing. Ivan Koudar, Ph.D. for the opportunity to deepen my analog circuit design expertise by writing a thesis about this contemporary topic under his guidance.

I am also grateful to Ing. Tomáš Rozkopal and other design engineers from Analog & Memory IP team at ON Design Czech s.r.o. whose seemingly endless willingness to provide advice was invaluable to me. Last but not least, I want to thank Ing. Vilém Kledrowetz, Ph.D. for his helpful advice on the formal aspects of the thesis.

CONTENTS

Introduction	20
1 Direct Time of Flight LIDAR	22
1.1 System overview	22
1.2 Imaging methods	23
1.2.1 Scanning LIDAR	23
1.2.2 Flash LIDAR	25
1.2.3 Comparison	26
1.3 Laser sources	26
1.3.1 Wavelength	26
1.3.2 Output power and eye safety	27
1.3.3 Diode lasers	28
1.4 Single Photon Avalanche Diodes	29
1.4.1 Avalanche Photodiode	29
1.4.2 Basic operation of SPAD	30
1.4.3 Timing characteristics	31
1.4.4 Dead time, quenching circuits and afterpulsing	32
1.4.5 photon detection efficiency	33
1.4.6 Dark count rate and crosstalk	35
1.4.7 SPAD imagers and Silicon Photomultipliers	36
1.5 Time Correlated Single Photon Counting	38
1.5.1 Coincidence counting	40
2 Time to Digital Converters	43
2.1 General considerations	43
2.1.1 Full scale and resolution	43
2.1.2 TDC array architecture	44
2.1.3 Reverse timing scheme	45
2.1.4 Sliding scale technique	46
2.2 Counter based TDCs	47
2.3 Delay Locked Loops	48
2.3.1 Delay Line Unit circuits	51
2.4 Propagation delay based TDCs	53
2.5 Sub-gate delay based TDCs	55
2.6 Clock distribution schemes	58
2.6.1 Global counting	58
2.6.2 Local counting	60

2.6.3	Summary	61
3	Injection Locked Oscillators	62
3.1	Oscillation criteria	62
3.2	LC tank based ILO	64
3.2.1	Phasor analysis	65
3.2.2	Locking range derivation	67
3.2.3	Paradox of locking and phase noise performance	70
3.3	Injection Locked Ring Oscillators	71
3.3.1	Ring oscillator time domain analysis	72
3.3.2	Ring oscillator under injection	75
3.3.3	Limitations of the analytical model	79
3.4	Injection Locked Ring Oscillator based TDCs	81
4	Injection Locked Ring Oscillator Modelling	83
4.1	Motivation and goals	83
4.2	Model overview	84
4.2.1	Differential inverter model	84
4.2.2	Injection signal	85
4.2.3	Default simulation parameters	86
4.3	Simulation outputs	87
4.3.1	Transient waveforms	87
4.3.2	Frequency domain	89
4.3.3	Spectrogram	90
4.4	Analyses	90
4.4.1	Locking range	90
4.4.2	Time domain measurements	93
4.4.3	Phase noise	96
4.4.4	Injection clock duty cycle	98
4.4.5	Injection pulse width	99
4.4.6	Summary	102
5	Injection Locked Ring Oscillator Design	104
5.1	Technology overview	104
5.2	Choice of ILRO operating frequency	105
5.3	TDC architecture overview	107
5.3.1	SPAD and TDC array architecture	107
5.3.2	TDC conversion timing scheme	110
5.3.3	Clock source, PLL and DLL	111
5.3.4	ILROs and clock phase latches	113

5.3.5	Coarse TDC counters and digital controller	116
5.3.6	Timing diagram	117
5.4	Simulation conditions	120
5.5	Injection locked ring oscillator design	121
5.5.1	Ring oscillator design	121
5.5.2	Biasing mirror design	127
5.5.3	Injection circuit design	129
5.5.4	Locking simulations	133
5.6	Delay Locked Loop design	138
5.6.1	Overview	138
5.6.2	Frequency Phase Detector and Charge Pump	140
5.6.3	Voltage to Current converter	146
5.6.4	Trimming	152
5.7	Top-level simulations	159
5.8	Alternative solutions	168
	Summary	171
	Bibliography	173
	Symbols and abbreviations	181
	List of appendices	187

LIST OF FIGURES

1.1	DToF LIDAR system diagram	22
1.2	Illustration of scanning LIDAR	23
1.3	Illustration of flash LIDAR	25
1.4	Computed NIR atmospheric transmission spectrum from ATRAN	27
1.5	Direct normal solar spectral irradiance at 1.5 air mass	28
1.6	SPAD I-V characteristic with key stages of the avalanche breakdown highlighted	30
1.7	Typical passive quenching SPAD circuit	30
1.8	Two-step PQAR circuit	33
1.9	PDE as a function of incident photon wavelength for two commercial SPAD based Silicon Photomultipliers, biased at $AP = 0.9$	34
1.10	Mean absorption depth of photons in silicon at 300 K as a function of wavelength	35
1.11	Simplified SiPM schematic	37
1.12	Standard SiPM readout with a TIA	38
1.13	Illustration of a TCSPC histogram	39
1.14	Illustration of coincidence counting	41
2.1	Time of flight of a reflected photon as a function of distance to the obstacle	44
2.2	Sliding scale TDC illustration	47
2.3	Asynchronous ripple counter	48
2.4	Delay Locked Loop illustration	49
2.5	DLL-assisted PVT immune VCDFLU biasing	49
2.6	Simplified charge pump based Delay Locked Loop control circuits	50
2.7	Current starved inverter	51
2.8	High linearity Voltage Controlled Delay Line Unit	52
2.9	Differential Current Controlled Delay Line Unit	52
2.10	Flash TDC illustration	53
2.11	Sliding scale flash TDC illustration	54
2.12	Gated ring oscillator based flash FTDC with a counter based CTDC	55
2.13	Simplified Vernier line SFTDC schematic	56
2.14	Simplified cyclic Vernier line SFTDC schematic	57
2.15	Global clock counting with single buffer	59
2.16	Global clock counting with multiple stage clock tree	59
2.17	Local clock counting with DLLs	60
2.18	Local clock counting with ring oscillators	61
3.1	General feedback system	62

3.2	General feedback system with additional phase shift	63
3.3	Phase response of a general feedback system with additional phase shift	64
3.4	Half circuit of a differential negative resistance LC tank based ILO . .	65
3.5	LC tank ILO current phasor diagram	66
3.6	LC tank ILO current phasor diagram at the edge of the locking range	69
3.7	Injection angle as a function of $\Delta\omega_0$ for various K	70
3.8	Phase noise of a free-running and injection locked oscillator	71
3.9	Differential injection locked ring oscillator model	72
3.10	Ring oscillator model voltage waveform	73
3.11	Time domain based model of injection locking	75
3.12	Locking range of ILRO for various number of stages	78
3.13	Ring oscillator injection sensitivity time window	79
3.14	Pulse injection circuit	80
3.15	Differential ILRO based 3-bit flash FTDC	81
3.16	Local clock counting with ILROs	82
4.1	Differential inverter model in Simulink	85
4.2	Injection pulse train generator	86
4.3	Injection pulse chirp generator	86
4.4	ILRO model period of oscillation for $f_{inj} < f_0$	88
4.5	ILRO model period of oscillation for $f_{inj} > f_0$	88
4.6	<i>PSD</i> of ILRO model when injection pulled and locked	89
4.7	Spectrogram of a clock phase first harmonic with chirp injection signal	91
4.8	Locking range of the ILRO model as a function of injection ratio . . .	92
4.9	Relative locking range of ILRO models as a function of injection ratio	93
4.10	Definition of time domain measurements	94
4.11	The phase shift Δ of the ILRO model as a function of f_{inj}	95
4.12	The injection angle ϑ of the ILRO model as a function of f_{inj}	95
4.13	The zero cross delay d of the ILRO model as a function of f_{inj}	97
4.14	The zero cross delay d of the ILRO model as a function of ϑ	97
4.15	Power spectral density of a free-running and injection locked ILRO .	98
4.16	The effect of injection clock duty cycle on injection pulse timing . . .	99
4.17	Relative locking range of the ILRO model as a function of injection clock duty cycle	100
4.18	Normalized relative locking range of ILRO models as a function of injection clock duty cycle	100
4.19	ILRO model period of oscillation for $DC = 40\%$, $f_{inj} > f_0$	101
4.20	Relative locking range of the ILRO model for various injection pulse sizes	102
5.1	Overview of an ILRO-based TDC architecture	108

5.2	SPAD & TDC array scanning and converting scheme	109
5.3	Differential ILRO clock phase latch example	115
5.4	Timing diagram	118
5.5	Realistic biasing current generator	121
5.6	Buffered differential CCDLU topology	122
5.7	Simulated propagation delay dependency on biasing current for low V_{th} (left) and normal V_{th} (right) minimum-sized CCDLU	122
5.8	ILRO connection overview	123
5.9	Simulated CCDLU typical transient signals	124
5.10	Free-running frequency f_0 as a function of I_{BIAS} for the designed ILRO	125
5.11	ILRO nominal biasing current for $f_0 = 625$ MHz	126
5.12	CCDL NMOS biasing current mirror	127
5.13	ILRO free-running frequency variation and mismatch for various bi- asing current mirror device sizes	128
5.14	Alternative injection methods	129
5.15	Designed injection circuit	130
5.16	Locking range of the designed ILRO for various sizes of R_{inj}	131
5.17	Propagation delay distribution within the ILRO with and without dummy switches, when free-running at $f_0 = 625$ MHz	132
5.18	Typical waveforms of the injection stage voltage, the injection pulse and the injected current for a locked ILRO of $f_0 = 590$ MHz	133
5.19	Various waveforms of injected current into ILROs of various f_0 when locked to 625 MHz	134
5.20	Charge injected by the injection circuit into an ILRO per cycle once locked to 625 MHz as a function of initial free-running frequency	135
5.21	Instantaneous frequency of ILROs with various f_0 during locking	136
5.22	Injection locking process	137
5.23	Simplified overview of the DLL	139
5.24	FPD signal sequence	140
5.25	Simplified schematic of the designed FPD and charge pump block	141
5.26	FPD input phase shift as a function of charge pump current source mismatch when connected in a DLL	143
5.27	<i>RMS</i> charge pump output current as a function of FPD input phase shift	145
5.28	<i>RMS</i> charge pump output current as a function of V_{CPO} when UP and DWN signals are both active	146
5.29	Standard V-to-I converter	147
5.30	Designed DLL V-to-I converter	148

5.31	Charge pump output voltage stability over temperature for nominal and cross resistor corners	150
5.32	Transconductance from V_{DD} to output current of V-to-I converter's current mirror	151
5.33	Designed DLL V-to-I converter with trimmed devices highlighted	154
5.34	Trimmed PMOS current mirror devices	156
5.35	Error of the injection stage propagation delay as a function of the free-running frequency error	158
5.36	ILRO trimming trim code and residual error histograms	158
5.37	DLL start-up waveforms	159
5.38	ILRO free-running frequency over VT variation when biased by a constant current versus a DLL	160
5.39	DLL trim code sweep across process corners	161
5.40	ILRO trim code sweep across process corners	162
5.41	Instantaneous frequency of ILRO across selected PVT corners during locking after trimming	164
5.42	PVT corner overview of free-running trimmed ILRO biased by a trimmed DLL	165
5.43	Free-running frequency of trimmed ILRO biased by a trimmed DLL as a function of V_{DD}	166
5.44	Jitter histogram of a locked and unlocked ILRO	168
5.45	PLL based ILRO biasing	169
5.46	Alternative dynamic ADC-based DLL trimming method	170
5.47	Alternative dynamic VCR-based V_{CPO} stabilizing method	170

LIST OF TABLES

4.1	Default simulation parameters	87
5.1	Benefits and detriments of higher number of ILRO stages	106
5.2	Clock phases of 16-stage differential ILRO	114
5.3	Default simulation conditions	120
5.4	CCDLU device sizes	124
5.5	ILRO performance	138
5.6	FPD performance	144
5.7	V-to-I converter performance	152
5.8	Overview of V-to-I converter's current mirror trimmed PMOS units .	157

INTRODUCTION

The invention of the *Single Photon Avalanche Diode* (SPAD) and its integration in standard *Complementary Metal Oxide Semiconductor* (CMOS) fabrication processes has enabled rapid growth of a wide range of new imaging applications. The single photon sensitivity and picosecond level temporal jitter of SPADs has been utilized for low illuminance imaging, positron emission tomography, various types of spectroscopy, fluorescence lifetime imaging microscopy, diffuse optical tomography or *Direct Time of Flight* (DToF) based *Light Detection and Ranging* (LIDAR).

The DToF based LIDAR measures the time interval between the instant when a laser pulse was transmitted, illuminating the scene, and the instant the reflected photons have returned and were detected by the receiver. The time interval is then used to calculate the distance to the obstacle. Compared to other 3D imaging and ranging techniques such as ultrasonic sensors, stereo vision cameras or millimetre wave radars, LIDAR can provide superior resolution and competitive dynamic range, while requiring less computing power and operating reliably in environments with uncontrolled ambient light or low illuminance conditions. This is achieved by utilizing *Time Correlated Single Photon Counting* (TCSPC), a statistical technique capable of restoring photon arrival times with picosecond resolution.

The superior spatial and longitudinal resolution of LIDAR has attracted attention from the automotive industry in particular, as it not only allows the detection of road traffic, but also relatively small obstacles such as pillars, wires or road defects at a distance. This information can be used by the *Advanced Driver Assistance System* (ADAS), which incorporates advanced cruise control, autonomous emergency braking, pedestrian detection, forward collision warning etc. As new car assessment programs across the world are expanding their safety rating criteria to include the safety of pedestrians as well as the passengers, it is expected that in the near future, safety systems like ADAS will be commonplace in cars across all price points. An even more advanced driving system, autonomous driving, is another application for LIDAR technology, as proven by Waymo, which uses a 360° LIDAR based computer vision to operate their driverless car prototypes [1].

The main disadvantage of LIDAR has historically been its cost and size. To scan the whole scene with a sufficient *Signal to Noise Ratio* (SNR), laser scanning is needed, which requires complex optomechanical systems. The recent efforts are therefore focused on bringing the cost and the size down by integrating the whole scanning system on a single CMOS integrated circuit.

A DToF measurement system consists of the following main parts: the laser source, the photodetector, the *Time to Digital Converter* (TDC) and the data processing circuits. The TDC measures the time interval between the transmitted laser

pulse and its reflected echo, and as such is one of the most critical parts, determining the dynamic range and the resolution of the system. The key challenges of TDC design are achieving the sub-nanosecond resolution, and the generation and routing of the counting signals in a power efficient manner.

The chapter 1 of this thesis will introduce the general principles of DToF based LIDAR systems, the operation of SPADs or the TCSPC technique. The chapter 2 will focus on TDCs, their key parameters, practical design techniques and their various implementations. In the chapter 3, a novel approach to TDC architecture, utilizing *Injection Locked Oscillators* (ILOs) to achieve power efficient clock generation and resolution on the order of tens of picoseconds will be described. In chapter 4, a MATLAB model of an *Injection Locked Ring Oscillator* (ILRO) will be presented. Finally, in chapter 5, an ILRO will be designed and simulated in a 65 nm CMOS processing technology.

1 DIRECT TIME OF FLIGHT LIDAR

This chapter will provide an overview of DToF LIDAR systems, as well as basic insight into imaging strategies, laser sources, photodetectors and signal processing techniques used in contemporary art. TDCs will be described later in chapter 2 in more detail.

1.1 System overview

The basic structure and operation of DToF LIDAR systems is illustrated on Figure 1.1.

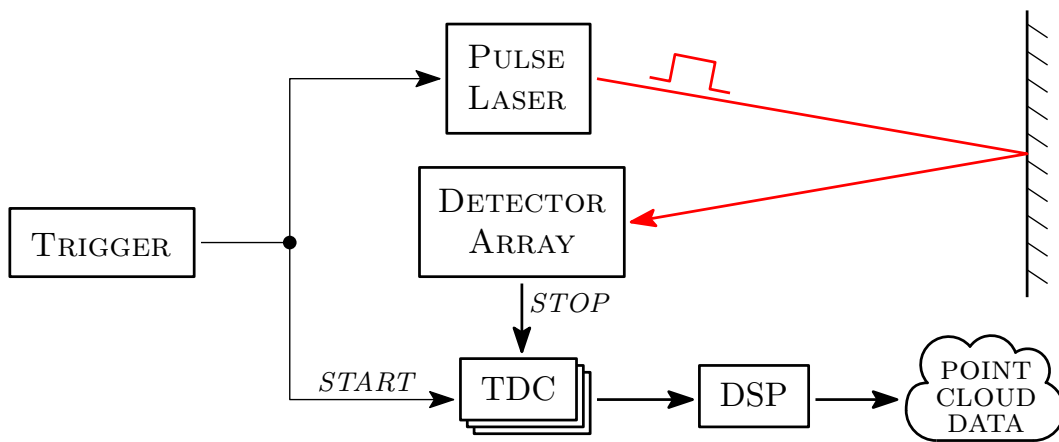


Fig. 1.1: DToF LIDAR system diagram

An electronically driven pulsed laser emits a light pulse, illuminating the scene, also known as the *Field of View* (FoV). The same trigger signal, which fired the laser pulse, is used to start the TDC measurement. As the travelling pulse hits objects inside the FoV, a portion of the pulse, which (amongst other factors) depends on the distance, the incident angle or the reflectivity of the surface of the object, is reflected back and hits the detector. Upon detection, the detector produces an electrical pulse, freezing the TDC. The TDC consequently outputs a digital value, corresponding to the time interval between the firing of the laser and the detection of its reflection. This time is called the *Time of Flight* (ToF), and can be used to determine the distance travelled by the light pulse, which is simply

$$d = \frac{c}{2} \cdot ToF \quad (1.1)$$

where d is the distance to the obstacle and c is the speed of light. The division by 2 needs to be included, because the reflected photons have travelled the distance between the LIDAR device and the obstacle twice.

A *Digital Signal Processing* (DSP) block performs various filtering and noise rejection techniques on the TDC readings, namely *Time Correlated Single Photon Counting* (TCSPC) which is described in more detail in section 1.5. The processed data can be then used to build a “point cloud”, which is the desired representation of the FoV with the distance measurements represented by points in a 3D coordinate space.

1.2 Imaging methods

Since the final output of the LIDAR system is a 3D point cloud of the objects in the FoV, and the FoV for applications such as automotive spans tens of degrees in angular size, the method of its illumination has to be carefully considered.

There are two main types of DToF LIDAR differing in their approach to FoV illumination. In this section, these methods will be briefly described and compared according to their advantages.

1.2.1 Scanning LIDAR

The “scanning LIDAR” utilizes a laser beam with low angular divergence to scan the FoV step by step using dual axis scanning. The beam needs to be steered across the FoV over time, which is a challenge of its own. The precision and speed of the scanning mechanism are now another factor determining the overall performance of the system, and depending on the implementation, the scanning system can add both significant costs and reliability issues.

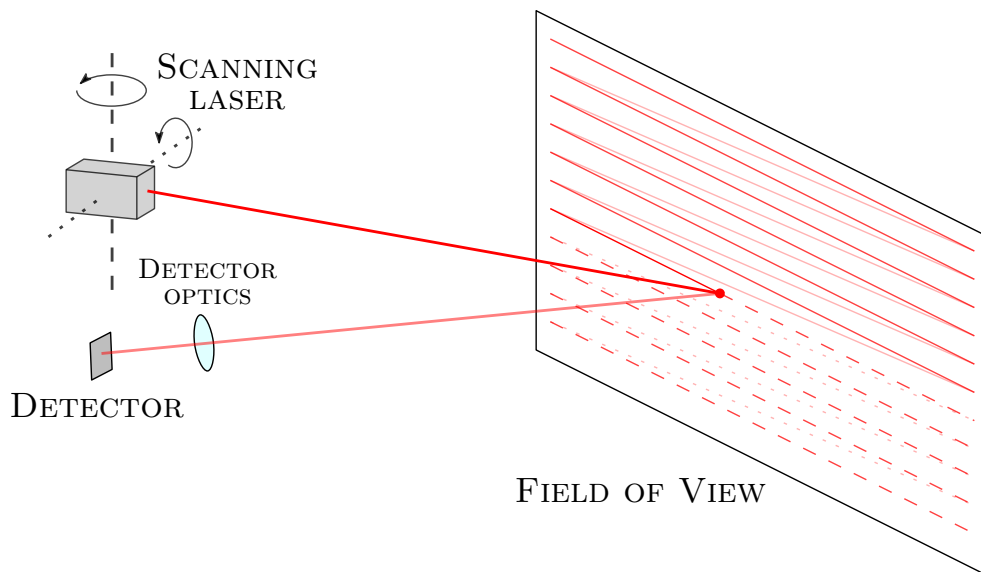


Fig. 1.2: Illustration of scanning LIDAR

There are two main types of scanners which utilize actual mechanical movement to direct the laser beam. The movement can be done with electrical stepper motors, which has the benefit of a wide FoV but brings larger cost, power consumption, size and reliability issues. In these cases, a polygon mirror reflecting the laser beam can be rotated instead of the whole laser or LIDAR assembly, as demonstrated in [2]. Despite the disadvantages, this is still the dominant method of scanning in automotive applications because of its long range performance and wide, up to 360° FoV [3, p.12].

A promising alternative are *Micro-electro-mechanical System* (MEMS) actuators, which steer the laser beam using electrically driven micromirrors. The advantages of such optical systems are clear: low weight, compactness, low cost and low power consumption. Although the technology is already proven at short to medium range (25 m range demonstrated by [4]), it has not been able to reach the FoV and the long range of rotational scanners. This technique is however getting increased attention and the company Innoviz Technologies is claiming they achieved 600 m range and $73^\circ \times 20^\circ$ FoV with their new product [5], which could be the best commercially released MEMS LIDAR so far.

Optical Phased Array (OPA) is a novel fully solid state method of beam steering. The operating principle is identical to phased array antennas, where the direction of the beam is determined by the phase difference between neighbouring transmitters, allowing constructive interference of the electromagnetic waves in the desired direction and destructive interference in others. OPA scanners share the same advantages of MEMS scanners, but since there are no moving parts at all in OPA scanners, they can reach even higher scanning speeds (100 kHz reported by [6, p.409]) and better reliability. However, the method does have issues with insertion power loss of the phase shifting waveguides. Quanergy is a company aiming to provide a commercial automotive OPA based LIDAR in the near future, claiming 120° FoV, 150 m range and *mean time between failure* (MTBF) of more than 100 000 h [7].

Finally, another novel option of implementing fully solid state beam steering is to use an array of *Vertical Cavity Surface Emitting Lasers* (VCSELs) and a special optical lens. An identical lens can be used for the detector array, which has the same size and number of pixels as the VCSEL array. Because the arrays and the optical systems match, by activating certain pixels of the laser array, a small section of the FoV is illuminated and the reflected photons arrive at the corresponding detector array pixels. Ibeo Automotive, a company which has implemented multiple products of this type, calls the method “sequential flash”, as it combines the philosophy of both scanning and flash LIDAR. Their product ibeoNEXT Generic contains 10240 pixels in both the VCSEL and detector array, and claims 250 m range, FoV ranging up to $60^\circ \times 30^\circ$ and angular resolution as good as 0.05° [8].

1.2.2 Flash LIDAR

The other approach known as “flash LIDAR” uses an optical diffuser to increase the angular divergence of the laser, illuminating the scene as a whole in a single pulse. The challenge is therefore shifted away from the transmitter to the detector, which needs to detect the reflections of the diffused laser, and the optical system, which needs to restore the directional information in order to determine the distance to obstacles in various regions of the FoV. Detectors positioned at the focal plane of a lens are called *Focal Plane Array* (FPA) detectors. A natural advantage of the flash LIDAR is the absence of moving parts and high refresh rate, as the whole FoV is illuminated at once.

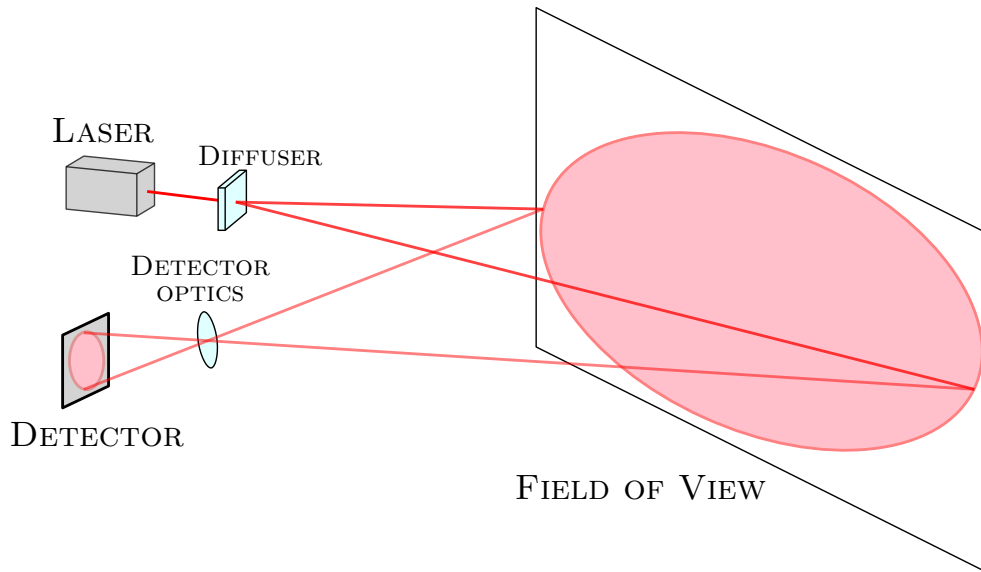


Fig. 1.3: Illustration of flash LIDAR

The detector optical system is designed to match the divergence of the laser diffuser, so that all the detector array pixels are illuminated simultaneously. The spatial resolution is therefore determined by the pixel density of the array, which is limited by the process node, the size of the chip and cost.

The main challenge with flash LIDAR is the SNR. Because the output power of the laser is spread across the whole FoV, the reflections are weaker. Therefore there is a clear inverse relationship between the size of the FoV and the maximum detectable range. Special attention has to be given to the detector array design in order to maximize its optical sensitivity and efficiency. Another potential issue is the power consumption and associated thermal dissipation. With flash LIDAR, a large number of TDCs need to be active simultaneously, requiring significant power and producing large amount of heat.

An example of a flash LIDAR system implementation is described in [9], where a 252×144 detector array was used, achieving 50 m range.

1.2.3 Comparison

Although currently most widely used automotive LIDAR systems are based on rotating mirror scanners, “there is a quite general agreement that mechanically scanning LIDARs need to move towards a solid-state version” [3, p.11]. While their up to 360° wide FoV is desirable, the size, cost and overall unreliability and maintenance complexities, which are critical factors in automotive systems, will outweigh the benefits in the future.

If solid state systems are to be compared, there are promising solutions on both the scanning and flash LIDAR side, as well as somewhere in between like the sequential flash technology. Generally though, it can be said that flash LIDAR seems to be better suited for short to medium range applications, which require high frame rate. The fact that the whole FoV is illuminated and detected virtually simultaneously is also beneficial for automotive, where the LIDAR system is moving relative to the surrounding objects at high velocity, as in such applications, scanning the FoV (an inherently slower process) can cause movement artefacts and distortions in the resulting image.

On the other hand, scanning LIDAR solutions promise higher maximum detectable range, as the output power of the laser is focused into a much smaller angular diameter, at the cost of slower frame rate. Further development of fast scanning OPA systems could however erase this disadvantage.

1.3 Laser sources

The laser source is a key part of the DToF LIDAR system, as its parameters define or contribute to many performance characteristics of the overall product. Quantities like peak output power, wavelength, pulse width, spectral purity, power efficiency, size, weight and many others are of interest.

1.3.1 Wavelength

Perhaps the most important of these specifications is the wavelength. Naturally, wavelengths invisible to human eyesight are necessary. In order to transmit the laser pulse as far as possible and subsequently differentiate the reflected photons from the surrounding ambient light, a wavelength with both high atmospheric transmittance (see Figure 1.4¹) and low solar irradiance (see Figure 1.5) needs to be chosen. These two conditions do not necessarily go hand in hand, therefore a compromise is necessary.

¹ATRAN parameters: altitude 2500 ft, latitude 39°, 2 atmospheric layers, zenith angle 20° [10]

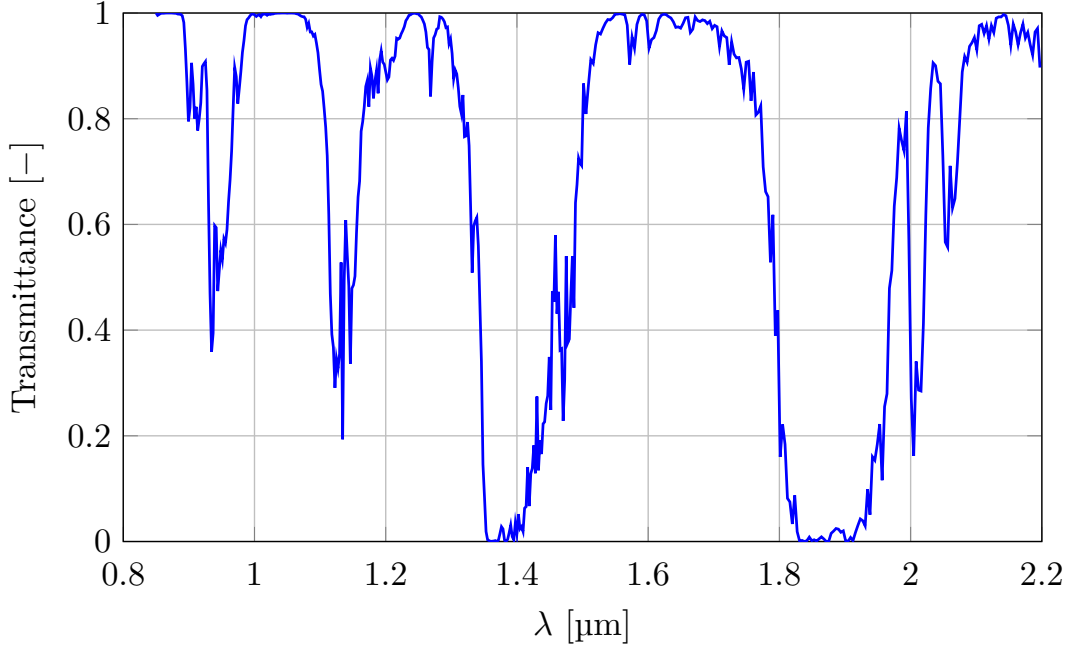


Fig. 1.4: Computed NIR atmospheric transmission spectrum from ATRAN [10]

There are three regions of interest: 0.8 to 0.95 μm , 1.06 μm and 1.55 μm [3, p.18]. The third region would be the ideal one in terms of the distance to the optical range (due to safety concerns, as will be detailed later in this section), the high atmospheric transmittance and the low solar irradiance in the region, but it is outside the optical window of silicon (more details in subsection 1.4.5) and requires InGaAs or InP detectors, which are expensive and due to processing technology limitations do not reach pixel densities comparable to traditional silicon nodes [11].

The most commonly utilized range is the 0.8 to 0.95 μm one. Although it suffers from power limitations due to eye safety, slightly worse atmospheric transmittance and potentially larger solar activity (although there is a noticeable dip around 0.93 μm , as visible on Figure 1.5), silicon based detectors are the most sensitive to electromagnetic radiation in this region and offer best sensitivity. The cost efficiency of CMOS processes is vastly superior to more exotic InGaAs technologies and as such lasers working at these wavelengths are chosen for primarily economical reasons.

1.3.2 Output power and eye safety

Another important specification of the laser source is the peak output power transmitted by the pulse. The higher the peak power transmitted by the laser, the higher the power of the reflection, and the higher the probability of its detection. This improves the SNR of the system, and effectively increases the maximum detectable range.

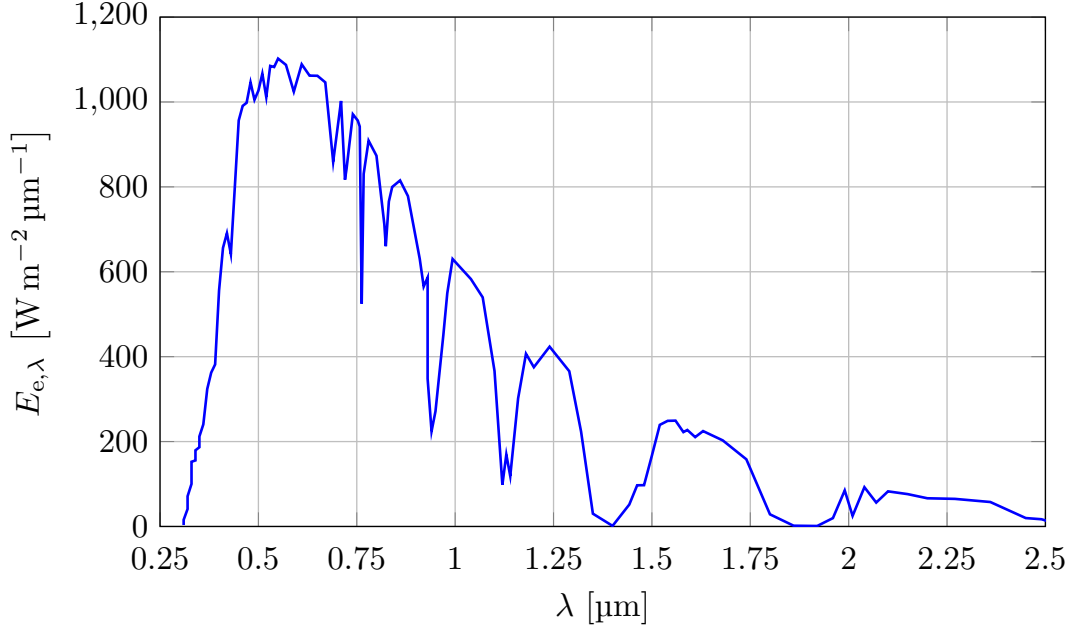


Fig. 1.5: Direct normal solar spectral irradiance at 1.5 air mass [12]

The relationship between the transmitted and collected power is described by the “LIDAR equation” [11, p.16]:

$$P_c = \frac{\rho}{\pi R^2} P_t A_d \cos(\theta) e^{-2\alpha R} \quad (1.2)$$

where P_c is the collected power, ρ is the reflection constant, R is the distance to the reflector, P_t is the transmitted power, A_d is the detector optical receiving area, θ is the incident angle respective to surface normal and α is the absorption coefficient. It is clear that R is the most dominant factor.

The limit to the peak optical power is however regulated by eye safety standards such as IEC 60825-1, which determine the *Maximum Permissible Exposure* (MPE) limits, commonly expressed as the maximum allowable exposure time for a given irradiance (given in Watts per cm^2) for the given wavelength.

Because LIDAR systems need to be safe under all conditions (Class 1 as defined by IEC 60825-1), the MPE limits are quite strict especially for the *Near Infrared* (NIR) range. To decrease the energy delivered to a unit area and meet the standards, either the pulse length can be decreased, or, for flash LIDAR systems, the diffusion angle can be increased so that the energy is spread over larger FoV.

1.3.3 Diode lasers

While the use of fiber or microchip lasers in LIDAR systems is possible (and they offer superior pulse rate and pulse widths as well [3, p.21]), the most widely used devices are diode lasers because of their compactness and price. This is especially

the case for systems relying on an array of lasers, as semiconductor diode lasers are the only economical option of fabricating laser arrays.

Semiconductor diode lasers come in two types: *Edge Emitting Laser* (EEL) and *Vertical Cavity Surface Emitting Laser* (VCSEL). EELs are well established diode lasers, based on PN or PIN junctions. The laser shines from the edge, i.e. in parallel to the plane of the silicon wafer it is constructed from. After fabrication, EELs need to be cleaved and coated with reflective materials on the sides to create a cavity with high optical gain. A disadvantage is that the beam they produce is not spherical and needs to be shaped with optical components [13].

VCSELs have been receiving increased attention recently. As the name suggests, the beam produced by the VCSEL device is emitted in a direction perpendicular to the top surface of the semiconductor die. Instead of using reflective coating, the mirrors are made from thin layers of planar Bragg reflectors. In [14], InGaAs Bragg reflectors were used to produce a 940 nm VCSEL.

The direction of the beam is a significant advantage for testing, as VCSEL devices can be tested before the wafer is cleaved, which is faster and more economical. The beam produced by VCSELs is also spherical even in the absence of additional lens [14] and less divergent. Finally, VCSEL arrays enable technologies like OPA scanners [15] or sequential flash imaging, which were discussed previously in subsection 1.2.1. The main challenge in current VCSEL development is the heat generation and transfer, as it is the factor limiting the maximum power or the VCSEL array size [16].

1.4 Single Photon Avalanche Diodes

The *Single Photon Avalanche Diode* (SPAD), also known as *Geiger-mode Avalanche Photodiode* (GAPD), is the dominant photodetector used in DToF LIDAR systems. It is capable of detecting a single photon with a high temporal resolution (on the order of low tens of picoseconds), while also being compatible with standard CMOS processes, which is critical for the viability of commercial products.

In this section, the operation and characteristics of SPADs will be described in more detail, in order to understand the implications for the LIDAR system architecture and performance.

1.4.1 Avalanche Photodiode

The SPAD is structurally nearly identical to the *Avalanche Photodiode* (APD). The APD consists of a PN junction, operated in the reverse region, close to but not above the SPAD breakdown voltage (V_{BD}), which is usually between 10 to 50 V [17].

A depletion region is created, where no charge carriers exist and virtually no current flows through the junction. As soon as an incoming photon hits the depletion region, it can be absorbed, generating an electron-hole pair, which is immediately separated by the electric field in the depletion region. If the reverse voltage and the power of the incident light are both high enough, the generated charge carriers multiply along the way via impact ionization, producing an avalanche.

The intensity of the avalanche is proportional the power of the incident light, therefore the APD provides both the ToF information as well as an analog current representation of the intensity of the reflection. In order to convert the current to voltage while preserving the resolution of both time and intensity, a *Trans-impedance Amplifier* (TIA) capable of satisfying high bandwidth, high gain and low noise requirements is required.

1.4.2 Basic operation of SPAD

Contrary to the APD, which is biased below V_{BD} , the SPAD is biased above V_{BD} in the so-called Geiger mode. The name comes from the Geiger-Müller tube, which operates in a similar manner. The difference between the applied reverse voltage and V_{BD} is the SPAD excess bias voltage (V_E), also known as the overvoltage, and it commonly ranges from 2 to 7 V.

While the avalanche in APDs eventually stops on its own, the avalanche in the SPAD devices is self-sustaining, as the PN junction breaks down completely and the impact ionization multiplication process produces an evergrowing number of charge carriers. In order to save the diode from overheating and self-destruction, the avalanche has to be quenched externally.

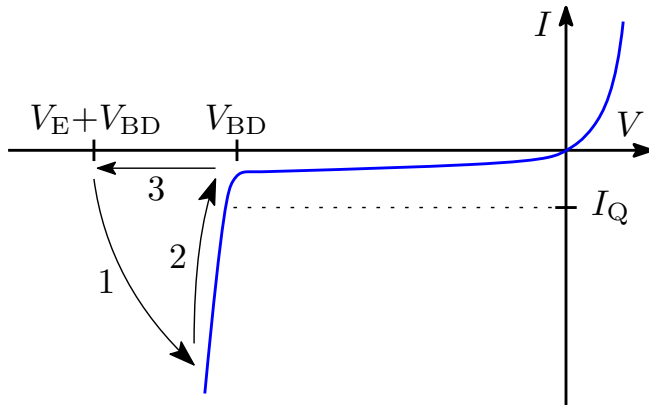


Fig. 1.6: SPAD I-V characteristic with key stages of the avalanche breakdown highlighted

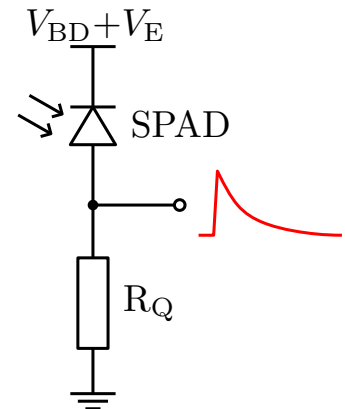


Fig. 1.7: Typical passive quenching SPAD circuit

The avalanche breakdown mechanism in SPADs will be described in more detail. Figure 1.6 depicts an I-V characteristic of a SPAD, including the three main phases

of its operation, and Figure 1.7 shows a typical SPAD circuit connection with passive quenching. At the start of the cycle, there is no current flowing through the diode and the full $V_{BD}+V_E$ voltage is applied across the junction in reverse. While this voltage exceeds V_{BD} , the diode can exist in this pseudo-stable state for an extended period of time, as long as no charge carriers exist inside the depletion region. As soon as a charge carrier appears inside the depletion region, a run-away avalanche develops and the current through the diode spikes within a few tens of picoseconds, moving the operating point of the diode from the pseudo-stable state to the steady state I-V curve (phase 1 – seeding and spreading). The current spike is fed into the quenching resistor R_Q , producing a growing voltage drop at the expense of the voltage across the diode (phase 2 – quenching). As the voltage across the diode decreases, the diode current eventually becomes lower than the *latching current* (I_Q), the avalanche is no longer self-sustaining and it stops shortly thereafter. When there are no more charge carriers in the depletion region left, the junction capacitance of the diode is charged to $V_{BD}+V_E$ and the diode can detect incident photons again (phase 3 – recharge).

Since the optical gain of the SPAD is theoretically infinite (a single photon can produce a self-sustaining avalanche), the information about the power of the incident light provided by the APD is traded off for single photon sensitivity and superior temporal resolution (on the order of low tens of picoseconds). Because the SPAD is essentially a digital photodetector, the read-out circuitry is also significantly simpler. The digital STOP pulse (required by the TDC, as depicted by Figure 1.1) can be provided with a simple fast positive-feedback inverter gate connected directly to the SPAD as opposed to a more complex TIA.

1.4.3 Timing characteristics

Perhaps the key characteristic of the SPAD is the timing jitter, which quantifies the statistical fluctuation of the delay between the photon arrival time and the SPAD response. The timing jitter essentially defines the best possible achievable temporal resolution of the DToF LIDAR system containing the SPAD device. It is commonly expressed as the *Full Width at Half Maximum* (FWHM) of the statistical distribution of the SPAD reaction time. Most published SPAD implementations report under 100 ps FWHM jitter [17, p.7], and a FWHM timing jitter as low as 7.8 ps has been achieved in [18].

The timing uncertainty arises from several factors. According to [19, p.14], “the most important timing factor is whether the carrier is generated in the depletion region itself, or if it must diffuse into the depletion region”. This diffusion process follows exponential distribution. If and once the avalanche is started, the impact

ionization events follow Gaussian distribution (for example because of thermal vibrations etc.) and contributes some cumulative uncertainty [20, p.31], therefore it is important to detect the avalanche with the following circuitry as soon as possible to minimize the jitter accumulation.

1.4.4 Dead time, quenching circuits and afterpulsing

The time it takes for the diode to be quenched and recharged after detection is called the *dead time*, because during this time, the SPAD is unable to detect any incident photons.

If a passive quenching and recharging circuit as shown on Figure 1.7 is used, the quenching resistor R_Q has to be sized carefully. High values of the resistance speed up the quenching phase at the expense of the recharging phase and vice versa. There is an optimum value of resistance which minimizes the dead time, usually in the region of tens of $k\Omega$, but a dead time shorter than a few microseconds is impossible to achieve with resistor based passive quenching [21].

This is not necessarily a negative aspect, as having a quenching circuit which quenches and recharges the SPAD too quickly is undesirable because of an effect called *afterpulsing* [19, p.14]. Fabrication defects inside the PN junction can act as traps in the forbidden energy band, which become filled with charge carriers once the avalanche builds up. The relaxation time of these traps can be as long as tens of nanoseconds, depending on the quality of the fabricated SPAD, which places a minimum dead time constraint. For example, suppose the SPAD was quenched and recharged virtually instantly, i.e. well within a nanosecond. The excited traps would relax several nanoseconds later, releasing the trapped carriers back into the depletion region, starting another avalanche. This “false positive” afterpulse would be indistinguishable from avalanches caused by incident photos, acting as a time-correlated noise source.

Transistor current sources can be used for passive quenching as well, providing some level of control over the length of the phase, but thick oxide *Metal Oxide Semiconductor* (MOS) devices have to be used, as SPAD diodes are biased with relatively high voltage (10 V and more).

The current state of the art is commonly based on *Passive Quenching Active Recharge* (PQAR) circuits, an example of which from [22] is depicted on Figure 1.8. When the avalanche starts, the anode of the SPAD is rapidly pulled high, quenching the diode, and the inverter output goes low. The parasitic capacitance C_P , which represents the total capacitance of the devices connected to this node as well as the free charge carriers in the PN junction of the SPAD, is slowly discharged by the current I_{QCH} , which is smaller than I_Q , guaranteeing that afterpulsing cannot

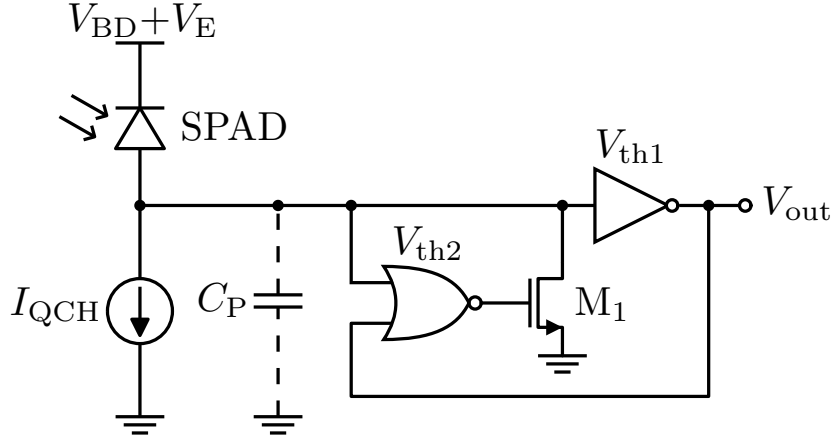


Fig. 1.8: Two-step PQAR circuit [22]

occur. This slow discharge phase continues until the voltage at the anode of the SPAD crosses V_{TH2} . At that point the NOR gate output goes high, turning M_1 on, which discharges the rest of the parasitic capacitance quickly, recharging the diode junction capacitance and completing the detection cycle as the inverter gate output goes low again, shutting M_1 off.

The advantage of this two-step PQAR circuit is the ability to control the dead time via I_{QCH} as necessary (low I_{QCH} safely prevents afterpulsing at the cost of longer dead time).

According to a survey of current art [23], the dead time of actively recharged SPADs usually ranges from 10 to 100 ns, and a 6 ns dead time has been achieved by [22]. It has to be noted that a large part of the decrease of dead time has been caused by the progress of SPAD manufacturing, as higher quality processes decreased the chance of afterpulsing dramatically and the recharge phase could be hastened [24]. This is demonstrated by [25], where a dead time of 8 ns was reached with an afterpulsing probability of 0.08%.

1.4.5 photon detection efficiency

To quantify the quality of a SPAD diode, its *photon detection efficiency* (PDE) can be evaluated. The PDE is defined as a product of the geometric *fill factor* (FF) and the *photon detection probability* (PDP), which is, in turn, a product of *avalanche probability* (AP) and *quantum efficiency* (QE) of the SPAD [24]:

$$PDE = FF \cdot PDP = FF \cdot AP \cdot QE \quad (1.3)$$

The fill factor is the ratio of the optically sensitive area of the diode and its total area. The quantum efficiency of a diode is a factor quantifying the probability of the photon penetrating into the silicon, being absorbed in the depletion region and

creating an electron-hole pair. However, even if an electron-hole pair is created in the correct area of the diode, it is still possible it will not create a self-sustaining avalanche due to statistical effects (thermal vibrations of the crystalline lattice etc.). This is quantified by the avalanche probability.

The fill factor is determined by the physical structure of the diode, its quenching and recharge circuits and the design of the pixel array. Usually, the larger the individual SPAD, the higher the FF . More complex quenching and recharging circuits or in-pixel TDCs take more space, reducing FF .

The avalanche probability is dependent on V_E . The higher the overvoltage, the higher the probability of a successful avalanche, as the electron-hole pair is accelerated by a higher electric field intensity. It is possible to reach AP of 1 with sufficient overvoltage. High AP however comes with a cost, as it also increases the possibility of undesired avalanches caused by crosstalk or the dark current.

Finally, the quantum efficiency of the diode is strongly dependent on the wavelength of the incident photon. For silicon SPAD, maximum PDE is usually reached for wavelengths between 400 to 500 nm, and less than 10% PDE is common for 900 nm photons, as seen on Figure 1.9.

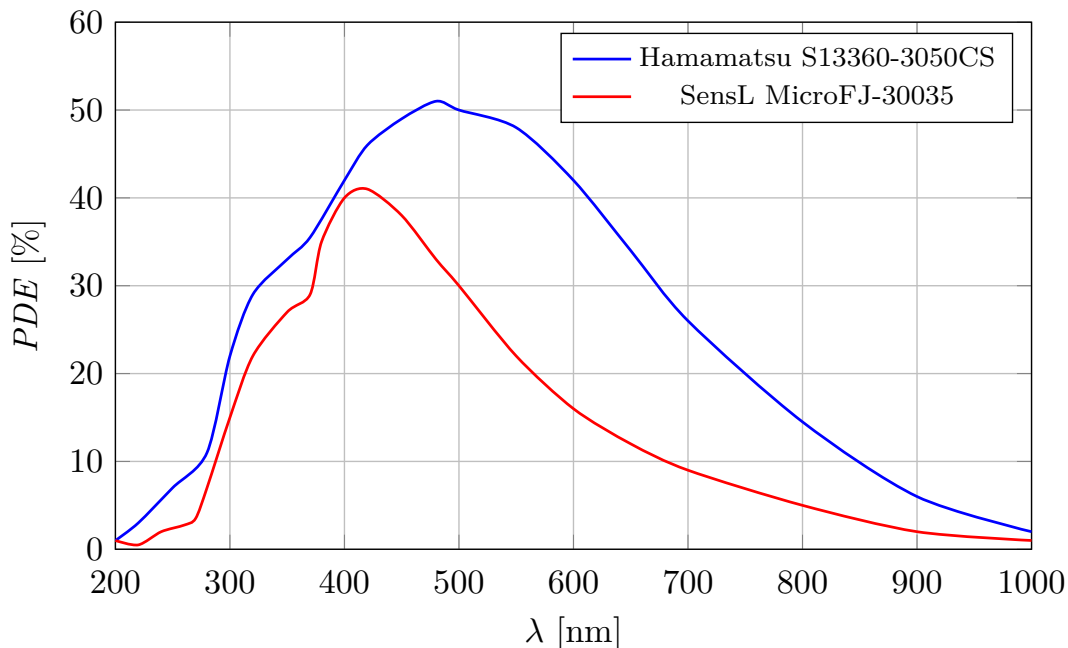


Fig. 1.9: PDE as a function of incident photon wavelength for two commercial SPAD based Silicon Photomultipliers, biased at $AP = 0.9$ [27]

The reason for this inefficiency with longer wavelengths is that the PN junction of the SPAD in common CMOS processes is manufactured as P^+/N -well junctions, which are at maximum only a few micrometers deep (depending on the process node) [17]. Photons with wavelengths longer than 790 nm have mean absorption depth in

silicon over $10\ \mu\text{m}$ (see Figure 1.10) and therefore simply pass through the depletion region or through the whole die most of the time.

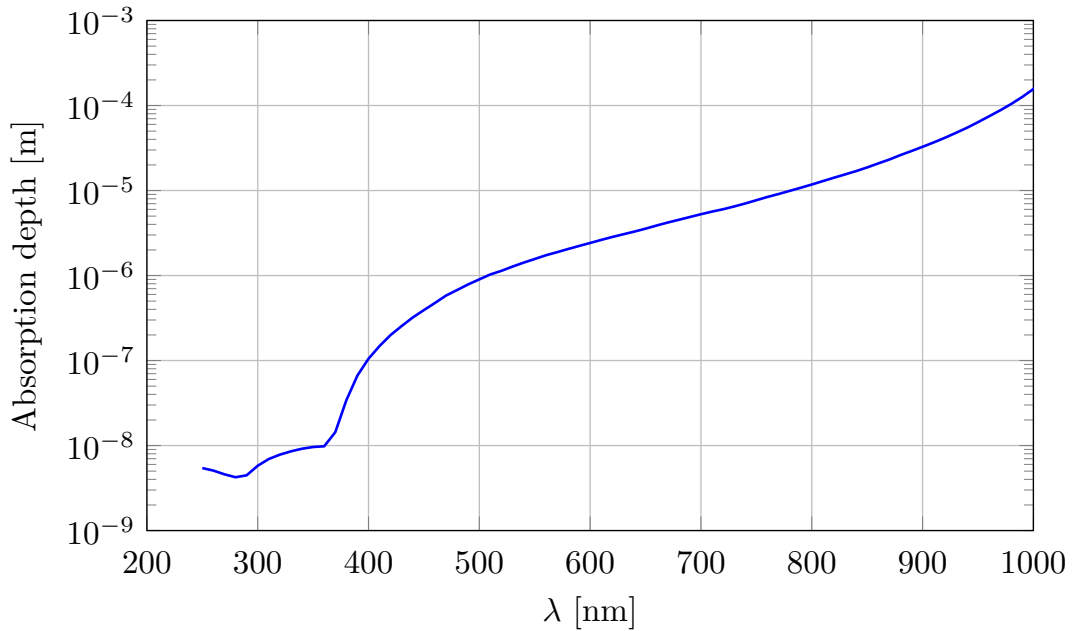


Fig. 1.10: Mean absorption depth of photons in silicon at 300 K as a function of wavelength [28]

To increase the depth of the depletion region, custom thick epitaxial layers were used in [26] to achieve PDE of 18% at 850 nm. Nearly 9% PDE at 900 nm was reached by [25] with the use of *Backside Illumination* (BSI) technology. More expensive materials such as InGaAs can be used for SPADs even more sensitive to NIR and *Infrared* (IR) light due to their different absorption coefficient characteristics, increasing the SNR and therefore the range of the LIDAR system, but this is not an economically viable option for many commercial products.

1.4.6 Dark count rate and crosstalk

Dark current is a well known phenomenon in photodetectors and image sensors and is defined as the current flowing through the detector when no photons are entering it. In relation to SPADs, the term *dark count rate* (DCR) is more commonly used, as SPADs are essentially digital detectors, and the unit of DCR is cps (counts per second).

Dark counts are avalanches triggered even when no incident photons are hitting the SPAD. There are two main causes of these avalanches: thermal generation and tunnelling.

Thermal generation occurs in any semiconductor even without any external forces applied, as it is caused simply by the energy of the crystalline lattice thermal vibrations. These vibrations can generate electron-hole pairs on their own, but most commonly via trap-assisted generation, also known as Shockley-Read-Hall generation, where the traps are formed by fabrication defects in the crystalline structure. As soon as there is an electron-hole pair in the depletion region, an avalanche can be started, producing a dark count. In a SPAD array, the pixels with traps can produce significantly more dark counts than the others, compromising the quality of the image. Due to the strong thermal dependency of this effect (DCR doubles for every 5 to 7°C of temperature increase [24]), LIDAR devices usually provide better image quality when cooled.

The second mechanism is tunnelling, which is dependent on electric field intensity. Strong fields are created when the PN junction is narrow because of high doping concentrations of the PN region. In that case, the energy barrier stopping the carriers from jumping from the valence band on one side of the junction to the conduction band on the other side is narrow as well and can be crossed via quantum tunnelling.

If the DCR of a SPAD is only weakly temperature dependent, it is clear that the tunnelling effect dominates, which is not optimal, as tunnelling can be minimized by better physical design of the SPAD device. Both causes of DCR are naturally proportional to V_E as well.

While DCR represents temporally uncorrelated noise, crosstalk between neighbouring SPADs acts as a correlated noise source (which is more difficult to filter out with DSP). Electrical crosstalk can occur when the electron-hole pair is generated deep within one device (beyond the depletion region, i.e. in the substrate), as it is possible for them to diffuse sideways and trigger an avalanche in a neighbouring SPAD [20, p.32]. It can also be triggered by other types of substrate noise or power supply noise [19, p.16]. This type of crosstalk can be reduced by substrate isolation, such as deep N-wells or P-wells, but this reduces the PDP at longer wavelengths.

Optical crosstalk is caused by photons created during an avalanche breakdown of the SPAD. Photons are emitted via electro-luminescence and can travel laterally, triggering an adjacent device. Optical crosstalk can be reduced by limiting the avalanche current, increasing the distance between the SPADs (the array pixel pitch) or by separating the neighbouring SPADs with deep trench isolation [20, p.33].

1.4.7 SPAD imagers and Silicon Photomultipliers

As was explained in section 1.2, *arrays* of detectors are used in DToF LIDAR systems so that high spatial resolution and/or high frame rate is achieved. There are two

possible ways of creating SPAD detector arrays. They do not differ in the connection of the SPADs themselves, but rather in what terminal or terminals do they consider the output of the array.

The first method does not have a specific widespread name in literature, and will be called simply “SPAD imager”. This is simply an array of SPADs and their quenching circuits, sharing the ground and the high voltage supply line, as depicted on Figure 1.7 or Figure 1.8. Each individual SPAD has its own digital output which is routed separately.

The other possibility contains SPADs connected together in the same way, but the individual SPADs no longer have their own output terminals. Instead, the output signal is either the current flowing through this parallel array of SPADs, or the voltage at the so-called “fast output”. This array is called a *Silicon Photomultiplier* (SiPM) and is depicted on Figure 1.11.

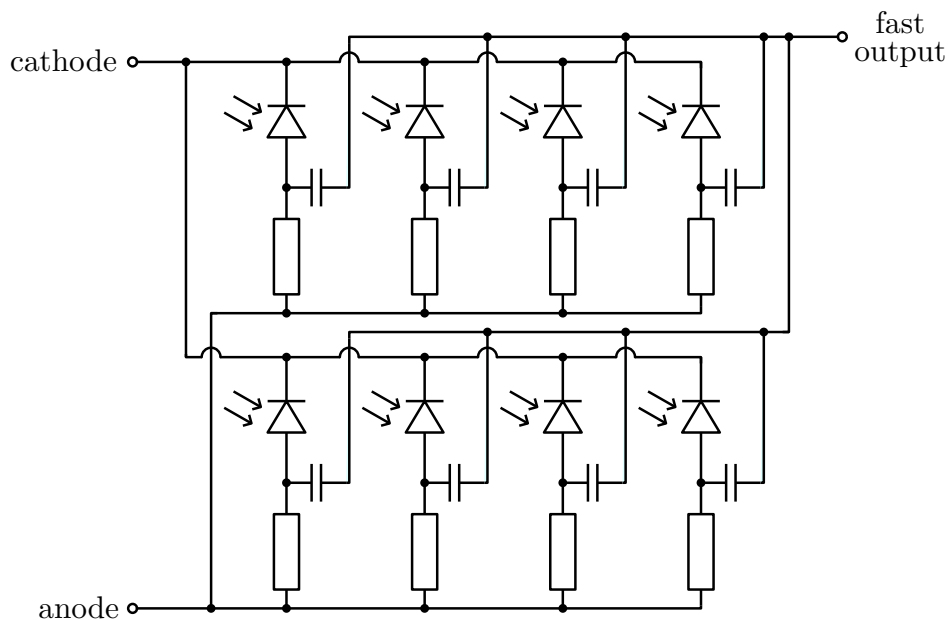


Fig. 1.11: Simplified SiPM schematic

By connecting the SPADs together, the SiPM is no longer an imaging device and does not resolve spatial information. Instead, by summing the current from all the SPADs, it provides analog representation of immediate incident optical signal intensity (which individual SPADs cannot do), and depending on the number of pixels, this representation is very linear to a limit (when the light intensity is high enough, the SPADs are mostly on all the time and the SiPM is saturated). The loss of spatial information makes SiPMs incompatible with flash LIDAR, but viable for scanning LIDAR, which only illuminates a section of the FoV at any given time.

While it is possible to restore some sense of spatial resolution by making an array of SiPMs, as demonstrated by [29], there are limits to this technique due to scale. If a resolution on the order of 100×100 pixels is required, and each pixel actually

consists of a SiPM with at least a few tens of individual SPADs, the number of devices can easily reach hundreds of thousands, which is very problematic due to cost, die size and yield of the fabrication process.

To convert the analog current signal of a SiPM to a voltage, a TIA is used, such as on Figure 1.12.

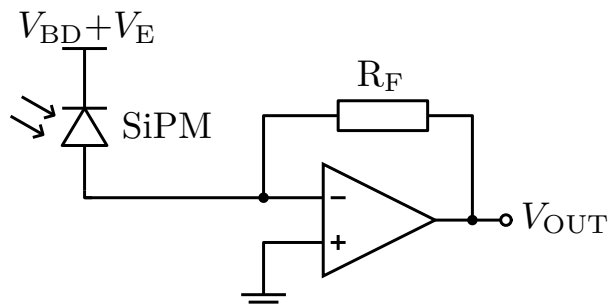


Fig. 1.12: Standard SiPM readout with a TIA

The problem with this type of analog readout is the capacitance of large SiPM arrays. The individual SPADs which are turned on and the ones which are turned off act as a capacitive divider, and further parasitic capacitance is provided by the metal routing [30]. The signal the TIA needs to amplify is therefore very small. If the picosecond level temporal resolution of the SPAD is to be preserved, the bandwidth of the TIA needs to be in the order of a few GHz at least, which combined with the mV-level input signal (high precision, low noise requirement) is extremely challenging.

The prediction in the industry is that SPAD imagers will be the photodetectors of choice in the future [31, p.13]. They preserve spatial information, allow flash LIDAR and interface with fully or nearly fully digital front-end electronics, which makes them better suited for contemporary, primarily digital CMOS processing technologies. While SPAD imagers are inherently digital detectors and cannot provide analog intensity information, digital processing can partially restore it via photon counting (more details will be provided in the following section 1.5).

An example of a contemporary state-of-the-art SPAD imager is described in [32].

1.5 Time Correlated Single Photon Counting

Time Correlated Single Photon Counting (TCSPC) is a statistical technique, allowing the restoration of the photon arrival times with picosecond resolution.

Let us suppose that our DToF LIDAR device consists of a single SPAD diode. A single laser pulse is sent and a photon is detected by the SPAD. The time interval between the laser pulse start and the photon detection is converted by the TDC and a ToF data point is created. In an ideal system with no noise sources, external

effects, timing jitter etc. such a reading would be valid, however, in a real world, it is practically worthless. The detection in the SPAD could have been a dark count caused by a thermal generation or a tunnelling event. Alternatively, the photon entering the detector could have been a photon originating from the Sun or other ambient light sources. And even if the photon did come from the laser pulse reflected from a distant target, the timing jitter of the SPAD diode and the TDC could have produced a significant random uncertainty in the final timestamp.

To solve this problem, the TCSPC technique performs a large number of ToF measurements and builds a statistically significant histogram out of them. To increase the statistical sample, several laser pulse cycles are performed and an array of SPADs and TDCs is active at any given time. Assuming all the noise sources are temporally uncorrelated to the laser pulses (which might not always be a valid assumption, see subsection 1.4.6), the invalid readings act like white noise in the histogram, while the reflected photons pile up over time and produce peaks which can be discerned. An illustration of the technique is on Figure 1.13.

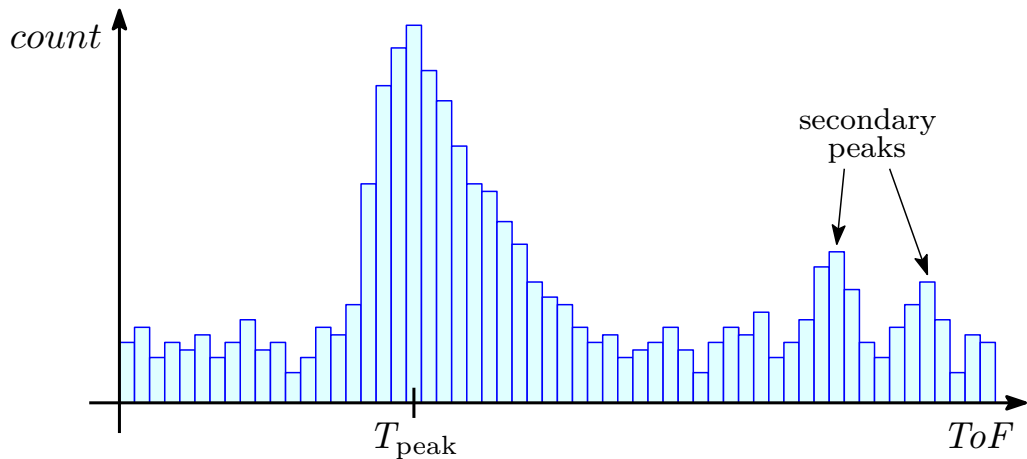


Fig. 1.13: Illustration of a TCSPC histogram

Usually the least computing-power intensive method of determining the correct ToF is finding the ToF corresponding to the highest peak in the histogram. Secondary peaks can also appear in the histogram, which is especially the case for Flash LIDAR systems where multiple objects are illuminated by the diffused laser pulse at a time. In some cases, it can be difficult to determine whether the secondary peaks correspond to real objects, which is why these systems frequently employ more complex and power and area intensive post-processing, such as *Finite Impulse Response* (FIR) filters applied to the histogram, using dynamic thresholds etc.

There is an obvious trade-off between the time spent acquiring the data for the histogram and the frame rate. Especially in applications where the LIDAR system is moving, it is important to build the histogram as soon as possible to prevent the peaks from smearing over time.

It has to be noted that while the histogram building logic does consume area and power, it is not a practical option to perform the histogram building elsewhere, i.e. on a separate microprocessor or an *Field Programmable Gate Array* (FPGA), as the amount of ToF conversions performed in larger arrays would require an extremely high bandwidth. For example, a 32×32 SPAD array with in-pixel 14-bit TDCs each producing conversions at 10 MS/s would produce 143 Gbit/s of raw ToF data in total. Building the histogram on-chip or on a nearby chip (3D chip stacking, multi-chip modules etc. can be used for great benefit in these applications) and only transmitting the histogram data or the estimated correct ToF makes the data rates much more manageable, saving power as well.

1.5.1 Coincidence counting

For LIDAR systems used outdoors, TCSPC might not be enough to produce an accurate ToF reading in a practical time period. While optical bandpass filters attenuating wavelengths outside the spectrum of the laser are commonly equipped in LIDAR systems, the disparity between the power of the laser and the solar ambient light can nevertheless prevent building a histogram with discernible peaks in time. When increasing the laser power, narrowing the optical filter or improving the photon collection efficiency are no longer an option, additional DSP techniques need to be employed.

Coincidence or concurrency counting is a noise rejection technique enhancing TCSPC. It takes advantage of the fact that reflected photons should be detected by the detector array at roughly the same time. A digital circuit called the discriminator or a coincidence detector is placed between the detector array and the TDCs and determines whether a sufficient amount of detections was detected in a defined time window. If the threshold condition is not met, the detections are ignored. If the number of detections is higher than the threshold, only then is the TDC activated and the result is added to the histogram.

An illustration of the technique is shown on Figure 1.14. Digital pulses produced by the SPADs are put out on the CH 1 to CH N bus. A hit is counted only when at least three detections are made within t_{window} , denoted by a green rectangle. Over a period of M laser pulse cycles, a histogram of successful hits is built, containing less noise and more easily discernible peaks in comparison to simple TCSPC.

In current literature, two main ways of implementing coincidence counting are common. The first method, used by [33], is to use a large OR tree to combine several SPAD output lines together, forming a unit called a macropixel. Whenever this combined line goes high, a timer is started. A detection is registered only if the subsequent pulses on the combined line cause a counter to reach its threshold before

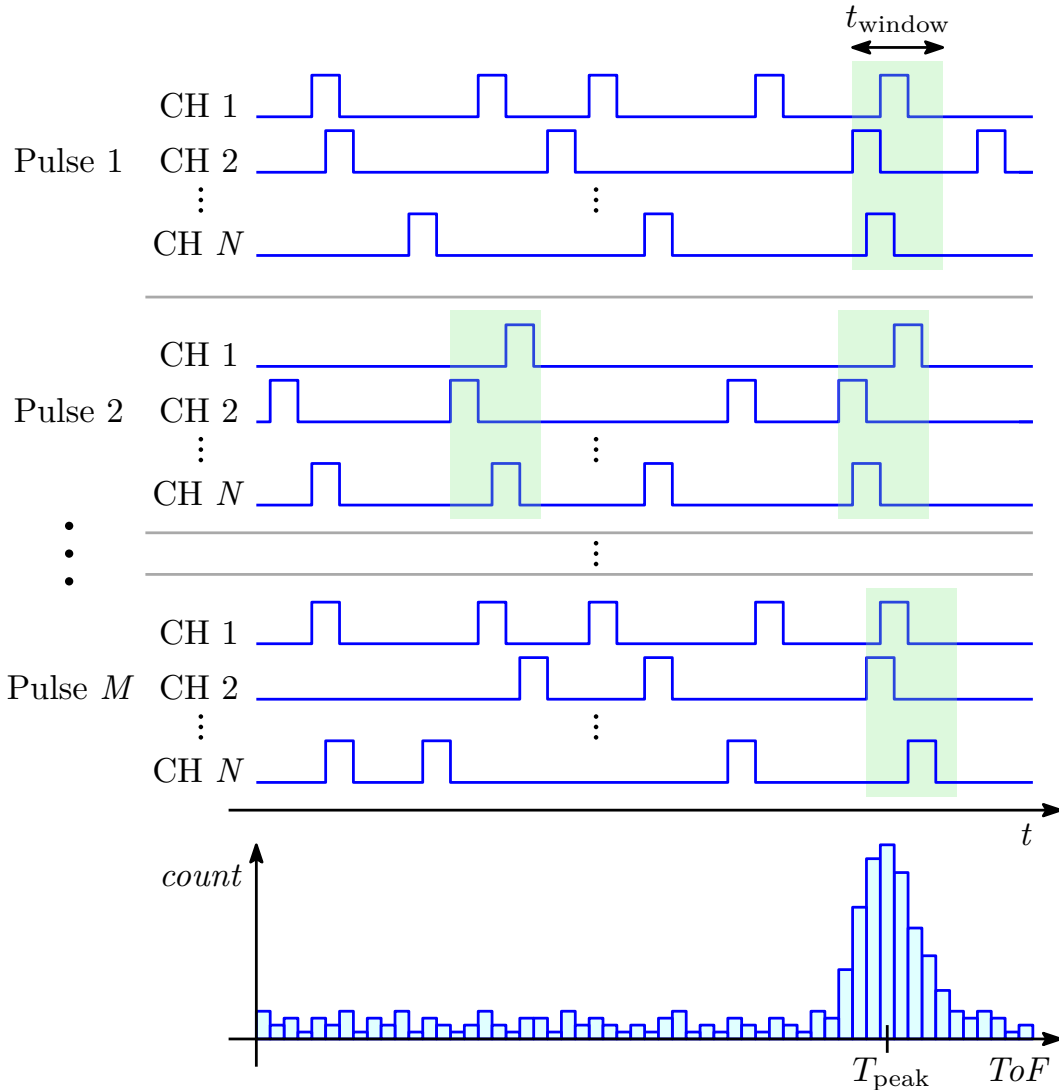


Fig. 1.14: Illustration of coincidence counting

the timer runs out.

A second method, employed by [34] or [2], is to shorten the SPAD pulses to a defined time (the coincidence window) and check their concurrency via combinational logic. In [34], a macropixel of 12 SPADs is checked for a concurrency of at least two pulses via a combinational logic network made from full adders, half adders and a large OR.

The advantage of the second method over the first one is that with D-latches and another adder, the number of concurrent detections is registered as well, which is a valuable piece of information for building the histogram in a time efficient manner. The first method loses the ability to count the number of concurrent pulses, as it combines the SPAD outputs into a single digital line and the pulses can overlap. This also decreases the spatial resolution of the array, as it is no longer determined by the size of an individual pixel, but rather the whole macropixel.

On the other hand, the second method requires a well balanced timing network for all the signal paths and is generally more area intensive and therefore harder to implement for larger SPAD arrays. Another advantage of the first method is its capability of changing the threshold digitally during the operation, possibly depending on the ambient light level, while the second method's concurrency threshold is fixed by the combinational logic design.

2 TIME TO DIGITAL CONVERTERS

Time to Digital Converters (TDCs) are a key part of the DToF LIDAR signal path. They do not only define the distance resolution and maximum measurable range. Just like *Analog to Digital Converters* (ADCs), they exhibit non-linearity, noise or a non-zero conversion time. And because modern DToF LIDAR detectors contain hundreds to thousands of pixels, the chip does not contain only one TDC, but commonly at least a few hundreds of them. Therefore, the TDCs represent a large part of the overall footprint and the power consumption. That is why the TDC design deserves special attention and a chapter of its own.

In this chapter, firstly, general requirements for the TDC will be discussed. Afterwards, three main types of TDCs will be presented. A section on *Delay Locked Loops* (DLLs) will be included as well, as these circuits are a crucial part of many popular TDC topologies today.

2.1 General considerations

In this section, general aspects of the TDC will be discussed. First, its full scale and resolution specifications, important for the topology of the TDC, will be described. A subsection detailing the overall architecture of the TDC array will follow. Finally, two subsection about more power efficient or linearity improving timing schemes will close this section.

2.1.1 Full scale and resolution

The Figure 2.1 plots the linear relationship between the distance to the obstacle and the time the photon travels to the obstacle and back, i.e. the ToF (see Equation 1.1).

For automotive applications, the desired maximum range usually lies in the order of a few hundred meters. This is the range required to be able to measure to distance to other cars on the highway, or to be able to brake in time in case an obstacle appears on the road.

On the other hand, the desired minimum resolution is on the order of a few centimetres or even sub-centimetre. This is not only because such LIDAR can be used as a parking assist, but also because modern ADAS do not use LIDAR data to measure the distance to obstacles or other cars only, but they can follow the time evolution of the 3D point cloud in order to gauge the velocity of the surrounding objects relative to the observer as well. For this velocity measuring ability, high resolution is beneficial.

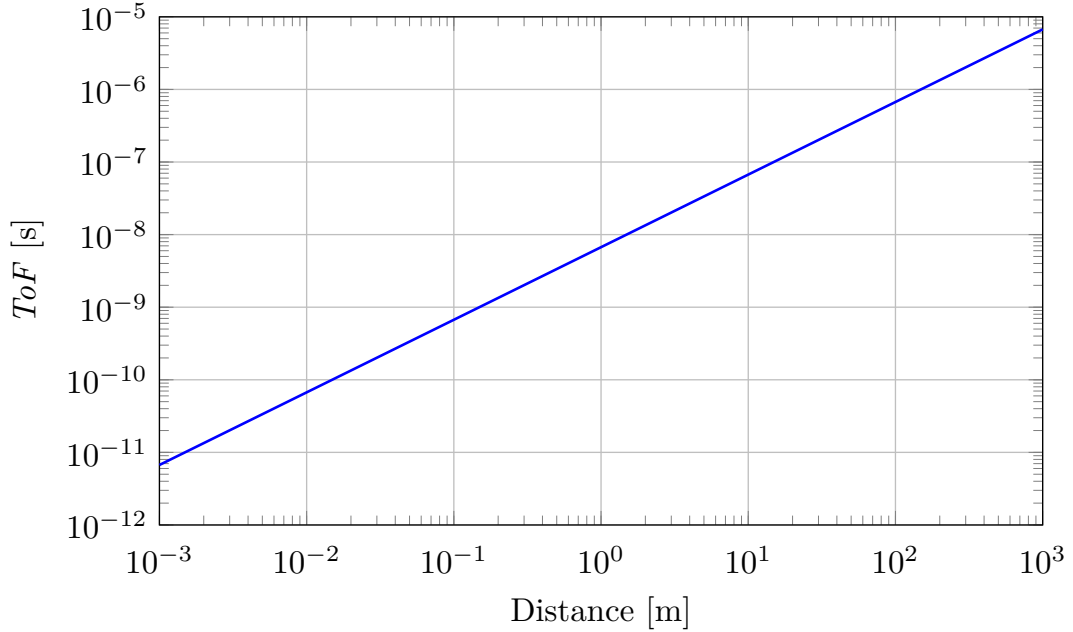


Fig. 2.1: Time of flight of a reflected photon as a function of distance to the obstacle

Assuming the maximum range required is 150 m with a resolution of 1 cm, the full scale of the TDC should correspond to 1 μ s with a resolution of 66 ps. This is a dynamic range of 83 dB, requiring a minimum of 14 bits.

Such a TDC is not trivial, and no single topology is capable of meeting such requirements in a power and area efficient manner. Instead, the problem is split into two smaller, more easily manageable parts: the so-called *Coarse Time to Digital Converter* (CTDC) and *Fine Time to Digital Converter* (FTDC). The CTDC measures time coarsely (i.e. with a low resolution), and determines the full scale range. The FTDC splits the *Least Significant Bit* (LSB) of the CTDC even more finely, achieving the required resolution of the overall TDC. The bits coming from the CTDC form the most significant bits of the overall TDC output bus, while the FTDC output bits are appended as the least significant bits.

By splitting the TDC into two distinct parts, optimal TDC topologies can be chosen for the CTDC and FTDC separately.

2.1.2 TDC array architecture

As was mentioned in the introduction to this chapter, due to the number of pixels in the detector array, having only one TDC is not viable. It could be theoretically feasible if the level of illumination of the detector was extremely low and the rate of incoming photons would be smaller than the conversion rate of the TDC, but for any daylight application, this is not the case.

There are two main approaches to the number of TDCs in the TDC array. Either

there could be one TDC for each pixel of the detector array (so-called in-pixel TDC), or there could be a smaller TDC bank, which will be somehow allocated to the detector array dynamically.

For flash LIDAR, it might seem that in-pixel TDCs is the optimal choice. While this approach makes sense, as there is a TDC available for every pixel at any time, and has been demonstrated in practice by [35] or [36], it has its shortcomings, which are mainly the comparatively large power consumption of the large number of concurrently running TDCs and the decrease of SPAD fill factor, as in-pixel TDCs take up pixel area.

Flash LIDAR using a TDC bank has been presented in [9], where 36288 pixels share a bank of 1728 TDCs. The dynamic allocation is done with a collision detection bus. This approach has allowed the authors to achieve a relatively high SPAD fill factor (28 %) on a single die, although they do not comment on the frame rate, the maximum allowable incoming photon rate and other related performance characteristics, which might potentially suffer, as the TDC allocation takes time which decreases the effective conversion rate.

Scanning or sequential flash LIDAR are more flexible. In-pixel TDCs can be implemented, but only small sections of the array can be activated at any given time, corresponding to the section being illuminated by the laser scanner. This decreases the power consumption significantly.

The inherent trade-off between SPAD fill factor and the in-pixel TDC area can be also solved with 3D stacking technologies [20, p.5], which allow the use of one die for SPADs only, while a second die includes all the readout and processing circuits. An advantage of this approach is not just the very high SPAD fill factor (approaching 100 %) but the fact that the two dies can be fabricated in different process technologies, optimizing the optical characteristics of SPAD and circuit characteristics for the digital CMOS die separately.

2.1.3 Reverse timing scheme

The START-STOP timing scheme described in section 1.1, where the START signal is produced by the laser trigger and the STOP signals are produced by the detector array, is intuitive and its implementation is straightforward.

However, let us suppose that there are 1000 TDCs in a TDC bank, all of which start counting the moment the laser is fired. Let us also suppose that only 300 SPADs detect a photon and stop their respective TDC before the time runs out and another laser pulse is fired. Clearly, 700 out of the 1000 TDCs did not produce any meaningful conversion, while still consuming power.

A reverse timing scheme fixes this issue. Instead of the TDCs being started by the laser trigger, they are started individually at various time instants by photon detections. The STOP signal is then synchronized to the next laser pulse. The equation for ToF for such a reversed timing scheme is no longer

$$ToF = t_{\text{STOP}} - t_{\text{START}} \quad (2.1)$$

but instead

$$ToF = T_{\text{cycle}} - (t_{\text{STOP}} - t_{\text{START}}) \quad (2.2)$$

where T_{cycle} is the period of the laser pulsing cycle.

This timing scheme is more energy efficient, as the TDCs spend most of the time in idle mode, waiting for the detection, and only consume significant power when producing valid conversions.

2.1.4 Sliding scale technique

If the reverse timing scheme (described in subsection 2.1.3) is used, the START signal is a pulse produced by a SPAD from the detector array, while the STOP signal is synchronized to the laser pulse trigger. Because the laser trigger is periodic, if the same ToF is measured (the same distance to an obstacle), the same part of the TDC range is exercised every single time. The measurements therefore have a systematic error due to the *integral non-linearity* (*INL*) at the given point of the transfer characteristic.

The sliding scale technique is a well known technique in ADC design, invented by Cottini et al. in 1963 [37]. Its purpose is to improve linearity at the cost of noise. A random but known analog noise is added to the converted signal and then subtracted in the digital domain. Even if the magnitude of the converted signal does not change, due to the noise, different parts of the converter range are utilized during repeated conversions. Therefore, the non-linearity can be averaged out over time.

The same method can be applied to TDCs. Sliding scale TDCs do not measure the time interval between the START and STOP signals, but instead measure the time intervals between the START and the reference signal, and the STOP and the reference signal. Because the reference signal is asynchronous with respect to the periodic STOP signal, different portions of the TDC range are utilized each measurement cycle and the non-linearity at various points of the transfer characteristic is converted into random noise of the resulting conversions.

An illustration of the sliding scale technique as applied to TDCs is depicted on Figure 2.2.

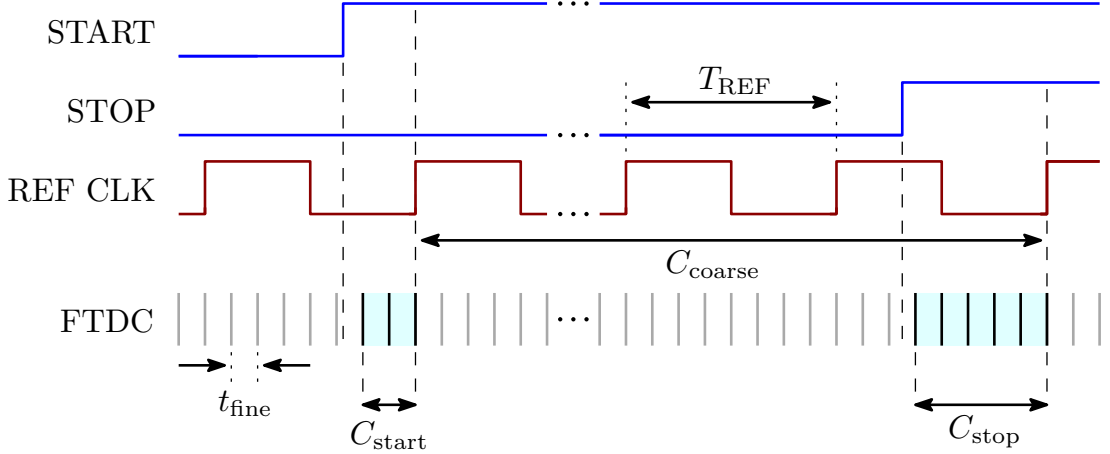


Fig. 2.2: Sliding scale TDC illustration

The illustration assumes that the TDC consists of coarse and fine TDCs, where the CTDC counts the number of reference clock periods from when START went high until STOP goes high. The depicted FTDC slices the reference clock period into 8 smaller slices (i.e. it is a 3-bit FTDC), and it counts the number of these slices since START went high until the first reference clock rising edge, and the same for the STOP signal. The final conversion result is therefore

$$T_{\text{result}} = T_{\text{REF}} \cdot C_{\text{coarse}} + t_{\text{fine}} \cdot (C_{\text{start}} - C_{\text{stop}}) \quad (2.3)$$

While this technique improves the linearity of the TDC, it only does so if repeated conversions of the same time interval are performed and averaged in the DSP. Moreover, the single shot precision is actually decreased, because there are two separate quantizing events, each contributing quantization noise. Finally, the technique also requires two FTDC copies in each TDC block. Although in the illustration on Figure 2.2 it might appear as if one FTDC is enough to count both C_{start} and C_{stop} , it is possible for the STOP rising edge to occur very shortly after the START rising edge, possibly even before C_{start} count is finished. In such a case, one FTDC would not be able to count both counts concurrently.

2.2 Counter based TDCs

Digital counters are the simplest TDCs imaginable and they are the implementation of choice for CTDCs. Their design is straightforward, they are power efficient and increasing their range is often as easy as adding another flip-flop.

On the other hand, due to non-zero setup and hold times, there is a maximum operating frequency they can reliably count at. This depends on the topology of the counter (synchronous/asynchronous), the standard cells used and the CMOS

processing technology itself. In either case, operating frequencies over a GHz and therefore resolution better than high hundreds of picoseconds are only possible in the most cutting edge processes.

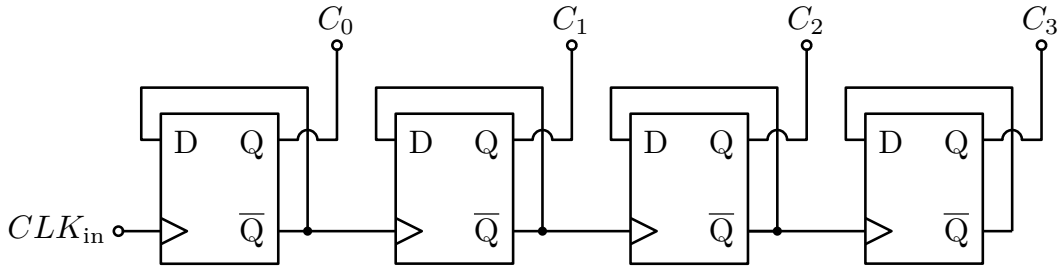


Fig. 2.3: Asynchronous ripple counter

A popular counter topology is the asynchronous ripple counter, depicted on Figure 2.3. Its main advantages are high operating frequency, low power consumption and small size, as it contains no combinational logic and scales extremely well. On the other hand, the counting signals count asynchronously and can be in invalid or metastable states until the counter settles. A synchronizing circuit, which is necessary to interface the counter with synchronous logic, might sample incorrect data which has not settled yet. A conversion to Gray code can be performed to decrease the size of the possible error, as with Gray code, the binary representation changes from one state to the next by flipping only one bit at a time.

Another option is to use a synchronous counter which counts in Gray code directly, as in [11, p.69]. Its maximum operating frequency is lower and it requires extra combinational logic and therefore has larger area and power consumption, but it does not require synchronization circuits.

2.3 Delay Locked Loops

Delay Locked Loops (DLLs) are a common part of popular FTDC topologies and it is necessary to provide a brief, qualitative description of their operation before proceeding.

DLLs share some commonalities with their better known counterparts, the *Phase Locked Loops* (PLLs). In comparison, DLLs are simpler, but also more limited in their application. They are mostly used for clock recovery, zero-delay clock buffers, multiphase clock generation or for *Process Voltage Temperature* (PVT) immune biasing of controllable *Delay Line Units* (DLUs). The last two applications are of interest for the purpose of implementing FTDCs.

The illustration on Figure 2.4 shows an example of a DLL. A clock signal is connected to a controllable delay line made up from four *Voltage Controlled Delay Line Units* (VCDLUs). Each VCDLU delays its input signal by t_d , which is variable.

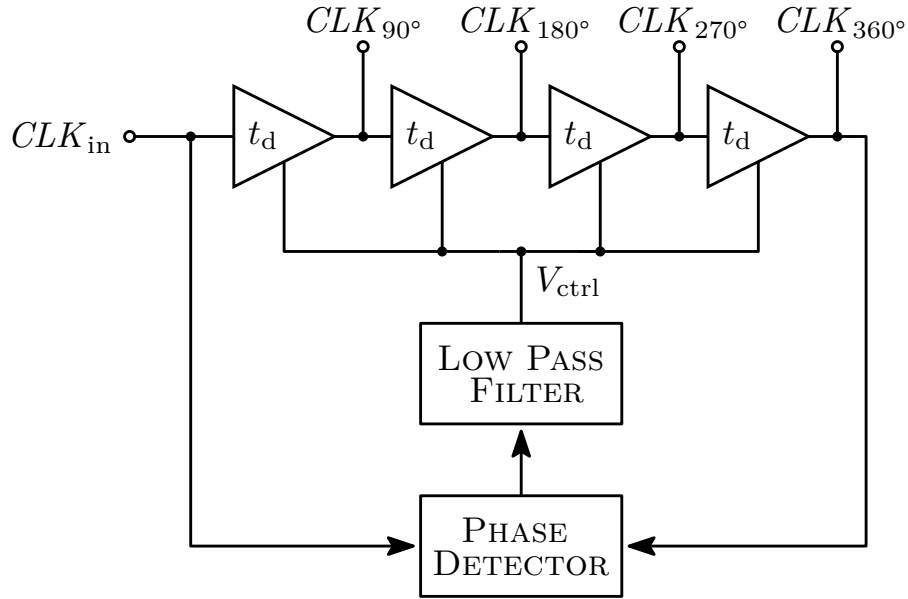


Fig. 2.4: Delay Locked Loop illustration

The output of the last VCDLU and the original clock input are both connected to a phase detector, which compares the phase shift between the two signals and produces a digital pulse proportional to the size of the phase shift at its output. The output of the phase detector is low pass filtered and applied to the VCDLUs as the control input. The negative feedback loop adjusts the delay of the VCDLUs, until the phase shift between the clock signal at the input and at the output of the delay line is 360° , i.e. the phase detector detects no difference. Since the delay line on Figure 2.4 consists of four elements, the intermediate signals of the delay line are phase shifted by $\frac{360^\circ}{4} = 90^\circ$. This particular DLL circuit therefore serves as a multiphase clock generator.

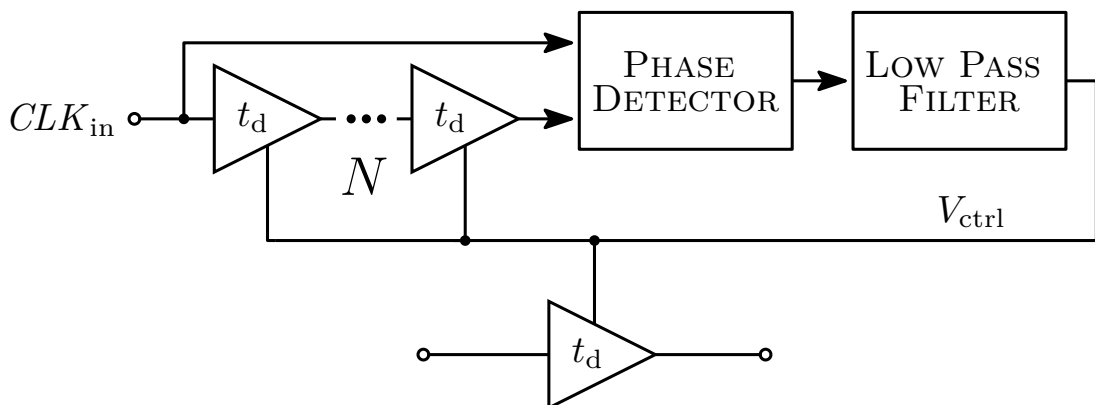


Fig. 2.5: DLL-assisted PVT immune VCDLU biasing

Another common use of DLLs in FTDCs is PVT immune DLU biasing, as shown on Figure 2.5. Assuming the loop is able to lock, the following equation can

be written

$$t_d = \frac{T_{\text{CLK}}}{N} = \frac{1}{f_{\text{CLK}} \cdot N} \quad (2.4)$$

and this relationship will be followed regardless of temperature, supply voltage or process skew. This method however requires high quality clock source and does not account for mismatch between the DLUs.

DLLs have a limited locking range. The limitation is determined by all three parts of the DLL: the delay line, the phase detector and the low pass filter. Firstly, for a DLL to be able to lock, the delay line needs to be able to produce a 2π phase shift for the given input clock frequency. However, it would be undesirable if the delay line was able to produce a phase shift of 4π or more, as in that case, it would be possible for the DLL to lock to the input signal delayed by two periods (instead of just one) by creating too large of a delay in the DLUs. This is called false locking and is problematic, as in such case, the DLUs present higher RC constants and attenuate the input clock, increasing jitter. More complex phase detector implementations or initialization of the DLUs at the lower end of their delay range can alleviate this [38, p.16]. Lastly, the bandwidth of the phase detector or the low pass filter can prevent locking to higher input clock frequencies, though that is something the designer should be able to avoid.

Contrary to a PLL, a DLL has only one pole, which is given by the filter (assuming first order approximation) [39, p.10]. It is therefore unconditionally stable. The commonly used filter is an integrator, as it guarantees zero error at DC. It also integrates well into the system. Phase detectors usually have two output bits: *UP* and *DWN*, which go active depending on whether the delay line is supposed to delay the input clock less or more. Therefore, implementing an integrating filter can be done with a simple charge pump, as presented on Figure 2.6.

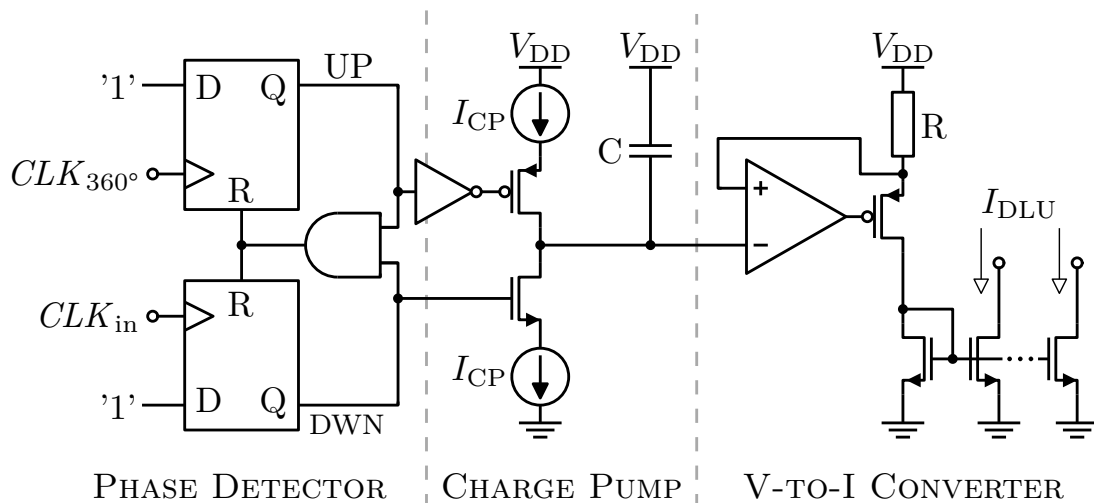


Fig. 2.6: Simplified charge pump based Delay Locked Loop control circuits

This particular illustration of DLL control circuits also depicts a V-to-I converter, used for driving *Current Controlled Delay Line Units* (CCDLUs). This implementation has its benefits, as replicating current with current mirrors and routing it to the point of use is simple and less sensitive to crosstalk, noise, *supply voltage* (V_{DD}) variation or IR drops than routing voltage signals.

2.3.1 Delay Line Unit circuits

There are many types of DLU implementations. DLUs can be voltage controlled (VCDLU) or current controlled (CCDLU) and the mechanism allowing the control of the propagation delay can be based on variable resistance, capacitance or current. A few typical DLU implementations will be presented.

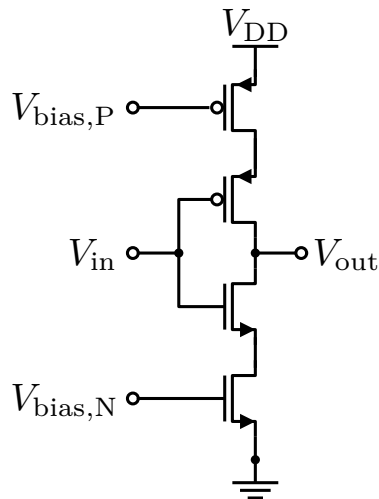


Fig. 2.7: Current starved inverter

A classic circuit called “the current starved inverter” is shown on Figure 2.7. The lowermost NMOS and the uppermost PMOS function as current sources, which can limit the current available to charge the output capacitance of the inverter. The biasing voltages determining the current are generated with current mirrors, which are not depicted.

A different approach is depicted on Figure 2.8. The control voltage adjusts the switching resistance of the NMOS connecting the lower PMOS diode to the output node. The PMOS diode diverts some of the current used to charge the output node capacitance to ground, therefore the higher the V_{ctrl} , the higher the delay. This VCDLU demonstrates high linearity of the $t_d = f(V_{ctrl})$ function (as long as $V_{ctrl} > V_{th,N}$), as proven by [40], where it was used to construct a 17 ps resolution TDC.

An alternative to the PMOS diode can be a MOS capacitor. In such implementations, increasing V_{ctrl} increases the effective capacitance at the output.

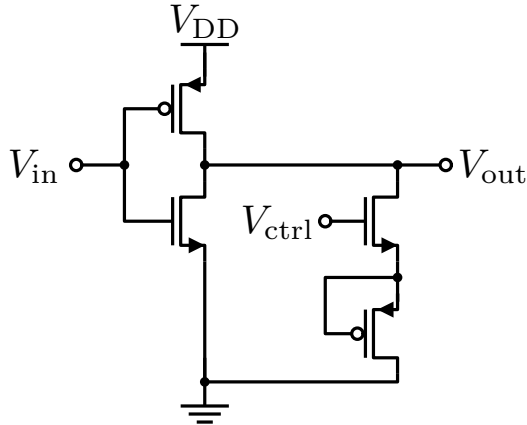


Fig. 2.8: High linearity Voltage Controlled Delay Line Unit

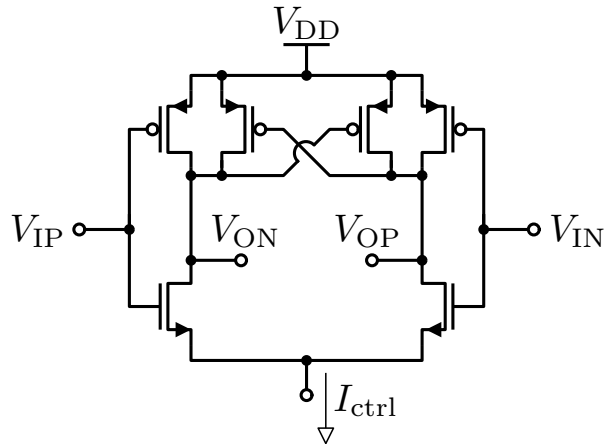


Fig. 2.9: Differential Current Controlled Delay Line Unit

Differential delay lines or differential ring oscillators offer some advantageous properties to single ended solutions, mainly in the area of common mode noise or power supply rejection. There are many options of implementing a differential delay unit, an example of which is on Figure 2.9 [41]. This cell consists of two inverters with additional positive feedback provided by the cross-connected PMOS transistors, which allows fast synchronized transitions. The delay of the cell is controlled by the current I_{ctrl} , therefore the principle is similar to the current starved inverter from Figure 2.7. Routing biasing currents over larger distances is beneficial to routing biasing voltages, as current signals are in general less sensitive to V_{DD} disturbances, noise, IR drops and other sources of error.

It has to be noted that the presence of the current sources in the current branch of both single-ended and differential current starved DLUs limits the input voltage range and different solutions might suit low voltage applications better.

2.4 Propagation delay based TDCs

Propagation delay based TDCs are the next step after counter based CTDCs, providing higher resolution not limited by setup and hold times of flip flops but by propagation delay of simple logic gates or DLUs. While the resolution of counter based CTDCs can reach higher hundreds of picoseconds at best, propagation delay based FTDCs can offer a LSB as small as tens of picoseconds.

The simplest propagation delay based TDC, known as tapped delay line TDC or flash TDC (due to its similarity to flash ADC), is shown on Figure 2.10.

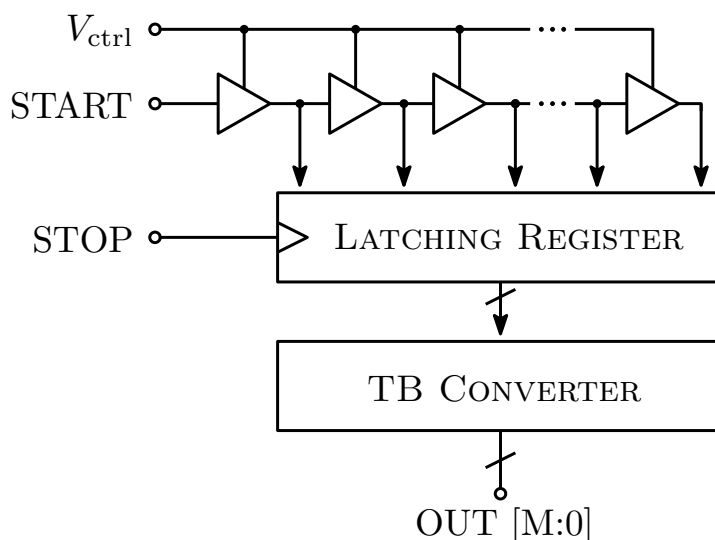


Fig. 2.10: Flash TDC illustration

The delay line, consisting of k DLUs, is initialized to zero and a START step is connected to its input. The step signal forces the output of the DLUs high as it gradually propagates through the delay line. When the STOP signal activates, the latching register latches the current state of the delay line. The number of active DLU outputs k_{on} corresponds to the time interval between the START and STOP signals t_{in} via $t_{in} = k_{on} \cdot t_d$, i.e. t_{out} is encoded in thermometric code, which can be decoded with a *Thermometric to Binary* (TB) converter.

The resolution of a flash TDC is equal to t_d , and the dynamic range is given by $t_{max} = k \cdot t_d$. The number of equivalent (binary encoded) bits is $\log_2(k + 1)$, i.e. to produce a 4-bit flash FTDC, 15 DLUs are necessary.

The precision of this FTDC is heavily dependent on t_d , which is why a DLL is commonly used to bias the DLUs, making their delays predictable and immune to PVT variation, as shown on Figure 2.5. Nevertheless, mismatch between the individual DLUs is still present, contributing to the *differential non-linearity* (DNL), respectively the *INL* of the FTDC.

Since the delay line outputs are asynchronous to the STOP signal, it is possible for the delay line transitions to not meet the setup and hold time conditions of the flip flops inside the latching register and cause a metastability at their output, which, once resolved, could settle into an incorrect state, producing a ± 1 LSB error. This would look like noise between consecutive measurements, whose noise power is related to the size of the setup and hold time interval. This is because the longer the setup and hold time, the higher the probability of the STOP signal violating them. It is therefore important to minimize the setup and hold time of the latching register, so that the noise contribution of metastability is comparable or smaller than the “naturally” present quantization noise.

A possible method of implementing the sliding scale technique (described in subsection 2.1.4) is shown on Figure 2.11.

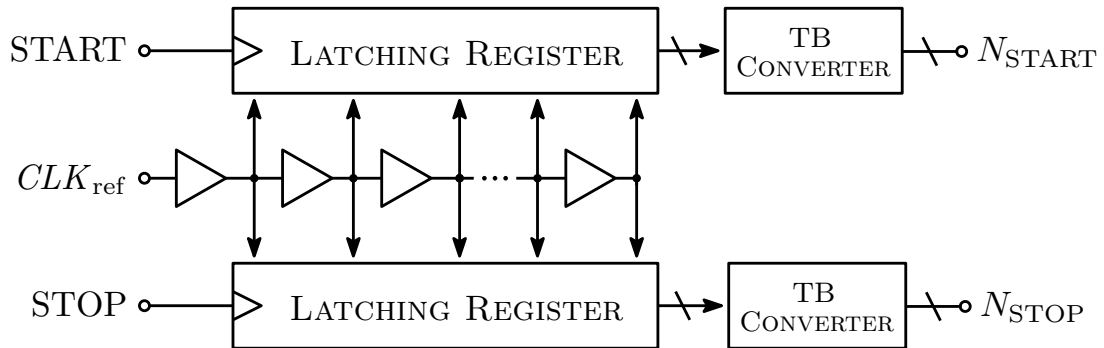


Fig. 2.11: Sliding scale flash TDC illustration

Instead of the START pulse being the signal which propagates through the delay line, a reference clock asynchronous to the START and STOP pulses is chosen instead. Both START and STOP pulses latch the state of the delay line at the moment of their assertion. The time interval between the pulses t_{int} is then calculated as $t_{\text{int}} = t_d \cdot (N_{\text{STOP}} - N_{\text{START}})$. This way, even if t_{int} is the same in each consecutive conversion, the sampled state of the delay line is different each time due to the asynchronism between the control pulses and the reference clock. The mismatch between the DLUs is converted into noise between conversions, which can be averaged.

An implementation of flash TDC integrating both CTDC and FTDC blocks is shown on Figure 2.12.

The delay line has been transformed to a gated ring oscillator. When EN goes high, the ring oscillator starts oscillating and the coarse counter starts counting, clocked by one of the ring oscillator output phases. As soon as EN goes low, the ring oscillator stops oscillating, the coarse counter is disabled and the latching register latches the current state of all the ring oscillator phases. The state of the ring oscillator can be decoded with a phase decoder to provide fine temporal resolution.

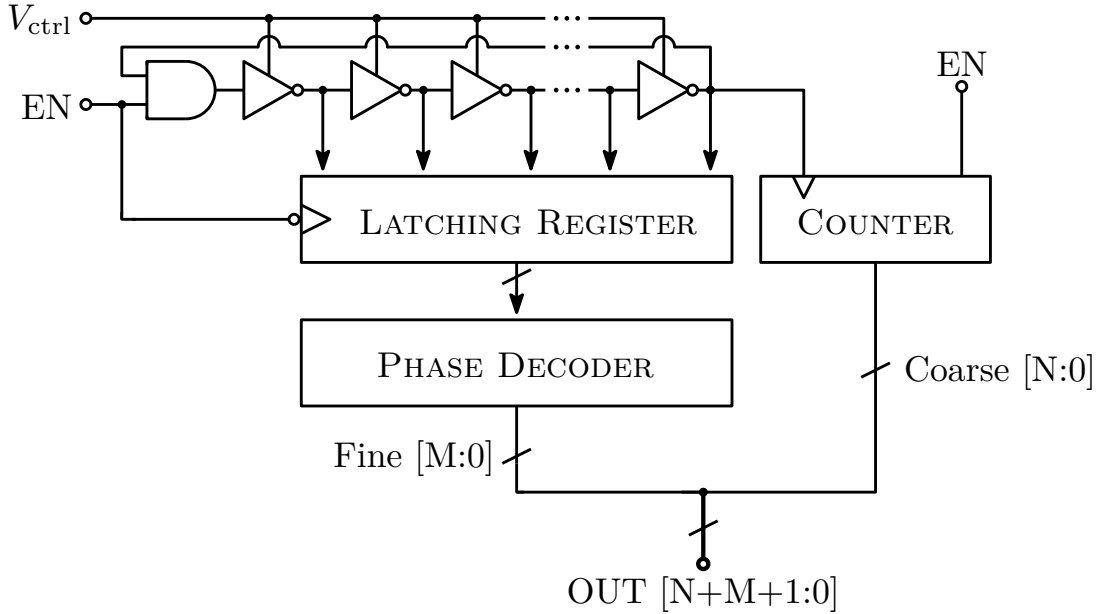


Fig. 2.12: Gated ring oscillator based flash FTDC with a counter based CTDC

The problem with using a single ended ring oscillator as a FTDC is that the number of unique states it can exist in during the oscillation period is $2k$, where k is the number of stages, which is an odd integer. Therefore, it is impossible to achieve a number of states equal to a power of two, which would be ideal for utilizing the whole binary bus. This issue can be solved with a differential ring oscillator, which can have an even amount of stages.

2.5 Sub-gate delay based TDCs

If the resolution provided by propagation delay based TDCs is not enough, so-called sub-gate delay based TDCs, also known as *Sub-Fine Time to Digital Converters* (SFTDCs), can be employed. SFTDCs provide temporal resolution better than low tens of picoseconds, and 1.2 ps resolution was achieved by [42]. There are multiple ways to implement a SFTDC, all of them however rely on the difference between DLU propagation delays.

A very popular SFTDC architecture is called the Vernier line, which is depicted on Figure 2.13.

In the Vernier line topology, two delay lines made from identical DLUs are constructed. Each delay line is however biased by a different control signal, i.e. their delays differ so that $t_{d1} > t_{d2}$. To achieve stability over PVT variation, the control signals are generated with DLLs (see Figure 2.5).

A START pulse propagates through the delay line with the longer delay t_{d1} . Later, a STOP pulse is sent through the faster delay line. Because of the difference

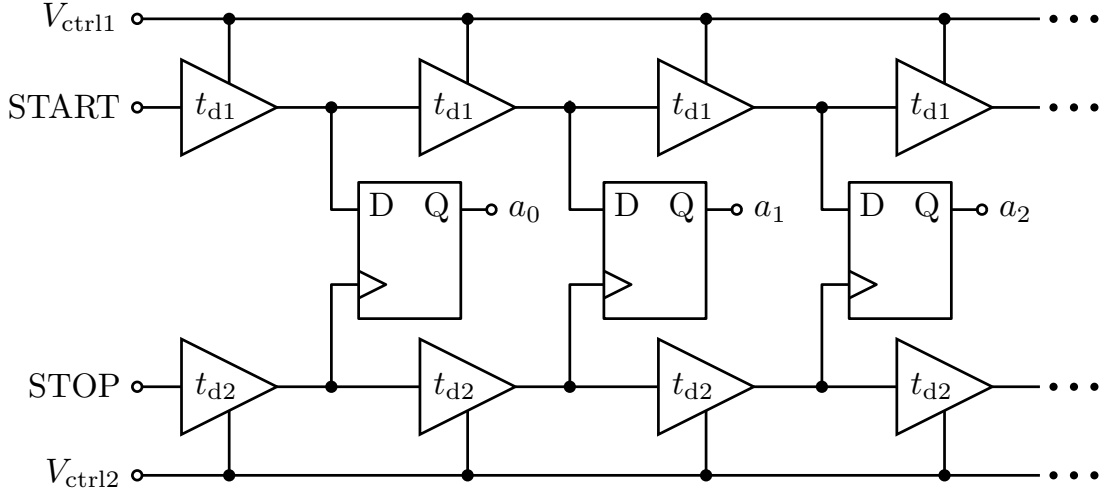


Fig. 2.13: Simplified Vernier line SFTDC schematic

between the two delays, the STOP pulse catches up to the START pulse eventually. The flip flops in-between the delay lines serve as arbiters – when the STOP pulse finally overtakes the START signal, the flip flop outputs of the following stages stay low. The bits a_0 to a_{k-1} are encoded in thermometer code, where the LSB corresponds to the temporal resolution of $t_{\text{dif}} = t_{d1} - t_{d2}$. A 30 ps resolution was achieved with a Vernier line based TDC by [43].

The dynamic range of the Vernier line SFTDC depends on the number of stages k via $t_{\text{max}} = k \cdot t_{\text{dif}}$, and the number of equivalent binary bits is $\log_2(k+1)$. However, the number of DLUs used is $2k$. This is a shortcoming of the architecture: to achieve dynamic range of four bits or more, the amount of DLUs required is rather large, contributing to large area occupation and power consumption.

A second issue of the Vernier line SFTDC is mismatch. Because the difference of propagation delays is so small, any mismatch between the delay cells contributes strongly to overall non-linearity. It also limits the minimum practical size of the LSB – the difference between t_{d1} and t_{d2} should be large enough to guarantee that the relationship $t_{d1} > t_{d2}$ always holds even under statistical variation between all DLU pairs.

A modified topology solving both of these issues is called the cyclic Vernier line, depicted on Figure 2.14 and presented in [40], [44] or [45].

Instead of using two lines of DLUs, the cyclic Vernier line folds the delay lines into single stage delay loops, i.e. ring oscillators. When START goes high, the coarse phase of the measurement cycle begins. The slower ring oscillator starts oscillating, incrementing the coarse counter N_S . As soon as STOP goes high, the fine measurement phase starts. The coarse counter is frozen, the slightly faster oscillator ($t_{d1} > t_{d2}$) starts oscillating and the fine counter N_F increments. A phase detector made up of the two flip flops and an AND gate activates as soon as the

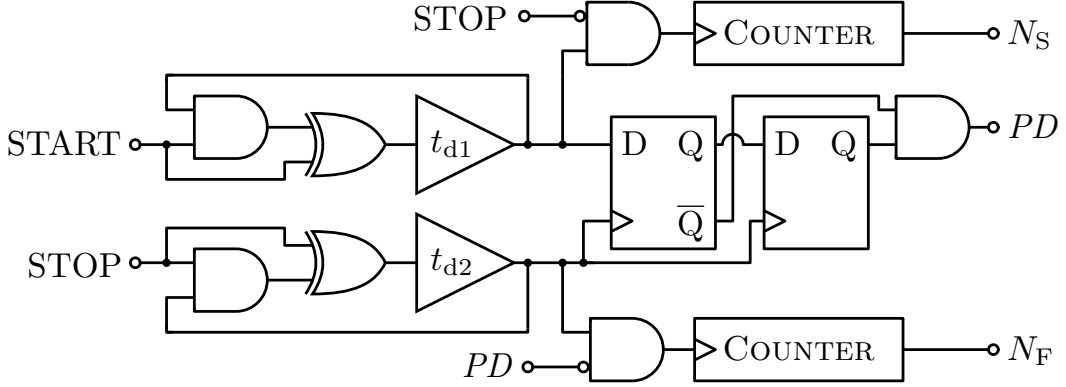


Fig. 2.14: Simplified cyclic Vernier line SFTDC schematic

faster oscillator overtakes the slower oscillator – at that point the fine counter stops counting as well and the conversion is complete.

The final conversion result is

$$t_{\text{out}} = N_S \cdot T_S + N_F \cdot (T_S - T_F) \quad (2.5)$$

where T_S is the period of the slower ring oscillator and T_F the period of the faster one [45, p.1512].

The advantage of the cyclic Vernier line SFTDC is that there is only one pair of DLUs which has to match. Any mismatch between the two does not translate to non-linearity, but a gain error instead, which can be calibrated [40, p.563]. Increasing the dynamic range of the coarse and sub-fine measurements is simply done by extending the counters.

A drawback of this TDC is that the fine measurement has a long conversion time. To resolve a time difference of $t_{d1} - t_{d2}$, one full period of the fast oscillator T_F needs to elapse. Moreover, the shorter the temporal resolution, the longer the conversion time of the fine measurement phase, as described by the following equation [46, p.44]:

$$t_{\text{conv, fine}} = (t_{\text{STOP}} - t_{\text{START}} - t_{\text{coarse}}) \cdot \frac{T_S}{t_{d1} - t_{d2}} \quad (2.6)$$

where t_{coarse} is equivalent to $N_S \cdot T_S$. The longer the residual time between t_{coarse} and the START-STOP time interval, the longer the fine measurement phase takes, i.e. the conversion time is dependent on the input signal level.

An implementation of a cyclic Vernier line TDC in 65 nm CMOS is presented in [44], where a resolution of 5.5 ps was achieved on die area as small as 0.006 mm². However, a lengthy (and therefore costly) calibration scheme had to be utilized.

There are other SFTDC topologies, most of which share either the drawbacks of the classic Vernier line (large area) or the cyclic Vernier line (long conversion time).

These are the cyclic pulse shrinking TDC [47], successive approximation TDC [42], pipelined TDC [48], noise shaping TDC [49] etc.

These drawbacks make these topologies unsuitable for DToF LIDAR applications requiring fast frame rates (short conversion time needed) and/or SPAD array based flash or sequential flash LIDAR (small area needed, as each SPAD has its own TDC). Picosecond level temporal resolution corresponds to millimetre or sub-millimetre distance resolution (see Figure 2.1), which are too short distances to be of interest in automotive LIDAR, and therefore for most applications, FTDCs are good enough.

2.6 Clock distribution schemes

It was already explained why DToF LIDAR systems contain multiple TDCs (hundreds or even low thousands), and why the TDCs consist of a CTDC and a FTDC (since SFTDC disadvantages outweigh the advantages for large pixel arrays and/or fast frame rates).

In section 2.4, two possible implementations of FTDCs were illustrated, both requiring N accurately phase shifted clock phases, where $N = 2^B$, B being the number of bits of the FTDC.

The clock phases needed for both the CTDC and FTDC have to be generated somewhere, and subsequently routed to the TDCs. Because of the size of the TDC array, the architecture of the clock distribution has big impact on the performance of the system, from power consumption to precision or uniformity.

2.6.1 Global counting

The first type of clocking scheme is to create the clock phases in one place of the chip and distribute them to all the blocks which require them. These schemes are called global counting. Their main advantage is the fact, that the clock signals received by one TDC should be in phase with the ones received by the others, as long as the routing is symmetrical.

The first, simplest example of a global counting scheme is shown on Figure 2.15, where all the clock phases are generated with a single DLL and distributed via a single large clock buffer.

The problem with this scheme are the parasitic capacitances and resistances. Firstly, assuming the TDC array is large, the length of the metal interconnects can be substantial. Long metal interconnects lead to large parasitics, which in turn lead to large dynamic power consumption via the well known CV^2f dependency. Since each interconnect is present N times, this can be a substantial amount of power.

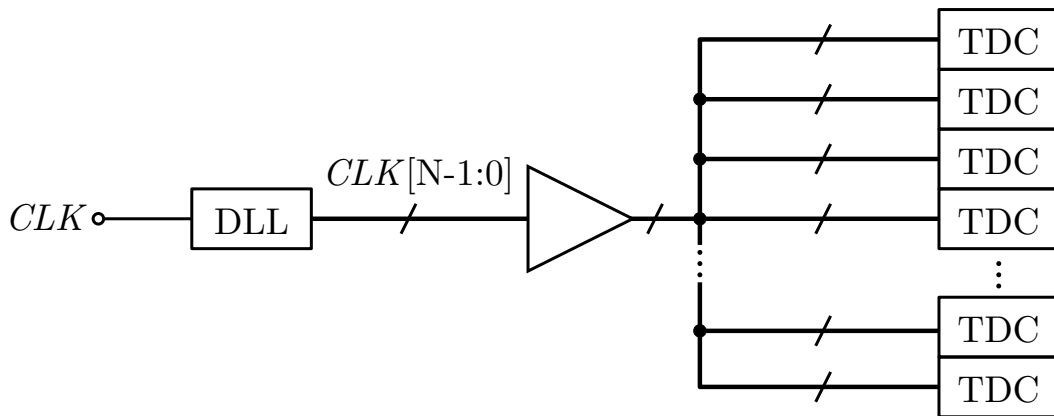


Fig. 2.15: Global clock counting with single buffer

Secondly, the large parasitics act as a distributed RC low-pass filter, slowing down the edges of the clock phases. This, along with interference and crosstalk picked up by the long interconnects, increases the jitter and therefore decreases the precision of the system.

The signal integrity issue can be solved with the multiple stage clock tree approach, as shown on Figure 2.16.

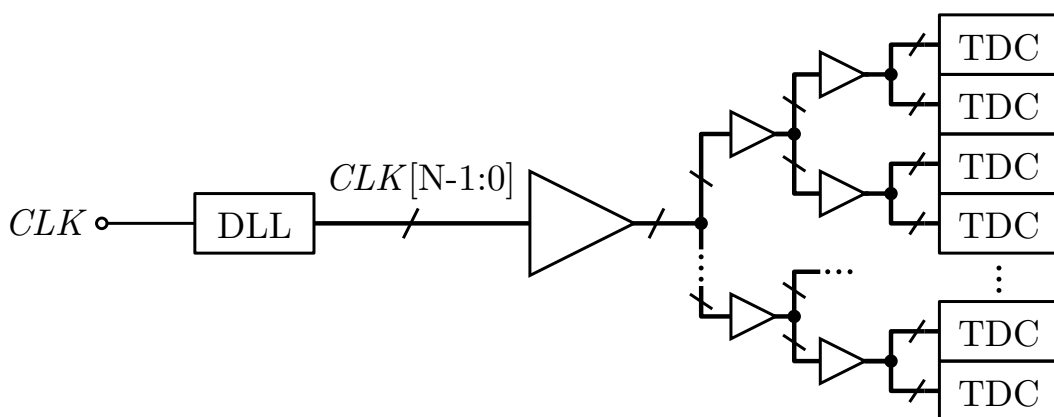


Fig. 2.16: Global clock counting with multiple stage clock tree

Instead of using a single clock buffer with high driving capability, a number of clock buffers is used, splitting the distribution path into segments. Each buffer restores the edge of the clock signal and decreases the loading of the previous buffer. The Figure 2.16 is just an illustration – the clock tree can be made from a higher number of stages, the last stage driving more than just two TDC etc. In practice, TDCs are often laid out in groups of four or eight.

On the other hand, this improvement is balanced by an additional increase of power consumption, since the number of clock buffers has increased significantly.

2.6.2 Local counting

Opposite to global counting stands the local counting method. Instead of generating the clock phases only once and routing them all over the chip, with local counting, the clock phases are generated in multiple blocks located at various places of the chip, each block distributing the phases to the nearest TDCs.

There are multiple possible ways of implementing local counting, one of them is shown on Figure 2.17.

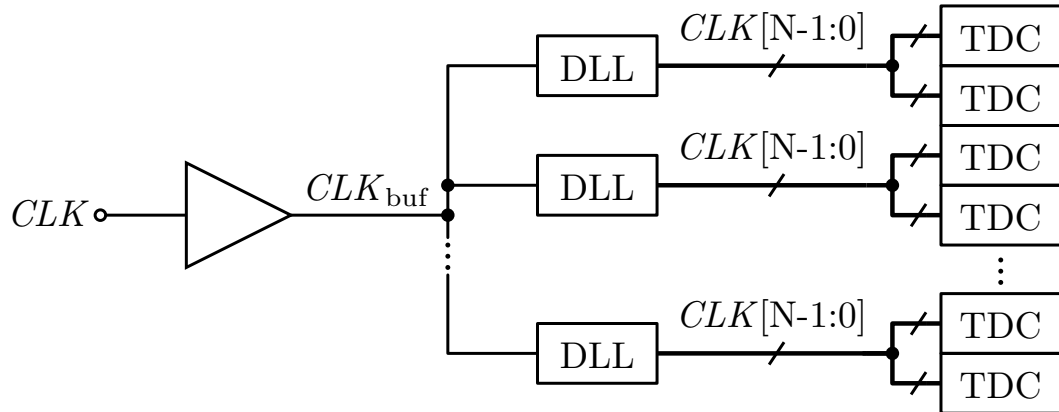


Fig. 2.17: Local clock counting with DLLs

This solution is only partly “local”, as each DLL which is generating all the required clock phases has to be driven by the global clock. Therefore the same problems as with global counting appear – the global clock needs to be distributed to the DLLs somehow. Since only one clock phase is routed this way, the parasitics are less of an issue than with global counting (especially for high N), but nevertheless, their contribution to the power consumption is not negligible. Furthermore, DLLs are not small blocks and take considerable amount of footprint, which is not ideal if there are a large number of them on the chip.

It has to be noted again that in practice, the size of the TDC group sharing clock sources is usually four or eight, although the pictures show only two member TDC groups for simplicity.

A truly local counting technique is shown on Figure 2.18.

In this scheme, the clock phases are generated in ring oscillators. A single, global DLL is used to bias the CCDLUs in the ring oscillators, guaranteeing PVT immune oscillation frequency. In order for this PVT stabilization to work, the DLL needs to be made from the same CCDLUs as the ring oscillators. The control signal routed to the ROs is current, because it is more immune to IR drops, noise or crosstalk. The current is essentially DC, therefore no dynamic power losses are generated.

Both of the local counting methods consume less power than the global counting schemes, as the interconnect parasitics issues are diminished, which is especially the

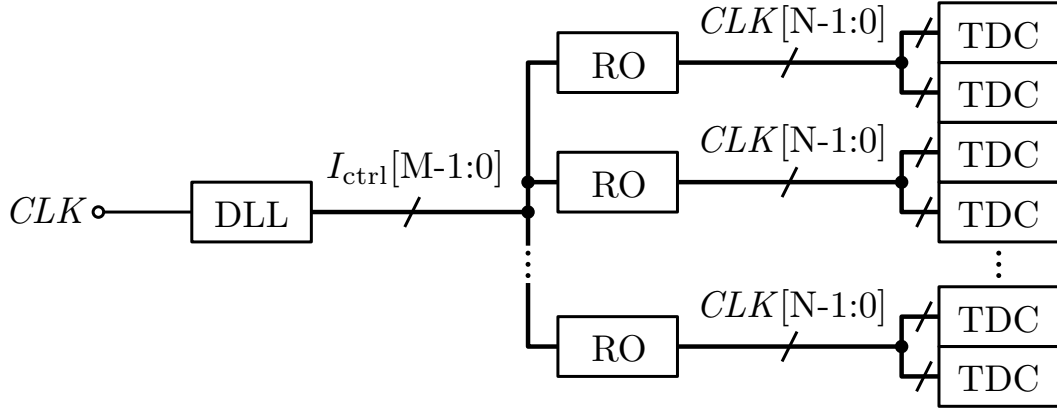


Fig. 2.18: Local clock counting with ring oscillators

case for the RO based local counting scheme. However, the local counting schemes also share the same problem with mismatch and phase asynchronism. While the PVT stabilization can compensate for global (shared) variations in process, voltage or temperature, mismatch between the DLLs or ROs can cause mismatch of the clock phases between the blocks, or even worse, as is the case with the RO based scheme, the oscillation frequency can vary between the ROs as well. The mismatch of the oscillation frequency can cause systematic error in the ToF measurements with the affected TDCs, and the phase shift between the clock signals can cause fixed pattern noise in the final depth map.

2.6.3 Summary

In this section, examples of both global and local counting schemes were presented.

The main advantage of the global counting clock distribution schemes is their simplicity and the synchronism, as all the TDCs should receive the same clock signals at any given instant, as long as the routing is symmetric. The main drawback, on the other hand, is the power consumption, as the clock distribution interconnect networks can possess large parasitic capacitance and resistance. Symmetric routing also takes a lot of area.

The local counting schemes, on the other hand, improve the power consumption at the cost of higher mismatch sensitivity and loss of synchronised clock phase inputs across the TDC array.

The ideal solution would be to implement the local counting scheme as shown on Figure 2.18, as it offers the best power efficiency, and add some power efficient mechanism capable of perfectly synchronising the oscillators. This can be achieved with the so-called injection locking technique, which will be described in the subsequent chapter 3.

3 INJECTION LOCKED OSCILLATORS

In this chapter, *Injection Locked Oscillators* (ILOs) will be described and analysed.

Firstly, a brief derivation of general oscillation principles and criteria will be presented. Afterwards, a closer look at harmonic LC tank based ILOs will follow, including phasor analysis and locking range derivation using its linear model.

A different, time domain based approach will be taken for the analysis of *Injection Locked Ring Oscillators* (ILROs), as they are highly non-linear.

Finally, ILRO based FTDC will be discussed, since its advantages are the main motivation of using ILROs in DToF LIDAR systems in the first place.

3.1 Oscillation criteria

A generalized feedback system consisting of a single block with a Laplace domain transfer function of $H(s)$ is shown on Figure 3.1.

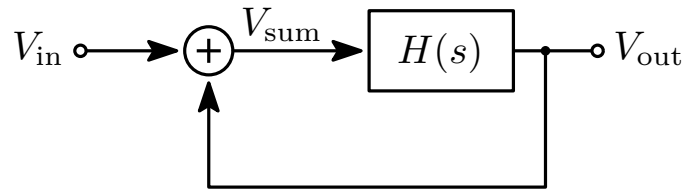


Fig. 3.1: General feedback system

The closed loop transfer function of the overall system can be derived

$$V_{\text{out}}(s) = H(s) \cdot V_{\text{sum}}(s) = H(s) \cdot [V_{\text{in}}(s) + V_{\text{out}}(s)] \quad (3.1)$$

$$V_{\text{out}}(s) = H(s) \cdot V_{\text{in}}(s) + H(s) \cdot V_{\text{out}}(s) \quad (3.2)$$

$$V_{\text{out}}(s) \cdot [1 - H(s)] = H(s) \cdot V_{\text{in}}(s) \quad (3.3)$$

$$\frac{V_{\text{out}}(s)}{V_{\text{in}}(s)} = \frac{H(s)}{1 - H(s)} \quad (3.4)$$

This is essentially identical to the well known Black's formula [50].

This chapter is devoted to oscillators, therefore, Equation 3.3 will be examined more closely. For oscillators, $V_{\text{in}}(s) = 0$, as the oscillator should be able to oscillate on its own without any external input signal. Therefore, we can write

$$V_{\text{out}}(s) \cdot [1 - H(s)] = 0 \quad (3.5)$$

This equation has two distinct solutions. The first one is $V_{\text{out}}(s) = 0$. This is not an interesting solution, as it only states that with the absence of any signal in the loop, the loop does not start oscillations on its own, whatever $H(s)$ may be. In physical reality, noise is always present and V_{out} is never exactly equal to zero.

The other solution, $H(s) = 1$, is the interesting one. One interpretation of it is that as long as $H(s) = 1$ for a given frequency ω_0 , $V_{\text{out}}(s)$ can be a non-zero signal indefinitely, i.e. oscillations can occur at the given frequency. To allow this, the following conditions need to be satisfied

$$|H(s)| = 1 \quad (3.6)$$

$$\angle H(s) = 2k\pi \quad (3.7)$$

where k is a non-negative integer.

These are the well known Barkhausen criteria. It has to be noted that these are *necessary*, but not *sufficient* criteria for sustained oscillations [51].

For reasons which will be explained in the following sections, the feedback system shown on Figure 3.2 will be analysed next.

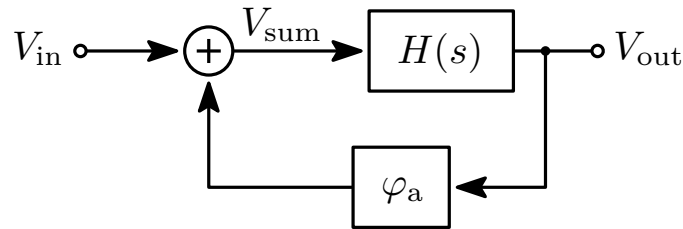


Fig. 3.2: General feedback system with additional phase shift

This system includes an additional phase shift of φ_a in the loop. The goal of the following analysis is to determine the way the additional phase shift affects the oscillation criteria. In the following derivation, $V_{\text{in}}(s) = 0$ will be assumed.

$$V_{\text{out}}(s) = H(s) \cdot V_{\text{sum}}(s) = H(s) \cdot V_{\text{out}}(s) \cdot e^{j\varphi_a} \quad (3.8)$$

$$V_{\text{out}}(s) \cdot [1 - H(s) \cdot e^{j\varphi_a}] = 0 \quad (3.9)$$

Equation 3.9 leads to similar conditions as previously, with a small adjustment.

$$|H(s)| = 1 \quad (3.10)$$

$$\angle H(s) + \varphi_a = 2k\pi \quad (3.11)$$

Finally, assuming the simplest case of $k = 0$, the phase condition for sustained oscillations can be rewritten as

$$\angle H(s) = -\varphi_a \quad (3.12)$$

To intuitively understand what this means for the oscillation frequency of the system, a plot of the phase response $\angle H(s)$ is useful. A generic plot of such a phase response is shown on Figure 3.3.

When a phase shift of φ_a is inserted into the feedback loop, the criteria for sustained oscillations change and they are no longer satisfied at the “natural” oscillation

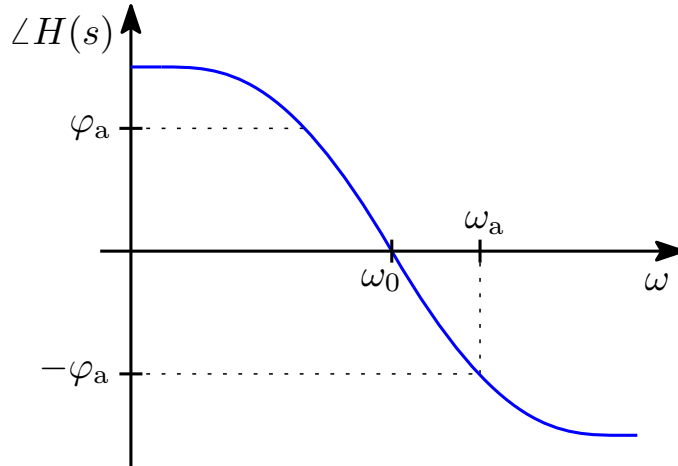


Fig. 3.3: Phase response of a general feedback system with additional phase shift

frequency ω_0 (where $\angle H(j\omega_0) = 0$, which would satisfy the original phase criterion from Equation 3.7). Instead, the condition is fulfilled at ω_a , where $\angle H(j\omega_a) = -\varphi_a$. In other words, an additional phase shift inserted into a closed loop oscillator shifts its oscillation frequency.

Injection Locked Oscillators are oscillators, whose oscillation frequency is locked to the frequency of an injected periodic signal. The injection mechanisms can differ in implementation, they are, however, always based on the previously described principle. The injected periodic signal somehow injects an additional phase shift into the feedback loop of the ILO, thereby changing its oscillation frequency.

3.2 LC tank based ILO

While the focus of this thesis is on ILROs, harmonic ILO analysis requires specific mathematical apparatus which provides different kind of insights, some of which are applicable to ILROs as well and which improve the general understanding of the injection locking phenomenon.

Since the seminal paper by Robert Adler from 1946 [52], LC tank based ILO has been used for explaining injection locking in most papers, theses and other publications on the topic, most notably [53] but also [54], [55], [56] etc.

To simplify the mathematical analysis, the circuit examined is an equivalent half circuit of a differential CMOS based negative resistance oscillator [57, p.21], as shown on Figure 3.4.

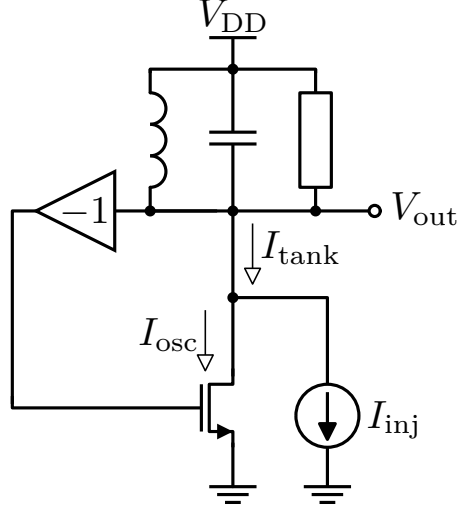


Fig. 3.4: Half circuit of a differential negative resistance LC tank based ILO

3.2.1 Phasor analysis

Without the injection current I_{inj} , the oscillator would oscillate at the resonant frequency of the LC tank

$$\omega_0 = \frac{1}{\sqrt{LC}} \quad (3.13)$$

sometimes also called the free-running frequency, and because a parallel RLC circuit contributes zero phase shift at its resonant frequency, $\angle V_{out} = \angle I_{tank}$.

Since the dynamics of injection pulling are very complex, let us now assume, that after connecting the harmonic injection current I_{inj} , the oscillator locks to its frequency ω_{inj} , where $\omega_{inj} \neq \omega_0$.

If the oscillator is locked, all its internal currents and voltages oscillate at ω_{inj} , including I_{tank} . However, because the phase response of a parallel RLC circuit is non-zero for frequencies $\omega \neq \omega_0$, we can write

$$\angle V_{out} = \angle I_{tank} + \varphi \quad (3.14)$$

where φ is the phase shift of the parallel RLC circuit at ω_{inj} .

Furthermore, assuming the negative resistance represented by the ideal inverting buffer and the NMOS device contributes no phase shift, we can also write

$$\angle V_{out} = \angle I_{osc} \quad (3.15)$$

Finally, simple nodal analysis provides

$$I_{tank} = I_{osc} + I_{inj} \quad (3.16)$$

Piecing these three equations together, it can be deduced that there is a non-zero phase shift between I_{osc} and I_{inj} . If there was no phase shift between the two

harmonic currents, I_{tank} would be in-phase with I_{osc} (because of Equation 3.16) and therefore with V_{osc} (via Equation 3.15), which we already know is not true from Equation 3.14. This knowledge allows us to draw a phasor diagram, as seen on Figure 3.5.

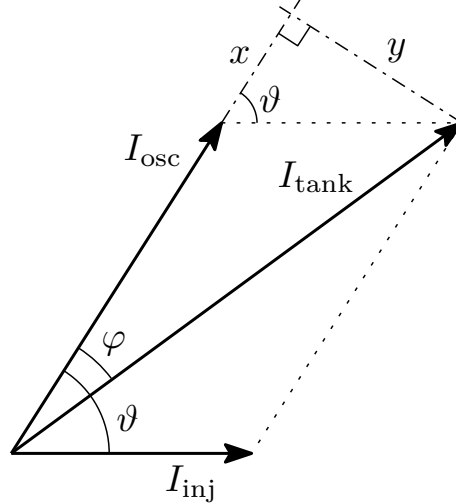


Fig. 3.5: LC tank ILO current phasor diagram

The angle between I_{osc} and I_{inj} , also known as *injection angle* and denoted as ϑ , reaches a value necessary to ensure that Equation 3.14 and Equation 3.16 are both satisfied.

To derive an equation for φ , a right triangle can be constructed by extending the I_{osc} phasor by x and constructing a line perpendicular to I_{osc} , connecting with the tip of the I_{tank} phasor. The length of this perpendicular line is y . Right triangle trigonometry leads to

$$\tan \varphi = \frac{y}{I_{\text{osc}} + x} \quad (3.17)$$

To derive the lengths of x and y in terms of known phasor lengths and angles, we can take advantage of the fact that the hypotenuse of the right triangle with legs x and y is equal to I_{inj} . Therefore

$$\sin \vartheta = \frac{y}{I_{\text{inj}}} \quad (3.18)$$

$$\cos \vartheta = \frac{x}{I_{\text{inj}}} \quad (3.19)$$

Combining with Equation 3.17 produces

$$\tan \varphi = \frac{I_{\text{inj}} \sin \vartheta}{I_{\text{osc}} + I_{\text{inj}} \cos \vartheta} = \frac{K \sin \vartheta}{1 + K \cos \vartheta} \quad (3.20)$$

where K is the so-called injection ratio, quantifying the strength of the injection current relative to the oscillator current, defined as

$$K = \frac{I_{\text{inj}}}{I_{\text{osc}}} \quad (3.21)$$

3.2.2 Locking range derivation

According to [53, p.1418], the phase shift of a parallel RLC circuit in the vicinity of ω_0 can be approximated by

$$\tan \varphi = \frac{2Q}{\omega_0}(\omega_0 - \omega_{\text{osc}}) \quad (3.22)$$

where Q is the quality factor, equal to $\frac{R}{\omega_0 L}$ for a parallel RLC circuit, and ω_{osc} is the frequency the RLC circuit currently oscillates at.

Combining Equation 3.22 and Equation 3.20 produces

$$\frac{2Q}{\omega_0}(\omega_0 - \omega_{\text{osc}}) = \frac{K \sin \vartheta}{1 + K \cos \vartheta} \quad (3.23)$$

$$\frac{2Q}{\omega_0} [(\omega_0 - \omega_{\text{inj}}) - (\omega_{\text{osc}} - \omega_{\text{inj}})] = \frac{K \sin \vartheta}{1 + K \cos \vartheta} \quad (3.24)$$

where $(\omega_0 - \omega_{\text{inj}}) = \Delta\omega_0$ is the difference between the free-running and injection frequency and $(\omega_{\text{osc}} - \omega_{\text{inj}})$ is the difference between the instantaneous frequency of the oscillations ω_{osc} and the injection frequency. Since instantaneous frequency is defined as the time derivative of phase, and the phase shift between the oscillations and the injection is ϑ , we can write

$$\frac{2Q}{\omega_0} \left(\Delta\omega_0 - \frac{\partial \vartheta}{\partial t} \right) = \frac{K \sin \vartheta}{1 + K \cos \vartheta} \quad (3.25)$$

$$\frac{\partial \vartheta}{\partial t} = \frac{\omega_0}{2Q} \cdot \frac{K \sin \vartheta}{1 + K \cos \vartheta} + \Delta\omega_0 \quad (3.26)$$

In the state of injection lock, $\omega_{\text{osc}} = \omega_{\text{inj}}$, therefore $\frac{\partial \vartheta}{\partial t} = 0$.

$$\Delta\omega_0 = \frac{\omega_0}{2Q} \cdot \frac{K \sin \vartheta}{1 + K \cos \vartheta} \quad (3.27)$$

To determine the minima and maxima of $\Delta\omega_0$, the function needs to be differentiated with respect to ϑ , the only variable which is not determined by design.

$$\begin{aligned} \frac{\partial \Delta\omega_0}{\partial \vartheta} &= \frac{\partial \left(\frac{\omega_0}{2Q} \cdot \frac{K \sin \vartheta}{1 + K \cos \vartheta} \right)}{\partial \vartheta} = \frac{K\omega_0}{2Q} \cdot \frac{\partial \left(\frac{\sin \vartheta}{1 + K \cos \vartheta} \right)}{\partial \vartheta} \\ &= \frac{K\omega_0}{2Q} \cdot \frac{\cos \vartheta \cdot (1 + K \cos \vartheta) - \sin \vartheta \cdot (-K \sin \vartheta)}{(1 + K \cos \vartheta)^2} \\ &= \frac{K\omega_0}{2Q} \cdot \frac{\cos \vartheta \cdot (1 + K \cos \vartheta) + K \sin^2 \vartheta}{(1 + K \cos \vartheta)^2} \end{aligned} \quad (3.28)$$

The minima and maxima can be found by finding the solution to $\frac{\partial \Delta\omega_0}{\partial \vartheta} = 0$. Therefore

$$\frac{K\omega_0}{2Q} \cdot \frac{\cos \vartheta \cdot (1 + K \cos \vartheta) + K \sin^2 \vartheta}{(1 + K \cos \vartheta)^2} = 0 \quad (3.29)$$

$$\frac{K\omega_0}{2Q} [\cos \vartheta \cdot (1 + K \cos \vartheta) + K \sin^2 \vartheta] = 0 \quad (3.30)$$

$$\frac{K\omega_0}{2Q} (\cos \vartheta + K \cos^2 \vartheta + K \sin^2 \vartheta) = 0 \quad (3.31)$$

$$\frac{K\omega_0}{2Q} [\cos \vartheta + K(\cos^2 \vartheta + \sin^2 \vartheta)] = 0 \quad (3.32)$$

Applying the well known trigonometric identity $\cos^2 x + \sin^2 x = 1$ we can continue

$$\frac{K\omega_0}{2Q} (\cos \vartheta + K) = 0 \quad (3.33)$$

$$\frac{K\omega_0}{2Q} \cos \vartheta = -\frac{K^2\omega_0}{2Q} \quad (3.34)$$

$$\cos \vartheta = -K \quad (3.35)$$

$$\vartheta = \pm \arccos(-K) \quad (3.36)$$

The angle ϑ can therefore reach $\pm \arccos(-K)$ at its extremes. Combining with Equation 3.27

$$\Delta\omega_{\text{LR}} = \frac{\omega_0}{2Q} \cdot \frac{K \sin(\pm \arccos(-K))}{1 + K \cos(\pm \arccos(-K))} \quad (3.37)$$

$$\Delta\omega_{\text{LR}} = \pm \frac{\omega_0}{2Q} \cdot \frac{K\sqrt{1-K^2}}{1-K^2} \quad (3.38)$$

$$\Delta\omega_{\text{LR}} = \pm \frac{\omega_0}{2Q} \cdot \frac{K}{\sqrt{1-K^2}} \quad (3.39)$$

The Equation 3.39 defines the *locking range* of the LC tank based ILO in terms of its free-running frequency ω_0 , the quality factor of the parallel RLC circuit Q and the injection ratio K . The locking range is the range of injected frequencies the oscillator can lock onto. It can be seen that by increasing K , the locking range can be widened. This seems intuitive, as the stronger the injection signal, the more easily it can “coerce” the oscillator to oscillate at the injection frequency.

It has to be noted that while Equation 3.39 works well for the so-called *weak injection* ($K \ll 1$), it does not encompass the effects of *strong injection*. A more general solution compatible with stronger injection signals is presented in [58]. The so-called perturbation model notes that when the injected signal is not negligible in magnitude, it does not only change the phase of the oscillation, but also modulates

the amplitude. The analysis requires understanding of large signal non-linearities of the given oscillator. In most practical applications, however, weak injection is the operating mode of interest, as the higher the K , the higher the power consumption.

To connect the results from Equation 3.39 with the phasor diagram, the phasor angles need to be examined. We can define $\psi = \vartheta - \varphi$, i.e. the angle between I_{inj} and I_{tank} . The maximum attainable value of ϑ , i.e. the value of ϑ at the edge of the locking range, is known from Equation 3.36. To quantify the maximum ψ , Equation 3.36 can be combined with Equation 3.20.

$$\begin{aligned}
\psi_{\max} &= \vartheta_{\max} - \varphi_{\max} = \arccos(-K) - \arctan\left(\frac{K}{\sqrt{1-K^2}}\right) \\
&= \arccos(-K) - \arcsin(K) \\
&= \pi - \arccos(K) - \left(\frac{\pi}{2} - \arccos(K)\right) \\
&= \pi - \arccos(K) - \frac{\pi}{2} + \arccos(K) = \pi
\end{aligned} \tag{3.40}$$

At the edge of the locking range, ψ reaches exactly 90° , as seen on the phasor diagram of an LC tank ILO at the edge of the locking range on Figure 3.6. This result makes intuitive sense. Assuming weak injection, φ is close to 0° at the edge of the locking range and $\vartheta \approx 90^\circ$. Since $I_{\text{tank}} \gg I_{\text{inj}}$, for I_{inj} to make the most difference to the oscillator, it needs to inject its energy when $I_{\text{tank}} = 0$, otherwise it would be easily overshadowed by it. The peaks of I_{inj} become synchronized with the zero crosses of I_{tank} when the phase shift between the two currents is precisely 90° . By shifting the phase of I_{inj} in either direction away from this point, its impact to the oscillator can only be lessened. Therefore, when I_{inj} is shifted by 90° at the edge of the locking range, it is already doing the most it possibly can for the given injection ratio.

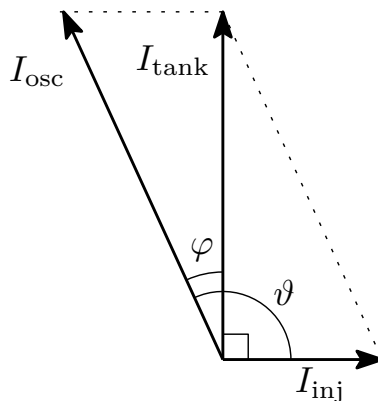


Fig. 3.6: LC tank ILO current phasor diagram at the edge of the locking range

A plot of the injection angle ϑ as a function of $\Delta\omega_0$ is shown on Figure 3.7. A quality factor Q of 4 was chosen for the plot. Per Equation 3.39, higher Q leads to

narrower locking range. The injection ratio K is a parameter in the chart, higher K leads to wider locking range and larger maximum ϑ .

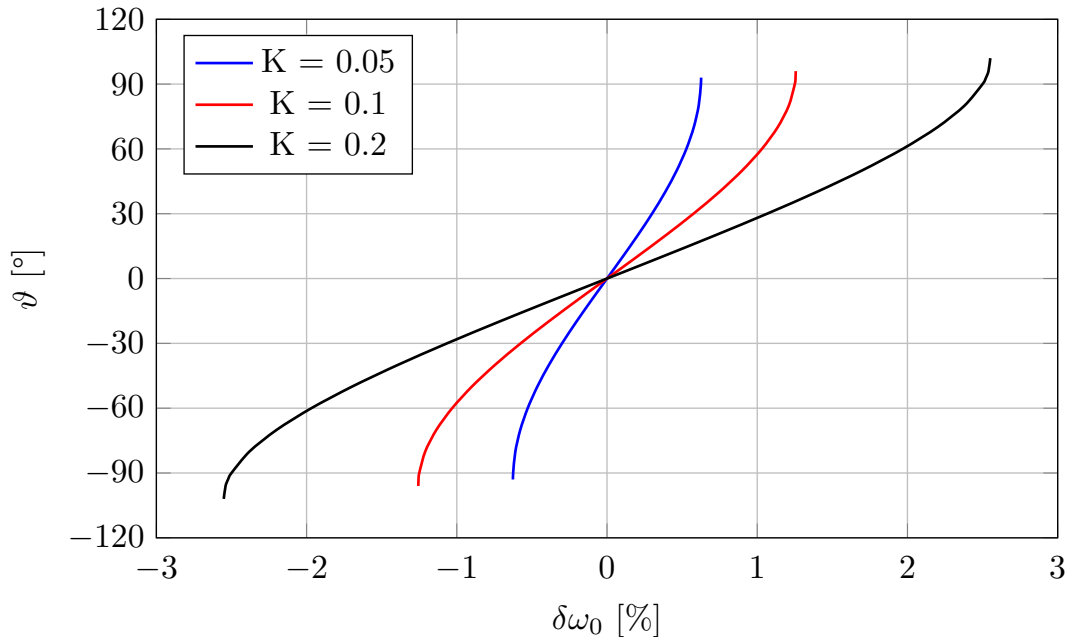


Fig. 3.7: Injection angle as a function of $\Delta\omega_0$ for various K

3.2.3 Paradox of locking and phase noise performance

An apparent paradox needs to be addressed. Assuming the ILO is a perfectly linear oscillator, via the superposition principle, the injected frequency ω_{inj} should simply add to oscillator, which would keep oscillating at ω_0 as well as ω_{inj} , responding to the input.

A unique analysis and resolution of this paradox is presented by Behzad Razavi in [53, p.1420]. According to Razavi, it is impossible to lock a perfectly linear oscillator onto an injected periodic signal. However, as long as even a mildly non-linear component exists in the system, the oscillator can be locked. In [53], the non-linearity is inserted via slightly non-linear negative resistance (represented by the inverting buffer and the NMOS in Figure 3.4), which is only natural, since the NMOS is a non-linear device. Needless to say, the role of the negative resistance is to replenish the energy dissipated by the resistor of the parallel RLC circuit.

The result of Razavi's analysis is that when $\omega_{\text{inj}} = \omega_0$, the feedback weakens the negative resistance effect. This is because when $\omega_{\text{inj}} = \omega_0$, $\vartheta = 0^\circ$ and the energy added to the oscillator by the injection signal is in-phase, therefore the negative resistance does not need to compensate for the full energy loss of the resistor. The negative resistance is weakened proportionally to the energy received by the oscillator by the injections.

On the other hand, when $\omega_{\text{inj}} = \omega_0 \pm \Delta\omega_{0,L}$, the injected energy comes with a 90° phase shift and therefore the negative resistance effect is at its maximum again, as it is needed to compensate the dissipated energy.

The main outcome of this analysis is that for injected frequencies somewhere in-between the two extremes, the negative resistance effect is weakened only slightly compared to the state with no injection. This means that the original oscillation at ω_0 gradually dies out (as the negative resistance no longer has the value necessary to sustain it), while sustained oscillation at ω_{inj} becomes possible, as the dissipated energy is replenished in-phase by the injection circuit.

Razavi subsequently uses this insight to explain one more useful effect. Since the overall impedance of the parallel RLC circuit (including the negative resistance effect) is now dependent on ω_{inj} and K (as explained in the previous paragraphs), the gain of the noise injected into the RLC circuit is dependent on the injection as well. In fact, if the oscillator is locked and the negative resistance effect is consequently weakened, the overall impedance of the RLC circuit is lowered (as the negative resistance no longer cancels out the normal, positive one in the RLC circuit) and therefore the noise is attenuated compared to the state with no injection. The result is that the phase noise of an injection locked oscillator is improved in the range of frequencies where the above holds true, i.e. the locking range, as seen on Figure 3.8 [53, p.1422].

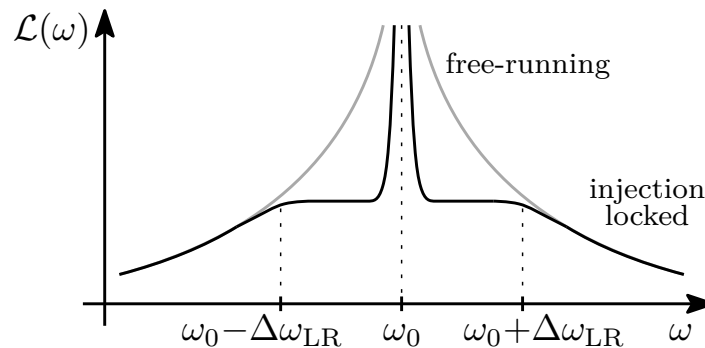


Fig. 3.8: Phase noise of a free-running and injection locked oscillator [53, p.1422]

3.3 Injection Locked Ring Oscillators

Ring oscillators, like any other oscillator, can be injection locked as well. However, since they are non-linear relaxation oscillators, the analysis of the locking phenomena has to take a different approach from the phasor based analysis of injection locking of a linear harmonic oscillator in section 3.2.

While linearised models of ring oscillators and their injection locking mechanism have been presented in literature, such as in [59], these approaches yield very limited accuracy when more realistic ring oscillator circuits are to be modelled.

A time domain based approach, which takes into account the inherent non-linearity of ring oscillators, has been presented in [60]. In this section, this approach will be explained and derived with more detail, including the mathematical assumptions not discussed in the paper. The main advantage of this approach is the ability to generalize it for any relaxation oscillator configuration with simple transient simulations, which will also be shown in chapter 4.

3.3.1 Ring oscillator time domain analysis

The starting point for this analysis is the differential ring oscillator model shown on Figure 3.9 (let us ignore the injection stage in this subsection). The differential mode of operation actually makes the analysis considerably simpler mathematically.

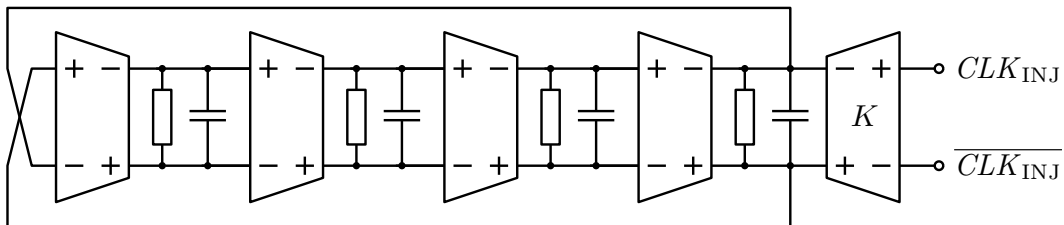


Fig. 3.9: Differential injection locked ring oscillator model [60]

The inverters in this model are represented by voltage sensing current outputting comparators driving a parallel RC load. The parallel RC load represents the output resistance of the driving stage and a combination of its output capacitance with the input capacitance of the following stage. When the comparator is in its high state (the non-inverting input voltage is higher than the inverting one), the current flowing out of the non-inverting output is I_{osc} , otherwise, it is $-I_{osc}$. The voltage waveform across the RC load while the comparator is in a constant state is a charging or discharging exponential, characterized by the well known time constant $\tau = RC$.

Since the implementation is differential, the next comparator in line transitions from its current state to the opposite state when the waveform of the RC load crosses zero in either direction. Because the ring oscillator has N stages, and each transitions when the output waveform of the previous one crosses zero, we can derive the frequency of the oscillations analytically.

Looking at the oscillation waveform on Figure 3.10, for the discharging part of the waveform ($0 \leq t \leq \frac{T}{2}$) we can write

$$v_{osc}(t) = -V_{osc,max} + (V_{osc,max} + V_{osc}) \cdot \exp\left(-\frac{t}{\tau}\right) \quad (3.41)$$

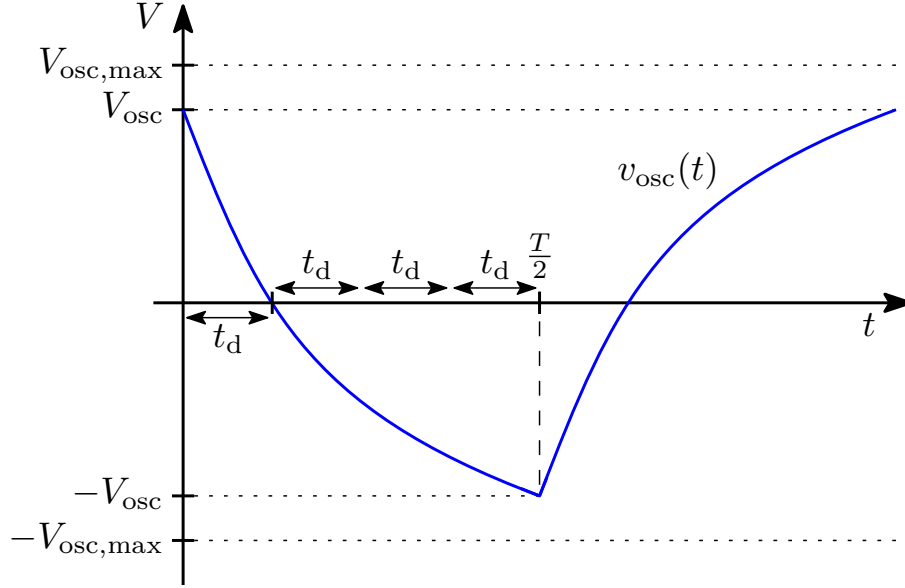


Fig. 3.10: Ring oscillator model voltage waveform [60]

where $v_{\text{osc}}(t)$ is the waveform of the oscillations, V_{osc} the amplitude of the oscillations and $V_{\text{osc,max}}$ the maximum possible amplitude of the oscillations, which is defined as $V_{\text{osc,max}} = I_{\text{comp}} \cdot R$. It is self-evident that the period of the oscillations can be defined as

$$T = 2Nt_d \quad (3.42)$$

where N is the number of stages and t_d the time it takes for a single stage to transition from $\pm V_{\text{osc}}$ to zero.

The equation for t_d can be derived analytically, since $v_{\text{osc}}(t_d) = 0$. Therefore

$$v_{\text{osc}}(t_d) = 0 \quad (3.43)$$

$$-V_{\text{osc,max}} + (V_{\text{osc,max}} + V_{\text{osc}}) \cdot \exp\left(-\frac{t_d}{\tau}\right) = 0 \quad (3.44)$$

$$\exp\left(-\frac{t_d}{\tau}\right) = \frac{V_{\text{osc,max}}}{V_{\text{osc,max}} + V_{\text{osc}}} \quad (3.45)$$

$$t_d = -\tau \ln\left(\frac{V_{\text{osc,max}}}{V_{\text{osc,max}} + V_{\text{osc}}}\right) \quad (3.46)$$

To evaluate t_d , it is necessary to find a relationship between N , V_{osc} and $V_{\text{osc,max}}$. For this analysis, Equation 3.41 and Equation 3.42 can be used, along with a final constraint, which is that $v_{\text{osc}}\left(\frac{T}{2}\right) = -V_{\text{osc}}$. Then, the following system of two equations can be written and rearranged.

$$v_{\text{osc}}(t_d) = 0 \quad (3.47a)$$

$$v_{\text{osc}}\left(\frac{T}{2}\right) = -V_{\text{osc}} \quad (3.47b)$$

$$-V_{\text{osc,max}} + (V_{\text{osc,max}} + V_{\text{osc}}) \cdot \exp\left(-\frac{t_d}{\tau}\right) = 0 \quad (3.48a)$$

$$-V_{\text{osc,max}} + (V_{\text{osc,max}} + V_{\text{osc}}) \cdot \exp\left(-\frac{T}{2\tau}\right) = -V_{\text{osc}} \quad (3.48b)$$

$$(V_{\text{osc,max}} + V_{\text{osc}}) \cdot \exp\left(-\frac{t_d}{\tau}\right) = V_{\text{osc,max}} \quad (3.49a)$$

$$(V_{\text{osc,max}} + V_{\text{osc}}) \cdot \exp\left(-\frac{2Nt_d}{2\tau}\right) = V_{\text{osc,max}} - V_{\text{osc}} \quad (3.49b)$$

$$\frac{V_{\text{osc,max}} + V_{\text{osc}}}{V_{\text{osc,max}}} \cdot \exp\left(-\frac{t_d}{\tau}\right) = 1 \quad (3.50a)$$

$$\frac{V_{\text{osc,max}} + V_{\text{osc}}}{V_{\text{osc,max}} - V_{\text{osc}}} \cdot \exp\left(-\frac{Nt_d}{\tau}\right) = 1 \quad (3.50b)$$

$$\frac{V_{\text{osc,max}} + V_{\text{osc}}}{V_{\text{osc,max}}} = \exp\left(\frac{t_d}{\tau}\right) \quad (3.51a)$$

$$\frac{V_{\text{osc,max}} + V_{\text{osc}}}{V_{\text{osc,max}} - V_{\text{osc}}} = \left[\exp\left(\frac{t_d}{\tau}\right)\right]^N \quad (3.51b)$$

$$\frac{V_{\text{osc,max}} + V_{\text{osc}}}{V_{\text{osc,max}} - V_{\text{osc}}} = \left(\frac{V_{\text{osc,max}} + V_{\text{osc}}}{V_{\text{osc,max}}}\right)^N \quad (3.52)$$

The Equation 3.52 can be solved for any given number of stages N , and the ratio $V_{\text{osc}}/V_{\text{osc,max}}$ can be obtained. For example, for $N = 3$, $V_{\text{osc}}/V_{\text{osc,max}} = 0.62$, while for $N = 5$, $V_{\text{osc}}/V_{\text{osc,max}} = 0.93$. This is intuitive: the higher the number of stages, the more time each individual stage has to settle, approaching $V_{\text{osc,max}}$ ever closer.

Since for $N \geq 5$ it can be approximated that $V_{\text{osc}} \approx V_{\text{osc,max}}$, returning to Equation 3.46 yields

$$t_d = -\tau \ln\left(\frac{V_{\text{osc,max}}}{V_{\text{osc,max}} + V_{\text{osc,max}}}\right) = -\tau \ln\left(\frac{1}{2}\right) = \tau \ln(2) \quad (3.53)$$

and therefore

$$f_{\text{osc}} = \frac{1}{2NRC \ln(2)} \quad (3.54)$$

As a side note, for ring oscillators used for FTDCs, the number of stages is often $N \geq 8$, since the larger the number of stages, the higher the number of output clock phases slicing the period and therefore the finer the resolution of the FTDCs. For these ring oscillators, Equation 3.54 is sufficiently accurate.

3.3.2 Ring oscillator under injection

At this point in the analysis, the injection stage from Figure 3.9 can be considered. Since the gain of the injection stage is K relative to the stages in the ring oscillator itself, its output current is $I_{inj} = \pm KI_{osc}$, depending on the state of the input clock CLK_{INJ} . Naturally, K is the injection ratio, which was already discussed in the context of harmonic oscillators in section 3.2, and it will be assumed that $K \ll 1$. In the context of this analysis, K can be also expressed as

$$K = \frac{V_{inj}}{V_{osc}} = \frac{V_{inj,max}}{V_{osc,max}} \quad (3.55)$$

The injection of a periodic square wave current I_{inj} into an RC load creates a periodic charging and discharging waveform $v_{inj}(t)$ very similar to $v_{osc}(t)$ with two differences: firstly, the amplitude of $v_{inj}(t)$, V_{inj} , and the maximum possible amplitude $V_{inj,max}$ are smaller than their $v_{osc}(t)$ counterparts. Secondly, there is a phase shift between $v_{inj}(t)$ and $v_{osc}(t)$, defined by a time interval Δ .

Via the superposition principle, the waveforms $v_{inj}(t)$ and $v_{osc}(t)$ add at the differential nodes where the injecting stage is connected, producing $v_{sum}(t)$. This is the physical waveform which consequently determines the time instant when the following stage flips its state. A diagram of this mechanism is shown on Figure 3.11.

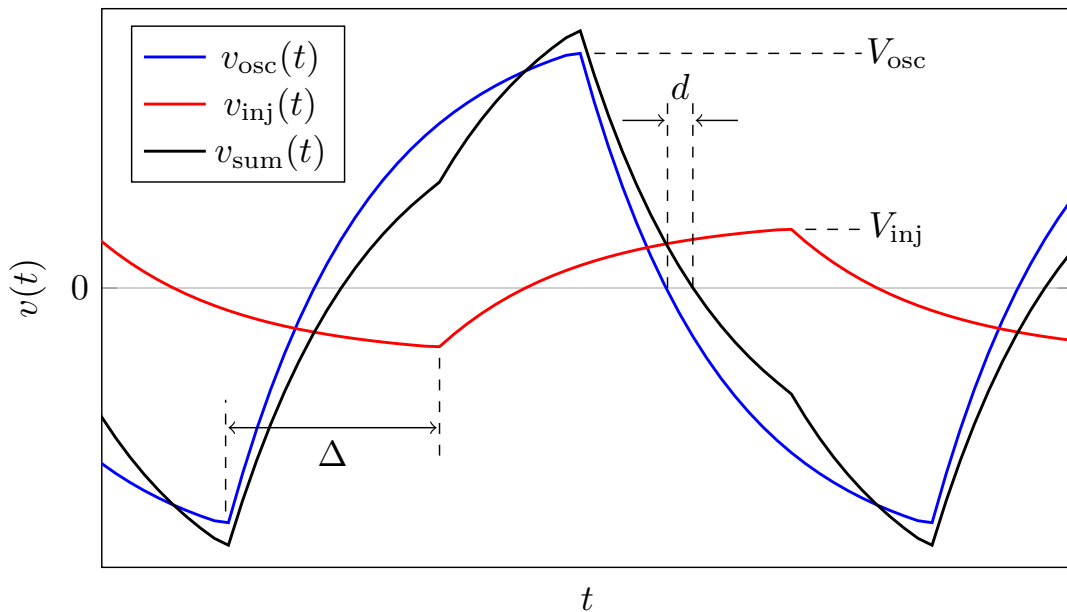


Fig. 3.11: Time domain based model of injection locking [60]

It can be seen that depending on the injection ratio and the phase shift Δ , the injection signal can either hasten or delay the zero-crossing point of $v_{osc}(t)$ by d . This can decrease, respectively increase the oscillation period. If the injection signal is strong enough, it can lock the oscillation period to its own.

Graphical analysis of Figure 3.11 yields that the new oscillatory period of the ILRO is $T_0 + 2d$, where T_0 is the free running period and d is the temporal difference between the original and new zero cross point of the ring oscillator stage under injection. Therefore, the locking range can be defined as ¹

$$T_0 + 2d_{\min} \leq T_{\text{inj}} \leq T_0 + 2d_{\max} \quad (3.56)$$

The challenge is therefore to find out what is the maximum and minimum attainable d . We can describe $v_{\text{inj}}(t)$ analogically to $v_{\text{osc}}(t)$ from Equation 3.41 as follows

$$v_{\text{inj}}(t) = -V_{\text{inj,max}} + (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{-t + \Delta}{\tau}\right) \quad (3.57)$$

and therefore we can write

$$\begin{aligned} v_{\text{sum}}(t) = v_{\text{osc}}(t) + v_{\text{inj}}(t) = & -V_{\text{osc,max}} + (V_{\text{osc,max}} + V_{\text{osc}}) \cdot \exp\left(\frac{-t}{\tau}\right) \\ & - V_{\text{inj,max}} + (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{-t + \Delta}{\tau}\right) \end{aligned} \quad (3.58)$$

For d , we can symbolically write

$$d(\Delta) = t_{\text{zc}}\{v_{\text{sum}}(t), \Delta \neq 0\} - t_{\text{zc}}\{v_{\text{sum}}(t), \Delta = 0\} \quad (3.59)$$

where $t_{\text{zc}}\{v_{\text{sum}}(t), \Delta \neq 0\}$ is the zero crossing time of $v_{\text{sum}}(t)$ when non-zero Δ is considered. This equation includes the insight that when $\Delta = 0$, the zero crossing of $v_{\text{sum}}(t)$ is not delayed with respect to $v_{\text{osc}}(t)$. This is because in such case the injection signal is perfectly in phase with the oscillation and can only affect the amplitude.

After several steps of mathematical rearranging, which are presented in section A.1 of the appendices, the following result can be obtained for $\Delta > 0$

$$d(\Delta)|_{\Delta > 0} = \tau \ln \left[\frac{V_{\text{osc}} + V_{\text{osc,max}} + (V_{\text{inj}} + V_{\text{inj,max}}) \cdot \exp\left(\frac{\Delta}{\tau}\right)}{V_{\text{osc}} + V_{\text{osc,max}} + V_{\text{inj}} + V_{\text{inj,max}}} \right] \quad (3.60)$$

and similar derivation for $\Delta < 0$ is presented in section A.2, resulting in

$$d(\Delta)|_{\Delta < 0} = \tau \ln \left[\frac{V_{\text{osc,max}} + V_{\text{osc}} - (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{\Delta}{\tau}\right)}{V_{\text{osc,max}} + V_{\text{osc}} - V_{\text{inj,max}} - V_{\text{inj}}} \right] \quad (3.61)$$

The final, but most mathematically tedious step of the determination of locking range is expressing the maximum and minimum values of the $d(\Delta)$ function.

¹It can be noted that the Equation 3.56 holds true regardless of what the oscillation or injection waveform actually looks like. As long as the relationship between d and Δ is known (analytically or simulated), Equation 3.56 can be used for expressing the locking range of any ILO.

A crucial finding helping in the process is the realization that $v_{\text{osc}}(\Delta_{\text{max}}) = -V_{\text{inj}}$. For $\Delta < \Delta_{\text{max}}$, the maximum of $v_{\text{inj}}(t)$ occurs sooner than the zero cross of $v_{\text{osc}}(t)$. Therefore, the full capability of delaying the zero cross has not been utilized. When $\Delta > \Delta_{\text{max}}$ (which is the case shown on Figure 3.11), the maximum of the $v_{\text{inj}}(t)$ waveform occurs too late, when $v_{\text{osc}}(t)$ has already crossed zero and its magnitude is much higher than the $v_{\text{inj}}(t)$ maximum.

However, when $\Delta = \Delta_{\text{max}}$, the maximum of $v_{\text{inj}}(t)$ perfectly cancels out the current value of $v_{\text{osc}}(t)$, i.e. $v_{\text{osc}}(\Delta_{\text{max}}) = -V_{\text{inj}}$ and $v_{\text{inj}}(\Delta_{\text{max}}) = V_{\text{inj}}$, therefore $v_{\text{sum}}(\Delta_{\text{max}}) = 0$ and the zero cross has been delayed as much as possible, i.e. $d_{\text{max}} = d(\Delta_{\text{max}})$. Similarly, $d_{\text{min}} = d(\Delta_{\text{min}})$, where $v_{\text{osc}}(\Delta_{\text{min}}) = V_{\text{inj}}$ and $v_{\text{inj}}(\Delta_{\text{min}}) = -V_{\text{inj}}$.

The results of the mathematical derivations and simplifications presented in section A.3 and section A.4 are

$$d_{\text{max}} = \tau \ln \left(\frac{1}{1 - K} \right) \quad (3.62)$$

$$d_{\text{min}} = \tau \ln \left(\frac{1}{1 + K} \right) \quad (3.63)$$

As explained in section A.3 and section A.4, these results are based on the assumption that $N \geq 5$. Otherwise, the expressions for d_{max} and d_{min} are much more complicated and impractical. Both Equation 3.62 and Equation 3.63 are useful because of their simplicity, as K is a design parameter and τ is a constant easily determined via simulation of the ILRO inverter cell.

Returning to Equation 3.56, the locking range of an ILRO is

$$T_0 + 2\tau \ln \left(\frac{1}{1 + K} \right) \leq T_{\text{inj}} \leq T_0 + 2\tau \ln \left(\frac{1}{1 - K} \right) \quad (3.64)$$

where T_0 , the free-running period, is defined by Equation 3.54.

When plotting Equation 3.64, the injection ratio K is definitely of interest and will be swept on the X axis. There are two possible choices for a parameter: τ and N (the number of stages N is “hidden” in the equation for T_0). Both parameters affect the absolute value of T_0 in the same way. Interestingly enough, when the maximum and minimum periods (or frequencies) are normalized to the free running period or frequency, varying τ does not affect normalized locking range at all. This is because τ appears in the equation for d as well as T_0 , and when normalizing, it cancels itself out. On the other hand, N appears only in the equation for T_0 and therefore causes variation even in the normalized locking range plot. This plot is shown on Figure 3.12.

The upper lines in Figure 3.12 show the upper limit of the locking range and the lower lines show the lower limit. There are two interesting insights this plot provides. First of all, increasing N decreases the locking range width. Secondly, the lower limit of the locking range is highly non-linear for high values of K . This is

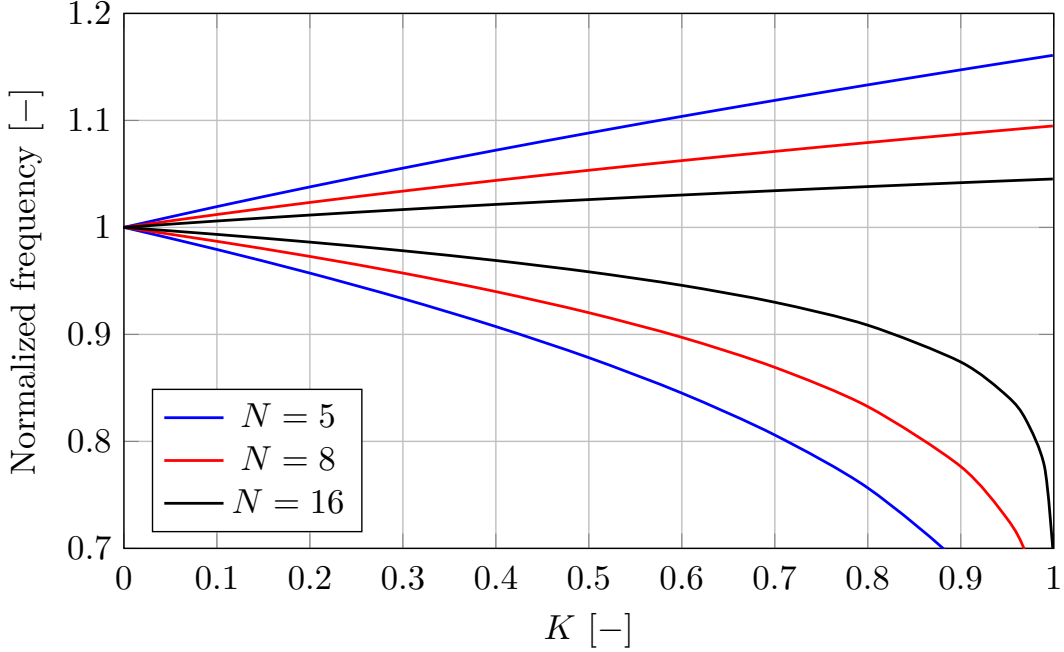


Fig. 3.12: Locking range of ILRO for various number of stages

because the denominator in Equation 3.62 approaches zero and the fraction grows in value rapidly. This non-linearity is not physical and is a mathematical artefact instead, serving as a reminder that these analyses are only valid for $K \ll 1$.

Before closing this subsection, one more concept needs to be addressed: it is theoretically possible to inject the signal into multiple stages of the ILRO, as proven in [59]. In this diploma thesis, however, only single-input injection will be considered. The main benefit of the multiple-input injection is that the locking range widens, which makes intuitive sense, as more energy is injected in total. The second benefit is that when the injection signal is injected into multiple inputs, the propagation delay of more stages rather than just one is adjusted, and therefore the necessary total propagation delay adjustment is more evenly distributed, leading to better *DNL* performance of the FTDC.

To give an example, let us assume that a 16-stage ILRO oscillates at 600 MHz. This means that each stage produces 52.1 ps delay (this can be calculated by rearranging Equation 3.54). When injecting 625 MHz into single input, assuming the ILRO locks, 15 stages still produce 52.1 ps delay, while the so-called injection stage has to produce 18.8 ps delay to compensate for the rest and produce total delay matching the period of the injections. However, if the injection signal was injected into two points rather than just one, 14 stages produce 52.1 ps delay, while the two injection stages produce 35.4 ps delay each. The linearity of the time-slicing action is therefore significantly improved.

However, there is also an important drawback: it is necessary to inject energy

into the multiple inputs in correct phase. Quoting [59, p.1908], “the locking range [for multi-input injection] decreases or becomes even worse than the case for the single-ended one as the input phase difference departs from the optimum value”. This is a problem for practical implementation, as it is difficult to guarantee precise phase alignment of digital signals. If the multiple injection waveforms were to be injected with incorrect phase shift, not only would the injection range suffer, but the phase noise of the oscillator as well, as the improperly injected energy could shape the waveform in such a way which would degrade its spectral purity. If the phase error between the inputs changed over time, this could also manifest as jitter. It is needless to say that creating the phase-shifted injection signals would also take up more area and consume more power. Therefore this concept will not be explored further in this thesis.

3.3.3 Limitations of the analytical model

While the results attained by the previous analysis are valid and useful, they do not necessarily translate perfectly to practical design. There are various reasons for this, some of which will be explained below.

First of all, this analysis is not directly applicable for single ended ring oscillators, although the basic qualitative insights provided by the analysis are transferable. Differential ring oscillators are however preferable for applications in DToF LIDAR, so this is not a big issue.

Secondly, it was assumed that the duty cycle of the injections is 50 %. This is not the most efficient implementation. In real ring oscillators, there exists a finite time-window t_{sens} where the inverter is sensitive to injections.

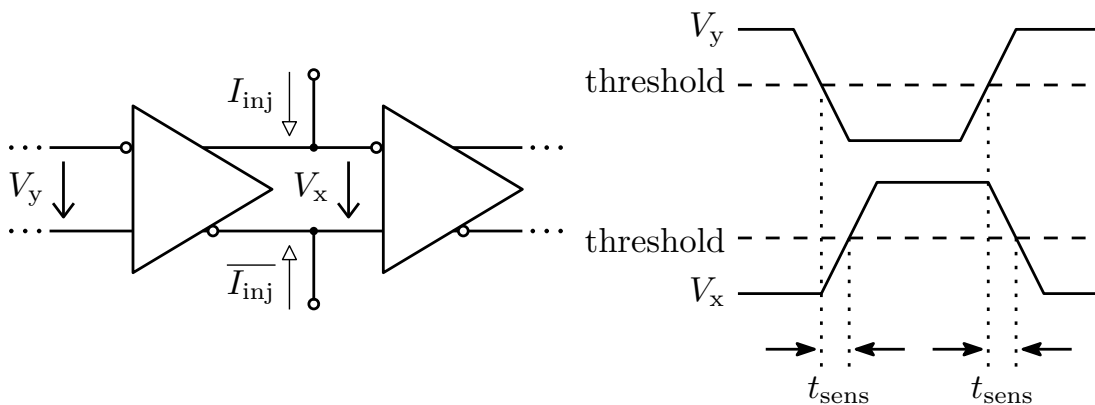


Fig. 3.13: Ring oscillator injection sensitivity time window

The reason for this sensitivity can be explained as follows. In the illustration on Figure 3.13, the injection stage signal V_x starts off low, and the physical voltages are very close to supply rails. Any charge injected during this state is quickly

absorbed by the supply rails and produces a rather small change of V_x ($K \ll 1$ is assumed again), as there now exists a low impedance path through the output driving transistors of the previous stage, some of which now operate in linear region. Once the previous stage voltage V_y crosses threshold, and the inverter driving the injection stage starts transitioning, the low impedance path to supply rails begins to weaken. The capacitance at the injection stage starts charging up and any charge injected during this time can hasten or delay the threshold crossing. Once V_x finally crosses the threshold, injecting further charge does not speed up the transitions of the following stages any more and when the injection stage settles at high, a low impedance path to the supply rails is formed once again, absorbing any injected charge rather easily.

Therefore, for real applications, injecting shorter pulses is more energy efficient, as less charge is injected during the periods when the injection stage sensitivity is weakened. An example of a practical injection circuit is shown on Figure 3.14 [41, p.2]. This circuit produces short, exponentially decreasing current pulses, whose magnitude (and therefore the total injected charge) depends on the size of the capacitors. The resistive feedback path opens up the bandwidth and improves the rise time of the pulses [61].

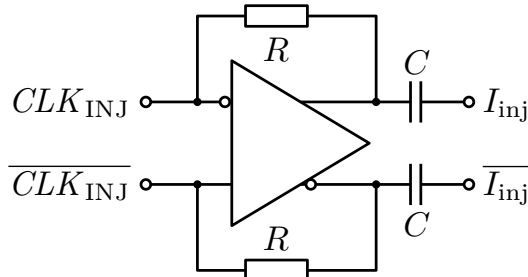


Fig. 3.14: Pulse injection circuit [41, p.2]

The reason why the previous analysis cannot account for this fact is that it models the oscillations as exponential charging and discharging of an RC circuit. This is not physically accurate and the dynamics are actually significantly more complex, as transistors are inherently non-linear devices.

Final limitation of the previously derived time domain based model of ILROs to be mentioned here is that often, the duty cycle of the injection clock cannot be guaranteed to be exactly or even close to 50 %. Even though a PLL and a divider are used to drive the injection circuit with a very symmetric clock signal, the duty cycle at the point of use can vary by several % due to parasitic capacitances, delays and non-ideal gate switching transients. Therefore the distance between the current pulses produced by the injection circuit can vary as well. The effect of this factor on the locking range, phase noise and other performance metrics of the ILRO needs

to be considered. However, so far no literature known to the author has provided analysis of the impact of injection clock duty cycle.

As mathematical analysis of these non-idealities would be impractical, a MATLAB model taking these factors into account will be described in chapter 4.

3.4 Injection Locked Ring Oscillator based TDCs

An example of an ILRO based 3-bit flash FTDC is shown on Figure 3.15.

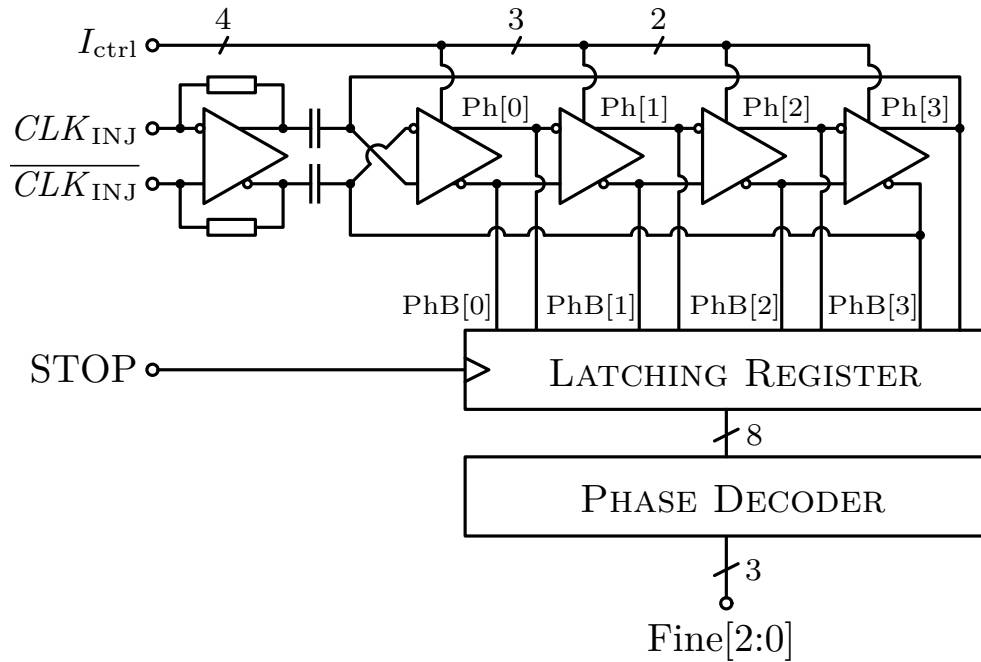


Fig. 3.15: Differential ILRO based 3-bit flash FTDC

The ILRO consists of 4 differential inverter stages in a ring oscillator configuration. These differential inverter stages are CCDLUs, and they can be implemented as shown on Figure 2.9. The current control, provided by a DLL (not shown) adjusts the free running frequency of the ring oscillator coarsely and brings it closer to its target value, so that the injection clock CLK_{INJ} can successfully lock the ILRO via the pulse injection circuit from Figure 3.14.

The way this ILRO based FTDC can fit into overall architecture is shown on Figure 3.16.

This scheme is similar to the one discussed in subsection 2.6.2, where the clock phases $CLK[N-1:0]$ are generated locally with ring oscillators. As was discussed previously, this scheme is power efficient, as the clock phases are only generated at the point of use and therefore their routing is short, minimizing dynamic power loss caused by interconnect parasitics. The ring oscillators were made of CCDLUs biased by a DLL, so that their free running frequency was close to target. The main issue

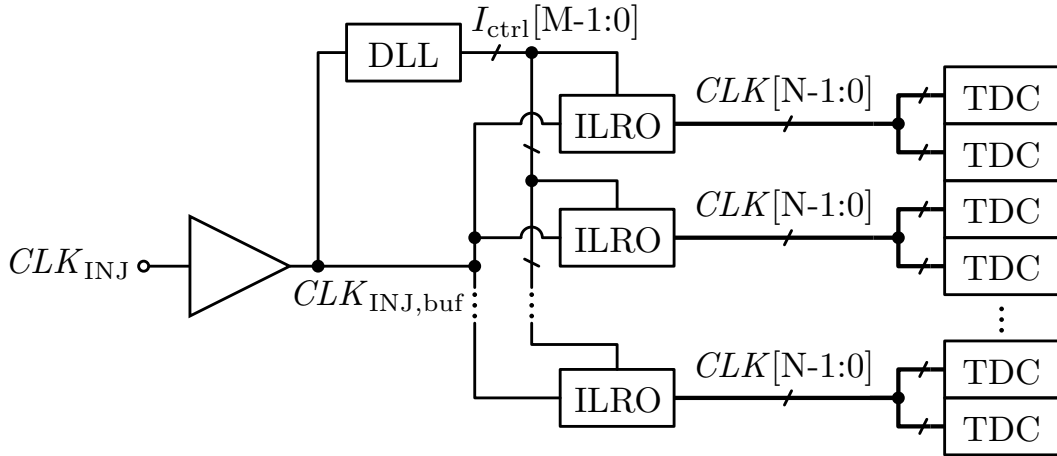


Fig. 3.16: Local clock counting with ILROs

of this local ring oscillator based counting scheme was the mismatch of the ROs, which would cause them to oscillate at slightly different frequencies and phases.

The scheme shown on Figure 3.16 fixes this issue, as all the ILROs are synchronized in frequency by injection locking. The ILRO mismatch therefore no longer affects the oscillation frequency, although it still affects the *DNL* (the CCDLUs inside each ILRO can vary relative to each other). As was discussed in subsection 3.2.3, the phase noise of clock phases also reduces, leading to less temporal jitter and higher precision.

The price paid for these improvements are additional dynamic losses caused by the routing of the injection clock, but since it is only a single phase which is routed (or two, for differential routing), the benefits outweigh the cost.

4 INJECTION LOCKED RING OSCILLATOR MODELLING

In this chapter, a MATLAB¹ model of a differential ILRO will be presented. In the first section, the motivation for the model will be discussed. Afterwards, the model itself will be described. Following that, the various simulation outputs of the model will be discussed, and in the penultimate section, the various analyses that can be performed using the model will be presented. Finally, the limitations of the model will be acknowledged.

4.1 Motivation and goals

It is important to discuss the motivation for the model, and the questions it should be able to answer.

MATLAB models of *Integrated Circuit* (IC) blocks are, in general, used for top-level block simulations from the system design point of view during architecture definition project stages. It is important to note that when a low level or a device level block is to be modelled, MATLAB models offer limited accuracy. The reason is that in reality, the real behaviour of these low level blocks is far from ideal, governed instead by complicated transistor device physics. The lower the level of the block, the more inaccurate are the idealized macromodels.

In this thesis, a MATLAB model of a differential ILRO will be created to obtain and/or confirm knowledge and intuition about the injection locking phenomenon, reveal some design trade-offs and answer some of the unanswered questions with regards to ILROs. The goal is to create a relatively simple model quickly, capable of helping the designer reach conceptual understanding of the ILRO during the architecture definition stage, i.e. before the design stage even begins. The actual physical implementation can be optimized later with better suited tools such as the Cadence Virtuoso & ADE suite.

There are a few questions related to the behaviour of an ILRO that the literature has not answered yet. These are the effects of injection clock duty cycle variation on the locking range, or the effect of injection pulse width on locking range or injection clock duty cycle sensitivity.

¹MATLAB R2019b and R2020b were used.

4.2 Model overview

There are two main ways a MATLAB model of a circuit can be created. Either the transient waveforms can be calculated step by step with explicit difference equation loops via a script, which is a very flexible but challenging and time-consuming method (as essentially a custom transient simulation engine needs to be created), or a Simulink model can be created using Simscape blocks schematically. The second approach yields results much faster and is therefore preferable for pre-design models such as the one in this thesis, although it does offer less control to the user.

There were two variants of the model created - an eight stage ILRO, as shown on Figure B.1 in the appendices, and a sixteen stage ILRO. The sixteen stage oscillator schematic is wide and not legible on an A4 paper, the full sized schematics are therefore available in the accompanying digital appendix (see chapter C).

4.2.1 Differential inverter model

Since an ILRO only consists of several inverters and an injection circuit, the choice of model for the inverter is important.

The simplest possible model of an inverter which still bears some semblance to its real counterpart is shown on Figure 4.1. This model is functionally identical to the one used for the mathematical analysis on Figure 3.9.

An input capacitance is followed by a voltage sensing double pole switch, connecting the output either to ground or the supply rail through a resistor. If the voltage on the capacitor crosses a specific threshold, the switch changes state in an opposite way, i.e. if the voltage rises above the threshold, the switch connects the output to ground and vice versa. Therefore, the input capacitance of the next inverter in the ring is charged or discharged through the output resistance of the previous inverter. The differential action is not implemented faithfully, as the differential inverter model simply contains two independent single ended inverters.

Clearly, this model has its shortcomings. It does not reflect any non-linearity of inverter gain or g_m and it is pseudo-differential at best. However, the model is simple enough to guarantee relatively fast simulation time. This is important, as when oscillators are simulated and relatively high frequency domain precision is desired, a large number of samples and periods is required (easily in the range of hundreds of thousands of sampled points over thousands of oscillation periods).

Several parameters of this model have been “masked” so that they can be set from the top hierarchical level in Simulink. Apart from the obvious ones, such as R , C , V_{DD} or $V_{th,sw}$, the initial voltage of the capacitor $V_{C,0}$ is crucial for “kickstarting” the oscillations as soon as possible during simulation runtime.

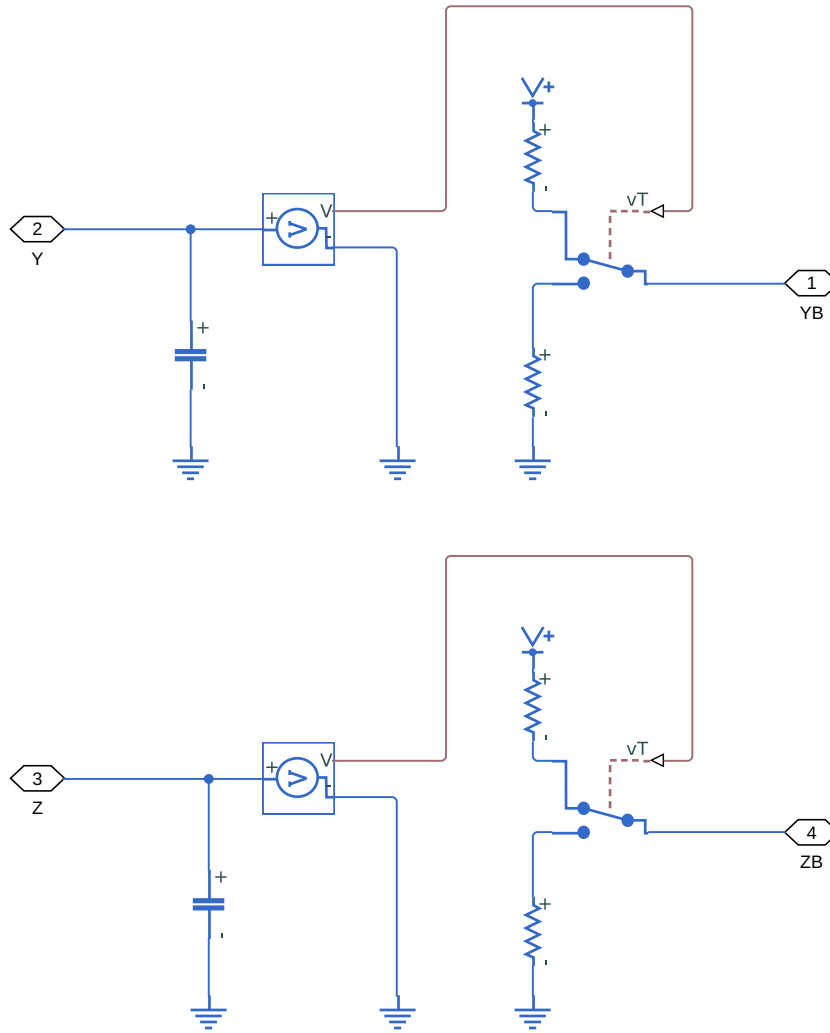


Fig. 4.1: Differential inverter model in Simulink

4.2.2 Injection signal

While it would be possible to recreate the injection circuit from Figure 3.14 in Simulink, it is not necessary. First of all, the inverter model is idealized as it is, and secondly, Simscape is not very efficient at solving feedback systems. It was determined that implementing the injection circuit from Figure 3.14 slowed down the simulation by as much as a factor of ten.

Instead, the current will be injected using ideal rectangular pulses via dependent current sources. It is expected that the dominant factor is not the shape of the pulse, but the charge injected in a given time period.

Default Simulink blocks are used to implement waveform shaping. Any zero-crossing waveform source block can be used at the input of the shaper to produce constant width rectangular pulses. The advantage of this approach is that the source block can be a pulse train block, where the duty cycle can be varied, or a chirp, sweeping through a range of frequencies over time. In both cases, the

waveform shaping blocks create a train of identical pulses at the times when the source waveform crosses zero, whatever its original shape.

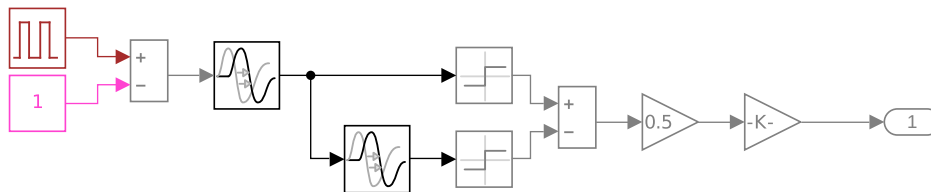


Fig. 4.2: Injection pulse train generator

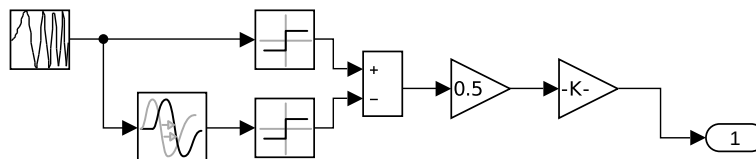


Fig. 4.3: Injection pulse chirp generator

The way this is done is shown on Figure 4.2 and Figure 4.3. First, the source waveform is delayed by pw . Then, the delayed and the original waveforms pass through the “sign” block, which returns 1 for positive values, -1 for negative values and 0 for zero. The results are subtracted, and the outcome of this subtraction is non-zero only during zero-crossings of the source waveform for the time specified by pw . This way, pulses of fixed length pw and amplitude ± 2 are created. The 0.5 gain block brings the amplitude to ± 1 , and the final gain block scales the size of the pulse by a_inj , which is the height of the current pulses in Amperes. This value is then fed into the Simscape current sources.

Additionally, a noise source is added to the rectangular pulses, as seen on Figure B.1 (the red white noise source block on the left). Even though the noise is injected along with the current pulses, it models the noise of the ring oscillator itself. This is useful for phase noise simulations.

4.2.3 Default simulation parameters

The simulation parameters listed in Table 4.1 were used for all simulations which will be discussed in the following sections, unless specified otherwise. Similarly, the sixteen stage ILRO model will be used by default.

Via Equation 3.54, the free running frequency of the eight stage ILRO model using the parameters listed in Table 4.1 is

$$f_{osc} = \frac{1}{2NRC \ln(2)} = \frac{1}{2 \cdot 8 \cdot 1000 \cdot 72.5 \cdot 10^{-15} \cdot \ln(2)} = 1.244 \text{ GHz} \quad (4.1)$$

and similarly for the sixteen stage ILRO model, 621.9 MHz (a half of the eight stage f_{osc}) can be calculated.

Tab. 4.1: Default simulation parameters

Parameter	Default value	Note
res	1000 Ω	inverter output resistance
cap	72.5 fF	inverter input capacitance
cap_esr	0 Ω	inverter input capacitance ESR
vdd	1.2 V	supply voltage
vth	0.6 V	inverter switch threshold
sw_g_open	1 nS	inverter switch open conductance
sw_r_closed	0.1 Ω	inverter switch closed resistance
pw	100 ps	injection pulse width
a_inj	200 μ A	injection pulse height
duty_cycle	50 %	injection clock duty cycle
noise_power	0	height of <i>PSD</i> of white noise
ts	10 ps	sampling period

4.3 Simulation outputs

There are multiple types of simulation outputs which the model offers. The transient data can be viewed on its own, transformed into the frequency domain or viewed from both time and frequency points of view simultaneously with a spectrogram.

It has to be noted that the model is sensitive to solver accuracy. Optimum performance was achieved using the “ode45” variable-step solver along with a relative tolerance no larger than 10^{-8} . Otherwise, the solver errors accumulate and the waveforms can even erroneously stop oscillating.

4.3.1 Transient waveforms

The primary type of output from Simulink models are transient waveforms, an example of which is shown on Figure 4.4 and Figure 4.5. On these two figures, i_{inj} is the differential injected current waveform, v_{prev} is the differential voltage at the stage preceding the injection stage and v_{inj} is the differential voltage at the injection stage (the reason their polarity is the same is that the preceding stage is cross connected to the injection stage, as seen on Figure B.1).

Transient waveforms can be used directly or for further post processing, as shown in subsection 4.4.2.

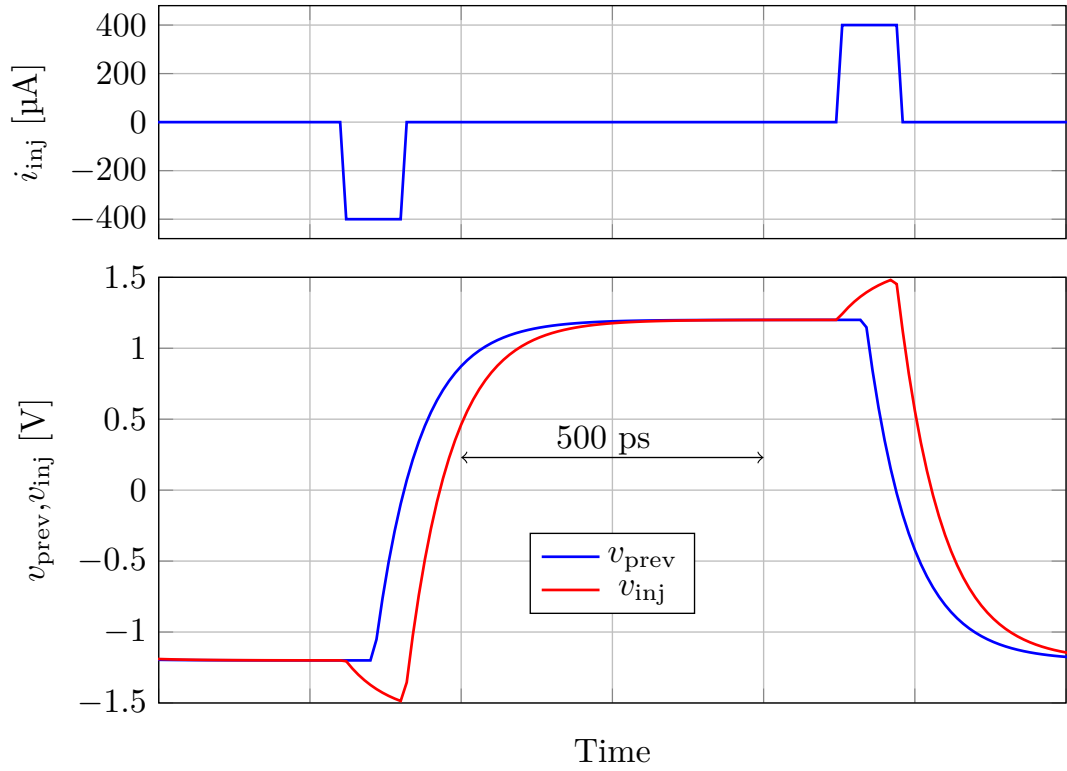


Fig. 4.4: ILRO model period of oscillation for $f_{inj} < f_0$

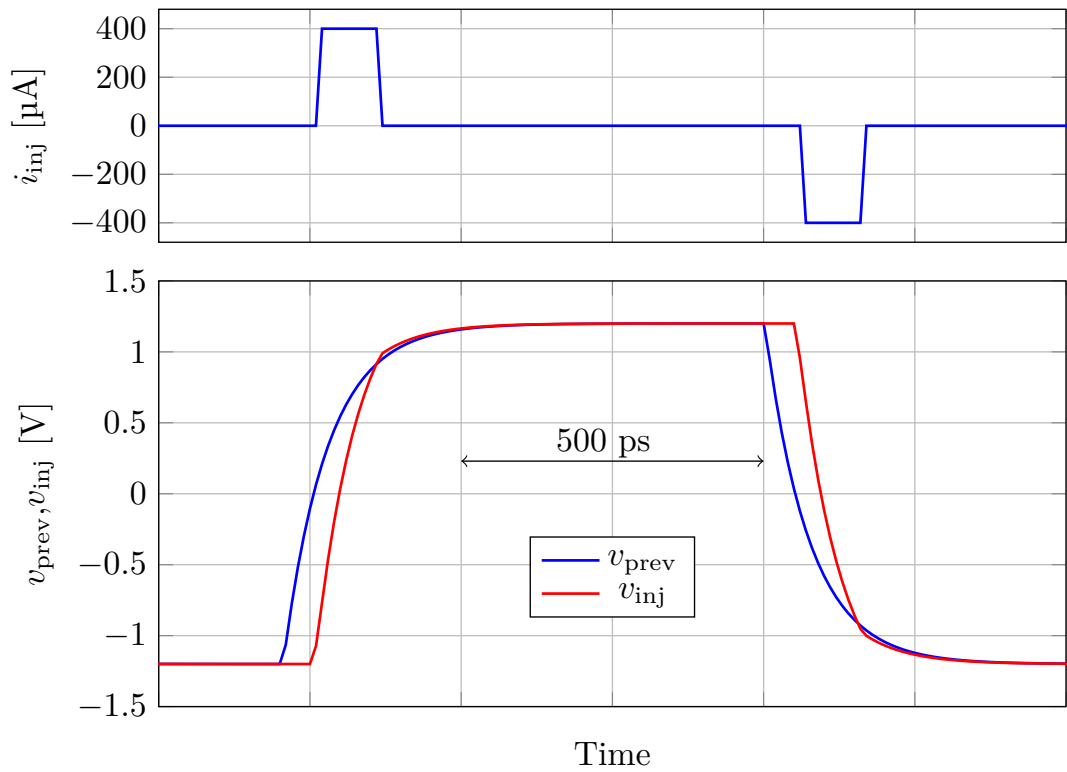


Fig. 4.5: ILRO model period of oscillation for $f_{inj} > f_0$

4.3.2 Frequency domain

Transient waveforms can be used to calculate their frequency domain representation via *Discrete Fourier Transform* (DFT). To obtain *power spectral density* (*PSD*), the `pwelch()` command can be used, utilizing Welch’s method of pre-processing the transient signal by cutting it into segments, windowing and transforming them into frequency domain separately and averaging the spectra. This method achieves easily readable low noise plot at the cost of lower frequency resolution.

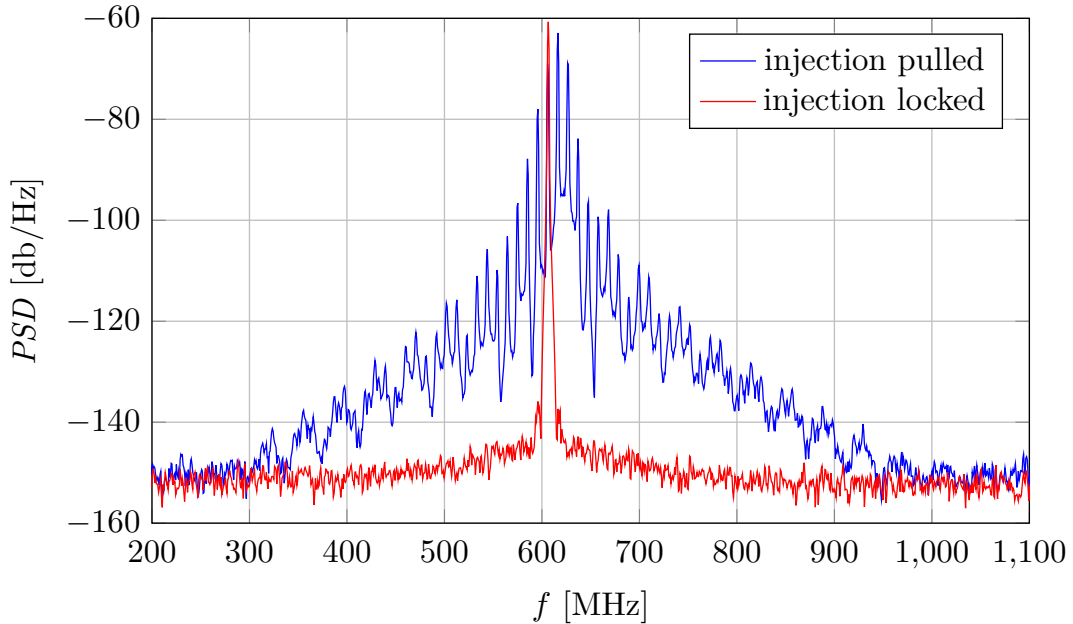


Fig. 4.6: *PSD* of ILRO model when injection pulled and locked

An example of a *PSD* calculated with the `pwelch()` command is shown on Figure 4.6, focusing on the first harmonic region of the frequency range. In the first case, $f_{\text{inj}} = 606$ MHz, which is just outside the lower edge of the locking range for the default simulation parameter (see Table 4.1) sixteen stage ILRO. The oscillator does not lock, but is pulled instead. The first harmonic frequency decreases and the spectrum smears, containing a large number of products of *injection frequency* (f_{inj}) and *free-running frequency* (f_0).

In the second case, $f_{\text{inj}} = 606.5$ MHz, which is a frequency only slightly different to the previous case, but “inside” the locking range. The oscillator is locked and its spectrum is clean, containing only a single harmonic at f_{inj} . This observation can be used for determining the size of the locking range.

4.3.3 Spectrogram

A spectrogram is shown on Figure 4.7. This spectrogram focuses on the first harmonic of the sixteen stage ILRO model, where the input injection signal is a chirp rising linearly in frequency over time.

From the start of the simulation until $t = 2.5 \mu\text{s}$, the oscillator is free running, oscillating at roughly 621.9 MHz, as calculated in subsection 4.2.3. At $t = 2.5 \mu\text{s}$, the injections start at roughly $f_{\text{inj}} = 604 \text{ MHz}$. This is outside the locking range of the oscillator, but close enough so that the oscillator is pulled to this frequency immediately. This is clearly visible, as the first harmonic frequency quickly decreases from 621.9 MHz to approximately 619.5 MHz, and additional frequency content appears, as discussed and shown previously in subsection 4.3.2.

As the frequency of the injections f_{inj} increases, the oscillator is pulled more and more, until $t \approx 6 \mu\text{s}$, when the oscillator locks at $f_{\text{inj}} \approx 607 \text{ MHz}$ and all other harmonics disappear. This clean harmonic rises in frequency along with f_{inj} , until f_{inj} crosses the free running frequency 621.9 MHz at $t \approx 30 \mu\text{s}$. This is accompanied by a slight but visible disturbance in the frequency contents, as the envelope of the oscillations suddenly changes shape (this is visible on the plots from subsection 4.3.1). Finally, at $t \approx 47 \mu\text{s}$, f_{inj} reaches approximately 634 MHz and the oscillator unlocks.

The spectrogram can be clearly used a crude way of determining the locking range, but because the number of points per period needed to achieve high frequency resolution is very high, and the length of the transient simulation to achieve good temporal resolution has to be very long, both the simulations and the spectrogram building take a long time.

4.4 Analyses

In the following subsections, analyses meant to provide the desired insights about the ILROs, such as the locking range, duty cycle sensitivity or injection pulse width dependencies will be presented.

Unless otherwise specified, all the following analyses were performed on the sixteen stage ILRO model by default.

4.4.1 Locking range

In this subsection, locking range of the ILRO models will be determined and compared to analytical predictions, which have been derived in section 3.3.

In order to determine the locking range, the locked state needs to be distinguished. There are two main approaches to this: the frequency domain approach and the time domain approach.

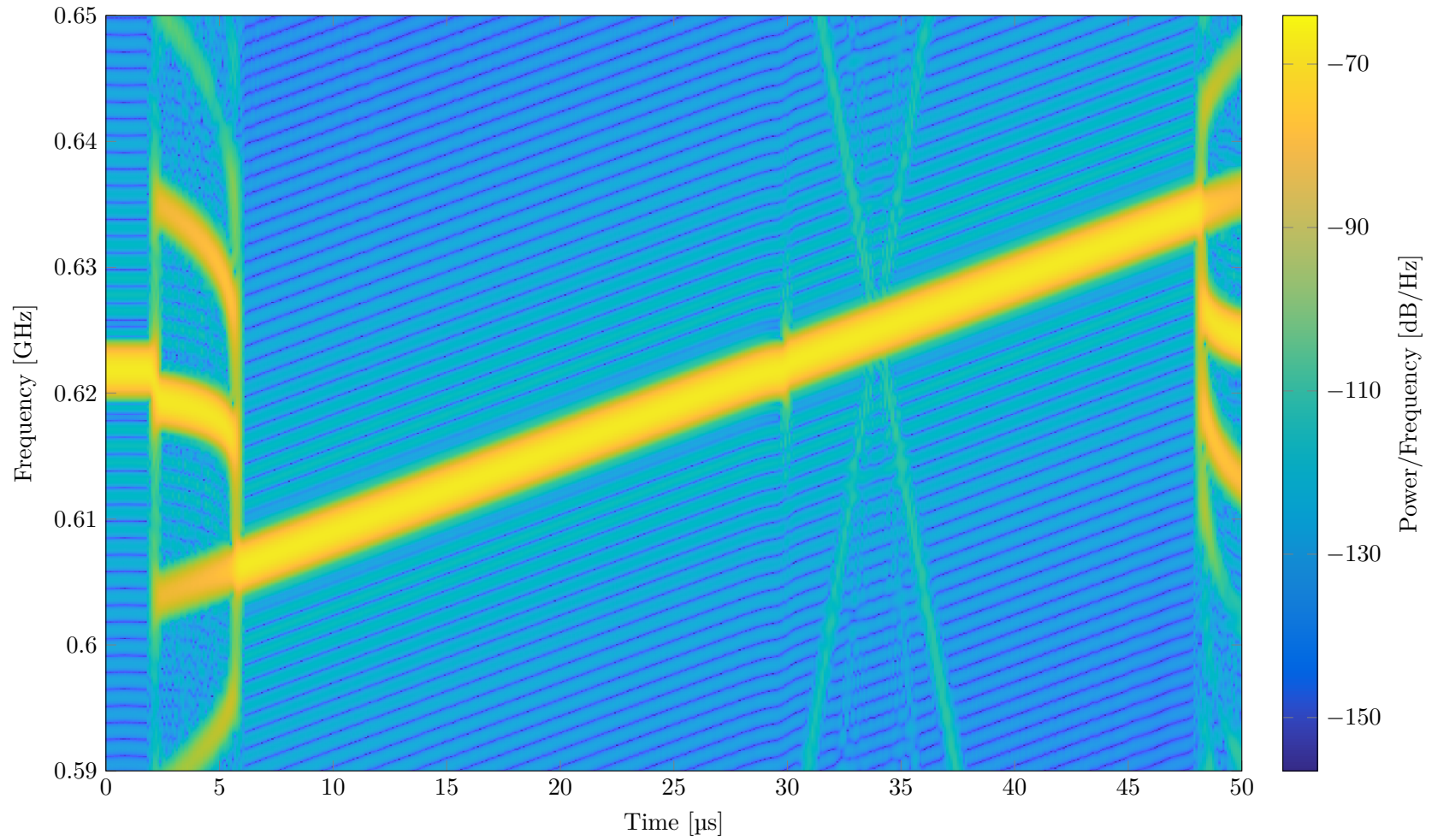


Fig. 4.7: Spectrogram of a clock phase first harmonic with chirp injection signal

The frequency domain approach was foreshadowed in subsection 4.3.2 or subsection 4.3.3. When the oscillator locks, the magnitude spectrum features a single clean tone in the first harmonic region of the frequency range at f_{inj} .

The time domain approach works more visually, as when the oscillator is not locked, the envelope of the injection stage voltage is periodically pulsating at a frequency significantly lower than f_{inj} due frequency pulling which resembles amplitude modulation.

Whatever the method used, the results for the locking range are the same. The absolute locking range of the sixteen stage ILRO model is shown on Figure 4.8, and the relative locking range of the two models is shown on Figure 4.9. The relative locking range is calculated as

$$\delta f_{LR} = \frac{\Delta f_{LR}}{f_0} = \frac{f_{inj,max} - f_{inj,min}}{f_0} \quad (4.2)$$

therefore the locking range can be also expressed as $\pm 0.5 \cdot \delta f_{LR} \cdot 100\%$ distributed approximately symmetrically around f_0 .

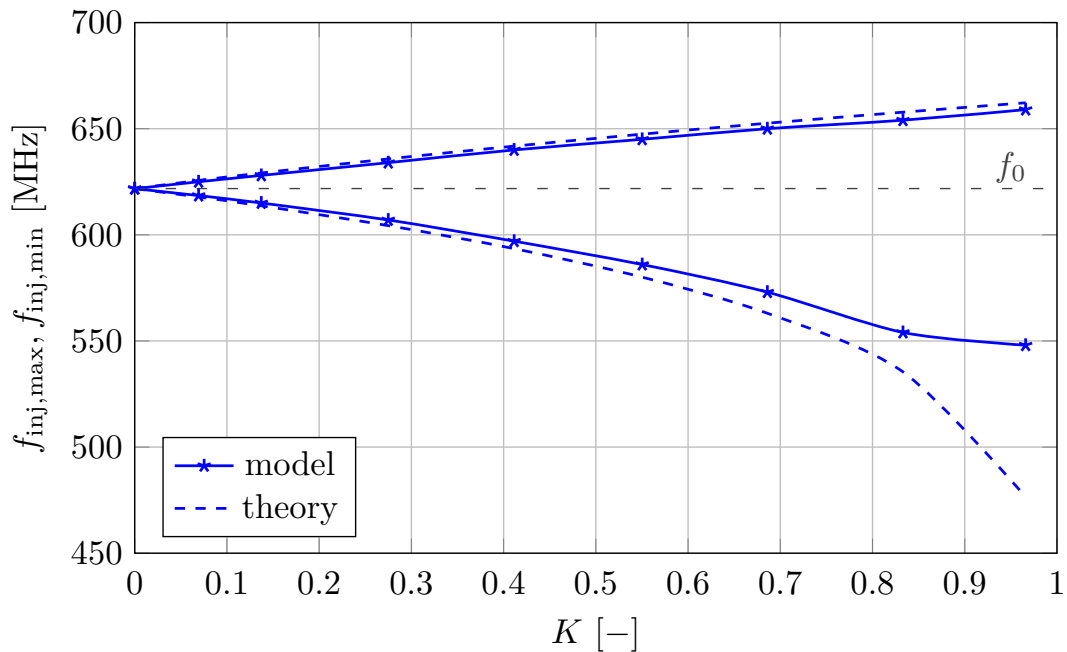


Fig. 4.8: Locking range of the ILRO model as a function of injection ratio

Based on the charts on Figure 4.8 and Figure 4.9, this symmetry of the locking range only holds for $K < 0.3$, which is the range of interest for most FTDC applications. The reason why for higher K the center of the locking range strays away from f_0 towards the lower frequencies is unknown. It could be a physical effect manifesting also in device level simulations, or it might be a quirk of this type of

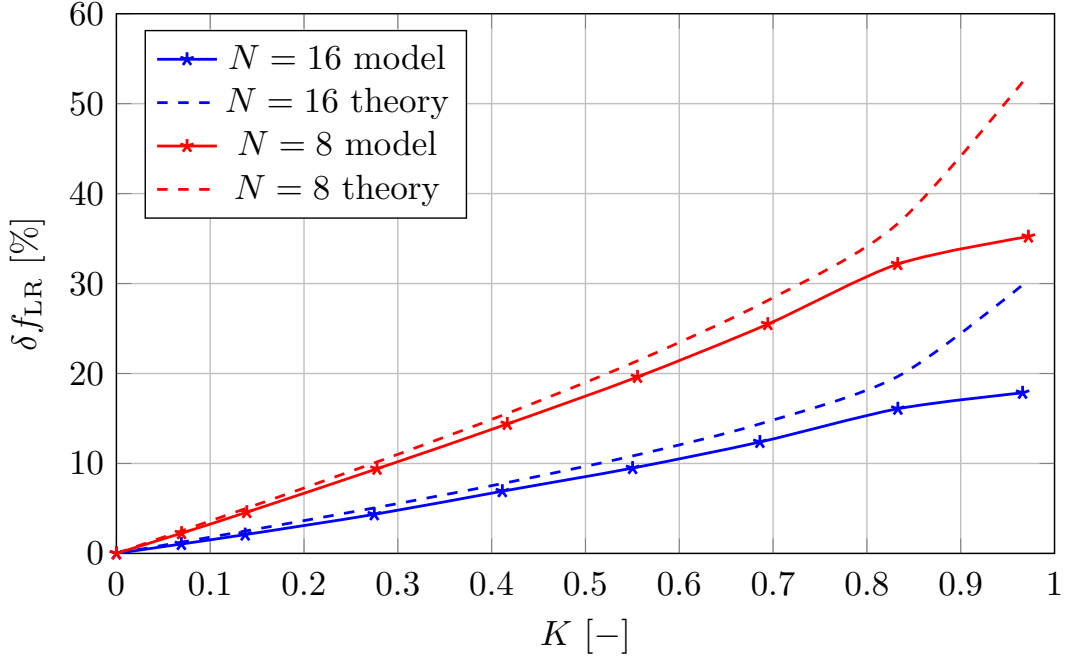


Fig. 4.9: Relative locking range of ILRO models as a function of injection ratio

models, as it is predicted by both the equation Equation 3.64 and this Simulink model.

The locking range predicted by Equation 3.64 and determined with the Simulink model match well for $K < 0.3$. The discrepancy between the analytical prediction and the model is significant only for the $f_{inj,min}$ limit when the injection ratio is high. The analytical model behaves hyperbolically in this region, which is clearly not a physical result, while the Simulink model clearly approaches a finite value.

The fact that the Simulink model and the analytical expression match so well is interesting, as the analytical derivation assumed a different shape of the injection signal - for that derivation, a wide 50 % duty cycle square wave was used, while the Simulink model features short injection pulses. While the shape of the injection waveform might not affect the width of the locking range significantly, as the deciding factor is the amount of charge injected in total during the “sensitive period” as discussed in subsection 3.3.3, it does affect other parameters, as will be discussed later in subsection 4.4.5.

4.4.2 Time domain measurements

In this subsection, the time domain metrics used by the analytic ILRO model from section 3.3 such as the injection waveform phase shift Δ and its equivalent injection angle ϑ or the zero cross delay caused by the injection d will be measured.

First, it is necessary to define these quantities in context of the different injection

waveform. This is showcased on Figure 4.10. The injection phase shift Δ , or the injection angle ϑ respectively, were measured with a MATLAB script as the distance between the zero crossing instant of the differential voltage signal of the falling edge of the previous stage and the rising edge of the injection pulse.

The zero cross delay d measurement is a bit more complicated. First, assuming the stage immediately preceding the so-called injection stage is the last stage, the time between the zero cross of the second to last stage and the last stage is measured by a MATLAB script. This time interval is unaffected by the injections in this model. Then, the time interval between the zero crossing of the previous stage and the injection stage is measured. The difference between these two time measurements is d .

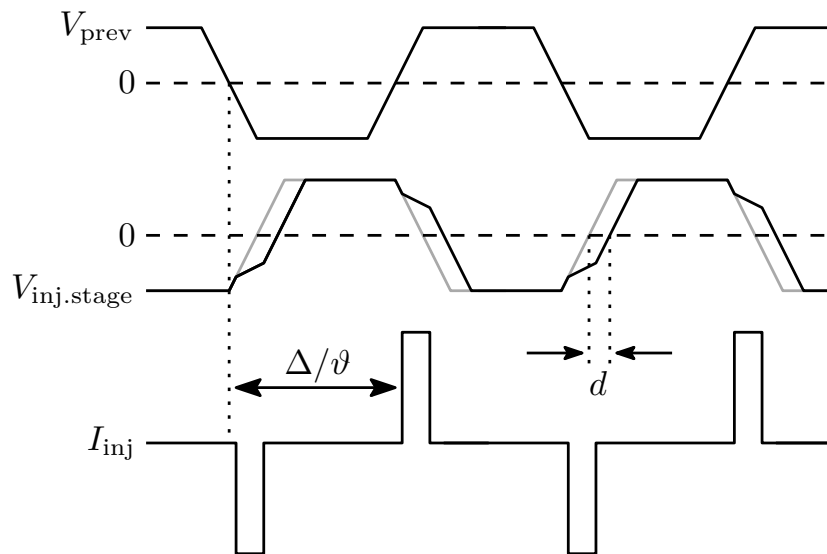


Fig. 4.10: Definition of time domain measurements

The charts mapping the various time domain metrics against f_{inj} are shown on Figure 4.11, Figure 4.12 and Figure 4.13. The injection frequency in these charts is swept from 607 MHz to 634 MHz, as that is the locking range for the sixteen stage ILRO model with default simulation parameters as listed in Table 4.1.

In the charts plotting Δ or ϑ , it is clearly visible that when $f_{inj} > f_0$, the injection pulses aligns in phase with the injection stage oscillation waveform and hasten its zero crossings. On the other side of the locking range, when $f_{inj} < f_0$, the injection waveform is in the opposite phase, delaying the zero-crossings of the injection stage. The transition between these two regions is rather sharp for reasons which will be explained shortly.

In contrast to the charts of Δ or ϑ stands Figure 4.13, depicting the $d = f(f_{inj})$ function, which is nearly perfectly linear (linear regression yields $R^2 = 0.9997$, signalling precise fit). This is not surprising. When the oscillation frequency of the

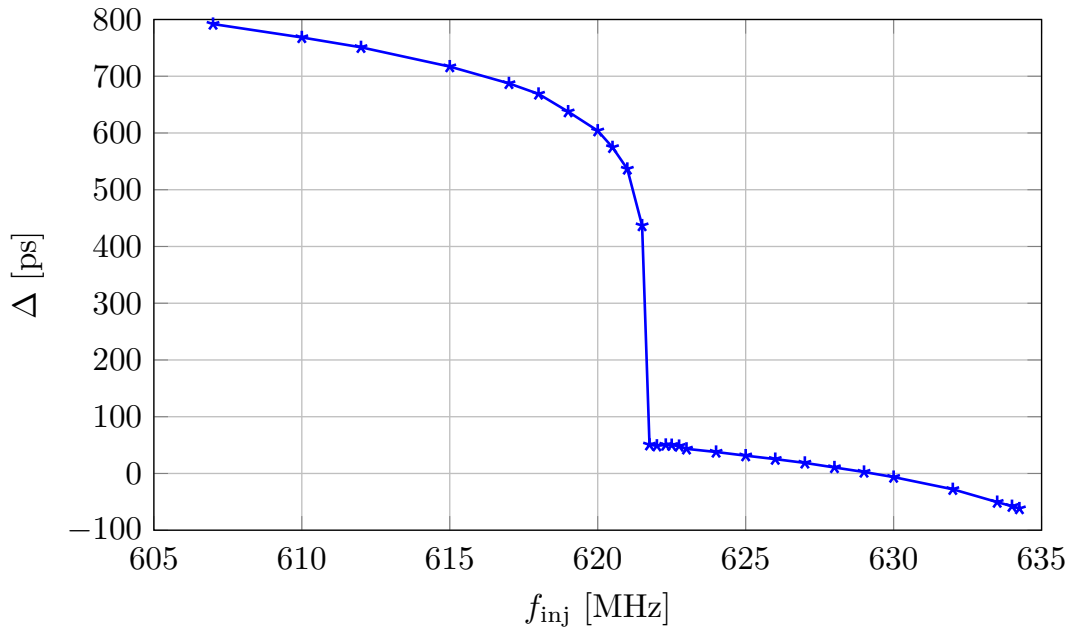


Fig. 4.11: The phase shift Δ of the ILRO model as a function of f_{inj}

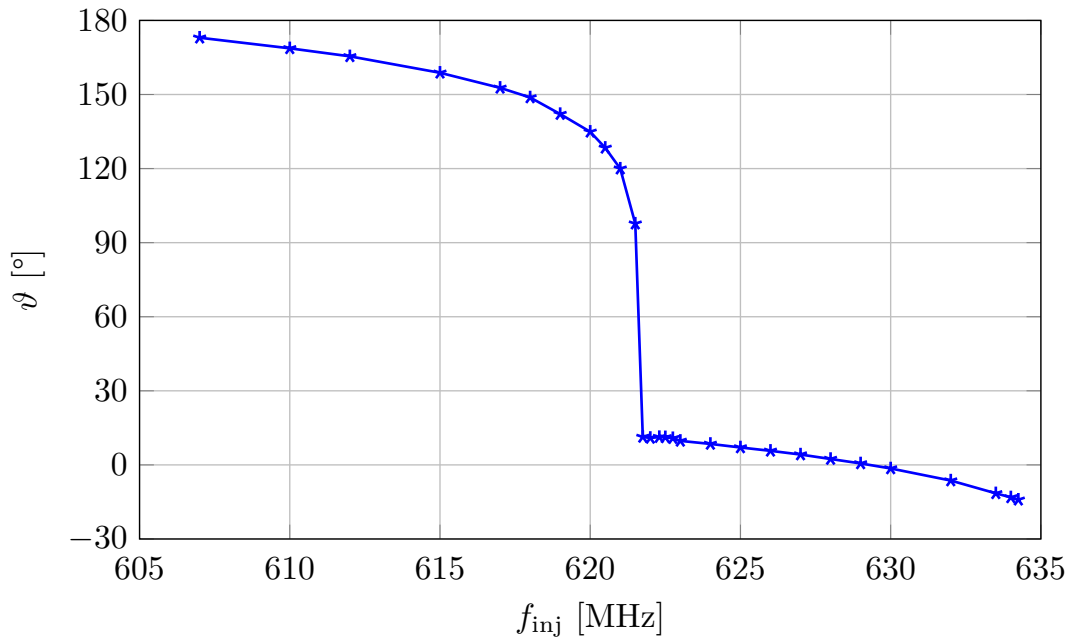


Fig. 4.12: The injection angle ϑ of the ILRO model as a function of f_{inj}

ILRO needs to match $f_{\text{inj}} \neq f_0$, a non-zero value of d is required, and the oscillatory period T_{osc} is a linear function of d as shown in Equation 3.56.

An interesting quirk of the chart on Figure 4.13 is that while at the lower end of the locking range, the d can reach nearly 20 ps, at the higher end of the locking range, d can only reach approximately 16 ps. The source of this asymmetry is unknown, but is clearly identical to the source of the asymmetry of the locking range itself, as discussed in subsection 4.4.1.

On the figure Figure 4.14, the function $d = f(\vartheta)$ is shown. This chart explains why the $\Delta = f(f_{\text{inj}})$ and $\vartheta = f(f_{\text{inj}})$ functions feature such an abrupt transition in the vicinity of f_0 . The reason is that the injection pulses can only really significantly affect the zero-crossing delay during the “sensitive period” as discussed in subsection 3.3.3. When $\vartheta \approx 0^\circ$, the injection pulse adds to the rising edges, when $\vartheta \approx 180^\circ$, the injection pulse subtracts from the rising edges. When ϑ is in-between (approximately 30° to 90°), the injection pulses align with the flat parts of the injection stage voltage waveform, where they have little to no effect, producing no significant change to d .

An interesting feature of the function shown on Figure 4.14 is that the function is asymmetric, i.e. the positive d and the negative d parts have a different shape (the positive d part is much less linear). The reason for this asymmetry is not known, but it might be a quirk of models approximating inverter gate transitions with exponential RC circuit charging waveforms, and/or might be linked to the source of the locking range asymmetry.

4.4.3 Phase noise

Another analysis that was performed is the *PSD* analysis. The goal here was to determine whether or not injection locking improves phase noise performance of the oscillator.

Two 50 μs long simulations of the differential voltage at the last stage before the injection stage were performed. Noise was injected into the injection stage with the “Band-Limited White Noise” block (`noise_power = 10-22`). In one case, the oscillator was free-running, in the other, it was locked to $f_{\text{inj}} = 618$ MHz. The first 10 μs of the simulations were cut off due to transients at the start of the oscillations. The rest of the signals formed 450002 points, which were processed with the `pwelch()` command to obtain Welch’s *PSD* estimate. A 112500 points long Hann window was used to provide a clearer plot. The result is shown on Figure 4.15, where the oscillation frequency was normalized to enable comparative analysis.

It can be clearly seen that injection locking does improve phase noise of ILROs in the vicinity of the main harmonic. This was explained for LC tank based ILOs

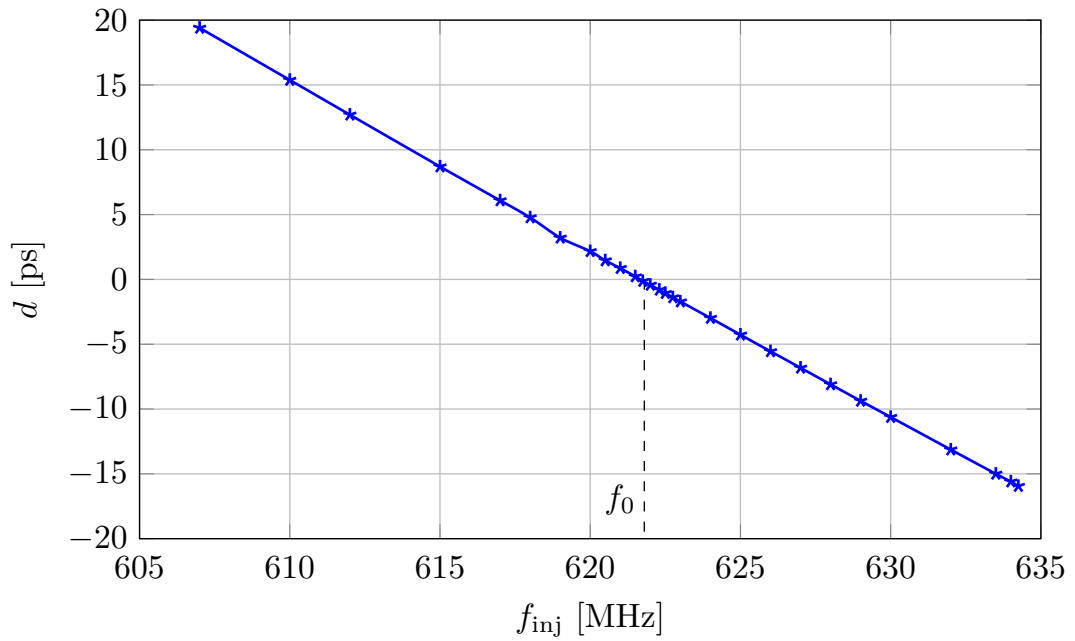


Fig. 4.13: The zero cross delay d of the ILRO model as a function of f_{inj}

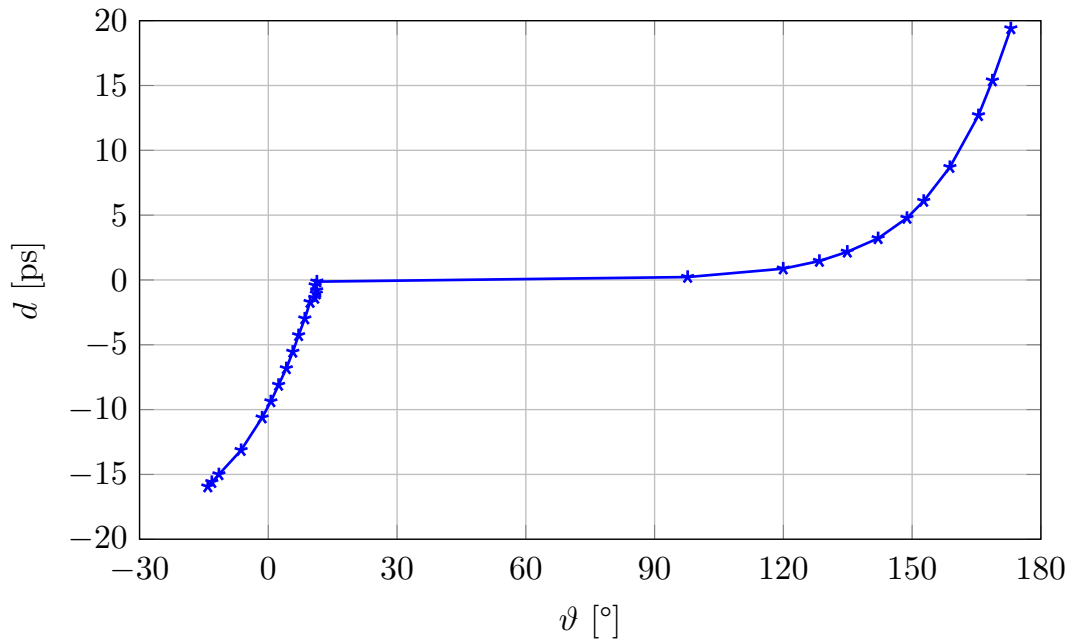


Fig. 4.14: The zero cross delay d of the ILRO model as a function of ϑ

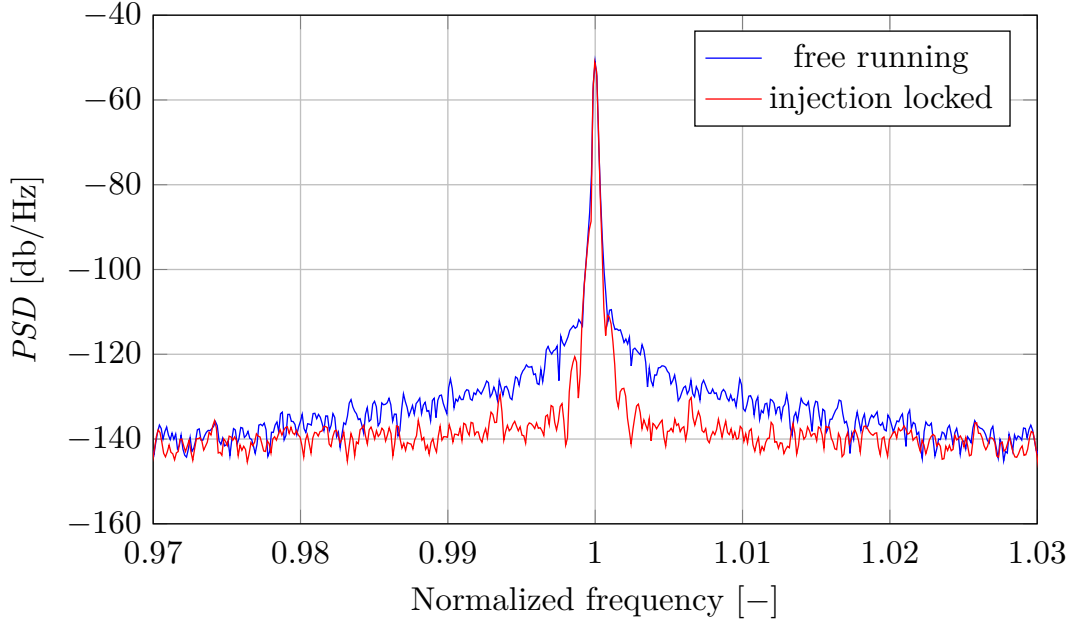


Fig. 4.15: Power spectral density of a free-running and injection locked ILRO

back in subsection 3.2.3, but was not discussed for ILROs.

The reason why the phase noise is improved is that without injection locking, ring oscillators accumulate jitter [56, p.46]. This means that the jitter of each transition, i.e. the difference between the ideal zero cross time and the real one, directly affects the zero cross time of the next stage and so on. Injection locking can overcome the jitter by forcing the transitions to occur at a specified frequency, essentially resetting the accumulated jitter with each injection. The noise is not removed altogether, but it does not accumulate in the vicinity of the oscillation harmonics any more.

In subsection 3.2.3, it was determined that the frequency region where the phase noise is improved corresponds to the locking range. According to the Simulink model, this is not the case. Instead, the region seems to be about half of the locking range.

4.4.4 Injection clock duty cycle

In this section, the sensitivity of the ILRO to the injection clock duty cycle variation will be examined. The Figure 4.16 examines the effect of the injection clock duty cycle variation on the injection pulses.

Based on the quantities highlighted on Figure 4.16, the duty cycle of the injection clock, and by extension also the duty cycle of the injection pulses, can be calculated with the following equation.

$$DC = \frac{T_1}{T_{inj}} = 1 - \frac{T_2}{T_{inj}} \quad (4.3)$$

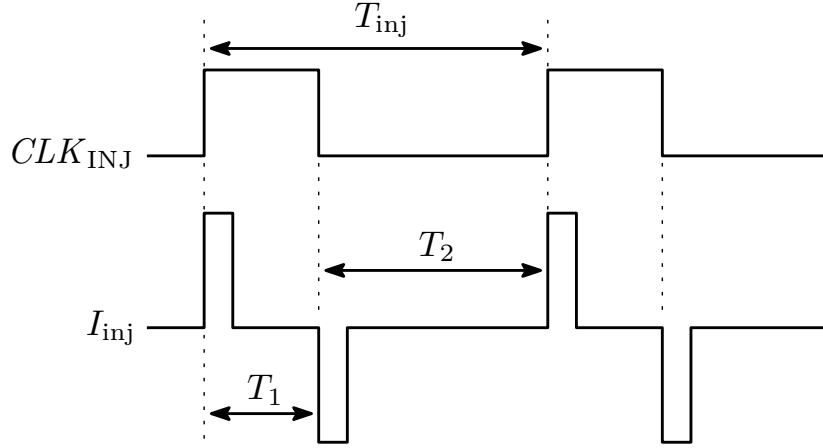


Fig. 4.16: The effect of injection clock duty cycle on injection pulse timing

The effect of the duty cycle variation on locking range is shown on Figure 4.17.

As expected, the duty cycle variation certainly does not benefit the locking range, but the shape of the curve is interesting. Naturally, it is symmetric. At first, it falls off quite rapidly within $\pm 10\%$ of duty cycle variation, but flattens from there onward. The reason is that for such asymmetric injection clocks, if the oscillator is able to lock at all, one polarity of the pulses completely misses the “sensitive window” of the oscillation waveform. Varying duty cycle even more does not seem to hurt the locking range as much, as the pulse polarity which is not delivering its charge efficiently only weakly affects the oscillation frequency of the ILRO.

The “single polarity locking” is showcased on Figure 4.19, where the transient waveforms of the ILRO are plotted. It is clearly visible that the oscillator locked to the negative polarity pulse, as the positive pulse occurs *after* v_{inj} crosses zero, thereby not affecting the zero cross timing d at all. On the other hand, the negative polarity pulse performs as intended, because it occurs before the zero is crossed.

The relative locking range of both eight stage and sixteen stage ILRO models as a function of duty cycle is shown on Figure 4.18. Since the two models have different relative locking range (as seen on Figure 4.9), their relative locking range has been normalized to 100% at $DC = 50\%$ to allow easier comparison. The plot shows a difference of shape - the eight stage model seems to be less sensitive to the injection clock duty cycle.

4.4.5 Injection pulse width

The final analysis of interest in this section is the analysis of the effects of varying injection pulse width. As was already discussed in subsection 4.4.1, the locking range does not seem to be affected by the shape of the injection pulses.

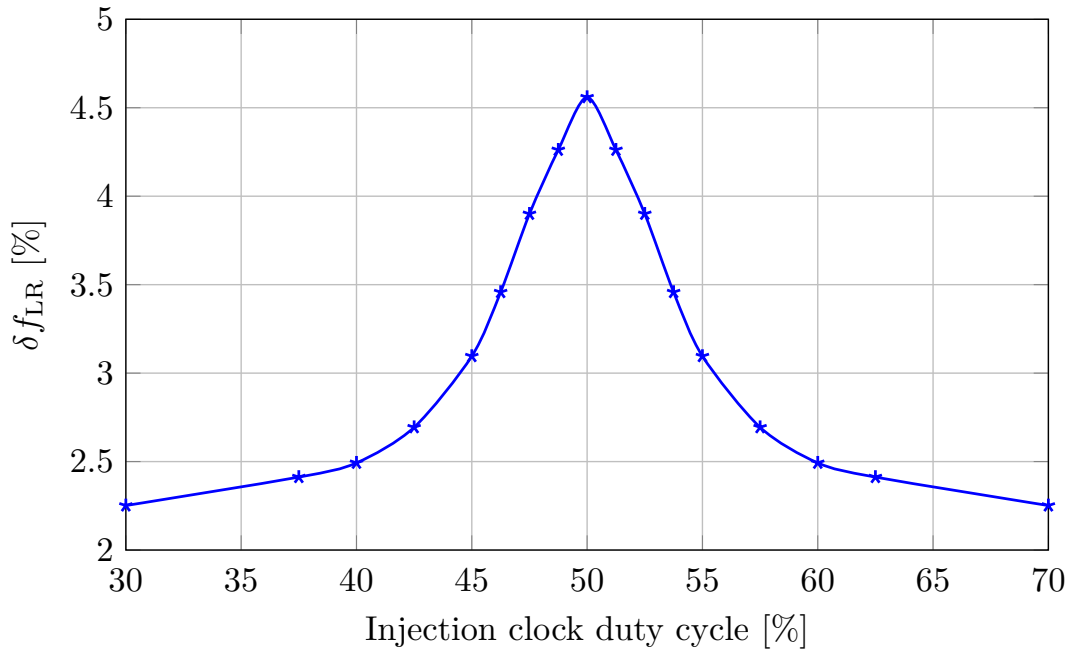


Fig. 4.17: Relative locking range of the ILRO model as a function of injection clock duty cycle

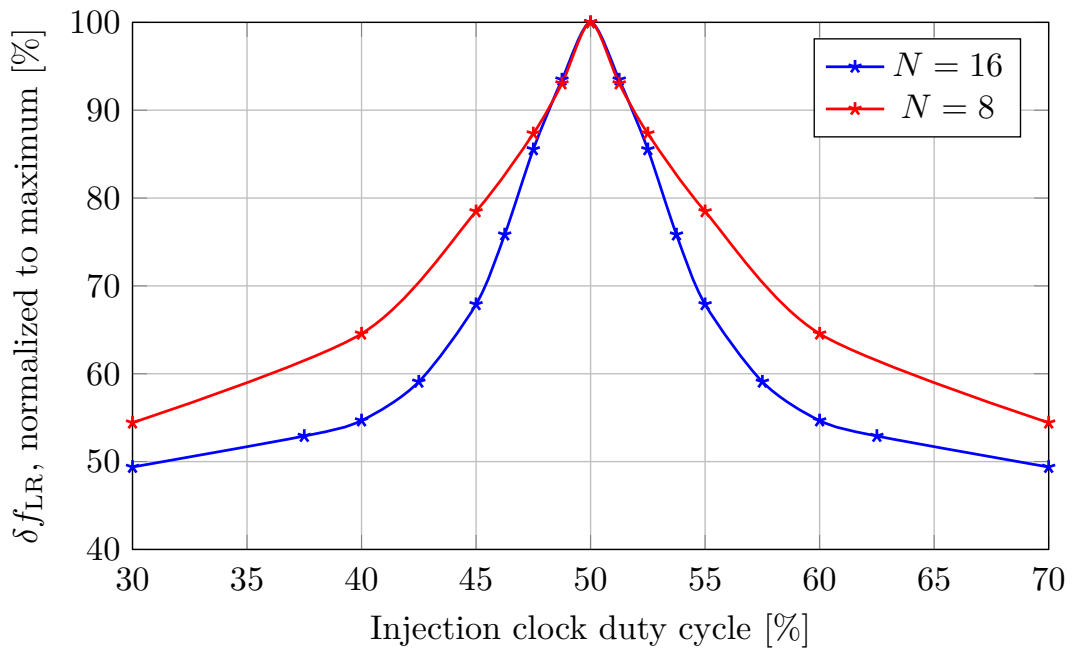


Fig. 4.18: Normalized relative locking range of ILRO models as a function of injection clock duty cycle

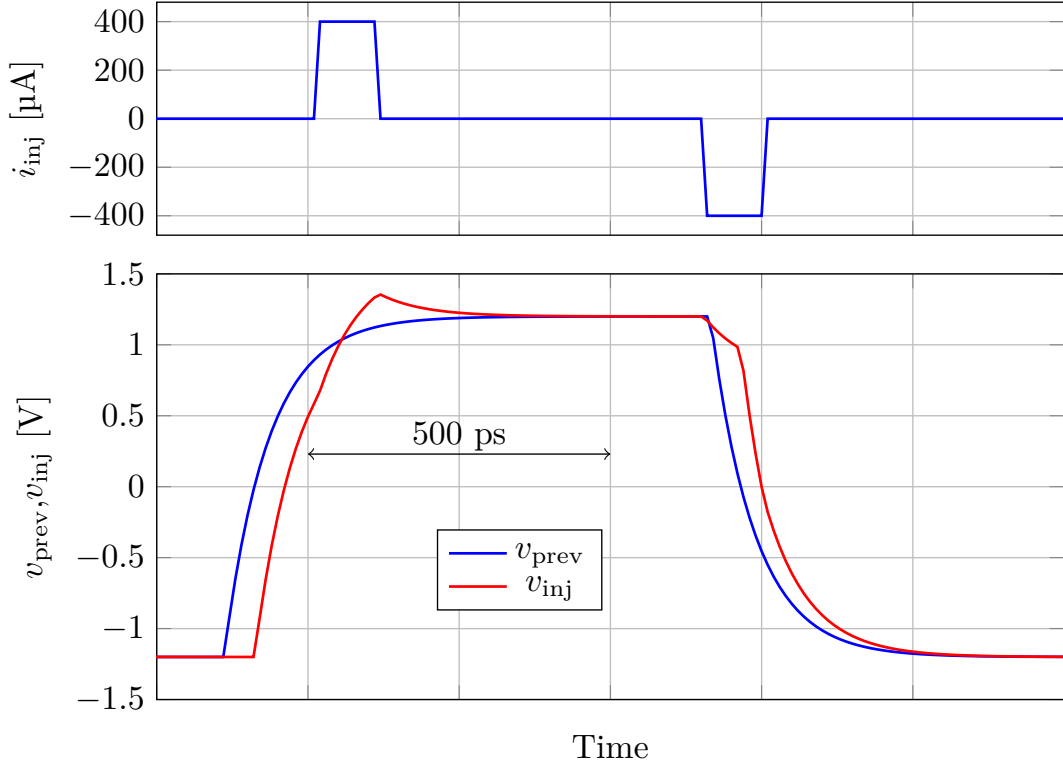


Fig. 4.19: ILRO model period of oscillation for $DC = 40\%$, $f_{inj} > f_0$

However, the pulse width can affect the sensitivity of the ILRO to injection clock duty cycle variation significantly, as shown on Figure 4.20.

For this analysis, the default pulse width $pw = 100$ ps and height $a_{inj} = 200$ μ A have been altered. Three variations of the pulse width were made. To achieve comparative results, a_{inj} was fine tuned so that the locking range stayed the same, i.e. for the 25 ps pulse, the amplitude had to be increased, while for the 400 ps pulse, it had to be decreased etc.

The results clearly show that wider pulse width helps massively with the sensitivity to the duty cycle. This can actually be explained with the “sensitive window” approach again - when the pulses are wide, altering the timing of one of the polarities within reasonable range still results in some overlap with the “sensitive window” of the oscillation waveform. On the other hand, some part of the injection pulses spends its charge outside of the window, burning power inefficiently.

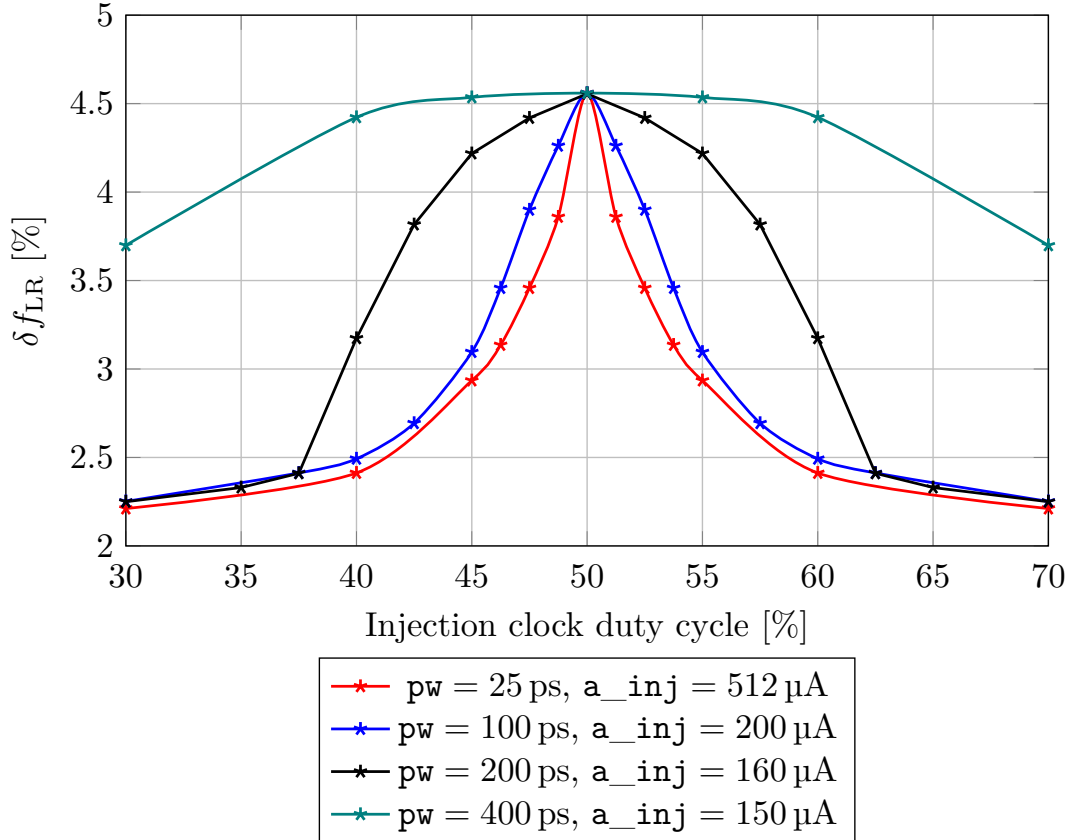


Fig. 4.20: Relative locking range of the ILRO model for various injection pulse sizes

4.4.6 Summary

In this section, several analyses have been discussed.

First of all, locking range equations derived in section 3.3 were verified. For $K < 0.3$, the equations match the Simulink model within $\approx 0.3\%$ for both the eight and sixteen stage models.

Secondly, the injection angle ϑ or the zero cross delay d were evaluated versus f_{inj} or against each other. These analyses confirmed the intuitive understanding of injection locking proposed by the “sensitive window” hypothesis (described in subsection 3.3.3), at least within the context of the RC circuit based ILRO models.

Afterwards, phase noise analysis was presented, again confirming the findings presented in subsection 3.2.3. The phase noise of the ILRO model improves when injection locked as expected, although the range of frequencies where this improvement occurs seems to be narrower than predicted by [53] for harmonic ILOs.

A novel analysis was described next, as the effect of the injection clock duty cycle was investigated. There are two key insights: asymmetric injection clock hurts the locking range of the ILRO, and this is more severe the higher the number of stages of the ILRO.

Finally, the effect of the width of the injection pulses was examined. Naturally,

the locking range changes when the pulse width is adjusted. This can be compensated for by also adjusting the height of the injected pulses. Although this fine tuning achieves the same locking range, the sensitivity to duty cycle variation of the injection clock is nevertheless different. Simulations have shown that wider injection pulses lead to less sensitivity to the injection clock at the cost of higher power consumption. This has been explained again with the use of the “sensitive window” model.

These last two analyses provide insights which could be valuable for device level design of the ILRO and its auxiliary blocks. Although the source of the injection clock is usually a PLL followed by a clock divider, and the injection clock is therefore symmetric enough, routing the clock through long multi-level clock trees can degrade the symmetry. The duty cycle variation can be compensated for during injection circuit design by widening the injected pulse width and decreasing the pulse height. This, however, increases the total power consumption, as the overall amount of injected charge is now higher. Power consumption of each ILRO is important though, as in TDC arrays, there will be a high number of them. A performance trade-off has been therefore identified before the device level design started.

5 INJECTION LOCKED RING OSCILLATOR DESIGN

In this chapter, the design of an ILRO based FTDC for application in LIDAR systems will be presented.

The aim of this thesis is to design and simulate an ILRO meant to be used in a DToF LIDAR FTDC along with its biasing circuits, which consist of a DLL. The assignment of this thesis does not contain the design of the CTDC, the PLL which drives the clock tree nor the latches and digital logic used for decoding the ILRO clock phases into a time stamp. These blocks will be however commented and key requirements for these blocks will be mentioned, as they operate in close conjunction with the circuits designed in this thesis.

Firstly, the manufacturing technology ONK65 will be briefly introduced. A few sections providing system level overview of the circuits to be designed as well as the way they could fit into an overarching architecture will follow and some high-level design decisions will be explained. The subsequent sections will delve deeper into device level circuitry of the ILRO itself and its operation with the DLL. Finally, the designed circuits will be simulated in conjunction to verify their performance.

5.1 Technology overview

The manufacturing technology ONK65, developed by Global Foundries, is a standard p-substrate based 65 nm CMOS process. The name of the technology library used in this project is *CMOS10LPe*, which is a library focused on low power digital applications. The library contains somewhat limited options for analog designers especially as far as bipolar devices are considered, which is the standard for common digital oriented process nodes. Further options for high performance analog circuitry such as high-gain asymmetric MOS devices, inductors etc. are included in the *Radio Frequency* (RF) design kit, which however requires additional masks and which will not be utilized in this thesis.

The name ONK65 does not actually come from the smallest gate length of the 1.2 V MOS devices (also known as thin oxide devices), as the minimum length is actually 60 nm. There are also thick oxide devices available, which operate with 1.8 V, 2.5 V or 3.3 V supplies and their minimum gate length is 260 nm, 280 nm or 400 nm respectively.

The process offers three different *threshold voltage* (V_{th}) options for thin oxide transistors (low, standard and high) and a native V_{th} option for thick oxide devices.

These variants can be useful for certain analog circuits where the voltage headroom is restricting.

There are four resistor types available. The N-well resistor *nwres* does not require any additional lithographic masks, while the N+ diffusion resistor *opndres*, P+ polysilicon resistor *opppres* and the high yield P+ polysilicon resistor *oprppres* do. The two polysilicon resistors are of particular interest for analog designers due to their low *thermal coefficient of resistance* (*TCR*) and high sheet resistance. The high yield version offers smaller process variation than its counterpart at the cost of yet another mask and will not be used. The process offers a variety of metal interconnect based capacitors as well, but since this application does not require extremely linear and stable capacitors, standard MOScaps will be used instead.

This process features from 4 to 9 metal interconnect layers, filled with low-*k* dielectric. All wiring levels are copper except for the final one, which is aluminium. The operating temperature range for devices made in this process is the standard automotive range of $-40\text{ }^{\circ}\text{C}$ to $125\text{ }^{\circ}\text{C}$.

The *Process Design Kit* (PDK) is well equipped for analog simulation purposes with statistical variation support for Monte Carlo runs as well as a variety of pre-defined fixed process corners for faster simulations at the extremes of the variation space. The mixed MOS corners FS and SF (corresponding to fast NMOS & slow PMOS or vice-versa respectively) correspond to 3σ variation, and the FFF and SSF corners (the last F stands for *functional*) are available in 3σ , 4σ , 5σ and 6σ varieties. Resistor and capacitor process corners are only available in 3σ varieties.

5.2 Choice of ILRO operating frequency

The thesis assignment provides a single specification, the resolution of the TDC, which is 50 ps (and which corresponds to 7.5 mm via Equation 1.1). This is a demanding specification to reliably meet especially over PVT variation, but should be ultimately achievable, possibly with some trimming.

The resolution specification directly translates to the ideal value of the propagation delay of a single ILRO delay cell. Therefore, assuming the ILRO will be implemented in a similar way to the example shown on Figure 3.15, by selecting the number of ILRO stages, the frequency of the ILRO can be defined. This in turn defines the frequency of the CTDC, as the LSB of the CTDC is supposed to be “sliced” by the ILRO clock phases and therefore the frequency of the ILRO should match the clock frequency incrementing the counter.

In practice, a double-counting CTDC can be used for various reasons, one of which is the benefit of running at half the frequency of the ILRO (this will be further

explained in section 5.3), but either way, there is a direct relationship between the frequency of the CTDC and the ILRO.

Since the delay of a single ILRO cell specified, the larger the amount of its stages, the lower the operating frequency of the ILRO and the CTDC and vice-versa. This can be formulated with the following equation

$$f = \frac{1}{2Nt_d} \quad (5.1)$$

where f is the frequency of the ring oscillator, t_d is the propagation delay of a single oscillator stage and N is the number of stages of the ring oscillator. This equation holds true for both single-ended and differential ring oscillator implementations, and was actually analytically derived back in subsection 3.3.1.

In order not to waste space, it is desirable to encode the resulting FTDC time stamp in binary code without unused states. Therefore, N should be a power of two, which is only possible for differential ring oscillators.

The choice of N is a very important one, as it bears implications not only for the ILRO itself, but also for the preceding analog circuits and the CTDC counter. A list of pros and cons of choosing a longer delay line for the ring oscillator (as opposed to a shorter one) is shown in Table 5.1.

Tab. 5.1: Benefits and detriments of higher number of ILRO stages

lower frequency	more delay line units
✓ relaxed CTDC counter specs	✗ larger area usage
✓ lower dynamic power losses	✗ higher static power consumption
✓ simpler PLL design	✗ worse <i>DNL</i> due to mismatch and gradients

Perhaps the most critical is the CTDC consideration. Choosing a ring oscillator length which is too short might result in a frequency which the CTDC is not able to reliably operate at (due to setup and hold time constraints). This is essentially a technology limitation, defining the shortest feasible ring oscillator length.

The second most important factor is the power consumption. While the longer delay lines consume more static power simply due to the larger amount of delay cells present (which all need to be biased with biasing currents etc.), it is reasonable to expect that the dynamic power losses will be dominant due to the large amount of ILROs and their clock phases routed throughout the chip. Therefore, doubling the length of the delay line and halving the dynamic power consumption (via $P_{\text{dyn}} \propto CV^2f$) for an increase in area could be a reasonable trade-off.

The number of units should not get too large, however. The more the units, the more space they occupy, which means that not only more die area is used, but also

that the processing gradients are going to have a larger effect on the units and an increase in mismatch and therefore DNL is to be expected.

In this thesis, a decision has been made to design a 16-stage ILRO, which would oscillate at 625 MHz (as per Equation 5.1). As mentioned previously, a double-counting CTDC (discussed more in the following section) can run at half the rate, which is 312.5 MHz in this case. This is a frequency the CTDC in this 65 nm technology can reliably run at, but high enough to not require unreasonably large number of delay cells, which would cost not only more area, but also hurt the linearity of the TDC.

5.3 TDC architecture overview

Although this thesis focuses on the design of the ILRO and its biasing DLL, it is important to keep the overarching architecture of the whole TDC signal chain in mind. An example of such an architecture is shown on Figure 5.1.¹ Since there is no architecture definition in the thesis assignment, this is the architecture which the circuits will be designed for.

Since the diagram is quite complex, its description will be divided into several smaller sub-sections for clarity.

5.3.1 SPAD and TDC array architecture

As discussed in chapter 1, LIDAR photodetectors commonly contain hundreds of pixels and therefore a large number of TDCs is needed. The architecture therefore needs to be designed in a scalable way.

The number of TDCs is given by the number of pixels active at any given time. This depends on the LIDAR architecture as a whole, as discussed in section 1.1. If the whole pixel array is illuminated simultaneously (i.e. flash LIDAR), the number of TDCs matches the number of pixels and the whole TDC array has to be active at the same time, which means that only relatively small spatial resolution is feasible, as running a large TDC array consumes high peak power and cooling becomes a serious issue. If only a part of the pixel array is illuminated at a time (i.e. sequential flash or scanning LIDAR), the number of active TDCs can be much lower.

Let us suppose sequential flash architecture is used and both the VCSEL and SPAD arrays have a resolution of 320×240 pixels, which is known as *Quarter Video Graphics Array* (QVGA). Let us also suppose that there are always four tiles of the VCSEL array active at a time, each 10×4 pixels large. In other words, the VCSEL

¹This is a simplified diagram. For the sake of clarity, blocks such as clock buffers or synchronisers and such have been omitted.

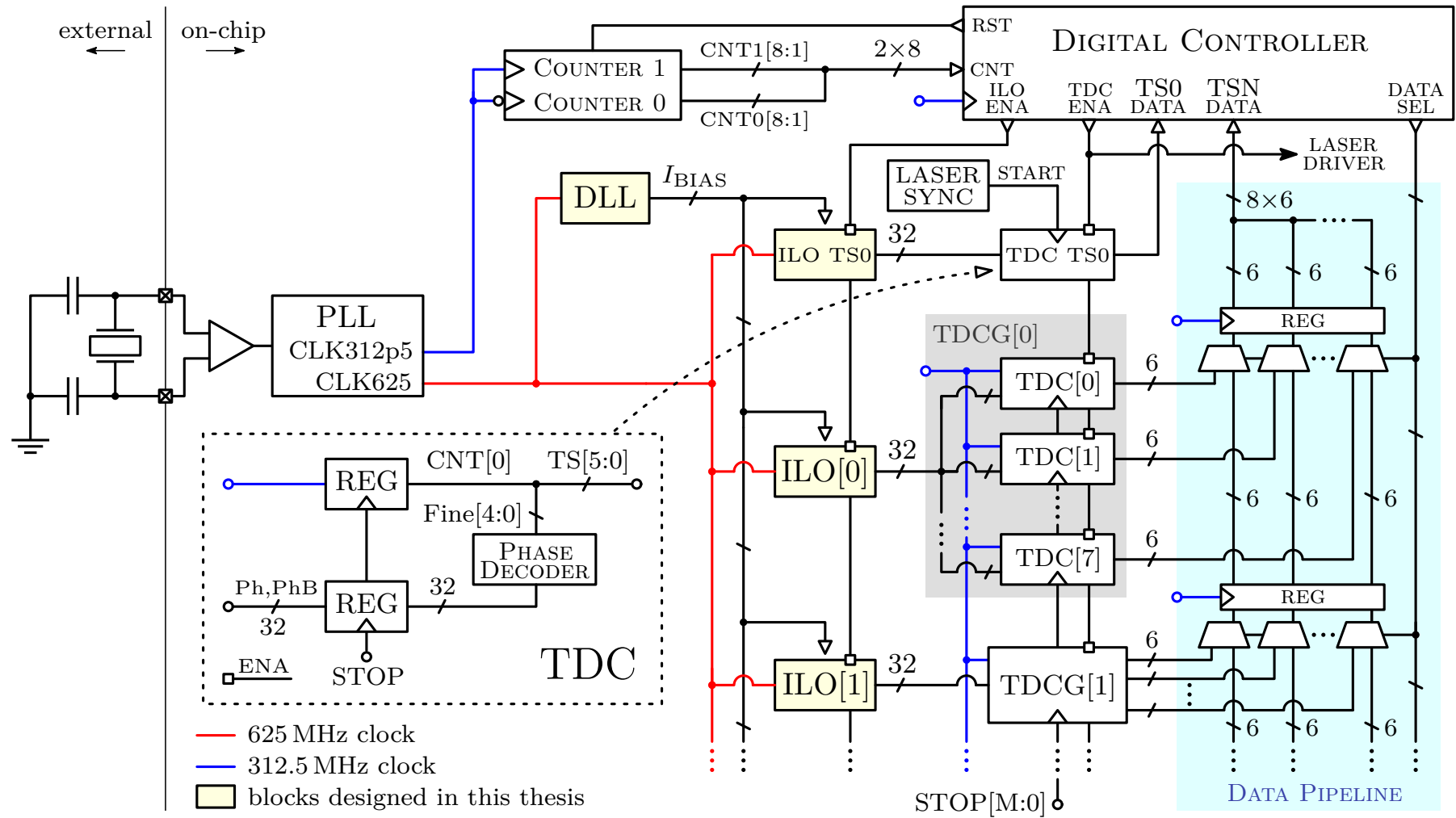


Fig. 5.1: Overview of an ILRO-based TDC architecture

array scans the FoV with four laser beams concurrently. Column by column, row by row, these 10×4 tile scan their sections of the array, their respective sections being called “clusters”. Since $320 \div 4 = 80$, each cluster is 80 pixels wide and 240 pixels tall. The scheme is shown on Figure 5.2.

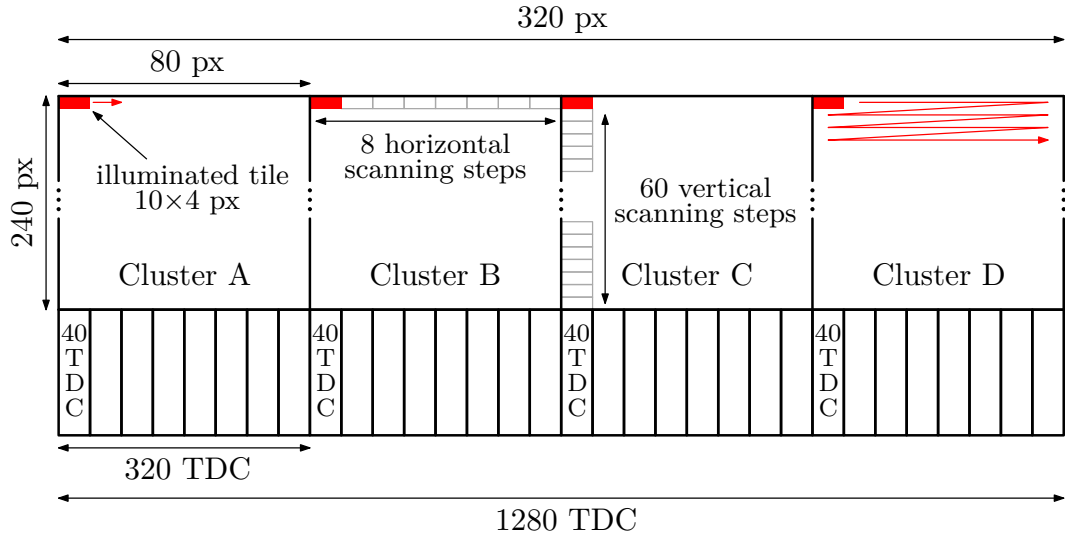


Fig. 5.2: SPAD & TDC array scanning and converting scheme

Upon reflection and arrival of the emitted photons, the photons should hit the corresponding area of the SPAD array (due to identical optical lensing systems, as previously described in section 1.1). Therefore, in each cluster of the SPAD array, only 40 SPADs are receiving valid signal at any given time, and therefore each cluster requires only 40 active TDCs.

Routing the SPADs to the TDC is a complex floorplanning problem. A reasonable approach is to implement an addressing scheme similar to memories, where each SPAD is addressable via column and row selecting buses. The detection signal from the SPADs needs to be sent to the TDCs in such a way to ensure that the propagation delay from each SPAD to each TDC is the same. Otherwise, a fixed pattern noise would appear, as some SPADs would always appear to trigger earlier/later than others, even if they absorbed photons concurrently. The routing and the logic along the way needs to be therefore perfectly symmetric.

Assuming the SPAD array is on a separate chip along with some basic readout digital circuitry, connected to the signal processing chip via 3D connections (stacked dies connected by through-silicon-vias or micro-bumps are the state of the art in SPAD imaging and signal processing systems [20, p.5]), the only way to achieve balanced routing known to the author is by implementing multiple TDC banks per cluster, each consisting of 40 TDCs, as routing all SPADs to one bank of TDCs would lead to serious issues with balancing the routing between various sections of

the SPAD cluster. This is because SPADs can be as wide as low tenths of micrometres and a 320 pixel wide array can be several millimetres wide. Techniques such as snaking to artificially lengthen the shorter paths would need to be employed, consuming area and most importantly increasing the potential for coupling of disturbances to the sensitive SPAD detection signal paths, which is something which needs to be avoided at all costs.

This means that each cluster requires not 40, but $8\times$ as many TDCs in total, one 40 TDC bank for each horizontal scanning step, as shown on Figure 5.2. Although this sounds like a high cost to preserve the propagation delay balance, the TDCs can be made rather small (they consist of only digital circuits, as will be discussed in the following sections) and only one bank per cluster is used at any given time, which means the others could be in power down mode to save power.

With regards to quantities like the refresh rate or the number of collected samples at each step of the scan, the following estimates can be calculated for the array shown on Figure 5.2. For example, let us assume that the target refresh rate is 10 Hz, which is reasonable for automotive applications. This means that the laser beams should be able to fully scan the pixel array in 0.1 s, and since there are $8 \times 60 = 480$ scanning steps, each step can last up to 208 μs . The length of an individual TDC conversion is defined by its maximum range – let us assume that it is roughly 1 μs , which corresponds to about 150 m maximum measurable distance (see Figure 2.1). In that case, approximately 200 laser pulses can be fired at each step of the scanning mechanism in order to build a statistically significant TCSPC histogram for each pixel of the array. If a larger histogram was desired, the refresh rate or the maximum measurable distance would have to be lowered.

5.3.2 TDC conversion timing scheme

The architecture presented on Figure 5.1 features standard START-STOP timing scheme, where the START signal will globally start the TDC conversion and individual STOP signals coming from the pixel array will stop their corresponding TDC, producing a timestamp. The reasons why the reversed timing scheme discussed in subsection 2.1.3 cannot be used are several. First of all, assuming each TDC has its own ILRO, enabling and locking each ILRO only after the corresponding START signal has triggered is problematic, as it takes some time for the ILRO to lock and taking a measurement of the ILRO clock phases before it locks onto the injected frequency would corrupt the measurement. Additionally, implementing one ILRO for each TDC is very area and power intensive. Instead, each ILRO will be used for several TDCs (as discussed in section 3.4) and therefore needs to be running at all times. To save power, only the ILROs driving the currently used TDC bank can be

enabled, as well as the ones which are going to drive the TDC bank which will be used in the next horizontal scanning step (see Figure 5.2), since these ILROs need to power up, settle and lock to target frequency in advance before their clock phases are used in the next step of the cycle.

While the reversed timing scheme from subsection 2.1.3 will not be considered in the proposed architecture, a certain modification of the sliding scale technique from subsection 2.1.4 will be. The reasoning is as follows. If a standard global START signal was used to fire a laser pulse and to start the TDC conversions at the same time, there would be a certain partly systematic, partly random error to the conversion. This is because there will always be a delay between the START signal triggering and the ideal *time stamp zero* ($TS0$). The ideal $TS0$ corresponds to the time instant the reflected laser pulse would be detected by the photodetector in case it was immediately reflected upon leaving the laser, i.e. when the distance from the LIDAR device to the obstacle would be effectively zero. The delay between the two is caused by the propagation delay of the START signal on its way to the TDCs and the laser driver, the inherent delay of the laser driver, the laser itself, the delay between the absorption of the photon in the SPAD and the activation of SPAD detection signal and the propagation delay of the detection signal on its way to the TDC. If the ideal value of $TS0$ was successfully acquired, it could be subtracted from all consequent time stamps produced by TDCs during normal operation to get a corrected timestamp, i.e. $TS_{\text{final}} = TSN - TS0$, where TSN are the time stamps produced by the TDCs during normal operation. This implements a sliding scale, as the reading is contained within the difference of two independent time stamps.

A novel way to acquire $TS0$ has been discovered during the work on this thesis. A patent application has been made and as such the invention cannot be publicly described until the application is processed. In Figure 5.1, the block is named “laser sync” and its implementation will not be presented. For the purposes of this thesis, which focuses on a different part of the TDC architecture anyway, it will be assumed that the START signal produced by this block is precisely the START signal needed to generate $TS0$, which can be then used to correct future time stamps in the current measurement cycle as defined above. More details on the operation of the TDCs themselves and the timing scheme of the whole architecture will be presented in the following sections.

5.3.3 Clock source, PLL and DLL

Before the description of the individual blocks in Figure 5.1 can start, the choice of the voltage domain will be addressed. The choice is ultimately quite simple. Since the circuitry making up the whole TDC signal chain needs to be as fast as possible

and operates in a somewhat digital or mixed-signal regime, the 1.2 V domain is clearly the best option and brings an additional benefit of lower area usage as well. The only classically analog circuit in the diagram is hidden inside the DLL block, and as will be shown in section 5.6, its design is feasible in 1.2 V domain as well.

The description will start from the beginning of the signal chain, which is the external clock source. This would be a crystal oscillator, which can provide a stable and spectrally clean clock signal, although at a relatively low frequency (higher tens or lower hundreds of MHz). This is why the first important block in the clock distribution scheme on-chip is a PLL. This PLL multiplies the frequency coming from the crystal oscillator to the double of the desired ILRO injection frequency (in this case 1.25 GHz), so that it can be halved again, producing a symmetric, close to 50% duty cycle clock.

This 625 MHz clock is the signal used to lock the ILROs and naturally also the DLL. It is also used for the double-counting CTDC, but only after its frequency is halved again to 312.5 MHz. The coarse counters will be discussed later in subsection 5.3.5.

The role of the DLL is to produce a PVT compensated biasing current I_{BIAS} necessary to set the total delay of a 16-stage delay line to 0.8 ns, which is the period corresponding to the chosen frequency of 625 MHz (not 1.6 ns, because there is a factor of 2 in Equation 5.1). In other words, the DLL produces a biasing current setting the propagation delay of each CCDLU to 50 ps, which is the specified resolution.

The design of the DLL will be presented in section 5.6, but the general principle of the PVT compensated biasing has been already explained in section 2.3 in chapter 2. It relies on the fact that the DLL delay line is essentially the same circuit made from the same number of identical CCDLUs as the ILRO. Therefore, the biasing current generated by the DLL to lock itself to 625 MHz is ideally the same as the current the ILRO needs to set its f_0 to 625 MHz as well.

In practice, this is obviously not the case. The primary reason is that the way the CCDLU is connected in DLL and ILRO topologies differs fundamentally, the DLL delay line being a pass-through circuit, while in the case of ILRO, the delay line is connected in a feedback system. This discrepancy can be decreased in magnitude by dummy cells helping to simulate the ILRO connection in the context of the DLL, but it is impossible to remove this discrepancy altogether. The consequence is that the free-running frequency of the ILRO biased by the same current which satisfied the DLL lock condition will not be precisely 625 MHz, but can actually differ by a few percent. Furthermore, even if this particular issue was somehow perfectly solved, there are more sources of error, such as the mismatch of the delay lines between the DLL and the ILROs themselves, the mismatch of the biasing current mirrors

distributing the current from the DLL, various error sources in the *Frequency Phase Detector* (FPD) and *Voltage-to-Current* (V-to-I) converter circuits and so on. The bottom line is that the DLL alone cannot set f_0 precisely.

Nevertheless, the DLL is still important in the context of the system, and that is because trimming needs to be included. There are two reasons for why trimming is necessary: first of all, trimming can eliminate the problems mentioned in the paragraph above and help the DLL supply the correct biasing current to the ILROs. Secondly, it solves the problem with propagation delay distribution in the ILRO, which was already foreshadowed in subsection 3.3.1. If the ILRO naturally oscillated at 600 MHz (either because of skewed biasing current or skew in the ILRO itself), its propagation delay would amount to 52.1 ps assuming perfect matching between the stages. When injection locked to 625 MHz, the propagation delay of the injection stage would need to adjust to 18.5 ps in order for the propagation delays to amount to half of the injection clock period. This would cause significant *DNL* of the FTDC. If the ILRO is somehow pre-trimmed to oscillate relatively close to the target frequency (for example within 2%), the *DNL* issue would be suppressed and as an added benefit, the ILRO would more easily lock to the target frequency, requiring less locking range and therefore less injected energy, conserving power.

Once trimming of individual ILROs is somehow incorporated in the system (more discussion on trimming implementation will follow in subsection 5.6.4), the DLL can fulfil its role of PVT immune biasing. This is because the problems causing the DLL to bias the ILROs with wrong current are static in nature and can be trimmed out. After trimming, the DLL can in principle correctly dynamically vary the biasing current depending on the current supply voltage or die temperature. This way the DLL helps to keep the ILROs locked even in extreme conditions. Simulations proving the usefulness of the DLL will be presented later on in section 5.7.

5.3.4 ILROs and clock phase latches

Each 16-stage differential ILRO outputs 32 clock phases, Ph[15:0] and PhB[15:0], where PhB are the negations of Ph. These clock phases can exist in 32 unique states, therefore they can be decoded into a 5-bit binary FTDC time stamp by the phase decoder, which is a piece of combinational logic in each TDC. The clock phases are shown in Table 5.2².

Each ILRO can be sampled by N TDCs. The precise value of N is mostly dependent on the layout. The goal is to route the clock phases only over short distances to minimize power loss and signal distortion. It is also necessary to distribute the

²The time step corresponds to the propagation delay of a single CCDLU. Negated phases PhB[15:0] are not shown.

Tab. 5.2: Clock phases of 16-stage differential ILRO

Time step / Code	Ph[0]	Ph[1]	Ph[2]	Ph[3]	Ph[4]	Ph[5]	Ph[6]	Ph[7]	Ph[8]	Ph[9]	Ph[10]	Ph[11]	Ph[12]	Ph[13]	Ph[14]	Ph[15]
0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
3	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
4	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
5	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
6	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
7	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
8	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
9	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
10	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
11	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
12	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
19	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
20	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
21	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
22	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
23	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
24	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
25	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0
26	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0
27	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0
28	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
29	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0
30	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
31	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

phases via balanced interconnects so that they arrive at the destination at the same time. This is easier if there are less TDCs per each ILO. On the other hand, having one ILO for each TDC takes a lot of area and power. A reasonable value for N seems to be 8, which will be assumed in this thesis and has been depicted in Figure 5.1, where each TDC group (TDCG) consists of 8 TDCs connected to the same ILRO. This also enables easy construction of the 40 TDC banks, as each bank would contain 5 such TDC groups. As a sidenote, this means that if the VCSEL and SPAD pixel arrays were implemented as described in the example from subsection 5.3.1, the total amount of ILROs would be $1280 \div 8 = 160$.

The ILRO clock phases pass through a series of buffers growing in strength and are finally connected to a latch in the TDC block. An example of what the latches can look like is shown on Figure 5.3. The differential implementation allows the latch to take advantage of the differential operation of the ILRO to improve the signal integrity and shorten the transition times.

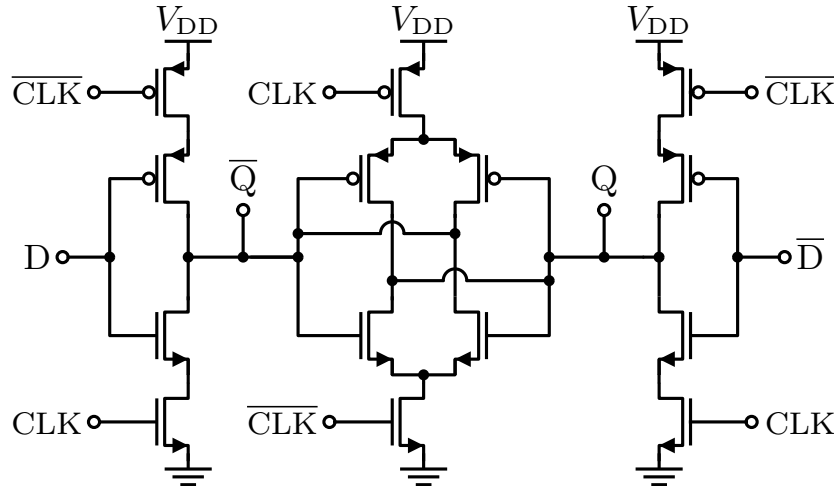


Fig. 5.3: Differential ILRO clock phase latch example

Since the STOP trigger is an asynchronous signal (in relation to the ILRO clock phases), there are two main goals for the latch design: minimize the metastable window (the sum of its setup and hold times) to reduce errors as much as possible as was already discussed in section 2.4, and in the case of setup and/or hold time violation, ensure the digital output quickly settles to a defined digital value as soon as possible and does not stay in the middle of the rails for a prolonged period of time. The circuit above can fulfil the latter requirement, and the worst case metastable window of the latch in ONK65 has been simulated to be 30 ps. This value is sadly above the theoretical quantization noise level (approximated by the well-known formula $LSB/\sqrt{12} = 50 \text{ ps}/\sqrt{12} = 14 \text{ ps}$), but this is ultimately limited by the technology. However, since the occurrence of metastability is essentially random, it also acts as noise, so in principle, it can be filtered out in the digital

domain by histogram building (see section 1.5), as long as a sufficiently large set of samples is taken. Either way, the design of the latch is out of the scope of the thesis assignment.

5.3.5 Coarse TDC counters and digital controller

The first design choice related to the CTDC counters is their bit width, as this will ultimately define the maximum measurable ToF and therefore the maximum distance. In practice, the maximum distance is defined by laser power, atmospheric conditions or obstacle reflectivity, so the coarse counters should be sized to ensure its size is not the limiting factor. In the architecture shown on Figure 5.1, a 9-bit CTDC is considered, which, assuming it runs at 625 MHz, corresponds to maximum measurable distance of 123 m (as per Equation 1.1). If the limitations of the optical system allowed practical measurements of longer distances, more bits could be added to the counters and the relevant digital circuits. This however takes more area and increases the requirements for the throughput of the data buses, histogram building circuits etc.

The actual implementation of the coarse counters is not the subject of this thesis, but some possibilities were presented in section 2.2. In this thesis, it is assumed the counters are synchronous.

The counters shown in Figure 5.1 implement a double-counting CTDC, as they count at a 180° phase angle in relation to each other. The leading counter CNT0 is active on the falling edge of the clock, while the trailing counter CNT1 is active on the rising edge. Therefore, CNT1 preserves the previous value of CNT0 for half a clock period after the CNT0 updates.

The main benefit of this scheme is the prevention of missing codes caused by the phase misalignment of the CTDC and the ILRO clock phases. While ideally the two blocks should be running in phase thanks to the injection locking mechanism, in reality, there always is some phase shift due to varying and possibly not-equal parasitic delays along the two diverging signal paths. When the double-counting scheme is used, a copy of the 312.5 MHz clock delayed by 90° (easily implemented with two flip-flops and the 625 MHz clock) can be used to decide which counter's value will be used for the final timestamp. In order to prevent the missing codes, the counter which has not updated in the last half of the clock period prior to the activation of the STOP signal has to be chosen. This will be shown on the timing diagram discussed in the following subsection 5.3.6.

As the design of the digital circuitry controlling the TDC is not the focus of this thesis, this technique will not be discussed further for the sake of brevity, but more details on the technique can be found in [11, p.69].

The LSB of the CTDC still corresponds to the ILRO frequency of 625 MHz, even though it runs at half the rate. The LSB reconstruction can be done by simply latching the state of the 312.5 MHz clock the moment the STOP signal triggers. This logic would be included in each individual TDC block, i.e. each TDC block has to accept the 312.5 MHz clock as well.

As depicted in Figure 5.1, a lot of functionality is offloaded to the digital controller. This is a block containing all the relevant digital logic necessary to produce the final timestamp, and its sequential circuits are running at 325 MHz. It includes the logic controlling the laser firing cycle and the TDC measurement cycle, as well as latches and logic registering and processing the values of the coarse counters. This digital controller then connects to more complex DSP circuits used for TCSPC histogram building. Some of its operation will be described in the following section.

5.3.6 Timing diagram

In this section the timing diagram shown on Figure 5.4 will be described. The diagram features the vast majority of signals depicted on Figure 5.1 and features a typical measurement cycle. It is heavily simplified, does not account for combinational logic delays and some signals might be optimized differently by digital engineers, as LIDAR systems are very complex systems and their digital timing controls are no exception. The point of the diagram is simply to show how the time stamp could be produced in principle, utilizing the circuits which will be designed in this thesis.

The first signal which has to activate is actually ILO ENA, which is not depicted on Figure 5.4. This signal enables the ILROs and the injections, and because starting the oscillator and locking it can take significant amount of time (simulations of the locking transient will be presented later in section 5.5), it needs to be done well in advance. In practice, there is no single ILO ENA bit. Instead, its a bus enabling the ILROs which are going to be needed in the next scanning cycle by the next TDC bank. Therefore, there are always 2 TDC banks (and therefore 10 ILROs) active at a time per TDC cluster – one of them being actively used and the other starting up in preparation for the next cycle. All the other banks can be kept in power-down.

When RST signal goes low, coarse counters CNT0 and CNT1 are released from reset and can start incrementing at the corresponding edges of the 312.5 MHz clock. At the same time, TDC ENA (also a bus enabling whole TDC banks at a time) goes high, enabling the sequential and combinational logic of the TDCs to operate. The digital controller managing the measurement cycle also signals the laser driver to fire a pulse, so this time instant is essentially the beginning of the conversion.

The first event of interest is the activation of the START signal. This signal

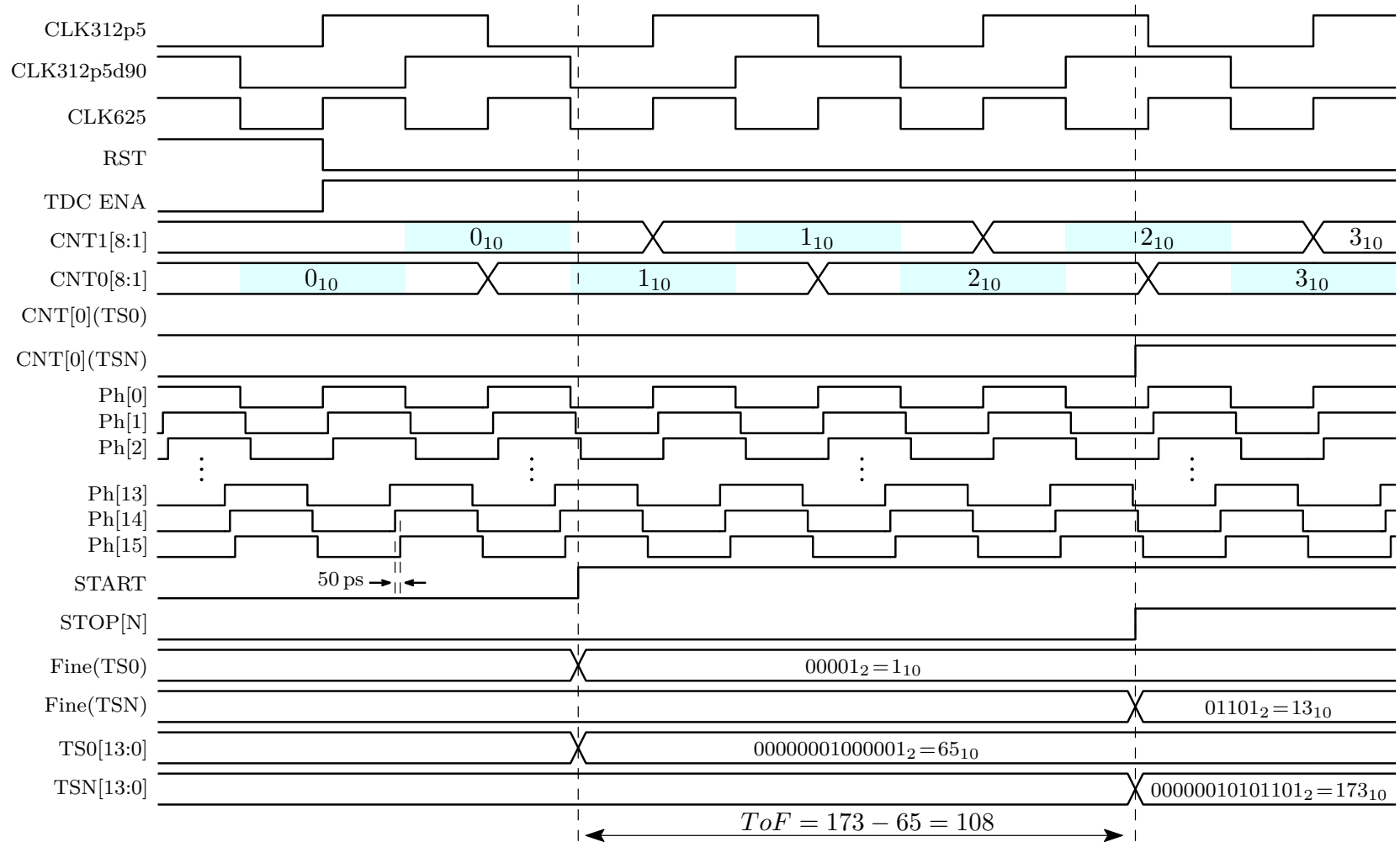


Fig. 5.4: Timing diagram

has been mentioned in subsection 5.3.2 and is produced by the “laser sync” block. Ideally, the instant the START signal activates is also the instant the earliest reflected photons could be detected by a SPAD in the pixel array, if they were reflected immediately upon leaving the optical system of the laser. Therefore, the START signal defines a true time zero and its timestamp TS0 is produced.

All timestamps consist of two main parts: the coarse part, i.e. the state of the coarse counter at the time of the START activation CNT[8:0], and the fine part Fine[4:0]. These are simply joined to form a timestamp TS[13:0], where the fine part is less significant than the coarse part.

In order to produce the coarse part, first, it needs to be determined whether to use the value stored in CNT0 or CNT1. This can be done by latching the state of CLK312p5d90 (the 90° shifted 312.5 MHz clock) – if it is low at the time of START activation, CNT0 is used, otherwise CNT1 is used³. This is done internally in the digital controller and is not depicted on Figure 5.1. In the case shown on Figure 5.4, CLK312p5d90 is low when START triggers, therefore the current state of CNT0 (1 in decimal) is used. This way the coarse counter bits CNT[8:1] have been determined, but the LSB still needs to be reconstructed. This is done in each TDC individually by latching the current state of the CLK312p5 clock. In the case shown on Figure 5.4, CNT[0] for the TS0 timestamp is zero. CNT[8:0] is therefore 000000010₂ or 2 in decimal representation.

At the same time, the state of the Ph[15:0], PhB[15:0] is latched and decoded in the phase decoder. Using Table 5.2, it can be decoded that the phases depicted on Figure 5.4 correspond to code 00001₂. Finally, by joining the coarse and fine parts, the timestamp TS0 can be constructed, amounting to 00000001000001₂ or 65₁₀.

The same process is repeated for the timestamp TSN, which is produced later when the STOP[N] signal (corresponding to the Nth pixel of the pixel array) fires, hopefully due to an incoming reflected photon. The only two differences in the timestamp production this time are that the LSB of the coarse counter happens to be high (as the state of the CLK312p5 clock is high when STOP[N] triggers), and CNT1 is chosen (as CLK312p5d90 is high at the moment of the trigger as well). The resulting timestamp TSN[13:0] is 173₁₀.

As was described in subsection 5.3.2, the ToF can be calculated by subtracting TS0 from TSN. For the case shown in Figure 5.4, the ToF code is 173 - 65 = 108, which amounts to 5.4 ns or 81 cm.

An important point that needs to be made is that since the sliding scale technique is used (subtracting TS0 from TSN), the measurement is immune to phase

³This is highlighted with the blue areas on Figure 5.4, which depict which counter’s state would be used if the trigger triggered at the given time.

misalignment between coarse counters and ILRO clock phases. If there is any misalignment, it will be present in both timestamps and subtracted out. This of course assumes that the phase misalignment does not change over time, which should be guaranteed by design, as thanks to injection locking, the frequency of the ILROs should be exactly two times the frequency incrementing the counters.

5.4 Simulation conditions

In the following sections, simulation data from the Spectre circuit simulator will be shown. Unless otherwise specified, the Table 5.3 lists the simulation inputs used for these simulations. As is standard, the supply voltage is varied by $\pm 10\%$ (reflecting the limited accuracy of the voltage regulator) and the temperature corners exceed the operating range of the process by additional 5°C , which provides a safety margin in case of inaccurate models at the temperature extremes.

Tab. 5.3: Default simulation conditions

Quantity	Min.	Typ.	Max.	Unit
V_{DDA1V2}	1.08	1.2	1.32	V
V_{BG}	1.225	1.25	1.275	V
temperature	-45	27	130	$^\circ\text{C}$
MOS corners	SSF3, SF, typ, FS, FFF3			
resistor corners	hi3s, typ, lo3s			

Capacitor corners will not be used as all capacitors used in the following design will be made out of MOS transistors. Metal capacitors are not needed, as no designed circuits require highly linear low variance capacitors. Instead, the occupied area is the main concern and MOS capacitors provide higher capacitance per area due to the thin oxide thickness.

A common occurrence in testbenches is the need for a biasing current. Unless otherwise specified, the circuit shown on Figure 5.5 is used for creating a realistic biasing current source.

The circuit consists of a *Direct Current* (DC) voltage source providing V_{BG} voltage (nominally 1.25 V), which is applied to a P+ polysilicon resistor. This resistor's resistance is set in such a way to ensure the nominal current is precisely the current desired, i.e. $R = 1.25 \text{ V}/I$. In PVT corner simulations, the value of V_{BG} is varied by $\pm 2\%$, and 3σ corners are used for the processing spread of the resistor, as listed in Table 5.3. This circuit therefore essentially simulates a realistic biasing current generator made from a bandgap voltage reference of limited precision and a polysilicon

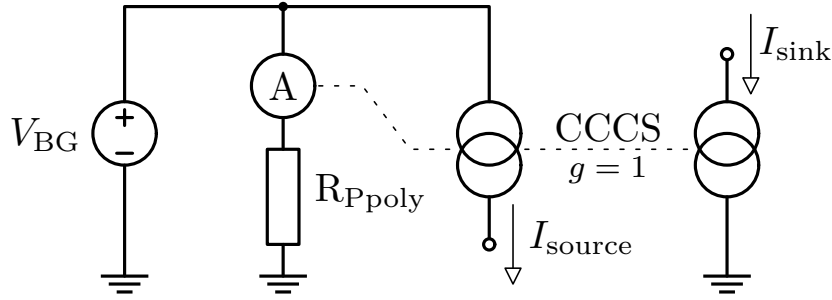


Fig. 5.5: Realistic biasing current generator

resistor. The current flowing through the resistor is sensed and copied by *Current Controlled Current Sources* (CCCSs), which finally feed this biasing current into the current mirror input diodes of the designed circuits.

5.5 Injection locked ring oscillator design

In this section, the ILRO circuitry will be presented. In the first subsection, the design of a single ring oscillator cell will be discussed, afterwards the biasing mirror design will be explained and the third subsection will focus on the implementation of the injection action itself. The final subsection will present simulations of the locking transient.

Although the design has to be described linearly, the real process was iterative over many cycles and therefore some design decisions might not be listed in chronological order.

5.5.1 Ring oscillator design

The choice of 16 ILRO stages requires the choice of a differential CCDLU topology. Such a topology has been already presented in subsection 2.3.1, and the final version of it is shown on Figure 5.6. Not depicted is the NMOS current mirror distributing I_{BIAS} for all CCDLUs of a given ILRO, which will be discussed later on in subsection 5.5.2.

Since the target for the propagation delay t_d is only 50 ps, which is a very short amount of time, it is important to be able to meet this specification with the lowest possible amount of current. Minimum-sized thin oxide MOS devices are the obvious candidate.

Furthermore, low V_{th} devices will be used for reasons illustrated on Figure 5.7⁴.

⁴The center line is the nominal corner, while the filled area is bounded by worst 3σ corners. The propagation delay has been measured between the inputs and the buffered outputs. Several CCDLUs have been connected in a delay line to simulate realistic driving and loading of each cell.

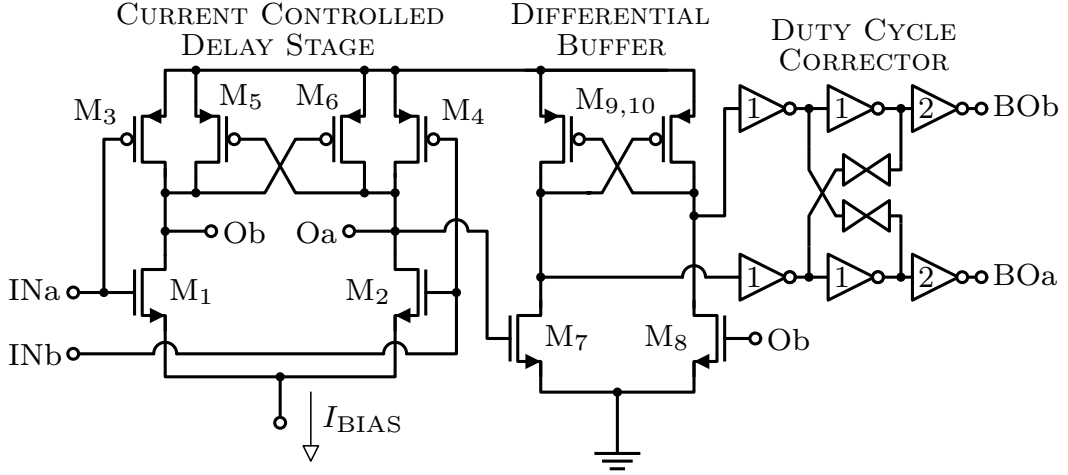


Fig. 5.6: Buffered differential CCDLU topology

Although the biasing current needed to reach $t_d = 50$ ps is quite close for the typical corner of both MOS device types (cca 11 to 13 μA), the problem is very visible for slow process corners (the upper bounds of the filled area). Not even I_{BIAS} of 50 μA can lower the propagation delay of a CCDLU made from normal V_{th} devices to 50 ps in case of a 3σ slow process skew. Using low V_{th} devices is the only practical possibility in this use case for the given specification, and even then the specification is clearly at the limit of the technology.

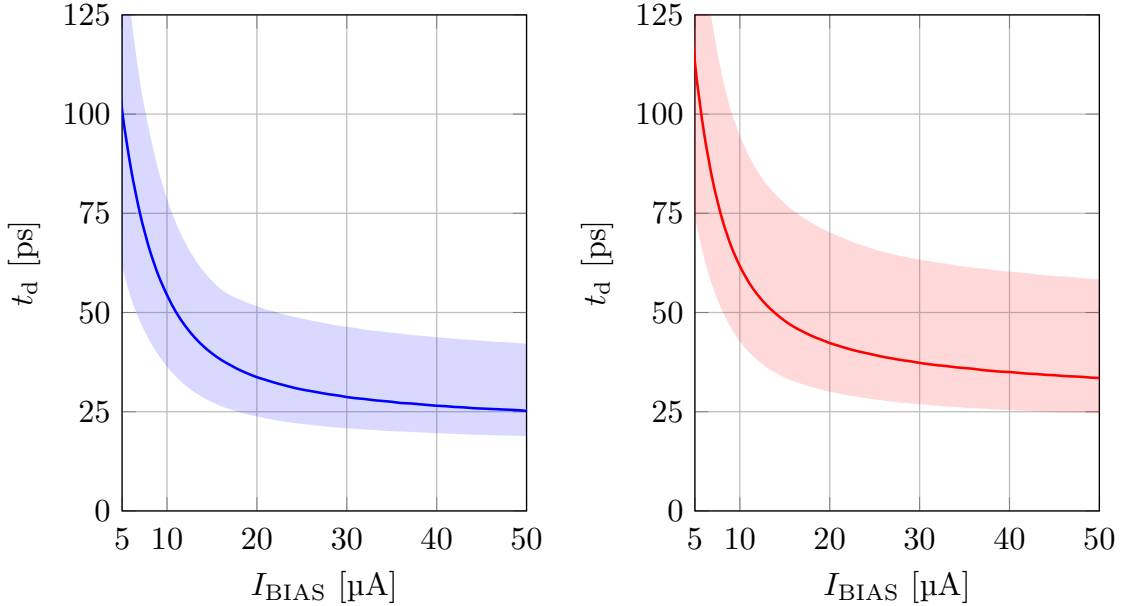


Fig. 5.7: Simulated propagation delay dependency on biasing current for low V_{th} (left) and normal V_{th} (right) minimum-sized CCDLU

A differential buffer stage is included each CCDLU, as depicted on Figure 5.6. The outputs of these buffers do not connect to the inputs of the next CCDLU in line, but instead connect to a duty cycle correcting circuit. The duty cycle correcting

circuit ensures that the duty cycle of a ring oscillator made from these CCDLUs is very close to 50%, as if one of the phases was faster than the other, the cross connected passgates would help the slower phase to transition as soon as possible, equalizing the duty cycle.

The last inverter in the duty cycle corrector has increased driving capability (level 2 instead of level 1). The output of the last inverter will be connected to another chain of inverters, each one having larger driving capability than the previous one, ultimately connecting to the inputs of the TDC latches, as depicted on Figure 5.1. Since there are multiple TDC's connected to each ILRO's outputs (as shown on Figure 5.1), the task of the buffers is to charge the non-negligible input capacitance of the TDC latches and minimize the capacitance connected to the internal unbuffered output nodes. Increasing this capacitance needlessly would slow down the oscillator, increase the power consumption and reduce the slew rate.

The connection of the ring oscillator is illustrated on Figure 5.8. Signals Ph[15:0], PhB[15:0] are the clock phases which ultimately connect to the TDC latches via the aforementioned chain of buffers.

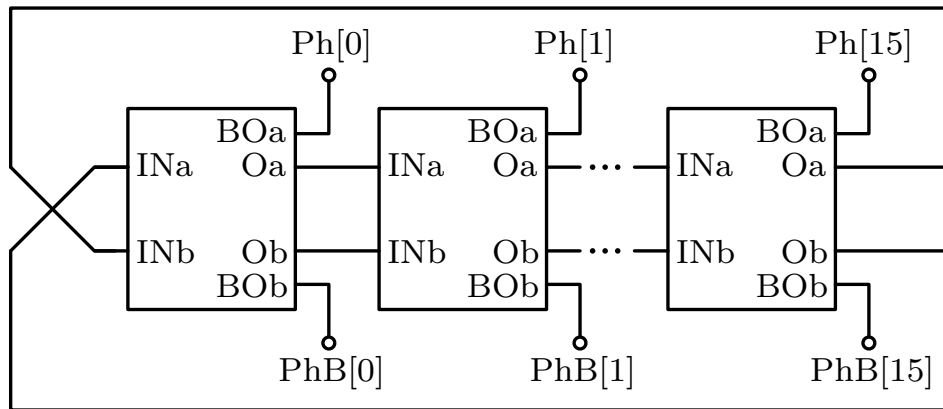


Fig. 5.8: ILRO connection overview

The problem with the minimum sized MOS devices within the CCDLUs is mismatch, which directly affects the *DNL* of the FTDC. It was simulated that when the *Current Controlled Delay Line* (CCDL) made from minimum sized MOS devices is connected as a free-running ring oscillator, the delay between the transitions of the neighbouring buffered clock phases varies by up to $\pm 90\%$ at 3σ , which is clearly unacceptable non-linearity. The usual rule-of-thumb is that the *DNL* should be kept below $\pm 50\%$ of the LSB.

In order to increase the precision of the LSB, the devices within the CCDLU need to be widened and/or lengthened. They cannot be, however, enlarged without consideration for the slowest PVT corners, as in such corners, large MOS devices within the CCDLU could prevent the possibility of oscillation at the desired frequency of 625 MHz, or only enable it for excessively large biasing currents. The

iterative optimization process was difficult, as meeting both criteria with any safety margin at all is on the very limit of the chosen processing technology, and the sizing has to be optimized in conjunction with the biasing mirror (which will be discussed in subsection 5.5.2), but eventually a satisfactory compromise has been found.

Tab. 5.4: CCDLU device sizes

Device	Width [nm]	Length [nm]
M _{1,2}	120	60
M _{3,4}	180	60
M _{5,6}	150	60
M _{7,8}	190	60
M _{9,10}	180	90

The device sizes achieving this compromise are listed in Table 5.4 (the device names match those depicted on Figure 5.6), and they were optimized in conjunction with statistical sensitivity analysis, which highlighted the most and least sensitive devices with regards to their contribution to the propagation delay mismatch. As Table 5.4 shows, the input NMOS pair M_{1,2} kept its minimum width and length, as its contribution to the propagation delay mismatch is minimal, while the transistors in the buffer stage (M₇ to M₁₀) had to be enlarged considerably, because the sensitivity analysis revealed their statistical variation plays a comparatively larger role. The mismatch of the propagation delay has been simulated via Monte Carlo, and its value is around 17%/σ, which for 3σ is right on the edge of 50%.

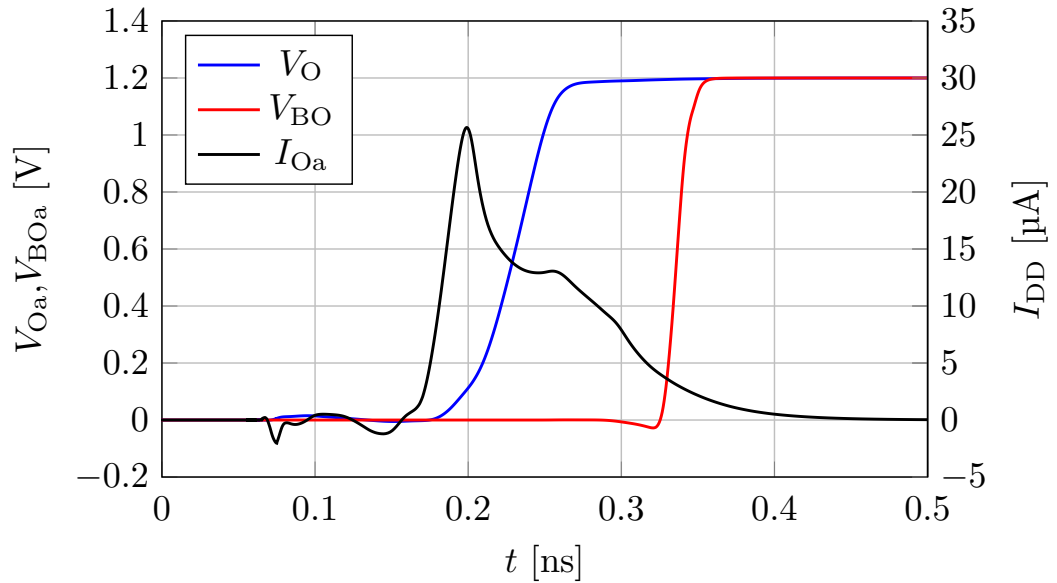


Fig. 5.9: Simulated CCDLU typical transient signals

The simulation depicted on Figure 5.9 showcases CCDLU transients. Naturally, the output of the buffered clock phases is delayed in relation to the unbuffered output which drives the next CCDLU in line.

Perhaps more interesting is the transient of the current drawn by the cell from the supply rail I_{DD} . The current quickly rises to around $25\ \mu\text{A}$ when it starts charging the capacitances of the differential inverter delay cell itself. After the output voltage of the differential inverter reaches a threshold of roughly $V_{DD}/2$, the current stabilizes around $13\ \mu\text{A}$ for a short while as it charges the buffer stage. The buffered clock phase transitions much more sharply than the unbuffered one, as its driving capability is stronger and the size of its capacitive load is smaller.

The most important takeaway here is that the biasing current behaves very dynamically, which is not surprising given that the whole transient takes roughly $250\ \text{ps}$. In this simulation, the CCDLU was biased by $11.5\ \mu\text{A}$, which is the current required to reach $50\ \text{ps}$ propagation delay as shown on Figure 5.7. However, on Figure 5.9, the current is clearly not limited by the value of the biasing current. Therefore it is important to keep in mind that feeding the CCDLU a certain biasing current does not mean the CCDLUs will be actually charged by this current. There is a relation between the biasing current fed into the input diode of the NMOS mirror distributing the current into each CCDLU and the actual charging current (as proven by Figure 5.7, because if there was no relation, it would not be possible to control the propagation delay via the biasing current), but it is not a simple equality and parasitic capacitances charging and discharging play a significant role in the transient.

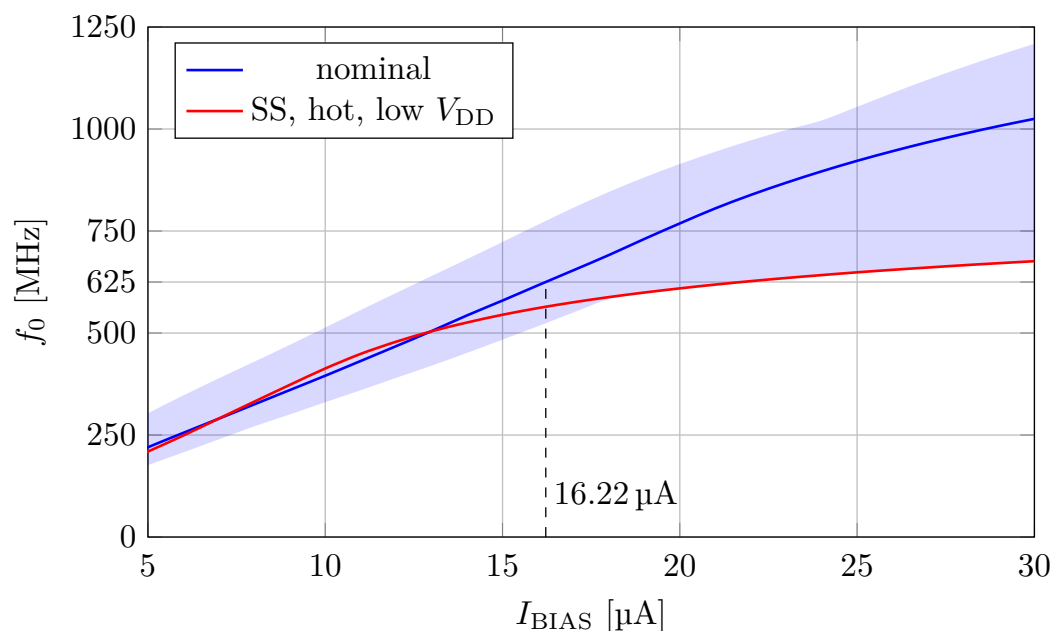


Fig. 5.10: Free-running frequency f_0 as a function of I_{BIAS} for the designed ILRO

An important chart on Figure 5.10 shows the free-running oscillation frequency of a ring oscillator made from the designed CCDLUs as a function of the biasing current. To achieve target $f_0 = 625$ MHz, a biasing current of roughly $16.22 \mu\text{A}$ is needed for the nominal corner. The filled area marks the boundaries of 3σ PVT corners again.

The red line on Figure 5.10 is the slowest worst case, which starts flattening out significantly just over 500 MHz. The worst corner would not be as slow if minimum sized devices were used in the CCDLU, but this is not possible due to maximum *DNL* requirements. This suggests that 625 MHz is at the practical limits of the technology, as faster oscillations would either require smaller devices with more mismatch, or much more biasing current for considerably less frequency gain, and due to the PVT variation, a wide range of biasing currents would be required, which would put demanding requirements on the V-to-I converter in the DLL.

It was discovered that the most critical factor for the worst case slowest corner is the process and the supply voltage, while the temperature plays comparatively smaller role. Depending on the project requirements, tighter V_{DD} specification could be used (for example $\pm 5\%$ instead of $\pm 10\%$), which would require more precise and/or trimmed voltage regulator, but could bring added benefit of ensuring the slowest ILRO corners are not as slow and many other analog circuits would pass PVT validation more easily and/or with a larger safety margin.

It also should be acknowledged that the corners do not actually correspond to 3σ probability (0.27%), as when combined with the V_{DD} variation and extreme temperature, this combination has significantly lower probability of occurring.

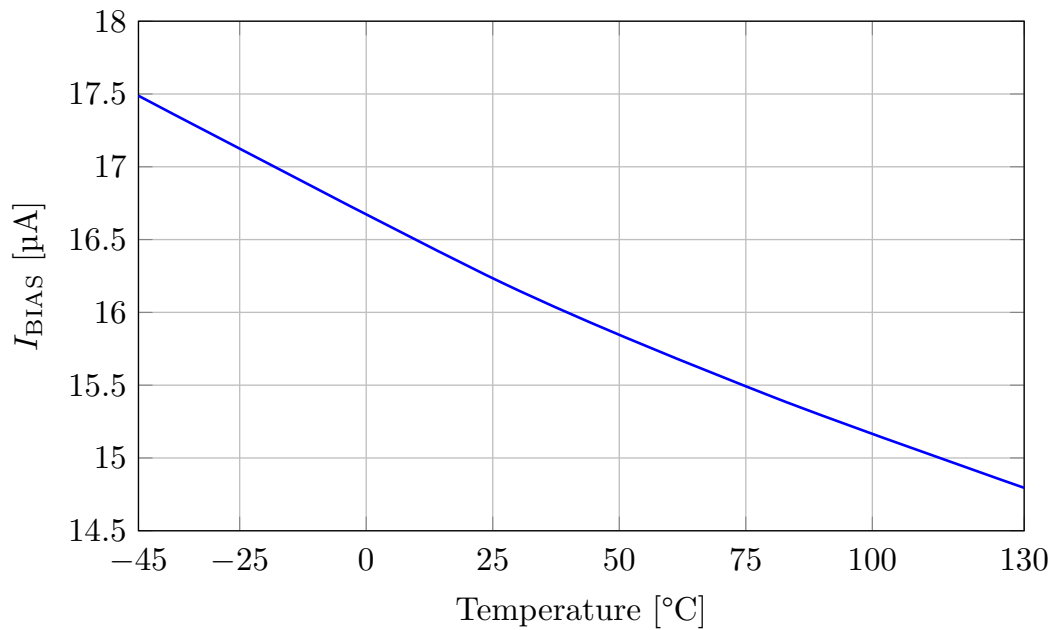


Fig. 5.11: ILRO nominal biasing current for $f_0 = 625$ MHz

The biasing current required to keep f_0 equal to 625 MHz across the temperature range is shown on Figure 5.11. It is relatively linear and can be approximated by a slope of cca $-15.4 \text{ nA}/^\circ\text{C}$. This will be taken into account when designing the DLL V-to-I converter in subsection 5.6.3.

5.5.2 Biasing mirror design

As already mentioned, each ILRO cell has a single input biasing current terminal, and this current needs to be mirrored into each of the 16 CCDL cells as shown on Figure 5.12.

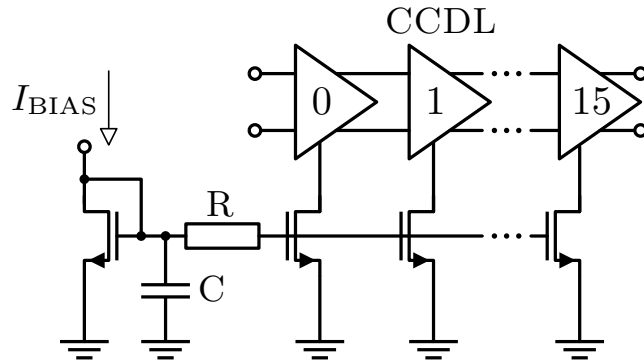


Fig. 5.12: CCDL NMOS biasing current mirror

To preserve linearity, it would be desirable to achieve good matching in the current mirror so that each cell would produce the same propagation delay. This is, however, not very practical, because neither of the two main techniques for improving matching – increasing device area and increasing overdrive – are suitable for the circuit.

Using large transistors is not suitable because of the need to minimize ILRO footprint. The architecture contains a rather large amount of ILROs which need to be routed in a particular way and the CCDLUs are otherwise quite small, as they are made out rather small thin oxide devices.

Increasing the overdrive of the current mirror too much is problematic as well because the current source supplying the current to the mirror (the V-to-I converter, which will be discussed in subsection 5.6.3) has limited compliance and especially in slow MOS, low V_{DD} and/or cold corners, the *gate-source voltage* (V_{GS}) of the ILRO biasing mirror could make the PMOS current source step out of saturation, lowering the output impedance as well as accuracy of the supplied current, which is undesirable. The current source compliance constraint defines the minimum W/L ratio for the NMOS biasing mirror devices, which was determined by simulation to be $5/8$, as longer and narrower devices were not able to meet the minimum 50 mV saturation margin (`vdsat_marg` device parameter) over 3σ PVT corners.

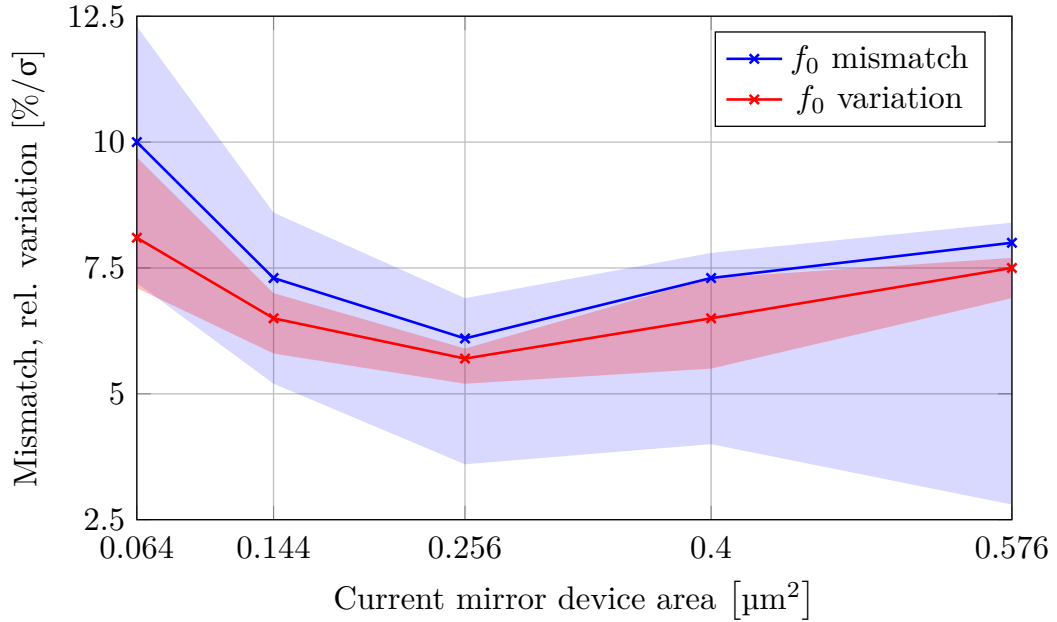


Fig. 5.13: ILRO free-running frequency variation and mismatch for various biasing current mirror device sizes

A Monte Carlo statistical analysis was performed to find the optimum size of the current mirror devices. The goal was to find a relatively small device size which minimizes the free-running frequency variation as well as mismatch. The following dimensions (width/length) were analysed: $0.2\ \mu\text{m}/0.32\ \mu\text{m}$, $0.3\ \mu\text{m}/0.48\ \mu\text{m}$, $0.4\ \mu\text{m}/0.64\ \mu\text{m}$, $0.5\ \mu\text{m}/0.8\ \mu\text{m}$ and $0.6\ \mu\text{m}/0.96\ \mu\text{m}$. The results of the analysis are shown on Figure 5.13⁵ (filled areas mark the span of the quantities across PVT corners).

Clearly, the optimum device size which minimizes both types of statistical variation is the third option, i.e. $0.4\ \mu\text{m}/0.64\ \mu\text{m}$. The reason for why larger devices than that actually worsen the matching is probably connected to the fact that the biasing mirror supplies current dynamically, and larger devices have larger parasitic capacitance, which can easily vary due to manufacturing variation.

The biasing current mirror shown on Figure 5.12 features an RC low-pass filter between the input diode and the 16 current sources. The reason for its inclusion is that each of the 16 current sources only supply current dynamically during short transients, as shown on Figure 5.9. The charging and/or discharging of their capacitance produced significant periodic spikes in the voltage at their gates, which propagated backwards as far as into the V-to-I converter which supplied the biasing current into the mirror (the V-to-I converter will be described in subsection 5.6.3).

⁵The term “ f_0 variation” refers to the statistical variation of the free-running frequency of a single ILRO instance, while the term “ f_0 mismatch” refers to the difference of free-running frequencies of two independent ILROs.

The period of these spikes is 50 ps, as the CCDLUs are designed to transition one by one after this period of time. The equivalent frequency of these spikes is 20 GHz and therefore a simple RC low-pass filter made from 10 k Ω P+ polysilicon resistor and a 500 fF NMOS capacitor is able to filter them significantly, as the cut-off frequency of such a filter is roughly 32 MHz.

5.5.3 Injection circuit design

A possible injection circuit has been shown in chapter 3 on Figure 3.14. This circuit has been published in [41] and was supposedly successfully used for locking an ILRO with a resolution of 52 ps. This circuit injects charge through capacitors directly connected to an internal ILRO node.

The authors of [41] do not specify the size of this capacitance. If small capacitance is used, the maximum charge which can be injected is limited and the locking range suffers. If high capacitance is used, large area is taken. In both cases, however, the extra capacitance connected to the injection node imbalances the symmetry of the oscillator and the propagation delay from one cell to the other can differ extremely even in free-running state, worsening the linearity of the FTDC significantly. Therefore, identical dummy capacitance must be connected to each internal node of the oscillator just to equalize the propagation delays again. However, this not only takes significant area (depending on the size of the capacitance), but it also necessitates the use of a larger biasing current, as the CCDLUs now have to charge larger total capacitance. Therefore, power consumption is increased as well.

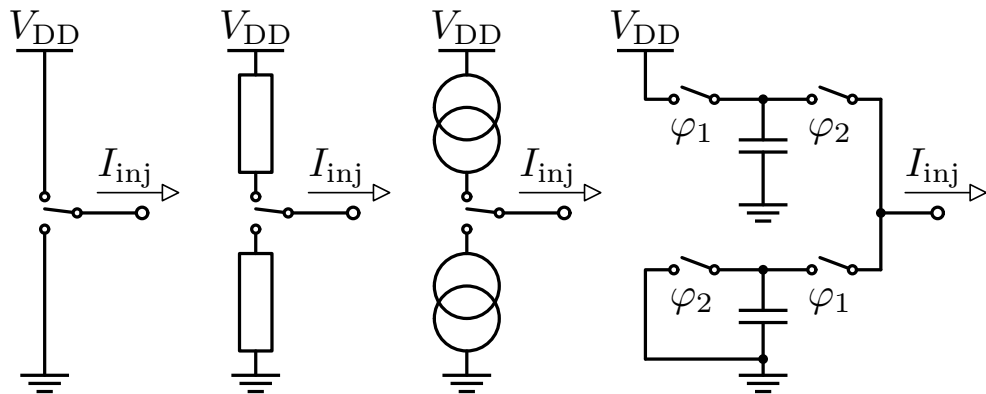


Fig. 5.14: Alternative injection methods

The injection circuit from Figure 3.14 has been thoroughly simulated, but it has been decided that its implications for the performance of the ILRO are too disadvantageous and a better solution has been sought after. Such a solution does not need to necessarily add zero capacitance to the injection node, as that is clearly

impossible. It should, however, provide a better trade-off between parasitic capacitance, area and locking range, as well as perhaps better PVT stability. Four such proposals are shown on Figure 5.14.

All of these methods share the same devices which are directly connected to the injection stage – minimum sized MOS switches. The switches represent relatively low capacitance (as low as realistically possible in a given technology) and also relatively low impedance to the flow of current in their “on” state. The minimum size of the switches also helps minimize charge injection and clock feed-through, as these effects could degrade the phase noise performance of the oscillator.

The four methods shown on Figure 5.14 were all simulated and compared according to various values such as power consumption, area, injected charge etc. It turned out that the simplest possible mechanism (switches only) is the best on paper, as it offers the largest possible locking range and the smallest area. Every other method such as injection through resistors, MOS current sources or switched capacitors is more area intensive and only lowers the locking range as it ultimately limits the maximum possible charge transferred.

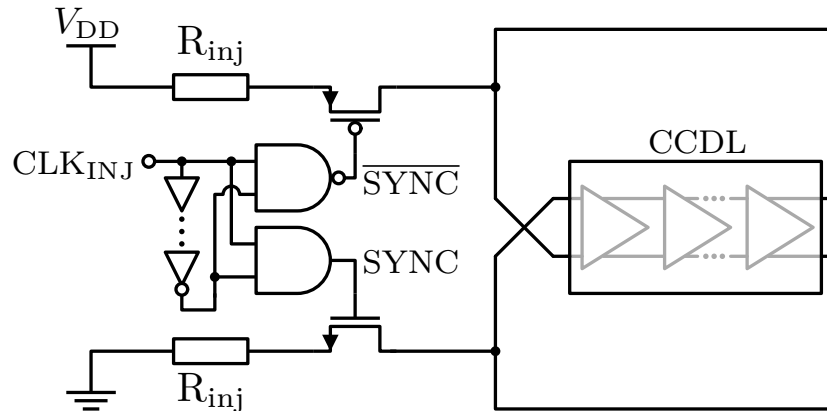


Fig. 5.15: Designed injection circuit

In the end, however, the resistor option has been chosen anyway and is depicted on Figure 5.15. There are two reasons for this choice. First of all, MOS switches on their own are quite sensitive to PVT variation, which could change the behaviour of the injection circuit significantly. Adding the resistors should in principle make the circuit more predictable. Secondly, in fast MOS corners where the drain-source on resistance $R_{DS,on}$ is unusually low, adding the resistors should reduce the harsh spikes experienced by the 1.2 V rail when the injection switches switch. These spikes could negatively affect the stability of the supply rail.

The precise value of the resistance of 1 k Ω has been chosen as it is the highest amount of resistance which does not affect the locking range of the ILRO signifi-

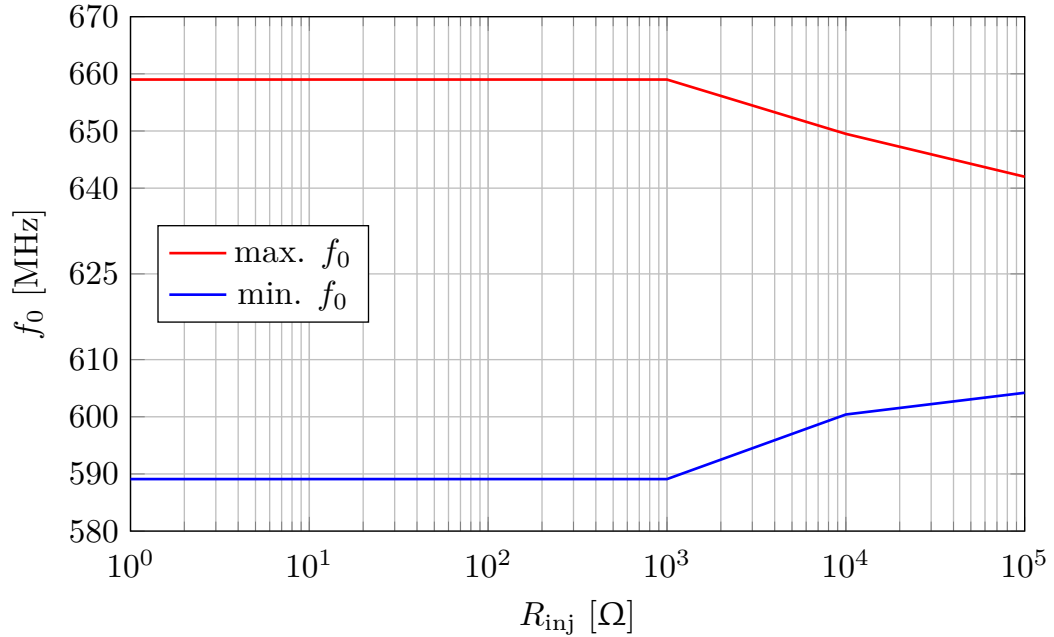


Fig. 5.16: Locking range of the designed ILRO for various sizes of R_{inj}

cantly, as shown on Figure 5.16⁶. The standard P+ polysilicon resistor device has been selected to implement the resistance.

As Figure 5.15 shows, the injection circuit is designed so that it injects energy after a rising edge of the injection clock. It does not, however, inject energy after a falling edge. This decision has several consequences. First of all, injecting only on one edge of the clock naturally lowers the locking range, as less energy is injected on average. However, this should not be an issue because the ILROs will be pre-trimmed quite close to target frequency, as was discussed in subsection 5.3.3. Injecting on one edge only should therefore save some power. Second reason is the area saving – injecting on one edge saves second instance of the switches, resistors and pulse shaping circuits, as well as additional dummy switches which would be required to balance the delay line. This leads us to the third reason, which is additional power saving – the additional dummies would increase the biasing current requirement of the ILRO, as the added capacitance would slow down the oscillations. The final reason is phase noise. If the duty cycle of the injection clock at the point of use is not very close to 50%, the ILRO would lock to only a single edge of the clock anyway. This was demonstrated in subsection 4.4.4. The edge which the ILRO would not lock onto will burn power needlessly and could potentially decrease the

⁶This chart is similar to Figure 3.12 and indeed the size of the resistance is inversely proportional to injection ratio introduced in chapter 3. The injection ratio is not evaluated in practical transistor level design because it varies significantly over PVT corners and because the non-linear behaviour of transistors produces irregular dynamic waveforms, which reduces the usefulness of the concept in this context.

spectral purity of the clock phases with its out-of-phase energy injections.

There is also a reason for why the resistors on Figure 5.15 connect to the supply rails and the switches connect them to the rest of the ILRO and not the other way around. While the MOS switches could drive the resistors better if they were connected to the supply rails instead, as the voltage drop on the resistors during the transient would not subtract from their V_{GS} , the resistors have some parasitic capacitance which differs from the parasitic capacitance of the minimum sized MOS switches. As the goal is to match the capacitive load of each ILRO stage in order to ensure that each stage has the same amount of propagation delay, dummy injection circuits need to be added to all stages. If the order of the resistor and the switch was reversed, identical resistors and switches would need to be added to each stage, as the resistor and switch combination acts as a Π RC circuit. This would take a lot of area due to the resistors. Instead, if the resistor is connected to the rails and the switches to the ILRO, only the dummy switches need to be added to all other nodes, excluding the resistor. This is because the resistor capacitance of the injection circuit is somewhat shielded from the ILRO by the switch. It has been simulated that this way the propagation delays match quite well as shown on Figure 5.17. For perfect matching, resistors would need to be added to the dummy injection circuits anyway, but the improvement in matching would be less than a picosecond, which is well below the size of the LSB and is simply not worth it.

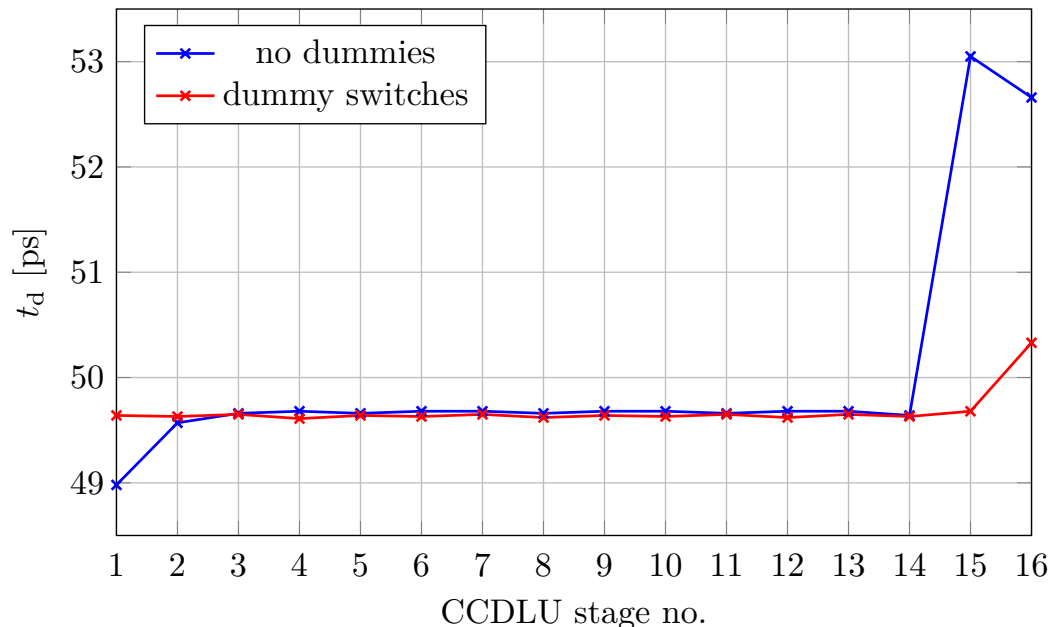


Fig. 5.17: Propagation delay distribution within the ILRO with and without dummy switches, when free-running at $f_0 = 625$ MHz

The circuit which shapes the injection pulses themselves is simply a rising edge detector with a pulse width set by the propagation delay of the standard cell buffers,

which are followed by an inverter, so that logic inversion is achieved. The circuits produce a pulse of roughly 80 ps length in typical case. This pulse width has been chosen so that the time roughly matches the length of the transition at the injection node, so that the “sensitive window” of the ILRO is covered (see subsection 3.3.3). The pulse width produced by the circuit varies over PVT, but thankfully the ILRO is quite insensitive to the variation of the pulse width. This is because the ILRO, once locked, always aligns its phase in such a way to accept the exact amount of energy it needs. Varying the pulse width affects this phase alignment between the injection clock and its injection stage, but the phase of the clock phases does not matter as much, because thanks to the sliding scale operation inbuilt into the architecture, the FTDCs only depends on the frequency of the output clock phases, which is unaffected (this benefit of the sliding scale technique was discussed in subsection 5.3.6).

5.5.4 Locking simulations

Since the ILROs will be biased by the DLL, in this section, only a few simulations will be discussed. A more thorough evaluation will be presented in section 5.7.

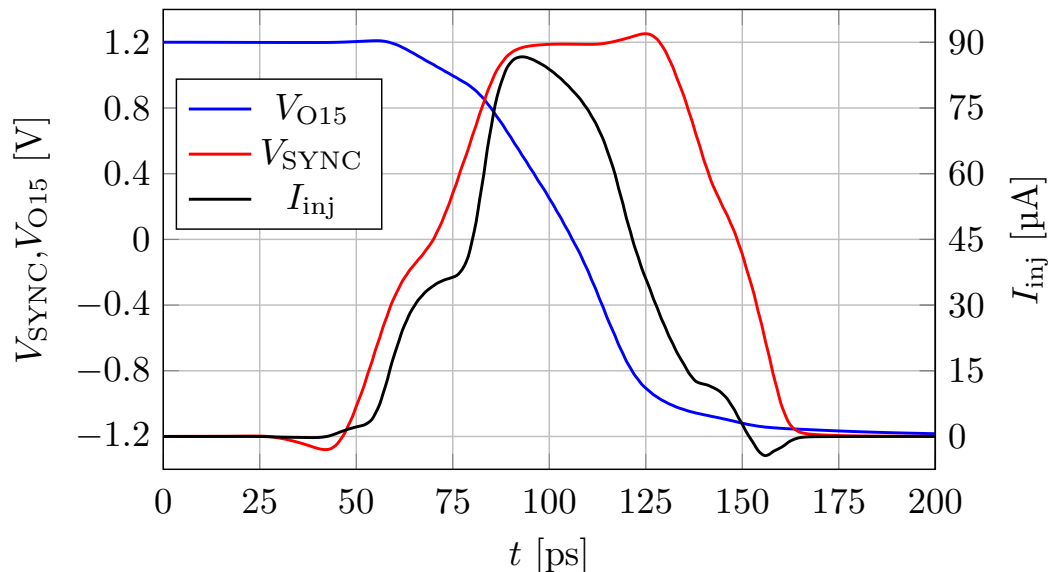


Fig. 5.18: Typical waveforms of the injection stage voltage, the injection pulse and the injected current for a locked ILRO of $f_0 = 590$ MHz

Typical transients of the unbuffered output voltage of the 15th stage (the node which is injected into), the injection switch signal SYNC and the injection current are shown on Figure 5.18. Unsurprisingly, the injection current is rather dynamic similarly to the CCDLU charging current from Figure 5.9. The width of the SYNC pulse roughly matches the length of the transition of the voltage at the injection node.

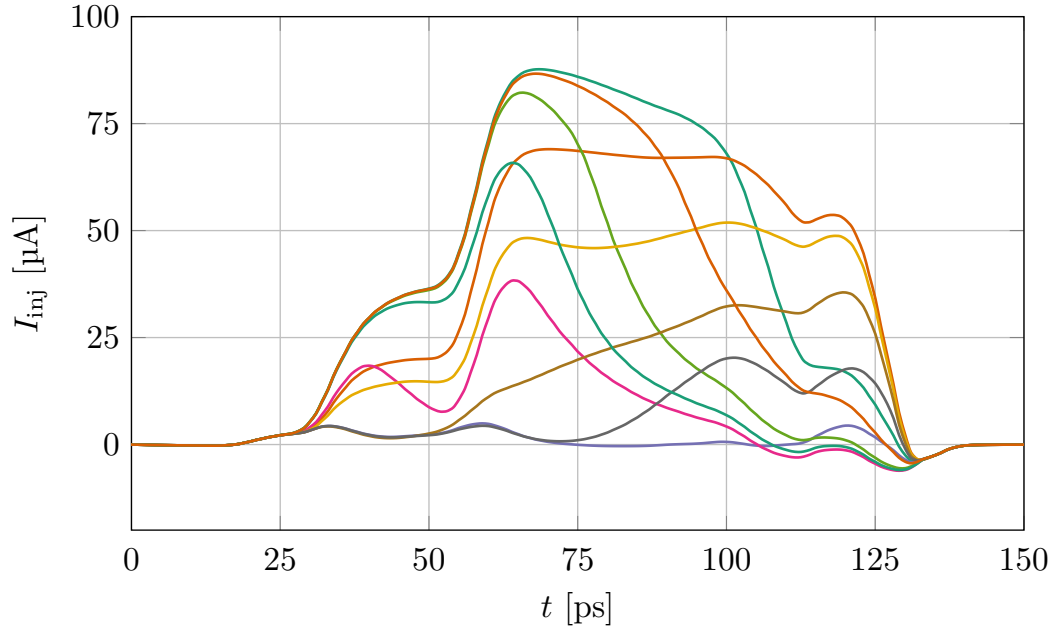


Fig. 5.19: Various waveforms of injected current into ILROs of various f_0 when locked to 625 MHz

The shape of the injection current waveform varies dramatically depending on the f_0 of the ILRO. In some cases, the ILRO's f_0 is quite close to the target, not a lot of charge is needed to lock it and the injected current is relatively flat. In other cases, however, the f_0 is on the edge of the locking range and the injection current waveform has to be relatively large and long in order to deliver the charge required to lock the oscillator. Several picked shapes of these injection waveforms are depicted on Figure 5.19.

The charge injected into the circuit for various sizes of the injection resistor as a function of the free-running frequency is shown on Figure 5.20. The dataset in the chart is limited to free-running frequencies the oscillator was able to lock onto (the locking range was already shown on Figure 5.16). An interesting takeaway is that for a given free-running frequency, the charge injected by the circuit with higher injection resistance is less. This seems to suggest that the injection circuits with less resistance inject energy needlessly, as less injected charge would have been sufficient. On the other hand, the higher the injection resistance, the narrower the locking range, which is the more important concern.

The second takeaway is the asymmetry between the right and left half of the chart. While for $f_0 > 625$ MHz the injected charge grows quite linearly, for $f_0 < 625$ MHz, the situation is very different. The source for this asymmetry has not been definitively identified, but it is most likely caused by the interaction of the asymmetry of the rise and fall times of the unbuffered CCDLU output (listed in Table 5.5) as well as the asymmetry of the injection circuit (the PMOS side and the

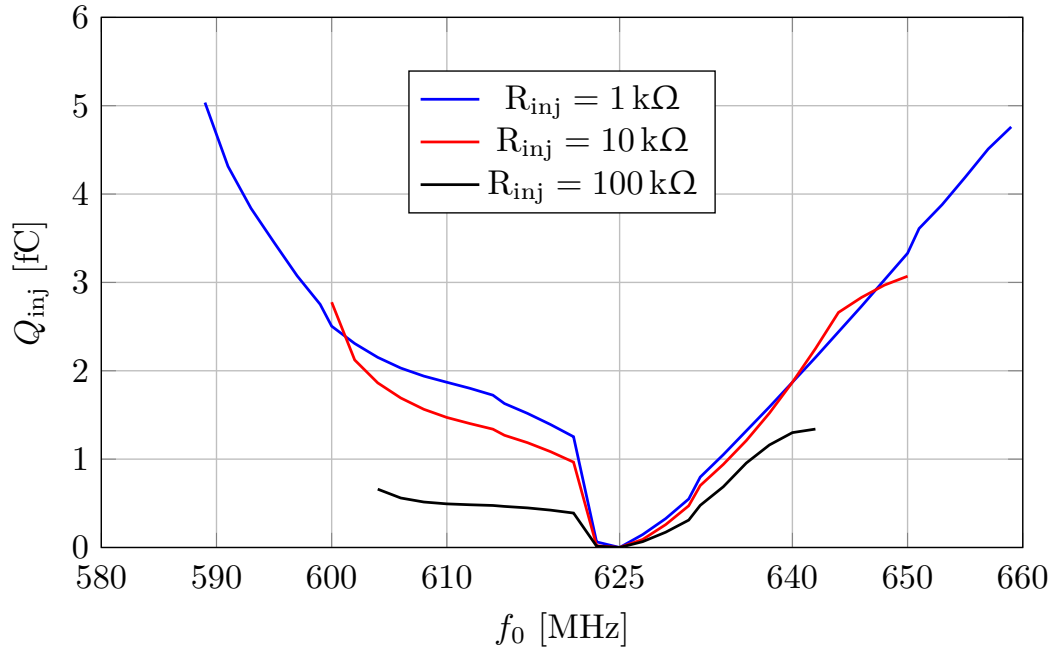


Fig. 5.20: Charge injected by the injection circuit into an ILRO per cycle once locked to 625 MHz as a function of initial free-running frequency

NMOS side of injections do not necessarily operate exactly the same way).

The instantaneous frequency of ILROs of various free-running frequencies under locking is shown on Figure 5.21. In the chart, injection starts at $t = 22 \text{ ns}$, which is why there is an abrupt change of instantaneous frequency, often caused by the injection pulse deforming the shape of the measured clock phase. This can even cause the frequency measurement function (`freq` from Cadence ADE) to erroneously return extremely high frequencies, which can be seen on Figure 5.21. The free-running frequency of the waveforms can be seen at roughly $t = 15 \text{ ns}$, before the injections start.

As a side note, the transient simulation shown on Figure 5.21 is precisely how locking range is actually determined. There is no other reliable way other than trying to lock the ILRO in transient simulation, letting it run for several hundreds of oscillatory cycles and seeing if the instantaneous frequency matches the injected one by the end of the simulation. This is a very time intensive simulation, which is why this simulation cannot be practically performed in Monte Carlo runs or for PVT corners in conjunction with f_0 variation. The locking range shown on Figure 5.16 has been therefore only determined for the nominal case (and the concept of locking range, when PVT corners are taken into account, is not as useful of a design tool any more anyway).

As Figure 5.21 shows, in all the depicted simulation runs, the ILROs are able to lock to the injected frequency eventually. The locking transient itself is very abrupt,

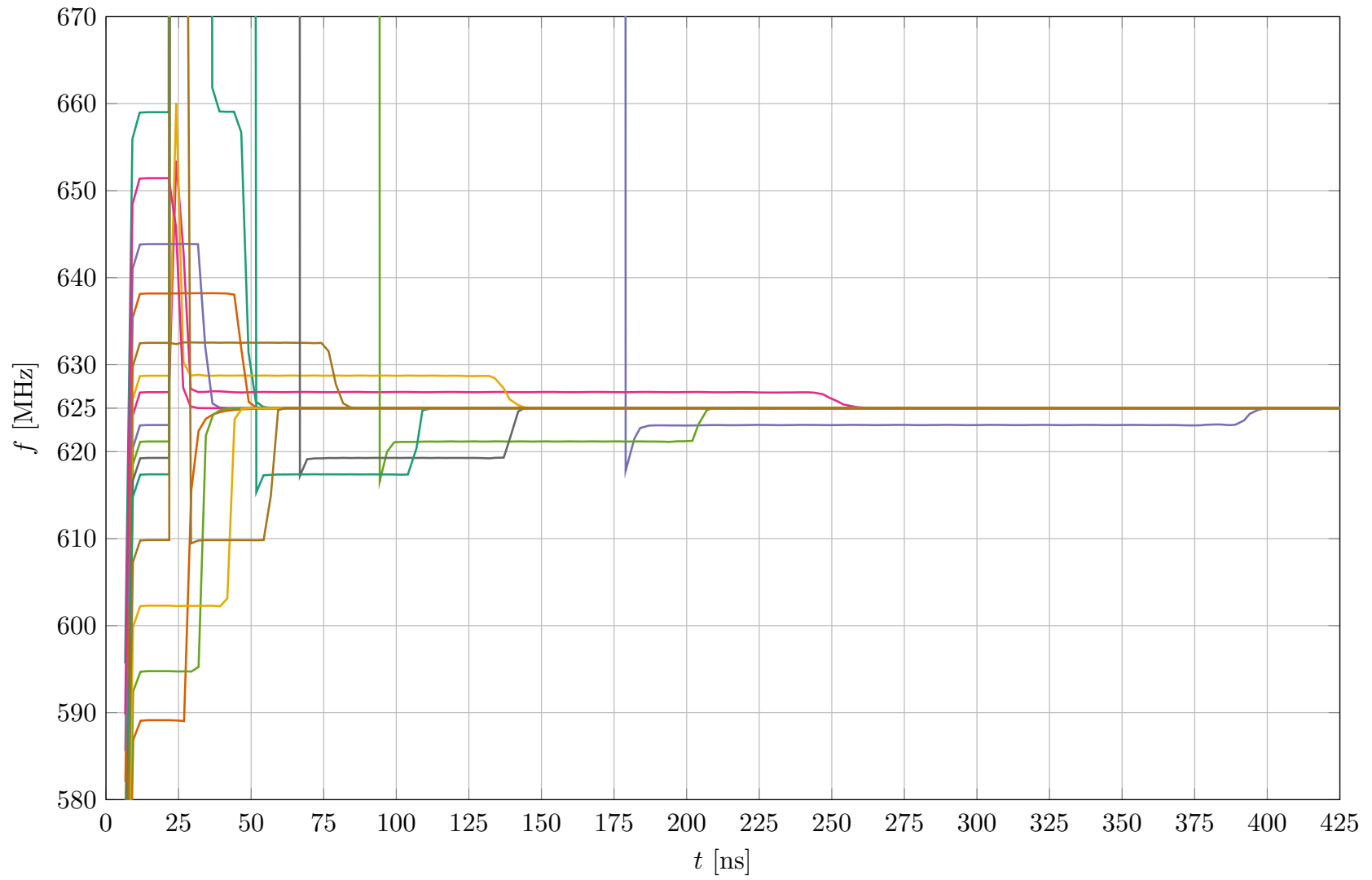


Fig. 5.21: Instantaneous frequency of ILROs with various f_0 during locking

but there can be a big difference in the time it takes for the ILRO to lock. In some runs, the ILRO was able to lock as soon as the injections start, while for others, it can take hundreds of oscillatory cycles to lock.

Upon closer inspection, there seems to be an inverse relationship between $f_0 - f_{inj}$ and the time it takes for the ILRO to lock. In other words, the closer the ILRO's natural oscillations are to the target frequency, the longer it actually takes to lock to it. This might appear counter-intuitive at first, but it can be explained in a simple manner, and the mechanism is shown on an idealized diagram on Figure 5.22.

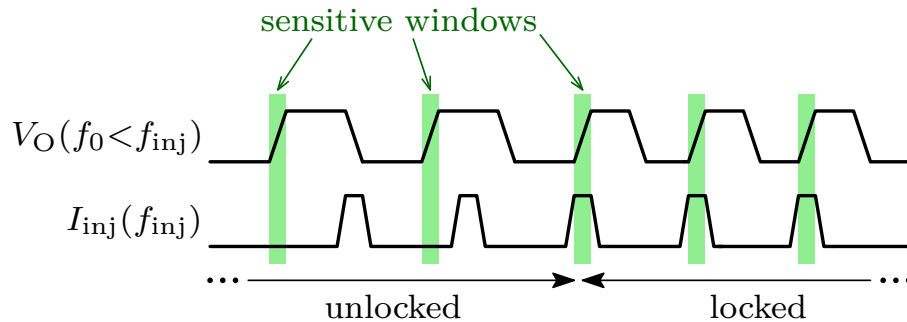


Fig. 5.22: Injection locking process

When the injections start, the phase shift between the injection pulse and the transition of the voltage at the injection stage it needs to synchronise with (the “sensitive window”) in order to lock is random. If the f_0 is several percent off the f_{inj} , this difference in frequency will ensure that the phase shift between the two waveforms will vary quickly and the injection pulse will line up with the sensitive window soon. Once the two waveforms line up correctly in time, the frequency of the ILRO locks to the injected frequency immediately and abruptly (as long as the frequency is within the locking range of the oscillator), as proven by Figure 5.21. If, however, the f_0 is very close to target frequency, the difference in frequencies is so small that it can take hundreds or even thousands of cycles before the phase shift between the injection pulse and the sensitive window adjusts properly.

Theoretically, if f_0 was equal to f_{inj} , the phase shift between the two waveforms would stay constant forever and the ILRO would actually never lock. In reality, this will not occur due to noise, temperature variations etc. but the locking process in these cases could take longer than usual.

Finally, some scalar quantities of interest related to the designed CCDLU or ILRO are listed in Table 5.5.

With regards to the listed areas, it has to be noted that they refer to the total device area of the designed circuits. After layout, the total area would be likely 50–100% larger, depending on the circuit, any special routing needs as well as the experience of the layout engineer.

Tab. 5.5: ILRO performance

Quantity	Min.	Typ.	Max.	Unit
CCDLU area		14.7		μm^2
ILRO area		342		μm^2
Supply current		502.4		μA
Power consumption		602.9		μW
t_d mismatch		17		$\%/\sigma$
Duty cycle variation		0.83		$\%/\sigma$
V_O rise time	75.9	87.1	102.8	ps
V_O fall time	40.6	48.6	54.6	ps
V_{BO} rise time	12.1	16.0	22.2	ps
V_{BO} fall time	12.1	16.0	22.2	ps
SYNC pulse width	59.8	78.7	86.6	ps

The supply current and power consumption figures listed in Table 5.5 are related to the nominal corner under locking. Minimum and maximum values are not provided for these quantities because the values depend too much on the way the ILRO is biased, which has not been discussed yet.

5.6 Delay Locked Loop design

In this section, the design of the DLL will be presented. First, the overview of the loop will be given. Afterwards, the FPD and charge pump cell will be designed, followed by the V-to-I converter. In the final subsection, the implementation of trimming will be thoroughly discussed.

5.6.1 Overview

A simplified overview of the designed DLL is shown on Figure 5.23. It consists of three larger blocks: the delay line, the FPD & charge pump block, and the V-to-I converter.

The role of the DLL is to provide a PVT compensated biasing current for the ILROs. Fortunately, much like the ILROs, the DLL will only ever be locked to 625 MHz and therefore many issues regarding its locking range, behaviour at various frequencies etc. are not relevant, which greatly simplifies the design process.

A key consideration for the design of the DLL is the need to connect the delay line in a way as similar as possible to the connection of the ILRO in order to ensure that the biasing current which satisfies the DLL lock condition is as close to the current

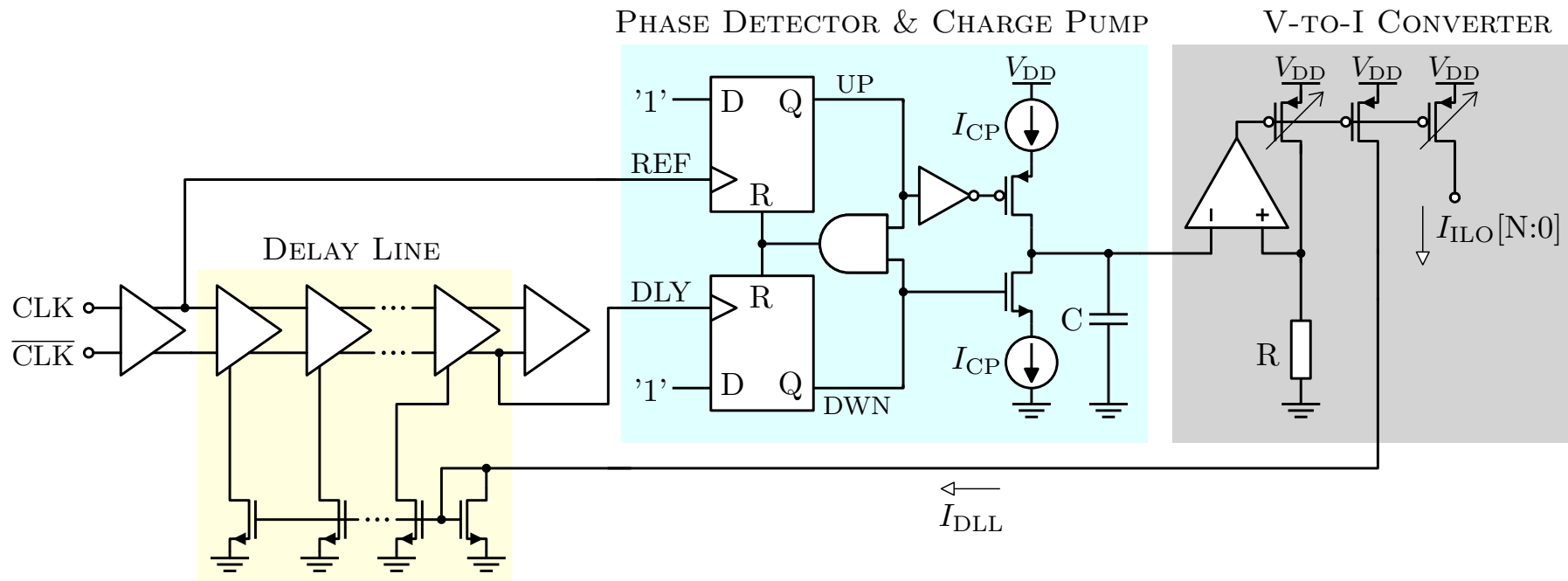


Fig. 5.23: Simplified overview of the DLL

needed to set the free-running frequency of the ILROs to 625 MHz as possible. There are only two things that can be done to achieve this, both of which are depicted on Figure 5.23: driving the delay line by a preceding delay line and loading the delay line by a dummy delay line as well. This is because the delay line in the ILROs is connected in a ring connection, driving and loading itself.

Other than that, there are no critical requirements for the DLL. Since the DLL is a block which will be only instantiated once, minimizing area is not as important as with other blocks like the ILRO. The speed of the start-up and the feedback loop of the DLL is also not as important, as the DLL has to be active at all times during the TDC conversion and will not be switched on and off dynamically during conversions. A few things which should be kept in mind, however, is prevention of false locking, good filtering of the charge pump, adequate speed of the V-to-I converter opamp loop that is able to follow the DLL and some attention should be also paid to power supply rejection (it is expected that the 1.2 V supply might be rather polluted with high frequency switching noise).

The design of the DLL blocks will be discussed in the following sections.

5.6.2 Frequency Phase Detector and Charge Pump

The role of the FPD and the charge pump, which form a single cell together, is to convert the phase difference between the clock input and its delayed counterpart into a corresponding change in voltage at the charge pump output. A variation of a common topology has been chosen as shown on Figure 5.25. The FPD consisting of two D-type flip-flops and a resetting NAND gate is shown in the upper left, while the rest of the schematic shows the charge pump along with its biasing current mirrors and pre-charge circuit.

If enabling signal EN is low, the circuit is disabled and does not react to the clocked inputs. As long as EN is high, the FPD operates as normal, and the sequence of the signals can be seen on Figure 5.24.

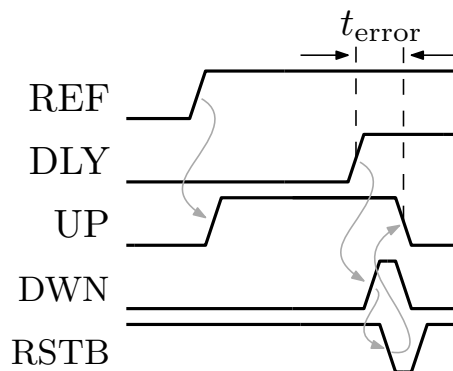


Fig. 5.24: FPD signal sequence

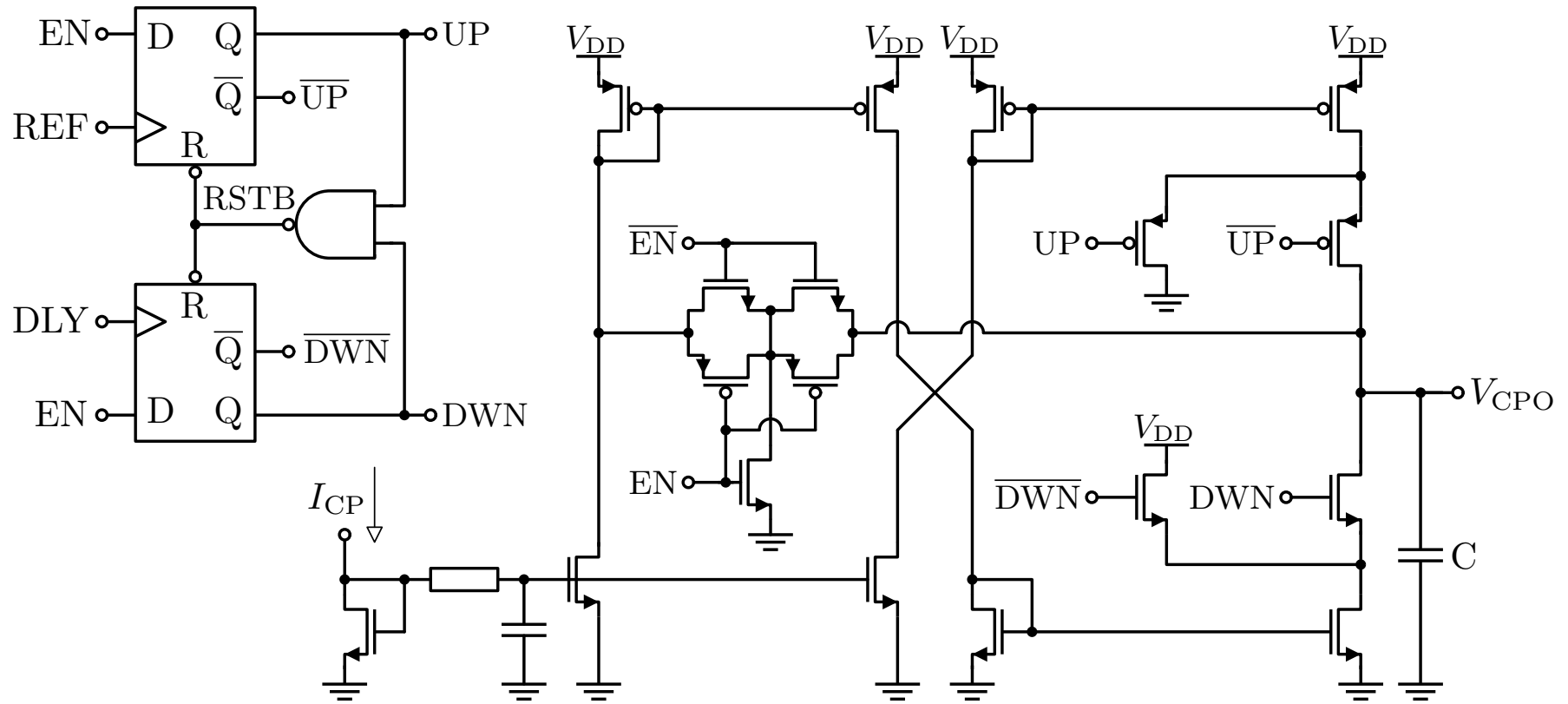


Fig. 5.25: Simplified schematic of the designed FPD and charge pump block

An important feature of the operation of the FPD is the delay between the moment when the second flip-flop activates and the moment the flip-flops are actually reset. This delay, on Figure 5.24 marked as t_{error} , is caused by finite speed of the digital gates.

There is a consequence to this error. First of all, the finite speed of the reset path means that there is a time period where the signals UP and DWN overlap. This could cause the charge pump to pull in both directions. Theoretically, if the charge pump current source and the current sink matched perfectly, this would have no effect on *charge pump output voltage* (V_{CPO}), but in reality, any mismatch can cause an erroneous change in V_{CPO} during this short period of time. Gating the signals to shorten the overlap would be possible, but gates have limited speed and do not match perfectly either.

The consequence of this erroneous change of V_{CPO} is that while the DLL will settle, the phase difference between the reference and delayed clock signals at the input of the FPD will not be zero. This is because the DLL tries to stabilize V_{CPO} by any means necessary, and if there is an error current I_{error} flowing into the charge pump capacitor due to mismatch, the loop will compensate it by inducing a non-zero phase shift at the input of the FPD. Other than increasing the size of the charge pump current mirror devices to improve matching, there is not much more that can be done to prevent this issue. Trimming, which will be discussed in subsection 5.6.4, can alleviate it though.

Simulating the exact relationship between the error current I_{error} and the phase shift seen at the input of the FPD is not straightforward. The following simulation has been devised to provide a reasonable approximation. An ideal DC current source representing the error current has been connected in parallel to the charge pump output capacitor, and its current has been swept. A transient simulation simulating the whole DLL operation has been performed for each swept point, and the input phase shift at the FPD has been saved. This phase shift then has to be re-evaluated, as the ideal DC current source provides current at all times, while the real error current only flows when both $\overline{\text{UP}}$ and DWN signals are active. The length of these signals when the input phase shift is zero is around 90 ps, while the period of the DLL input signal is 1.6 ns, i.e. the duty cycle of the error current is around 5.6%. Therefore the simulated phase shift is scaled accordingly by this factor to produce an estimate of phase shift seen at the input of the FPD based on the size of the error current I_{error} in relation to I_{CP} , as depicted on Figure 5.26.

The error current has been swept from -20% to 20% as a Monte Carlo simulation has shown that the designed charge pump current sources match to $6.4\%/\sigma$, which is nearly 20% for 3σ . The worst case phase shift at the input of the FPD caused by this mismatch is around 12 ps. This is about 1.5% of the target delay of 800 ps, which

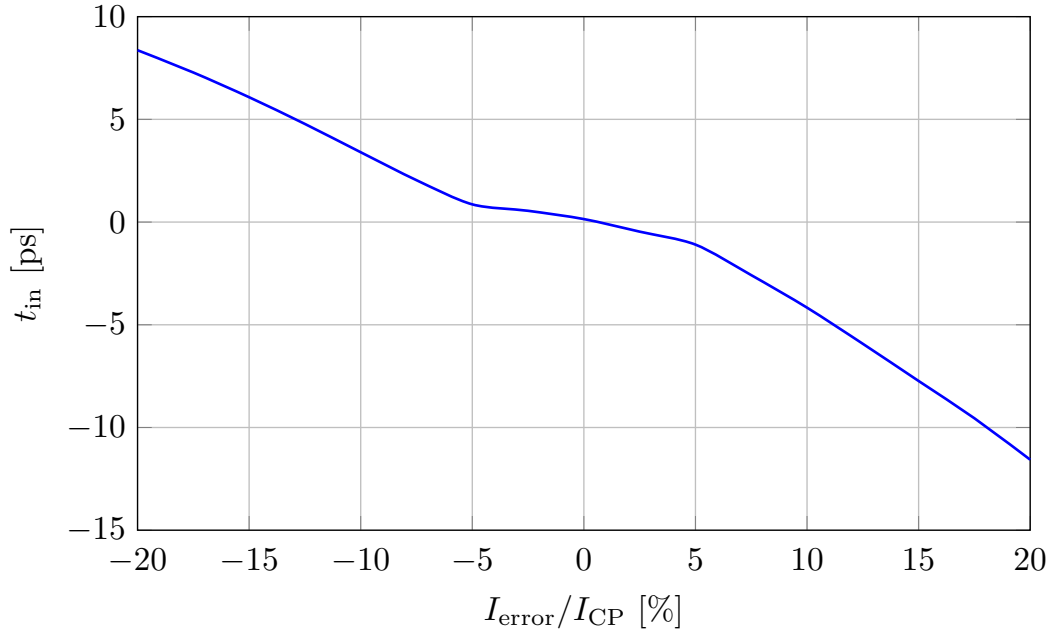


Fig. 5.26: FPD input phase shift as a function of charge pump current source mismatch when connected in a DLL

corresponds to a similar error of f_0 in an ILRO biased by this DLL, which is an error small enough to not matter significantly (this will be proven in subsection 5.6.4). Mismatch caused by fabrication is static and produces a static error that can be trimmed out, but it does not endanger the capability of the DLL to track *Voltage Temperature* (VT) variation.

There is actually a benefit to the limited speed of the digital gates. If the phase error at the input of the FPD was rather small, extremely fast digital gates could produce extremely short digital pulses which would be reset much sooner than the charge pump could react to them. This would lead to a dead-zone of the FPD & charge pump transfer characteristic in the vicinity of zero input phase error. The consequence would be that within the dead-zone, the delay of the DLL delay line and therefore the free-running frequency of the ILROs biased by this DLL could vary very slightly without any correction of the DLL loop, as the dead-zone would make this small variation essentially invisible. This would naturally result in degraded phase noise performance. As long as t_{error} is larger or similar to the bandwidth of the charge pump switches, the dead-zone is not present and the FPD & charge pump system can in principle react to very small phase errors at its input properly.

The values of t_{error} and some other scalar quantities of interest are listed in Table 5.6. The quantities were evaluated for zero input phase error, which is the state the FPD is expected to work in most of the time when DLL settles.

The size of the charge pump current has to be selected in conjunction with the output capacitor. The goal is to choose a capacitor large enough to sufficiently

Tab. 5.6: FPD performance

Quantity	Min.	Typ.	Max.	Unit
Device area		228		μm^2
Supply current	55.7	77.3	103.9	μW
Power consumption	60.1	92.7	137.2	μW
t_{RSTB}	51.9	85.6	164.5	ps
t_{error}	62.1	102.2	194.5	ps
$I_{\text{error}}/I_{\text{CP}}$		6.4		$\%/\sigma$

smooth out the transitions at the output, but not too large either, as that would increase the footprint. Luckily there is only one DLL in the system (another advantage over the architecture shown on Figure 2.17) and therefore a rather large NMOS capacitor can be used. In this design, the values 1 pF and 4 μA were chosen for the capacitor and the current sources respectively.

The start-up time of the circuit has been improved with a circuit shown on Figure 5.25. When EN is low, the circuit pre-charges the capacitor to a voltage one V_{GS} drop below the supply rail, which should be (very roughly) $0.5 \times V_{\text{DD}}$. This hastens the locking of the DLL, as such a voltage should, after being converted to a biasing current by the V-to-I converter, produce a reasonable starting delay in the delay line.

The schematic on Figure 5.25 features twice the amount of current mirrors needed. The motivation for their inclusion is to stop transient disturbances from travelling backwards from the charge pump output switches to the biasing circuits. On the other hand, the additional mirrors increase the overall current mismatch. The current mirror devices were therefore scaled accordingly to minimize it.

There are two switches connected to each MOS transistor sourcing/sinking current to/from the charge pump capacitor. The additional switches (the PMOS switch connected to UP signal, and the NMOS switch connected to $\overline{\text{DWN}}$ signal) are not necessary for correct operation of the FPD, but they serve a purpose nonetheless. Their purpose is to keep the current sources supplying current at all times (when the FPD is enabled). If they were not included, the other switches would turn off the current paths completely and no current would flow. While this would save power, it would also slow down the operation, as some time would have to be spent on charging various parasitic capacitances before the current sourcing MOS transistors would supply current to the output capacitor again.

The transfer characteristic of the FPD & charge pump cell is shown on Figure 5.27. The input frequency was 625 MHz, which corresponds to a period of 1.6 ns.

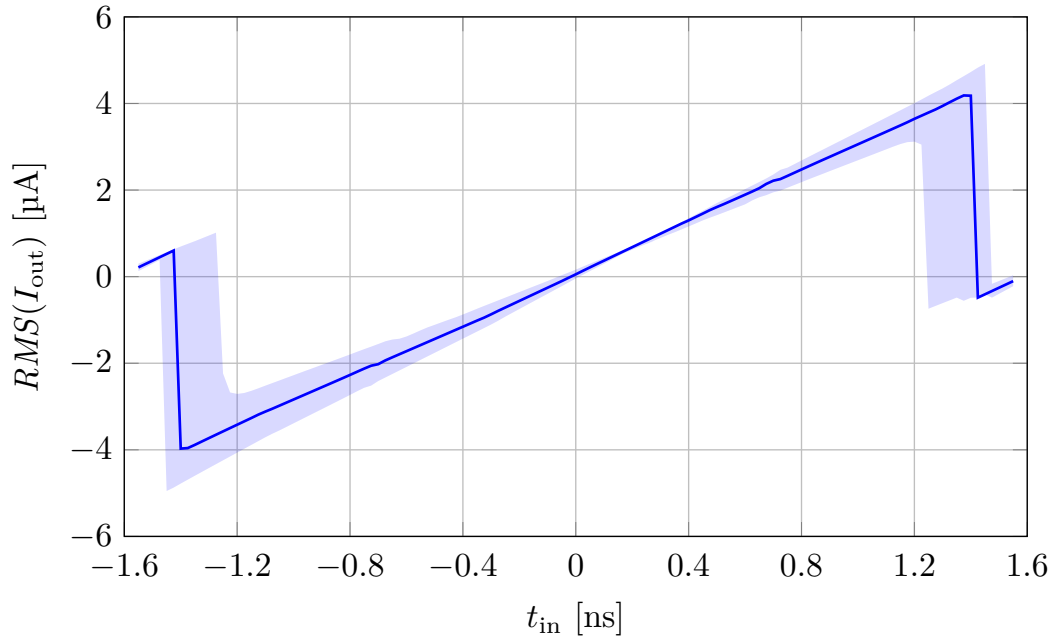


Fig. 5.27: *RMS* charge pump output current as a function of FPD input phase shift

As the span of the filled area shows, the transfer characteristic is somewhat PVT sensitive (which ultimately does not matter from a DC standpoint, as it only really affects how quickly the DLL reacts to changes in input phase), but it also features an abrupt change as the input phase shift gets closer to $\pm T_{\text{in}}$. This sharp edge is caused by the finite speed of the digital cells (when the phase shift approaches $\pm T_{\text{in}}$, the reset pulse can overlap with incoming input signal edges and prevent the FPD from reacting to them) and is unavoidable in the given topology. It could in principle cause false-locking of the DLL, which is why pre-charging the charge pump capacitor is important, as it ensures that the default state of the charge pump output creates a phase shift not too far from the target. Trimming of the DLL, which will be discussed in subsection 5.6.4, also helps prevent harmonic false locking.

The chart on Figure 5.28 shows how the V_{CPO} affects the charge pump output current. In this simulation, both $\overline{\text{UP}}$ and DWN signals were made active, and the desired value of output current is therefore zero, as the current supplied by the upper current source should be consumed by the lower current sink. The DC value of V_{CPO} , however, imbalances these current sources due to their finite output impedance and an error current flows out of the charge pump.

The chart shows that in order to minimize this error current, V_{CPO} should be kept somewhere in-between 0.5 V and 0.8 V, as in this region, both current sources are saturated and their output impedance is at its maximum. This is important for the design of the V-to-I converter (which ultimately defines the V_{CPO} the DLL will settle at) as well as trimming.

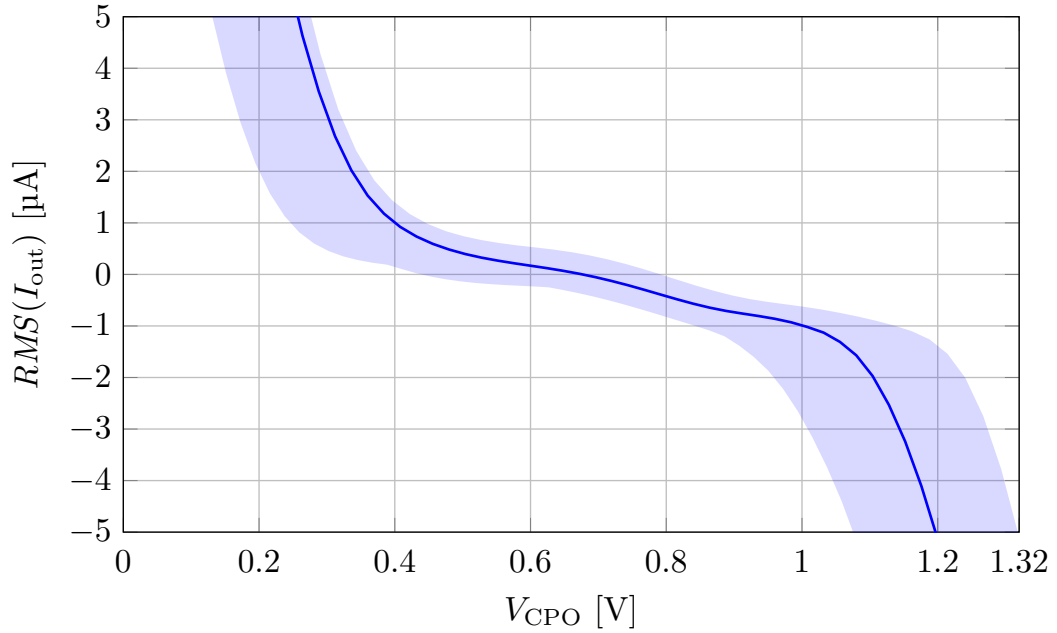


Fig. 5.28: *RMS* charge pump output current as a function of V_{CPO} when UP and DWN signals are both active

5.6.3 Voltage to Current converter

The task of this V-to-I circuit is to convert the V_{CPO} to a current, which is in turn used to bias the CCDLs and set their propagation delay correctly.

The obvious choice of converting voltage to current is the well-known opamp and resistor circuit shown on Figure 5.29. The primary problem of this circuit, the resistor's process variation, is not a critical issue thanks to the DLL locking mechanism. As long as the circuit is capable of producing the required current (which is a function of the resistor's resistance, opamp gain, V-to-I converter's current mirror mismatch etc), the V_{CPO} will adjust to accommodate it. Therefore, the resistor's process variation is transferred to charge pump output variation. As was already discussed in the previous section, though, this is a problem as well, as it is desired to keep V_{CPO} reliably in a certain region to maximize charge pump output impedance.

The first important design choice is the choice of the voltage domain. While all the other cells of the TDC signal chain have to be designed with thin oxide 1.2V devices due to the frequencies they operate at, the operation of the V-to-I circuit is fundamentally analog and could potentially benefit from thick oxide 2.5V devices with regards to quantities like *power supply rejection ratio (PSRR)*, current mirror output impedance etc. Furthermore, in the settled state, the circuit should be effectively DC and speed is not critical. On the other hand, 1.2V devices take up less area and offer faster reactions to voltage or temperature variation as well as faster start-up for a given biasing current. Additionally, keeping everything in a single

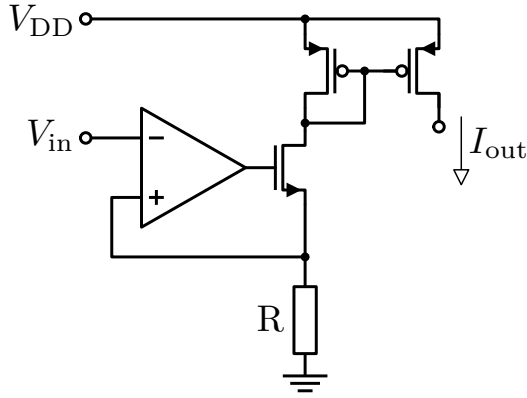


Fig. 5.29: Standard V-to-I converter

voltage domain prevents lots of issues with voltage domain separation, guarding (saving even more area), *Safe Operating Area* (SOA) violations etc. This is why the 1.2 V domain was chosen as well for the designed V-to-I converter.

Finding a topology suitable for 1.2 V domain which would pass PVT corners, especially with regards to sufficient ($> 50\text{ mV}$) saturation margin of each MOS device, is not obvious. The classic topology shown on Figure 5.29 is not suitable, as the voltage across the resistor should be close to $V_{DD}/2$ (which is the ideal operating point of the charge pump output) and the V_{GS} of the NMOS therefore pushes the opamp output voltage extremely close to V_{DD} where it loses gain regardless of the topology.

Driving a PMOS device instead of an NMOS one leads to voltage headroom issues with regards to the topmost PMOS current mirror. Assuming the opamp output voltage is reasonably low (cca $0.25 \times V_{DD}$), the source of the PMOS is one V_{GS} higher, and the remaining voltage needs to accommodate the PMOS mirror. It has been simulated that this is not feasible due to PVT variation.

A suitable topology is depicted on Figure 5.30, where the opamp is driving the topmost PMOS mirror directly.

The opamp output is a V_{GS} below V_{DD} , which for normal V_{th} thin oxide PMOS devices amounts to roughly middle of the rail. This enables the use of a folded cascode topology, which brings many benefits such as high gain, decent input and output voltage range and output dominant pole stabilization. This is especially beneficial as the opamp drives a lot of PMOS mirror devices, which all contribute with their gate capacitance to the compensation capacitance stabilizing the opamp loop. In order to improve the input voltage range even further, low V_{th} NMOS devices have been used for the input differential pair of the opamp.

In the case of this diploma thesis, only one ILRO current branch was implemented in the V-to-I converter' current mirror for simplicity. In reality, there would be tens or hundreds of ILRO current branches, each supplying biasing current to its ILRO.

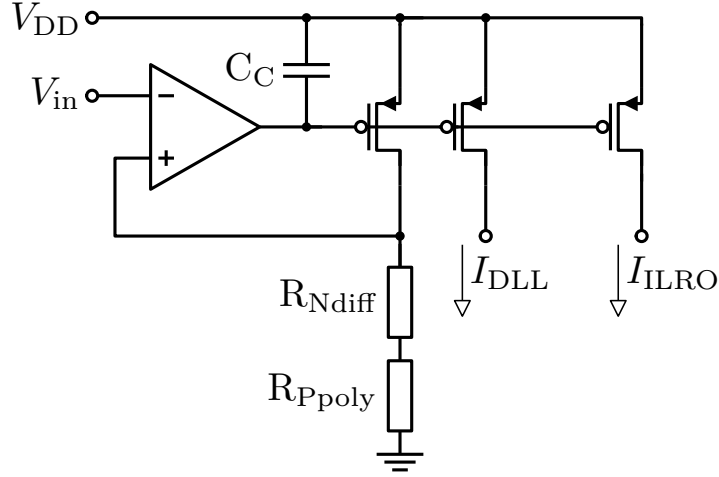


Fig. 5.30: Designed DLL V-to-I converter

These extra current branches would act as extra opamp output capacitance, as mentioned previously, so in this particular design, the compensation capacitor had to be increased proportionally to stabilize the loop. The capacitance used for stabilizing the loop across PVT corners was about 2 pF. Once the full amount of ILRO branches would be included, the capacitor's size could be decreased dramatically.

The sizing of the PMOS current mirror devices is very important, as they will be trimmed, which will be discussed in subsection 5.6.4. The goal was to ensure that the V_{GS} of the devices in nominal case is roughly 0.6 V, so that the opamp output voltage is also around 0.6 V and its output cascodes are therefore safely in saturation. Mismatch is also a concern and minimum length was therefore not viable – less than 10% mismatch at 3σ was sought after. The W/L size which satisfied both conditions is $9\ \mu\text{m}/0.75\ \mu\text{m}$. This is a rather wide and short transistor, which is not ideal for a current mirror (matching current mirrors is more area efficient for larger V_{GS}), but due to the low voltage supply, such a compromise needs to be made. The actual sizes of the PMOS transistors will be discussed in subsection 5.6.4.

The resistor which effectively converts the charge pump voltage to current is actually made up of two separate devices. The reason is the optimization of the TCR . As was shown on Figure 5.11, the amount of biasing current necessary to set the ILRO free-running frequency to 625 MHz decreases with temperature at a rate of cca $-15.4\ \text{nA}/^\circ\text{C}$. As a constant V_{CPO} is desired, this requires the resistor to have a positive TCR . An approximation for the desired TCR can be calculated as follows. First, the ratio of the ideal resistance value at hot ($135\ ^\circ\text{C}$) and cold ($-45\ ^\circ\text{C}$) will be evaluated, utilizing the values of the required current from Figure 5.11.

$$\frac{R_{\text{hot}}}{R_{\text{cold}}} = \frac{V_{\text{CPO}}}{I_{\text{hot}}} \cdot \frac{I_{\text{cold}}}{V_{\text{CPO}}} = \frac{I_{\text{cold}}}{I_{\text{hot}}} = \frac{17.5\ \mu\text{A}}{14.8\ \mu\text{A}} \approx 118.2\% \quad (5.2)$$

The result means that the resistor must grow by 18.2% over the temperature

range. The desired TCR can be estimated by the following equation.

$$TCR \approx \frac{18.2\%}{130^\circ\text{C} - (-45^\circ\text{C})} \approx 1040 \text{ ppm K}^{-1} \quad (5.3)$$

In ONK65, the P+ polysilicon resistor has a TCR of -154 ppm K^{-1} , while the N diffusion resistor has a TCR of 1400 ppm K^{-1} . The desired TCR can be implemented by combining the two resistors in series in a certain ratio r , which can be calculated using the following equations.

$$TCR = r \cdot TCR_{\text{Ndif}} + (1 - r) \cdot TCR_{\text{Ppoly}} \quad (5.4)$$

$$r = \frac{TCR - TCR_{\text{Ppoly}}}{TCR_{\text{Ndif}} - TCR_{\text{Ppoly}}} = \frac{1040 - (-154)}{1400 - (-154)} = 0.768 \quad (5.5)$$

The value of r was then optimized by simulation to 0.8 as this ratio seemed to produce more stable V_{CPO} when V_{DD} variation was taken into account as well.

The total resistance of the combination was calculated as follows

$$R_{\text{total}} = \frac{V_{\text{CPO}}}{I_{\text{BIAS}}} = \frac{0.6 \text{ V}}{16.22 \text{ }\mu\text{A}} = 37 \text{ k}\Omega \quad (5.6)$$

and this value was optimized by simulation to $34 \text{ k}\Omega$. The reason for the difference is the previously discussed fundamental discrepancy between the way the CCDL is connected in an ILRO and in the DLL. The DLL delay line actually requires more current to produce the desired 800 ps delay than the ILRO requires to produce the desired 625 MHz oscillations.

There is a problem that needs to be considered when a combination of two resistors manufactured by different processing steps is utilized – these resistors are uncorrelated. It is perfectly possible for the P+ polysilicon resistor to skew towards higher resistance, while the N diffusion resistors skews towards lower resistance and vice versa. These “cross” corners (the resistors skew in opposite directions) will affect the overall TCR and it is necessary to validate that the system operates correctly in these corners as well.

The problem with validating these cross corners is that the resistor corner model files only define two corners: hi3s and lo3s, which vary the resistors in the same direction as if they were perfectly correlated. Correct simulation of the uncorrelated variation can be done via Monte Carlo simulations, but because simulating the whole DLL is very time intensive and Monte Carlo simulations require hundreds of runs, it is not practical.

Instead, a separate short Monte Carlo simulation was run to determine the variation of these resistors on their own. It was simulated that the P+ polysilicon resistor varies by $5\%/\sigma$, while the N diffusion resistor varies by $6.7\%/\sigma$. For 3σ , this corresponds to 15% and 20% variation respectively.

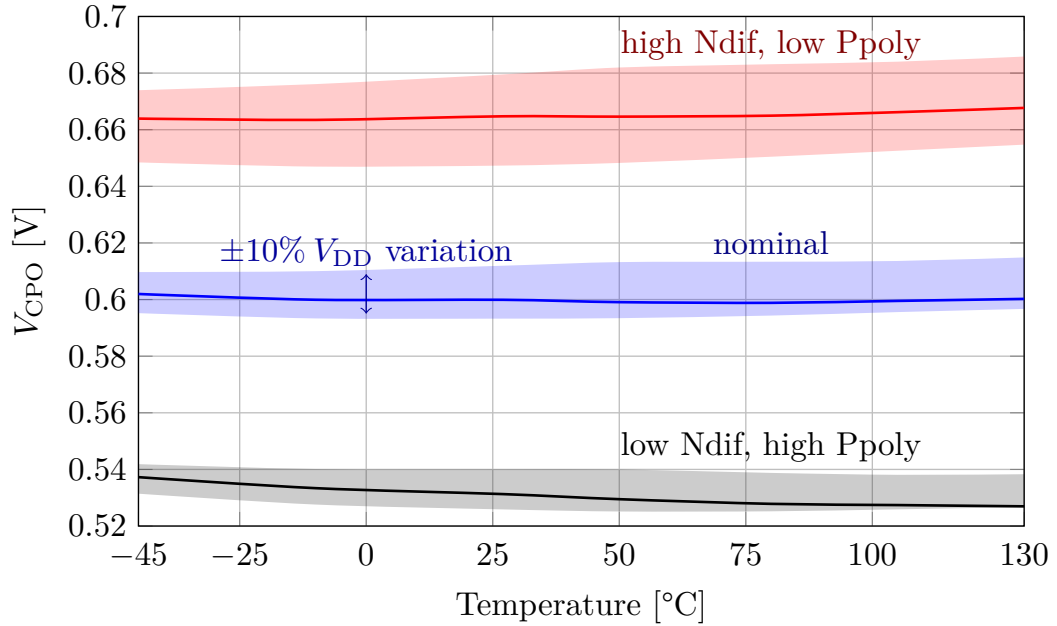


Fig. 5.31: Charge pump output voltage stability over temperature for nominal and cross resistor corners

In order to validate the cross corners, the resistors were manually skewed by the aforementioned 15% and 20% in opposite directions, and the resulting value of the V_{CPO} is shown on Figure 5.31. This simulation considers only VT variation, MOS process corners were not applied as MOS process variation would skew the mean V_{CPO} value heavily and trimming will trim out this static source of variation anyway (as will be discussed in subsection 5.6.4).

While the mean value of the V_{CPO} varies in these cross corners as the total resistance of the series combination changes, this will not be an issue due to trimming (which will be discussed in the following subsection 5.6.4). The important takeaway here is that thanks to TCR compensation, V_{CPO} is remarkably stable over temperature even in the cross resistor corners. Furthermore, as these resistors are uncorrelated, skewing both resistors by 3σ actually corresponds to much lower probability. In this particular case, since the probability of a skew larger than 3σ in either direction is around 0.27%, the probability of two such skews happening independently and concurrently is $0.0027^2 = 0.000729\%$, which actually corresponds to nearly 4.5σ . The cross corner cases depicted on Figure 5.31 are therefore overly pessimistic.

The *transconductance* (g_m) from V_{DD} to output current of V-to-I converter's current mirror is shown on Figure 5.32 (this is similar to $PSRR$, but for an output current quantity and not relative, in this case). The transconductance starts rising once the opamp loop starts losing its gain and grows to rather large values at high frequencies. The special frequencies of interest are the harmonics of 625 MHz,

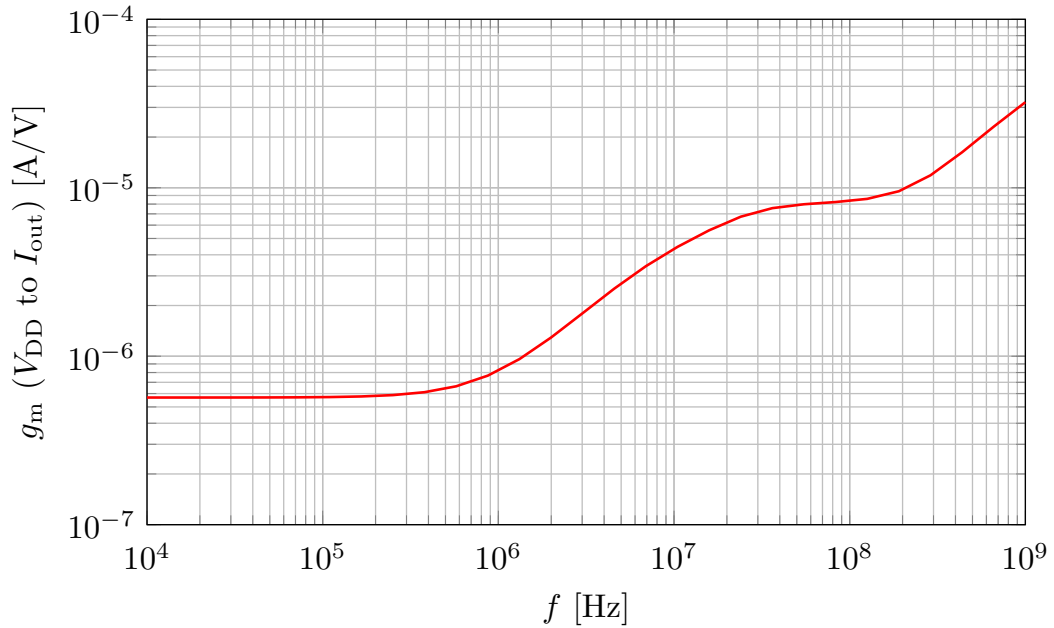


Fig. 5.32: Transconductance from V_{DD} to output current of V-to-I converter's current mirror

where the transconductance is obviously far from ideal. The problem is that with only 1.2 V supply voltage, better power supply rejection is difficult to implement. At hundreds of megahertz, in this processing technology, it is unrealistic to expect that disturbances this fast can be effectively blocked. The designers and layouters should be focused instead on preventing the coupling of high frequency disturbances to sensitive parts of the supply rail in the first place. This can be achieved with proper layout techniques such as increasing distances between noisy and sensitive nodes or by implementing RC low pass filters for the V_{DD} on chip to filter both external and internal noise.

A list of scalar quantities related to the designed V-to-I converter is shown in Table 5.7.

The input voltage range of the opamp was defined as the range of input voltages for which the systematic offset does not exceed ± 3 mV. This was set empirically, as outside this range, the systematic offset increases rapidly due to loss of gain.

The offset of the opamp (and therefore the open loop gain as well) is not critical in this application, as the only consequence of it will be the offset of V_{CPO} . While V_{CPO} should be kept around middle of the rail in order to balance the charge pump current sources, it is not so sensitive that a few millivolts would make any significant difference.

The saturation margins and stability were investigated for extreme input voltages of 0.4 V, 0.6 V and 0.8 V, as well as for extreme trim code combinations (trimming will be discussed in the following subsection 5.6.4). The results listed in Table 5.7

Tab. 5.7: V-to-I converter performance

Analysis	Quantity	Min.	Typ.	Max.	Unit
PMOS current mirror	Random mismatch		2.8		$\%/\sigma$
	Systematic offset	-4.2	0	3.3	$\%$
Opamp	Device area		219		μm^2
	Supply current	10.7	13.8	17.9	μA
	Power consumption	11.5	16.6	23.6	μW
	Input voltage range	0.25		0.95	V
	Systematic offset	-2.7	0	0.75	mV
	Random offset		2		mV/ σ
	Feedback loop	Open loop gain	55.1	69.2	74.7
GBW		4.2	8.9	15.7	MHz
Phase margin		60.5	82	96.8	$^\circ$
Gain margin		18.5	24.7	32.2	dB

are therefore overly pessimistic, but still satisfactory, which proves the robustness of the design.

The *gain bandwidth product* (GBW) of the feedback loop is important for the settling time, and it should not be slower than the charge pump output – otherwise, the DLL could actually ring for a long time. In this design, there is no ringing of V_{CPO} , which will be shown later in section 5.7, and if there was, either the charge pump capacitor could be increased or the charge pump current sources could be made to supply less current. This would slow down the charge pump output in relation to the opamp loop, so that it is not the bottleneck any more.

5.6.4 Trimming

As was discussed previously in subsection 5.3.3, trimming is a necessary addition to the system.

The first reason for this is the inability of the DLL to bias the ILROs accurately on its own. As was already discussed in previous sections, this is because the delay line in the two circuits, while identical, is connected in a different manner. Furthermore, mismatch between the DLL and ILRO delay lines, FPD non-idealities, mismatch in the V-to-I converter current mirror and many other sources of error contribute to the discrepancy. Trimming each ILRO individually is necessary to bring their free-running frequencies closer to target 625 MHz so that they can be reliably locked in all conditions. An equally important benefit of the ILRO trimming is the reduction of systematic DNL error, as was also discussed in subsection 5.3.3.

The second reason is the need for trimming the DLL itself. In some process corners, the output of the charge pump could settle to a voltage that is close to the ground or the supply voltage. As was foreshadowed in subsection 5.6.2, in such cases, the charge pump current sources would be heavily imbalanced and the charge pump would therefore integrate some error current, which would lead to the DLL not producing the correct amount of delay in the delay line and therefore the supplied biasing current to the ILRO would not be (close to) the current needed to achieve the desired free-running frequency. Furthermore, even if the charge pump were to be ideal and perfectly balanced, a charge pump output voltage too close to the rails could be problematic due to limited V-to-I converter input voltage range.

An added benefit of trimming the DLL is the prevention of false harmonic locking of the DLL. The harmonic locking can occur easily for example when the process variation sets the V-to-I converter's resistor's resistance rather high and/or makes the MOS transistors (and therefore the delay lines) rather slow. In such a case, the DLL could by-default generate smaller than usual biasing current because of the unusually high resistance, and the resulting unusually low current would produce a very long delay in the unusually slow delay line. In such case, it is possible for the DLL to settle incorrectly, producing not one clock period worth of delay in the delay line, but two clock period delay instead. Consequently, the biasing current produced by the falsely-locked DLL would, when biasing the ILROs, produce very slow oscillations. These oscillations would be too slow to enable the ILROs to be locked to the target frequency. Trimming the DLL itself can prevent this by trimming out this process skew and steering the DLL to the desired operating point, i.e. when the charge pump output is roughly in the middle of the supply rails, as was discussed previously in subsection 5.6.2. During the trimming process correct locking can be verified by also measuring the signals entering the FPD, as long as these signals are made accessible to the tester.

Since all the aforementioned problems are caused by fabrication variation (mismatch and process), these errors are static, i.e. they are set at the point of fabrication and should not change significantly over time. Therefore they can be in principle trimmed out and trimming is a valid option.

A decision has been made to trim the circuits in two steps: first, the DLL will be trimmed to set the operating point of the charge pump close to the ideal middle of the supply rail. Then the ILROs will be trimmed individually to set their f_0 as close to target (625 MHz) as possible.

Trimming will be performed in a controlled environment at a specified temperature and with an accurate power supply. In practical applications, the temperature and the supply voltage will vary, but this variance should be compensated by the DLL to ensure that the ILROs still lock properly, which will be verified by simula-

tion.

The implementation of both trimming steps is shown on Figure 5.33.- Both steps are implemented by trimming the PMOS transistors in the V-to-I converter's current mirror.

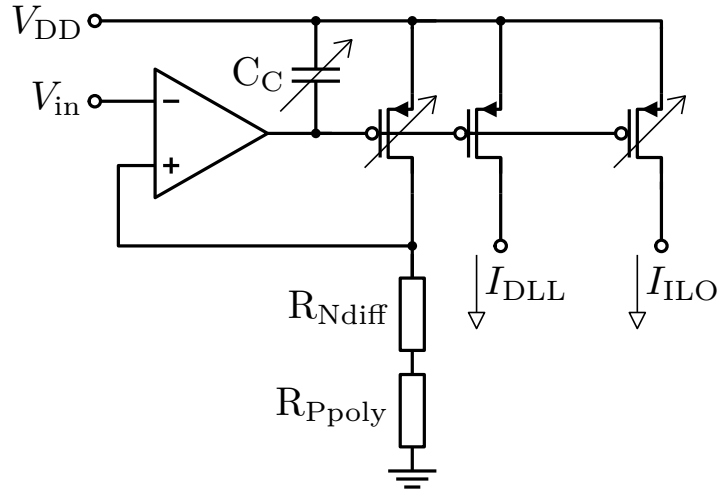


Fig. 5.33: Designed DLL V-to-I converter with trimmed devices highlighted

The DLL trimming is done by trimming the width of the PMOS transistor in the resistor branch. By trimming the width of this transistor, the current gain of the mirror from this reference current branch to all the other branches (both the ones supplying current to the DLL delay line as well as the numerous branches supplying the ILROs) can be adjusted. This can be therefore thought of as a kind of global trimming, as it affects both the DLL as well as all the ILROs, and therefore it clearly needs to occur first. The goal here is to adjust the width of the trimmed PMOS transistor so that the output voltage of the charge pump settles at 0.6 V, which is the ideal default operating point of the DLL.

This could also be achieved by trimming the V-to-I converter's resistors, but that would be a more problematic way to implement this trimming step, as the switches in the resistor trimming network would need to be relatively wide, and since there are two independent resistors which should be weighed in a certain ratio in order to implement target TCR , the network would be large and complex. Connecting or disconnecting PMOS units to the opamp output node is a simpler method in comparison. Both trimming methods are non-linear though (both the resistor and the PMOS transistor in the resistor branch are in the denominator of the equation defining the current in the DLL or ILRO branch).

The individual ILRO trimming also occurs in the PMOS current mirror and is also done by trimming the width of the PMOS transistor supplying current to the trimmed ILRO instance, but the trimming could have been also done within the ILRO cell itself, for example in its NMOS biasing mirror. There are a few reasons

why trimming the ILRO biasing mirror was not chosen. First of all, the NMOS biasing mirror has sixteen branches, one for each CCDLU of the delay line, so trimming each of them would take a lot of area. Trimming the current mirror input diode would be possible, but it would also be non-linear. More importantly, however, this would mean that more routing would need to be connected to each ILRO. It is critical that the routing of the clock phases generated by the ILROs to the TDCs is as balanced as possible, and having more routing in this area could complicate it needlessly. Furthermore, more routing next to the ILROs would lead to more noise coupled into the ILROs, which should be prevented. Therefore the ILROs biasing current will be trimmed in the V-to-I converter PMOS mirror as well. The goal of this trimming is to individually adjust the width of the PMOS transistors supplying the biasing current to the ILROs in order to get the free-running frequency of the unlocked ILROs as close to 625 MHz as possible.

A problem with this trimming technique is that the stability of the V-to-I opamp feedback loop is now trimming code dependent. This is because when the DLL trimming code is high, the width of the PMOS transistor feeding the resistors is increased, therefore its g_m is high and the loop has higher gain. Higher gain means the loop's GBW is higher and its phase margin at the GBW frequency can be worse than desired. Widening the PMOS transistor increases the total capacitance connected and therefore slightly decreases the frequency of the dominant pole, but not enough to offset the stability reduced by the higher gain. The loop is also dependent on the ILRO trimming codes, as each transistor connected via a switched-on passagete represents an RC load, adding additional pole and zero into the transfer function of the loop.

A solution is to add extra capacitance to *compensation capacitance* (C_C) in proportion to the trimming codes, and therefore the compensation capacitance is also trimmed and made up of binary weighted units. This is represented on Figure 5.30. The goal was to make the stability of the loop independent on the code (it is not desirable to overcompensate the loop either), which was successfully achieved as proven by Table 5.7.

There is a second issue related to DLL trimming in particular – adjusting the width of the PMOS transistor supplying the V-to-I converter's resistors also adjusts their V_{GS} , as the total current flowing through the PMOS parallel combination is constant. This means that for low trimming codes, the V_{GS} can be very high, pushing the output voltage of the opamp low and vice versa. This can make the output cascodes of the opamp step out of saturation. In this particular design, all the opamp devices are kept safely in saturation over all PVT corners with a minimum 50 mV margin, but if a wider trimming range was desired, this could be an issue.

In order to not make the layout of the PMOS mirror too complex, it was decided that no more than 5 bits will be used for the trimming. More bits would require splitting the PMOS transistors into a larger number of narrower pieces, which would increase the routing overhead, reduce matching, increase noise coupling etc. Binary weighing the segments is an obvious decision for the aforementioned reasons.

Determining the necessary range of the trimming circuits is not simple, as there are a lot of error sources that need to be trimmed out at once with this trimming and the trimming range should be able to correct all of them once combined. The DLL trimming range should primarily cover MOS process corners (as they heavily affect the delay lines), V-to-I converter resistor variation, V-to-I converter PMOS mirror mismatch, the mismatch of the NMOS mirror in the DLL delay line, the mismatch of the CCDLUs inside the DLL delay line, V-to-I converter opamp offset and FPD & charge pump mismatch. The individual ILRO should then trim out the V-to-I converter PMOS mirror mismatch as well as the ILRO NMOS mirror mismatch and the mismatch of the CCDLUs themselves (process variation of the delay line is shared by the delay line in the DLL and should be theoretically trimmed out by the DLL trimming). The trimming range was therefore adjusted by simulation in such a way to ensure that the ideal trimming codes for the nominal corners are roughly in the middle of the range and that the trimming circuits can correct both 3σ process and mismatch variation, ideally with a margin of a few codes left to spare.

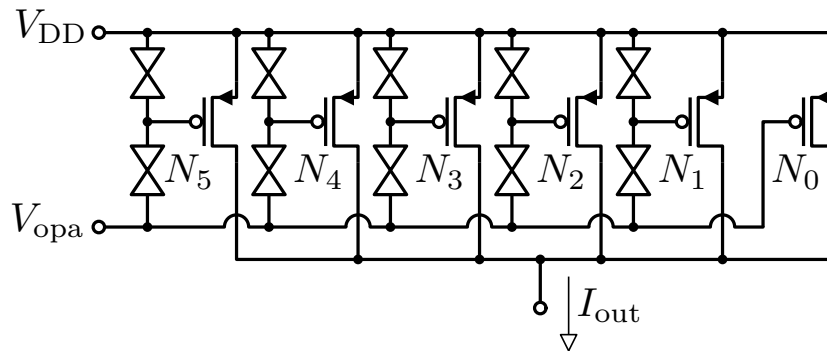


Fig. 5.34: Trimmed PMOS current mirror devices

Both the DLL trimming and ILRO trimming PMOS transistors are made from a parallel combination of binary weighted 175 nm/750 nm units which are either switched to the supply rail or the opamp output node by minimum sized passgates, as shown on Figure 5.34.

The number of parallel units connected by each binary weighted bit is listed in Table 5.8. It can be seen that the DLL trimming PMOS transistor is trimmed more coarsely, as its LSB consists of two parallel 175 nm/750 nm units, while the ILRO trimming step is made from a single 175 nm/750 nm unit.

Tab. 5.8: Overview of V-to-I converter's current mirror trimmed PMOS units

Current branch	N_0	N_1	N_2	N_3	N_4	N_5	Range
Resistors	32	2	4	8	16	32	32-94
DLL	32						62
ILRO	44	1	2	4	8	16	44-75

This makes sense because DLL trimming is global, occurs first, needs to trim out process variation and the trimming does not have to be very precise (as Figure 5.28 proves, the charge pump operates well in a relatively wide range of V_{CPO} voltages). On the other hand, ILRO trimming is finer and the target is to trim the f_0 within only a few percent of target frequency.

The reason for that was foreshadowed in subsection 3.3.2 and discussed in subsection 5.3.3. If a perfectly balanced ILRO oscillates at f_0 , the propagation delay of each individual cell is exactly

$$t_{d,0} = \frac{1}{2Nf_0} \quad (5.7)$$

where N is the number of stages, and when this ILRO is locked to $f_{\text{inj}} \neq f_0$, the average propagation delay \bar{t}_d needs to fit the following equation.

$$\bar{t}_d = \frac{1}{2Nf_{\text{inj}}} \quad (5.8)$$

In order to accommodate this requirement, the propagation delay of the injection stage adjusts. If there is only one injection stage like in this case, the average propagation delay can be written as follows.

$$\bar{t}_d = \frac{(N-1) \cdot t_{d,0} + t_{d,\text{inj}}}{N} \quad (5.9)$$

where $t_{d,\text{inj}}$ is the propagation delay of the injection stage. After rearranging, the following equation can be written for $t_{d,\text{inj}}$.

$$t_{d,\text{inj}} = \frac{1}{2f_{\text{inj}}} - \frac{N-1}{2Nf_0} \quad (5.10)$$

Since in our case both f_{inj} and N are known, we can plot the relationship between the relative error of f_0 (its ideal value is equal to $f_{\text{inj}} = 625$ MHz) and $t_{d,\text{inj}}$ (the ideal value being 50 ps), as shown on Figure 5.35.

Since the goal is to keep the DNL of the FTDC timestamp below 50%, f_0 needs to be trimmed within 3% of target. To get some safety margin when the CCDLU mismatch is unfavourable or the VT conditions cause the f_0 to skew, it would be better to trim f_0 within 1%. On the other hand, this can be difficult to achieve with only 5-bit trimming.

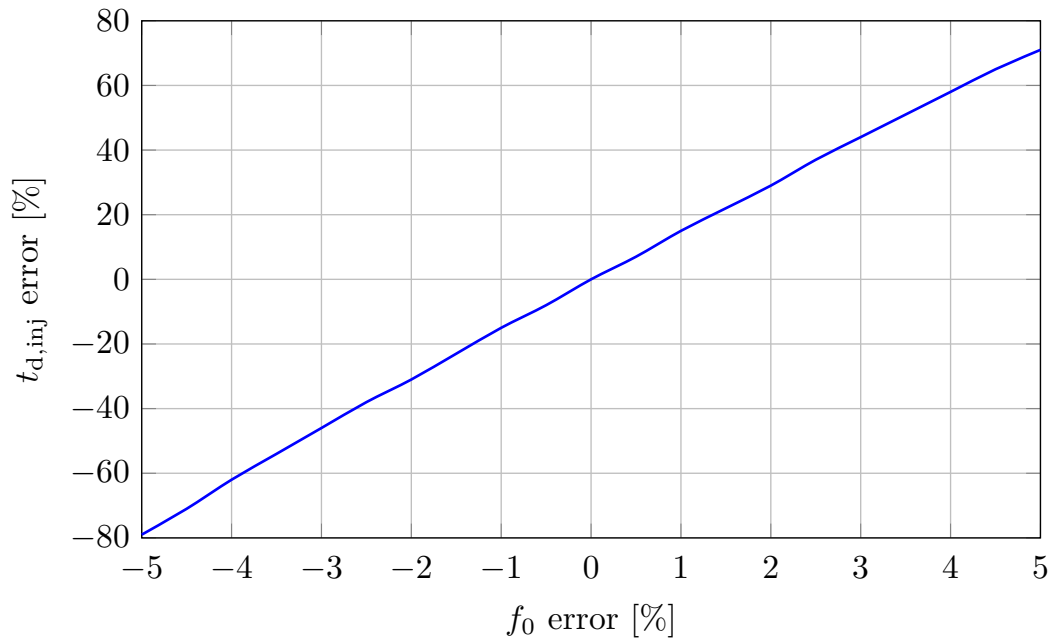


Fig. 5.35: Error of the injection stage propagation delay as a function of the free-running frequency error

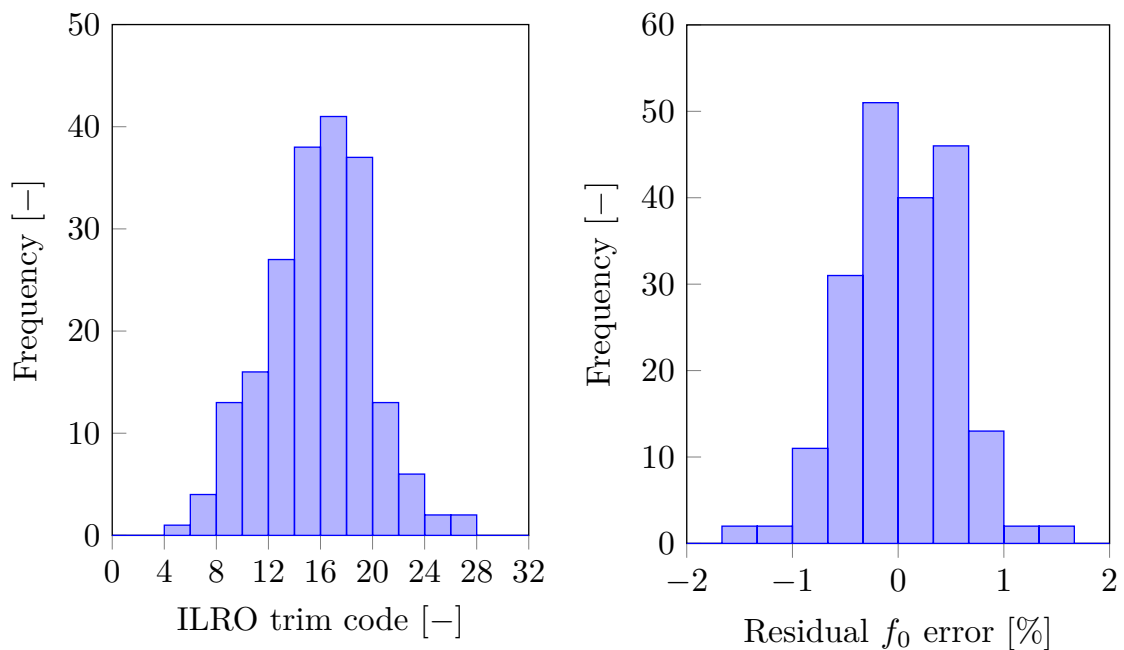


Fig. 5.36: ILRO trimming trim code and residual error histograms

The results of a Monte Carlo simulation simulating the ILRO trimming process on its own (without the DLL biasing) are shown on Figure 5.36. On the left chart it can be seen that all the trim codes are well within the limits with a trim code standard deviation of 3.9 codes. The residual f_0 error after trimming is shown on the right. The standard deviation is 0.51%, i.e. roughly 95% (2σ) units are within $\pm 1\%$ of f_0 error.

There is no straightforward way to define the specification for DLL trimming, so its trimming circuit was designed iteratively based on simulations in order to guarantee that the trimming range is wide enough to ensure that all PVT corners could be trimmed successfully. The DLL trimming simulations will be shown in the following section 5.7.

5.7 Top-level simulations

In this section, a few top level simulations will be shown in order to prove that the designed circuits function properly not only when separated into individual blocks, but also at fully connected top-level.

Because simulating the DLL takes a long amount of time (because of the need to simulate picosecond-level time steps along with microsecond-level time spans), Monte Carlo simulations have not been performed on the top-level. Instead, PVT corners were utilized. As long as mismatch can be neglected (or is taken into account by separate simulations), this should prove the robustness of the design successfully, as the PVT corners are very pessimistic (unlikely) combinations of extreme values.

The start-up of the DLL (nominal corner) is depicted on Figure 5.37.

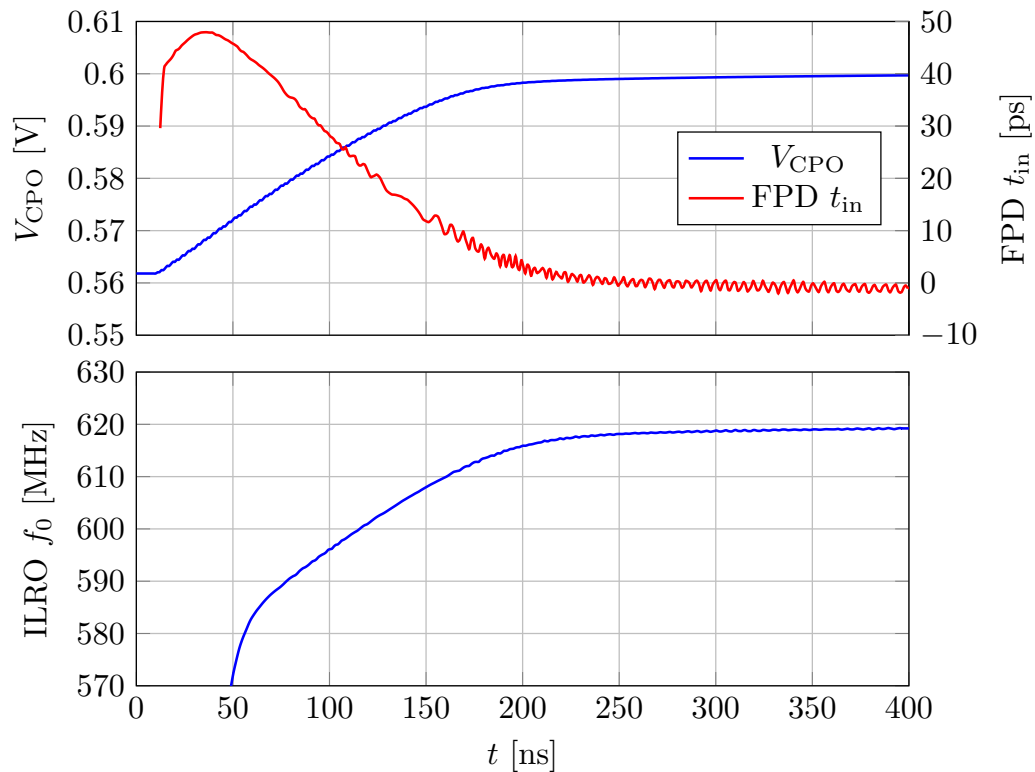


Fig. 5.37: DLL start-up waveforms

At $t = 0$, the charge pump voltage V_{CPO} is pre-charged to roughly $V_{DD}/2$ (0.562 V

to be precise) via the circuit shown on Figure 5.25. This biases the DLL delay line quite close to the ideal point, as the FPD input phase shift t_{in} shows. The DLL then adjusts the input phase shift close to roughly -1 ps. This is not exactly zero due to various FPD related errors discussed in subsection 5.6.2, but its close enough. The free-running frequency of an ILRO biased by the current produced by this DLL is shown on the lower half of the chart. It stabilizes around 620 MHz, which is close enough to the target frequency to ensure locking (typical locking range was depicted on Figure 5.16).

A comparison between *Constant Current* (CC) biasing and DLL biasing of an ILRO is shown on Figure 5.38. This simulation does not include process variation.

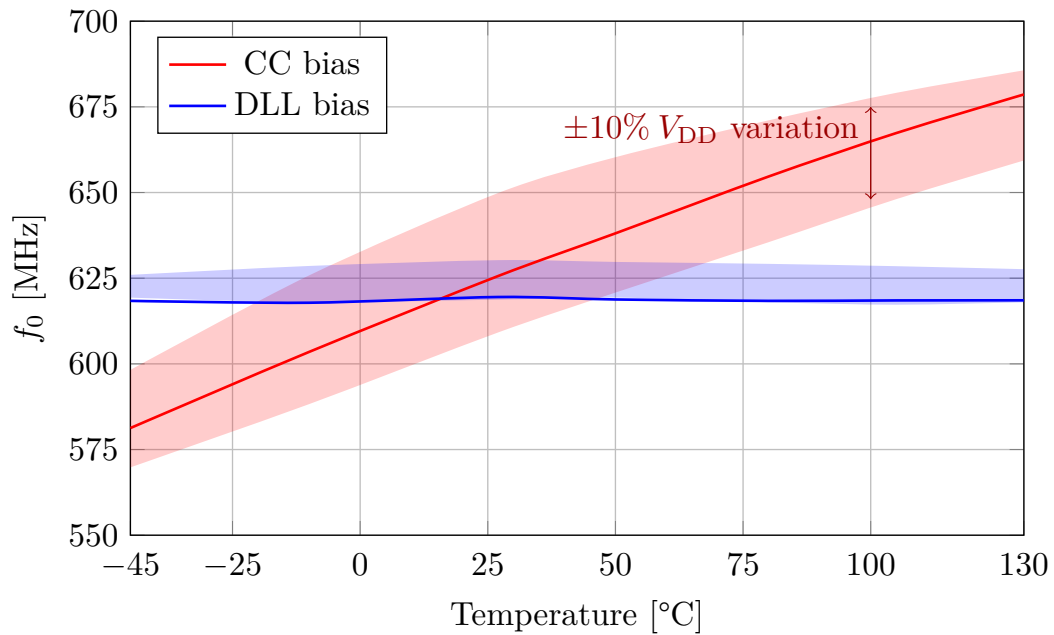


Fig. 5.38: ILRO free-running frequency over VT variation when biased by a constant current versus a DLL

The CC biasing approximates band gap reference & P+ polysilicon resistor biasing, which would be a much less complex alternative to the DLL. As the chart proves, such biasing would be not only much more temperature sensitive, but V_{DD} sensitive as well. This chart therefore justifies the added complexity of the DLL, as its operation stabilizes the free-running frequency of the ILRO significantly.

A full trimming simulation has been run for PVT corners. The simulation plan was as follows:

1. Set a MOS and resistor process corner, keep V_{DD} and temperature nominal
 - VT can be kept nominal as trimming in production would be performed in controlled VT conditions
2. Run a transient simulation for each DLL trimming code

- Each transient run has to be long enough for the DLL to settle (cca 500 ns)
3. Evaluate V_{CPO} at the end of each run
 4. Save the DLL trim code which minimizes $|V_{\text{CPO}} - 0.6 \text{ V}|$
 5. Run a transient simulation for each ILRO trimming code, utilizing the saved DLL trim code
 - Each transient run has to be long enough for the ILRO to settle (cca 500 ns as well)
 - The ILRO is free-running in this simulation (injections disabled)
 6. Evaluate ILRO instantaneous frequency at the end of each run
 7. Save the ILRO trim code which minimizes $|f_0 - 625 \text{ MHz}|$
 8. Run a transient simulation of the whole DLL + ILRO system, utilizing both saved trim codes, for each V_{DD} and temperature corner
 - VT corners are swept now, because trimming has been completed at this point and this simulation should verify functionality for all possible operating conditions
 9. Evaluate instantaneous frequency of the ILRO and verify it locks, evaluate propagation delay between each ILRO stage
 10. Repeat for all process corners

The DLL trimming sweeps are shown on Figure 5.39. All corners are trimmed within 8 mV of target. While the span of the trim codes is rather wide (from code 5 to code 29), it has to be acknowledged once again that these process corners are quite pessimistic and each line actually represents a combined 4.5σ corner.

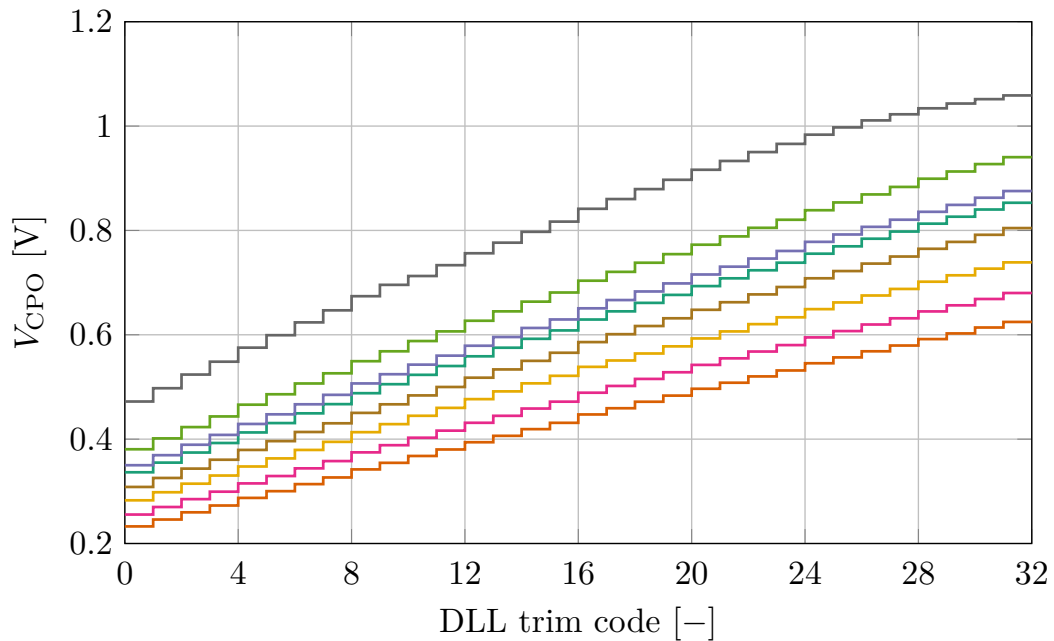


Fig. 5.39: DLL trim code sweep across process corners

Nevertheless, it is possible that the DLL trimming range could be insufficient once mismatch of the DLL delay line is considered. A mismatch-only Monte Carlo simulation has been run to identify what range of biasing currents is necessary to keep the propagation delay through the whole delay line equal to 800 ps, which is what the DLL feedback tries to achieve. It has been evaluated that the current can vary by up to $6.5\%/σ$ to satisfy this condition. For $3σ$ (i.e. 20%) biasing current variation, the trimming code of the DLL trimming circuit has to change by up to 8 codes (this can be calculated based on the ratios of trimmed PMOS units from Table 5.8).

If the process corners can require codes as high as 29 and mismatch can require up to 8 codes to compensate, it is definitely possible for a process & mismatch combination to be so skewed that the DLL trimming circuit's trimming range could be insufficient to bring V_{CPO} close enough to target value. Increasing the DLL trimming circuit's range has been attempted, but this led to problems with the opamp operating point, which is an issue discussed in the previous subsection 5.6.4. Solving this would probably require 2.5 V supply voltage domain for the V-to-I converter, as it is essentially a voltage headroom problem, which depending on the project specifications might or might not be possible.

It has to be noted once more, however, that the probability of a $4.5σ$ process corner along with $3σ$ delay line mismatch is very low and is definitely much lower than the inverse of yield of the manufacturing process itself, which is usually around 90% for mature processes, so it is not a critical issue which would require a redesign automatically.

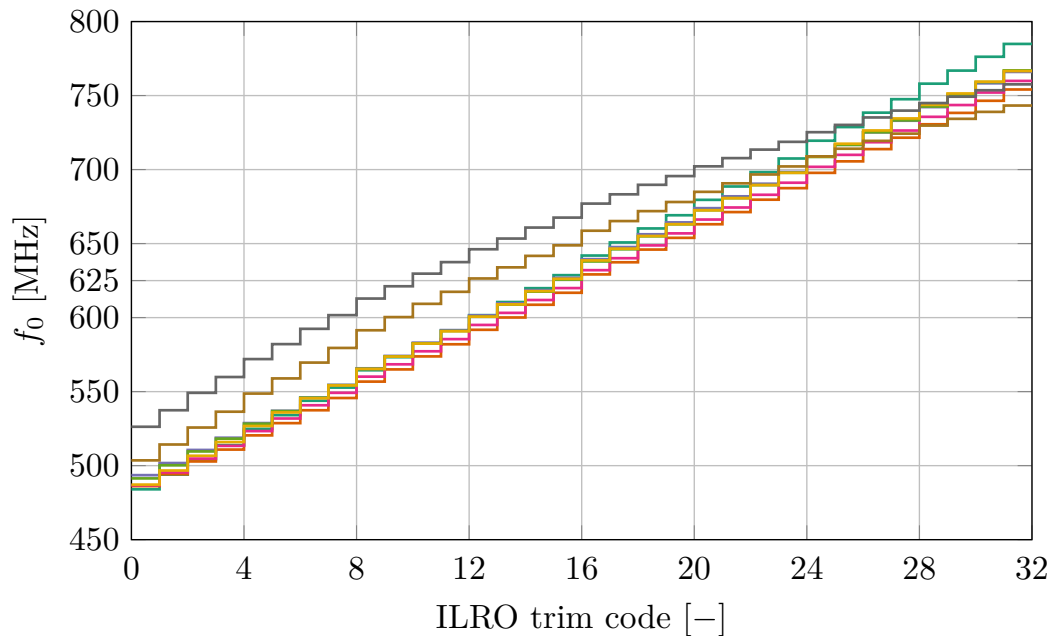


Fig. 5.40: ILRO trim code sweep across process corners

The ILRO trimming sweeps are shown on Figure 5.40. The span of required codes over process corners is much narrower and when combined with the span of codes required to compensate mismatch, as simulated on Figure 5.36, there should be no problems trimming the ILRO. The worst case residual f_0 error in this simulation is 5.1 MHz.

Two corners from Figure 5.40 stand out, however – they are less linear and require lower trim codes than the rest. These are the slow MOS corners. This process corner has been highlighted on Figure 5.10 and its curvature is related to the fact that the CCDLU transistor devices could not have been kept minimum sized because of mismatch requirements. This is a compromise that needed to be made in order to meet the 50 ps specification as well as realistically possible in the given processing technology.

The instantaneous frequency of trimmed ILROs over selected PVT corners is shown on Figure 5.41. In this simulation, the ILROs are left free-running until $t = 0.3 \mu\text{s}$, when injections start. It can be seen right away that a single corner does not lock at all and shows a typical periodic frequency pattern which occurs in such cases.

This problem was analysed further. The charts on Figure 5.42 show a PVT corner overview of a simulation, where a trimmed ILRO biased by a trimmed DLL is free-running. It is essentially the same simulation as before, but without injections. This allows us to inspect the free-running frequency f_0 better. The top chart focuses on the ILRO, while the other three charts focus on the operation of the DLL itself.

As Figure 5.42 shows, nominal, FF and FS corners operate very well, as the f_0 is close to target, so is V_{CPO} , the biasing currents are within expected range and the input phase shift seen by the FPD is close to zero. The situation is slightly worse for SF corners, but not by much. The problems are however clearly visible for SS corners. The f_0 spreads widely, V_{CPO} can be off target by up to 0.3 V, the biasing currents produced by the DLL can be very large and the FPD input phase shift is not close to zero any more for some SS corners.

There are actually two different groups of problem SS corners, denoted SSa and SSb and shown on Figure 5.42 with a different mark shape. The group SSa are corners where the DLL itself does not operate properly. It appears that the DLL delay line in these corners requires a lot of current to produce the desired delay, which causes the feedback to increase V_{CPO} significantly. This, in turn, squeezes the voltage headroom of the upper charge pump current source, which produces a mismatch between the upper and lower charge pump current sources, causing the loop to correct it with non-zero FPD input phase shift, which can be as much as 25 ps. This means that the delay produced by the DLL delay line is not very close to 800 ps any more, which worsens the VT tracking capability of the DLL, causing

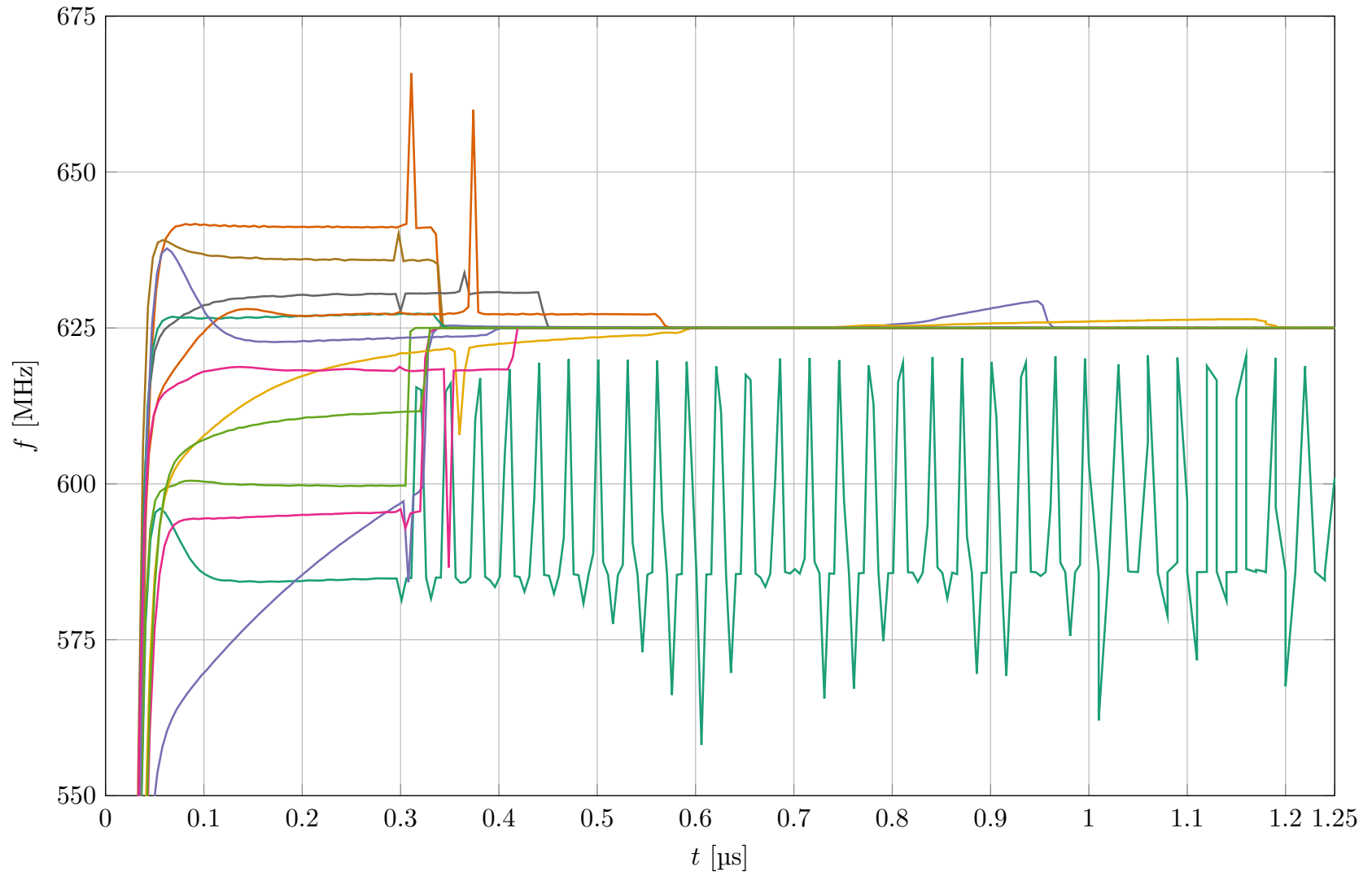


Fig. 5.41: Instantaneous frequency of ILRO across selected PVT corners during locking after trimming

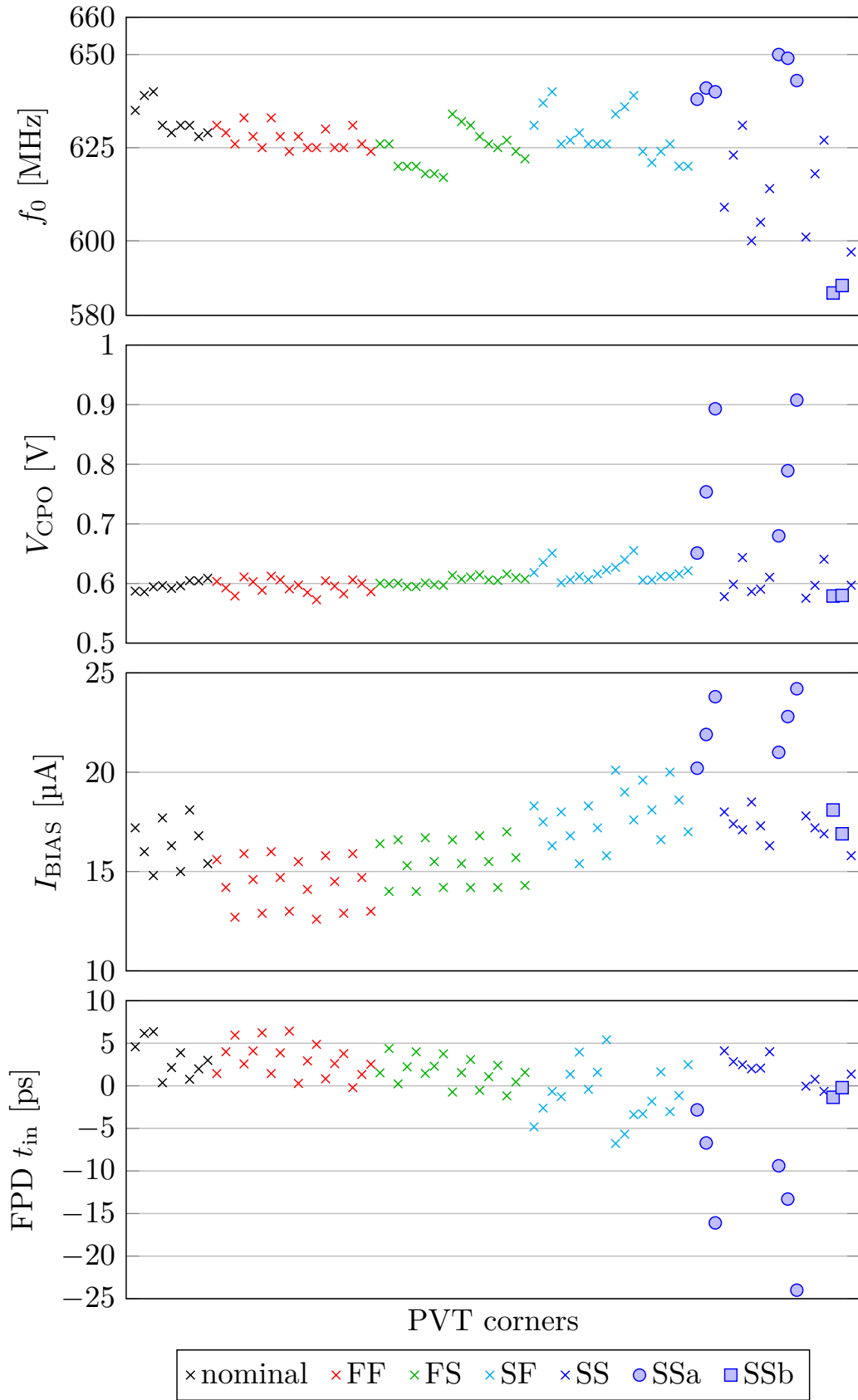


Fig. 5.42: PVT corner overview of free-running trimmed ILRO biased by a trimmed DLL

the f_0 to skew away from 625 MHz. These corners are able to lock to 625 MHz nevertheless.

The reason for why these corners require so much current is shown on Figure 5.10, as these are the slow corners for which the $f_0 = f(I_{\text{BIAS}})$ function is flatter than usual, requiring a wider range of currents to meet the target f_0 . The flatness of these corners is caused by the increased size of the CCDLU transistors, which was required in order to lower the mismatch so as to meet the DNL criteria. Fixing this corner would therefore lead to a different, statistically more likely issue.

The corner group SSb is the more problematic one. As the lower three charts show, the DLL operates perfectly fine in these three corners and none of the three lower charts suggest there should be any issue at all. However, the f_0 of the ILRO in these corners is so low that it is outside the locking range and these corners are not able to lock to 625 MHz. The explanation is that the fundamental discrepancy between the DLL and ILRO is exacerbated in these corners, preventing a perfectly functioning DLL from biasing ILRO correctly.

Another point of view on the situation is shown on Figure 5.43. This compares the sensitivity of f_0 to V_{DD} for SS corners versus all the other corners.

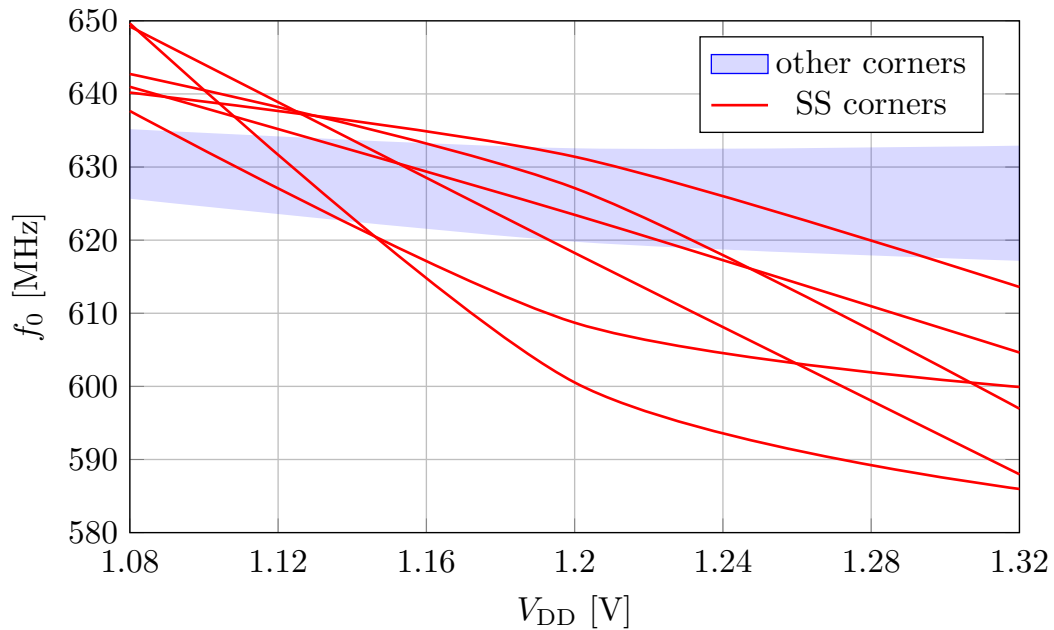


Fig. 5.43: Free-running frequency of trimmed ILRO biased by a trimmed DLL as a function of V_{DD}

It can be seen that while in all the other corners the DLL is able to compensate the effect of V_{DD} very well, stabilizing the f_0 , the situation is different for SS corners, which are much more sensitive. When V_{DD} skews high, the f_0 decreases. In these slow MOS corners, the g_m of the transistors is low and/or their V_{th} is high. When V_{DD} is high, output capacitance of the CCDLU stage have to charge up to a higher

voltage in order to trigger the following buffer stage, which occurs more slowly. A possible solution would be to counteract this by increasing the W/L ratio of the transistors, but this would increase the parasitic capacitances and slow down the delay line, requiring more biasing current for a given oscillation frequency. As was discussed in subsection 5.5.1 and also a few paragraphs earlier, increasing the size of the transistors in the CCDLU also flattens the $f_0 = f(I_{\text{BIAS}})$ function of the slow corners, which would exacerbate the issue seen in the corner group SSa. None of these problems were seen when minimum sized transistors were used in a test-version of the CCDLU, but due to *DNL* specifications, this is not a viable option. In this processing technology, it appears that possibilities for further improvement of this particular architecture are very limited. A completely different way to approach the ILRO biasing scheme might solve the issue, though, and it will be briefly discussed in section 5.8.

That being said, all corners but the SSb group are able to lock to 625 MHz correctly anyway. As was discussed previously, this would not affect the yield enough to be a serious issue requiring a complete redesign, as the probability of such variation is significantly lower than the inverse of the yield of the manufacturing process itself (cca 90%). It is however something that would have to be kept in mind in future designs.

With regards to the propagation delay distribution across the 16 ILRO stages, the delays range from 44.4 ps to 103.3 ps for the stages adjacent to the injection point and from 48.2 ps to 52.1 ps for the remaining stages. Only 2 out of 33 corners exceed 75 ps though, and these outlier corners are the slowest ones from corner group SSb mentioned above. These results are expected and apart from the two overly slow f_0 corners, the propagation delay is within bounds.

The final simulation which will be discussed in this thesis is a transient noise simulation analysing the jitter of an unlocked and locked ILRO. To be specific, transient noise modelling was enabled and the clock jitter was measured at the buffered output of the last stage, which performs the worst due to jitter accumulation (as discussed in subsection 4.4.3). The histogram of jitter is shown on Figure 5.44. The standard deviation of unlocked ILRO jitter is 1.94 ps, while the standard deviation of a locked ILRO jitter is 0.82 ps. This is basically an alternative way of looking at the spectral purity of the oscillator, proving the theory discussed in subsection 3.2.3 and the validity of the MATLAB model output shown on Figure 4.15. In other words, injection locking significantly improves the phase noise of an oscillator and reduces the clock jitter by half on average.

Further simulations would need to be run post layout on schematics including extracted parasitics, possibly with Spectre RF or equivalent high-frequency oriented simulators in order to analyse the phase noise performance of the ILROs properly.

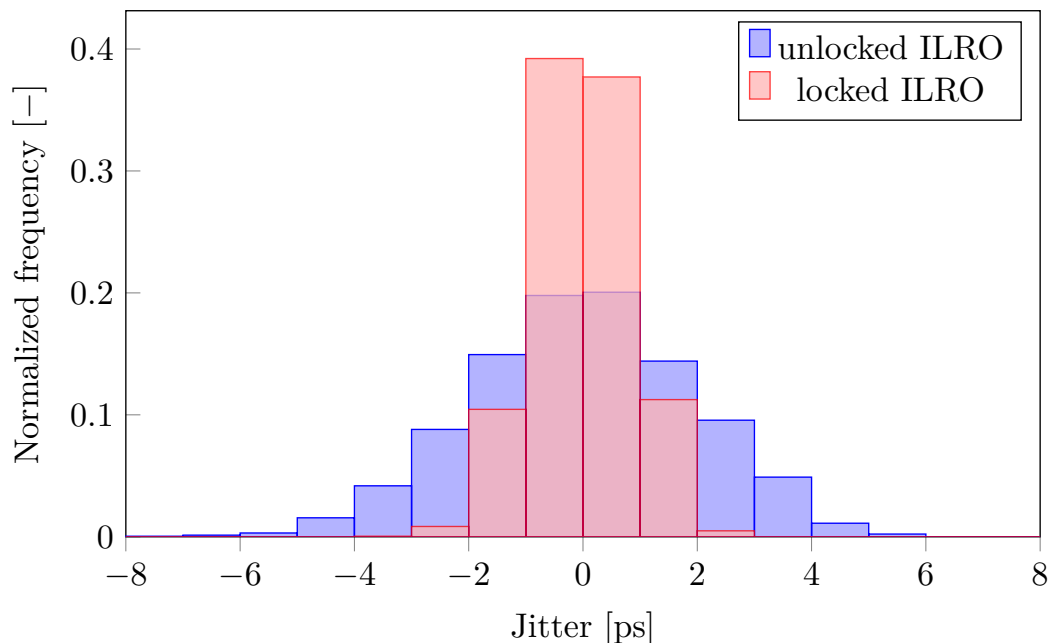


Fig. 5.44: Jitter histogram of a locked and unlocked ILRO

Because of the speed of the designed circuits and the required small time resolution, the circuits will be very sensitive to parasitics and optimizations to the design post-layout are to be expected. For the purposes this thesis, however, layout and post-layout simulations are not a part of the assignment.

5.8 Alternative solutions

During the design, a few ideas about alternative solutions to the encountered problems were discovered and they will be listed in this short section.

One of the problems of the designed system which was mentioned numerous times is the inherent problem in DLL based ILRO biasing, which is caused by the discrepancy between the connection of a delay line as a pass-through delay element and as a ring oscillator. This issue can contribute to the unsatisfactory DLL biasing performance seen in some PVT corners discussed in the previous section 5.7.

An obvious solution to this problem is PLL-based biasing, as shown on Figure 5.45, where all delay lines are connected as ring oscillators.

The PLL based approach has some shortcomings though. First of all, PLL design in general is much more difficult. This is primarily because PLLs are not conditionally stable unlike DLLs. While a DLL has a single pole at zero frequency due to the integration action at the charge pump output, a PLL actually integrates twice, because when a phase detector is utilized to control *Voltage Controlled Oscillator* (VCO) frequency, it means that its frequency is integrated (because phase is a time integral of frequency).

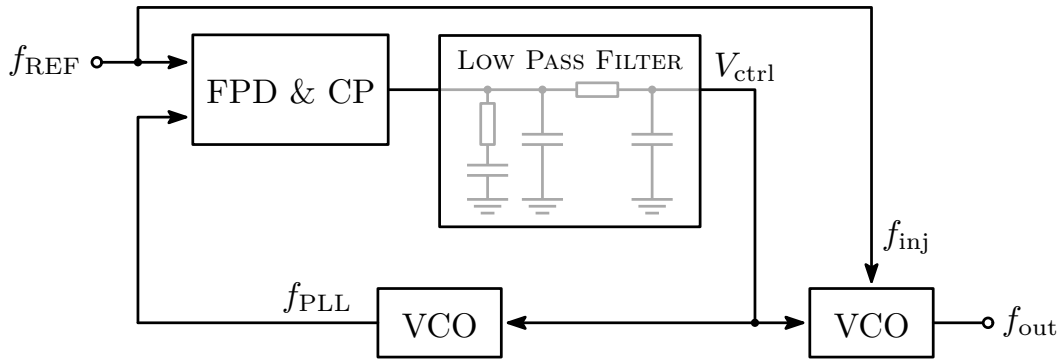


Fig. 5.45: PLL based ILRO biasing

This means that the whole stabilization scheme has to be much more robust, commonly requiring second or third order passive filters. The issue is not only the design complexity, but also the area: the sizes of capacitances required by PLL compensation circuits are significantly higher (tenfold or even more). Because of the need to not spend area needlessly, even compensated PLL settling waveforms often overshoot and ring for a comparatively long time, lengthening settling time and prolonging start-up.

On the other hand, it is expected that the PLL will bias ILROs more accurately over PVT conditions, as the connection of a ring oscillator within the PLL and within the ILRO blocks is absolutely identical. This would likely help to prevent the issues seen in some corners which were discussed in the previous section 5.7. Using a PLL to bias ILRO is a legitimate technique, but as always, its benefits and disadvantages need to be weighed carefully with regards to the particular project specifications.

The second area of design where an alternative approach could have been taken is the interface between the charge pump and the V-to-I converter. As was discussed previously, due to limited output impedance of the charge pump (caused by voltage headroom issues in the 1.2 V supply domain), it is important to keep V_{CPO} close to the middle of the supply rail to keep both charge pump current sources well within the saturation region. In the design, this was accounted for by trimming. Trimming is, however, a costly operation which takes time during testing, requires measuring equipment etc, so if there are any alternatives, they should be explored.

Two alternative methods will be proposed. The first method is shown on Figure 5.46. This proposal takes advantage of the fact that the target V_{CPO} does not have to be very precise and that its value is roughly $V_{DD}/2$, which can be easily made on-chip. A 5-bit ADC can be designed to trim V_{CPO} dynamically across all operating conditions. The added complexity of an ADC can outweigh the costs of the trimming operation while also bringing the benefit of preventing V_{CPO} skewing away from target in some specific conditions.

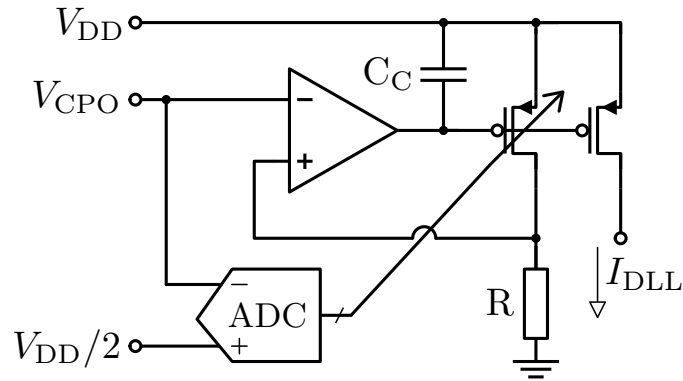


Fig. 5.46: Alternative dynamic ADC-based DLL trimming method

On the other hand, this is an additional feedback loop implemented inside another feedback loop and such a circuit needs to be approached carefully. The ADC does not have to be very precise or fast, but it has to be strictly monotonic. Furthermore, it should not compromise the signal integrity of the nearby circuits because of switching noise coupling etc.

The circuit could also be designed in such a way that the resistor is trimmed instead of the PMOS transistor. This is because when this dynamic trimming is employed, the *TCR* of the resistor is not important any more and this could have an added benefit of keeping the output voltage of the opamp stable across trim codes. It might not be as area efficient, though.

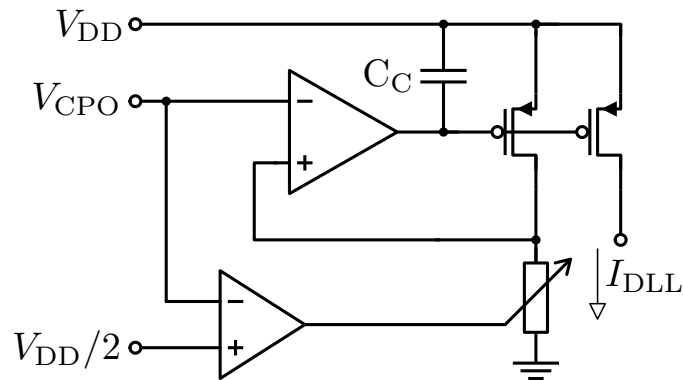


Fig. 5.47: Alternative dynamic VCR-based V_{CPO} stabilizing method

An alternative solution to the same problem is depicted on Figure 5.47. In this proposal, the resistor is replaced by a *Voltage Controlled Resistor* (VCR) (implemented with MOS transistors). The VCR is controlled by an opamp feedback loop in such a way to ensure that V_{CPO} is stabilized around $V_{DD}/2$. This is probably a more difficult solution to stabilize, as it is a fully analog feedback loop, but it could be the most area efficient method as well as simpler than designing an ADC.

SUMMARY

In chapter 1, general principles and components of DToF LIDAR systems were discussed. Scanning and flash LIDAR technologies were compared based on performance or commercial viability, laser sources and the constraints put on them by atmospheric effects or eye safety levels were discussed. A large part of the chapter was dedicated to SPADs, as these devices enable LIDAR technology in the first place. Their implementation, operation and non-idealities were discussed in sufficient detail necessary to understand the key performance characteristics and trade offs of LIDAR systems. The chapter closed by discussing advanced DSP techniques used for retrieving the ToF of photons with picosecond level of precision even in noisy daylight environments.

The chapter 2 was dedicated to TDCs, the key components of DToF LIDAR systems, as the actual ToF measurement is performed by these components. Common techniques optimizing TDC implementations such as the reverse timing scheme or sliding scale technique were discussed. Then, various types of TDC implementations were presented and compared, starting from counter based CTDCs, propagation delay based FTDCs and even sub-gate delay based SFTDCs. Since many FTDCs and SFTDCs implementations rely on DLLs, several pages were dedicated to their operation as well. At the end of the chapter, clock distribution schemes for LIDAR system TDC arrays were compared, highlighting the issue that many schemes trade off clock signal integrity and uniformity for power efficiency or vice versa. It was foreshadowed that ILOs can be used to optimize this trade off.

In chapter 3, ILOs were discussed and explained. LC tank based ILO model was used, enabling phasor analysis to achieve intuitive understanding of the injection locking phenomenon. Afterwards, a time-domain based mathematical model of an ILRO was presented and the equation for its locking range was derived. This simple mathematical model was, however, unable to predict the behaviour of ILROs when important non-idealities such as the injection clock duty cycle variation were considered. Finally, ILO based TDC architecture was presented, optimizing the performance trade offs discussed in chapter 2.

In order to understand the behaviour of ILROs better, a MATLAB Simulink macro model of an ILRO was created in chapter 4. Various types of analyses were performed using the model, confirming the findings derived in chapter 3 and expanding on the previous knowledge with new insights such the exploration of the sensitivity of the ILRO to the injection clock duty cycle variation. It was found that asymmetric injection clock leads to sub-optimal locking range span, but this sensitivity can be diminished if the injection pulses are wide enough to cover the so-called “sensitive window” of the ILRO. This, however, increases the total energy

spent by the injections, which is an important consideration for DToF LIDAR systems which contain hundreds or thousands of TDCs. Therefore, one of the design trade offs was identified before the start of the design phase.

In the final chapter 5, a DLL-biased ILRO capable of achieving 50 ps time resolution has been designed and simulated. First, the processing technology ONK65 was introduced and a key design decision on the ILRO operating frequency and the number of stages was explained. Afterwards, several subsections were dedicated to the discussion of overall architecture of an ILRO based LIDAR TDC signal chain, because without the understanding of architecture, educated decisions with regards to design trade-offs are not possible. The design started with the ILRO itself, focusing on the optimization of the CCDLUs, the biasing mirror or the injection circuit. The second phase of design focused on the DLL blocks such as the FPD or the V-to-I converter. A detailed subsection was dedicated to trimming, which is necessary in order to ensure proper operation of the circuits in all operating conditions.

The designed circuits were verified by various types of simulation. Apart from a single PVT corner, the designed ILROs are able to lock to the injected frequency properly. The problematic PVT corner, while statistically very unlikely and having minimal negative impact on the manufacturing yield, was analysed, and possible solutions of the problem were discussed along with their drawbacks. The time resolution offered by the ILROs has been simulated to be $50 \text{ ps} \pm 50\%$ (at 3σ), which meets the assigned specification. A transient noise simulation confirmed the expected improvement in spectral purity of the generated clock phases, which is another key benefit of the ILRO based TDCs. In the final section, some alternative techniques discovered during the design and potentially leading to better performance were briefly presented.

BIBLIOGRAPHY

- [1] JEYACHANDRAN, Satish. *Introducing the 5th-generation Waymo Driver: Informed by experience, designed for scale, engineered to tackle more environments* [online]. 2020 [cited 20-9-2020]. Available at: <<https://blog.waymo.com/2020/03/introducing-5th-generation-waymo-driver.html>>
- [2] NICLASS, Cristiano, Mineki SOGA, Hiroyuki MATSUBARA et al. A 0.18 μm CMOS SoC for a 100-m-Range 10-Frame/s 200 \times 96-Pixel Time-of-Flight Depth Sensor. *IEEE Journal of Solid-State Circuits* [online]. IEEE, 2014, 49(1), 315-330 [cit. 2020-09-29]. ISSN 0018-9200. DOI:10.1109/JSSC.2013.2284352
- [3] ROYO, Santiago and Maria BALLESTA-GARCIA. An Overview of Lidar Imaging Systems for Autonomous Vehicles. *Applied sciences* [online]. MDPI, 2019, 9(19), 4093 [cit. 2020-09-30]. DOI:10.3390/app9194093
- [4] ITO, Kota, Cristiano NICLASS, Isao AOYAGI et al. System Design and Performance Characterization of a MEMS-Based Laser Scanning Time-of-Flight Sensor Based on a 256 \times 64-pixel Single-Photon Imager. *IEEE Photonics Journal* [online]. IEEE, 2013, 5(2), 6800114-6800114 [cit. 2020-10-02]. Dostupné z: doi:10.1109/JPHOT.2013.2247586
- [5] Innoviz Technologies. [online datasheet]. *Innoviz Pro*. Innoviz Technologies: © 2020 [cit. 2020-10-3]. Available at: <<https://innoviz.tech/innovizpro>>
- [6] YOO, Han, Norbert DRUML, David BRUNNER et al. MEMS-based lidar for autonomous driving. *Elektrotechnik und Informationstechnik* [online]. Vienna: Springer Vienna, 2018, 135(6), 408-415 [cit. 2020-10-02]. ISSN 0932-383X. DOI:10.1007/s00502-018-0635-2
- [7] Quanergy Systems, Inc. Solid State LiDAR Sensors: The Future of Autonomous Vehicles. [conference presentation]. Burlingame, California: I&T International Symposium on Electronic Imaging 2019, Autonomous Vehicles and Machines, 2019-01-16. In: *imaging.org* [online]. Quanergy Systems, Inc.: © 2019. [cit. 2020-10-3]. Available at: <https://www.imaging.org/Site/PDFS/Conferences/ElectronicImaging/EI2019/AVMKeynotes/LouayEldada_Quanergy_AVM%202019.pdf>
- [8] Ibeo Automotive Systems GmbH. ibeoNEXT Generic Solid State LiDAR [online product page]. Ibeo Automotive Systems GmbH: © 2020 [cit 2020-10-03]. Available at: <<https://www.ibeo-as.com/en/products/sensoren/ibeoNEXTgeneric>>

- [9] LINDNER, Scott, Chao ZHANG, Ivan Michel ANTOLOVIC et al. A 252×144 SPAD Pixel Flash Lidar with 1728 Dual-Clock 48.8 ps TDCs, Integrated Histogramming and 14.9-to-1 Compression in 180 nm CMOS Technology. In: *2018 IEEE Symposium on VLSI Circuits* [online]. IEEE, 2018, s. 69-70 [cit. 2020-10-03]. ISBN 9781538667002. DOI:10.1109/VLSIC.2018.8502386
- [10] National Aeronautics and Space Administration. Web-based input form for ATRAN. [online web tool]. National Aeronautics and Space Administration: © 2020. [cit. 2020-10-04]. Available at: <<https://atran.arc.nasa.gov/cgi-bin/atran/atran.cgi>>
- [11] CHEN, Chih-Yuan. *A Sub-Centimeter Ranging Precision LIDAR Sensor Prototype Based on ILO-TDC*. [online]. Texas, 2016-06-28. Master's thesis. Texas A&M University, Department of Electrical and Computer Engineering. Available at: <<http://hdl.handle.net/1969.1/157844>>
- [12] HULSTROM, Roland, Richard BIRD and Carol RIORDAN. Spectral solar irradiance data sets for selected terrestrial conditions. *Solar cells* [online]. Elsevier B.V, 1985, 15(4), 365-391 [cit. 2020-09-30]. ISSN 0379-6787. DOI:10.1016/0379-6787(85)90052-3
- [13] Denton Vacuum, LLC. *Comparing Edge-Emitting Lasers and VCSELs*. [online article]. 2019-07-23. Denton Vacuum, LLC: © 2020. [cit. 2020-10-04]. Available at: <<https://www.dentonvacuum.com/comparing-edge-emitting-lasers-and-vcseles/>>
- [14] CHIH-CHIANG, Shen, Huai-yung WANG, Lu YUN-TING et al. 850/940-nm VCSEL for optical communication and 3D sensing. *Opto-Electronic Advances* [online]. Chengdu: Editorial Office of Opto-Electronic Advances, 2018, 1(3), 180005 [cit. 2020-10-04]. ISSN 20964579. DOI:10.29026/oea.2018.180005
- [15] JOHNSON, T. Matthew, Dominic F. SIRIANI, Kent D. CHOQUETTE et al. High-Speed Beam Steering With Phased Vertical Cavity Laser Arrays. *IEEE Journal of Selected Topics in Quantum Electronics* [online]. IEEE, 2013, 19(4), 1701006-1701006 [cit. 2020-10-04]. ISSN 1077-260X. DOI:10.1109/JSTQE.2013.2244574
- [16] D'ASARO, L. Arthur, Jean-francois SEURIN, James WYNN et al. High-Power, High-Efficiency VCSELs Pursue the Goal. *Photonics Spectra* [online]. 2005, 39(2), 62 [cit. 2020-10-04]. ISSN 0731-1230. Available at: <<http://search.proquest.com/docview/29124163/>>

- [17] RHIM, Jinsoo, Xiaoge ZENG, Zhihong HUANG et al. Monolithically-Integrated Single-Photon Avalanche Diode in a Zero-Change Standard CMOS Process for Low-Cost and Low-Voltage LiDAR Application. *Instruments* [online]. MDPI, 2019, 3(2), 33 [cit. 2020-09-26]. DOI: 10.3390/instruments3020033
- [18] NOLET, Frédéric, Samuel PARENT, Nicolas ROY et al. Quenching Circuit and SPAD Integrated in CMOS 65 nm with 7.8 ps FWHM Single Photon Timing Resolution. *Instruments* [online]. MDPI, 2018, 2(4), 19 [cit. 2020-10-04]. ISSN Instruments. DOI:10.3390/instruments2040019
- [19] CHARBON, Edoardo, Matt FISHBURN, Richard WALKER et al. SPAD-Based Sensors. In: REMONDINO, Fabio, David STOPPA. TOF Range-Imaging Cameras. Berlin, Heidelberg: Springer, 2013. 11-38. ISBN 978-3-642-27523-4. DOI:10.1007/978-3-642-27523-42
- [20] ZHANG, Chao. *CMOS SPAD Sensors for 3D Time-of-Flight Imaging, LiDAR and Ultra-High Speed Cameras* [online]. Delft, 2019-05-13. Doctoral thesis. Delft University of Technology. [cit. 2020-10-04]. DOI: 10.4233/uuid:f2e8ac06-33c0-423e-9617-6eaa87f7abd8
- [21] MITA, Rosario, Gaetano PALUMBO and Giorgio FALLICA. A fast active quenching and recharging circuit for single-photon avalanche diodes. In: *Proceedings of the 2005 European Conference on Circuit Theory and Design, 2005* [online]. IEEE, 2005, III/385-III/388 vol. 3 [cit. 2020-09-23]. ISBN 0780390660. DOI:10.1109/ECCTD.2005.1523141
- [22] NICLASS, Cristiano and Mineki SOGA. A miniature actively recharged single-photon detector free of afterpulsing effects with 6ns dead time in a 0.18 μ m CMOS technology. In: *2010 International Electron Devices Meeting* [online]. IEEE, 2010, 14.3.1-14.3.4 [cit. 2020-09-23]. ISBN 9781442474185. ISSN 01631918. DOI: 10.1109/IEDM.2010.5703360
- [23] VORNICU, Ion, Ricardo CARMONA-GALÁN, Belén PÉREZ-VERDÚ et al. Compact CMOS active quenching/recharge circuit for SPAD arrays. In: *International Journal of Circuit Theory and Applications* [online]. 2016, Vol. 44(4), 917-928 [cit. 2020-09-23]. ISSN 0098-9886. DOI: 10.1002/cta.2113
- [24] PIATEK, Slawomir S. Silicon Photomultipliers: theory & practice [conference presentation]. San Francisco: SPIE Photonics West, 2017-01-11. In: *YouTube* [online]. [cit. 2020-09-24]. Recording available at: <<https://www.youtube.com/watch?v=BH768QRfzFA>>

- [25] LINDNER, Scott, Sara PELLEGRINI, Yann HENRION et al. A High-PDE, Backside-Illuminated SPAD in 65/40-nm 3D IC CMOS Pixel With Cascoded Passive Quenching and Active Recharge. In: *IEEE Electron Device Letters* [online]. IEEE, 2017, 38(11), 1547-1550 [cit. 2020-09-24]. ISSN 0741-3106. DOI: 10.1109/LED.2017.2755989
- [26] ACERBI, Fabio, Giovanni PATERNOSTER, Alberto GOLLA et al. Silicon photomultipliers and single-photon avalanche diodes with enhanced NIR detection efficiency at FBK. In: *Nuclear instruments & methods in physics research. Section A, Accelerators, spectrometers, detectors and associated equipment* [online]. Elsevier B.V, 2018, 912, 309-314 [cit. 2020-09-26]. ISSN 0168-9002. DOI: 10.1016/j.nima.2017.11.098
- [27] OTTE, Adam Nepomuk, Distefano GARCIA, Thanh NGUYEN et al. Characterization of three high efficiency and blue sensitive silicon photomultipliers. In: *Nuclear instruments & methods in physics research. Section A, Accelerators, spectrometers, detectors and associated equipment* [online]. Elsevier B.V, 2017, 846(C), 106-125 [cit. 2020-09-26]. ISSN 0168-9002. DOI: 10.1016/j.nima.2016.09.053
- [28] GREEN, Martin A. and Mark J. KEEVERS. Optical properties of intrinsic silicon at 300 K. In: *Progress in Photovoltaics: Research and Applications* [online]. New York: Wiley Subscription Services, Inc., A Wiley Company, 1995, 3(3), 189-192 [cit. 2020-09-26]. ISSN 1062-7995. DOI: 10.1002/pip.4670030303
- [29] Hamamatsu Photonics K.K. [online datasheet]. *S13361-2050 series*. Hamamatsu Photonics K.K.: © 2018. [cit. 2020-10-04]. Available at: <https://www.hamamatsu.com/resources/pdf/ssd/s13361-2050_series_kapd1055e.pdf>
- [30] CORSI, Francesco, Cristoforo MARZOCCA, Angelo DRAGONE et al. Electrical Characterization of Silicon Photo-Multiplier Detectors for Optimal Front-End Design. In: *2006 IEEE Nuclear Science Symposium Conference Record* [online]. IEEE, 2006, s. 1276-1280 [cit. 2020-10-04]. ISBN 1424405602. ISSN 10957863. DOI:10.1109/NSSMIC.2006.356076
- [31] SensL. SiPM and SPAD Arrays for Next Generation LiDAR. [conference presentation]. Les Diablerets, Switzerland: 2018 International SPAD Sensor Workshop, 2018-01-26. In: *imagesensors.org* [online]. SensL: © 2018. [cit. 2020-10-4]. Available at: <http://imagesensors.org/wp-content/uploads/2018/05/Salvatore_Gnecchi.pdf>

- [32] HUTCHINGS, Sam W., Nick JOHNSTON, Istvan GYONGY et al. A Reconfigurable 3-D-Stacked SPAD Imager With In-Pixel Histogramming for Flash LIDAR or High-Speed Time-of-Flight Imaging. *IEEE Journal of Solid-State Circuits* [online]. IEEE, 2019, 54(11), 2947-2956 [cit. 2020-10-04]. ISSN 0018-9200. DOI:10.1109/JSSC.2019.2939083
- [33] PERENZONI, Matteo, Daniele PERENZONI and David STOPPA. A 64×64 Pixels Digital Silicon Photomultiplier Direct TOF Sensor With 100-MPhotons/s/pixel Background Rejection and Imaging/Altimeter Mode With 0.14% Precision Up To 6 km for Spacecraft Navigation and Landing. *IEEE Journal of Solid-State Circuits* [online]. IEEE, 2017, 52(1), 151-160 [cit. 2020-09-29]. ISSN 0018-9200. DOI:10.1109/JSSC.2016.2623635
- [34] NICLASS, Cristiano, Mineki SOGA, Hiroyuki MATSUBARA et al. A 100 m Range 10-Frame/s 340×96 -Pixel Time-of-Flight Depth Sensor in $0.18 \mu\text{m}$ CMOS. *IEEE Journal of Solid-State Circuits* [online]. IEEE, 2013, 48(2), 559-572 [cit. 2020-09-29]. ISSN 0018-9200. DOI:10.1109/JSSC.2012.2227607
- [35] BEER, Maik, Charles THATTIL, Jan F. HAASE et al. 2×192 Pixel CMOS SPAD-Based Flash LiDAR Sensor with Adjustable Background Rejection. In: *2018 25th IEEE International Conference on Electronics, Circuits and Systems (ICECS)* [online]. IEEE, 2018, s. 17-20 [cit. 2020-10-30]. DOI:10.1109/ICECS.2018.8617905
- [36] MORRISON, Daniel, Simon KENNEDY, Dennis DELIC et al. A 64×64 SPAD Flash LIDAR Sensor using a Triple Integration Timing Technique with 1.95 mm Depth Resolution. *IEEE Sensors Journal* [online]. IEEE, 2020, , 1-1 [cit. 2020-10-30]. ISSN 1530-437X. DOI:10.1109/JSEN.2020.3030788
- [37] COTTINI, Carlo, Emilio GATTI and Vito SVELTO. A new method for analog to digital conversion. In: *Nuclear Instruments and Methods*. 1963, vol. 24, 241-242. ISSN 0029-554X. DOI:10.1016/0029-554X(63)90314-8
- [38] YANG, Chih-Kong Ken. Delay Locked Loops - An Overview. In: RAZAVI, Behzad. *Phase-Locking in High-Performance Systems: From Devices to Architectures*. Hoboken: IEEE Press, John Wiley & Sons, 2003. 13-22. ISBN 0-471-44727-7. DOI:10.1109/9780470545492.ch2
- [39] JIA, Cheng. *A Delay-Locked Loop for Multiple Clock Phases/Delays Generation* [online]. Georgia, 2005-08-24. Doctoral thesis. Georgia Institute of Technology, School of Electrical and Computer Engineering. [cit. 2020-10-10]. URI: <<http://hdl.handle.net/1853/7470>>

- [40] MARKOVIC, Bojan, Simone TISA, Federica A. VILLA et al. A High-Linearity, 17 ps Precision Time-to-Digital Converter Based on a Single-Stage Vernier Delay Loop Fine Interpolation. *IEEE Transactions on Circuits and Systems I: Regular Papers* [online]. IEEE, 2013, 60(3), 557-569 [cit. 2020-10-10]. ISSN 1549-8328. DOI:10.1109/TCSI.2012.2215737
- [41] CHEN, Chih-yuan, Cheng LI, Marco FIORENTINO et al. A 52 ps resolution ILO-based time-to-digital converter array for LIDAR sensors. In: *2016 IEEE Dallas Circuits and Systems Conference (DCAS)* [online]. IEEE, 2016, s. 1-4 [cit. 2020-10-10]. DOI:10.1109/DCAS.2016.7791148
- [42] MÄNTYNIEMI, Antti, Timo RAHKONEN a Juha KOSTAMOVARA. A CMOS Time-to-Digital Converter (TDC) Based On a Cyclic Time Domain Successive Approximation Interpolation Method. *IEEE Journal of Solid-State Circuits* [online]. IEEE, 2009, 44(11), 3067-3078 [cit. 2020-10-11]. ISSN 0018-9200. DOI:10.1109/JSSC.2009.2032260
- [43] DUDEK, Piotr, Stanislaw SZCZEPANSKI a John V. HATFIELD. A high-resolution CMOS time-to-digital converter utilizing a Vernier delay line. *IEEE Journal of Solid-State Circuits* [online]. IEEE, 2000, 35(2), 240-247 [cit. 2020-10-11]. ISSN 0018-9200. DOI:10.1109/4.823449
- [44] PARK, Youngmin and David D. WENTZLOFF. A Cyclic Vernier TDC for ADPLLs Synthesized From a Standard Cell Library. *IEEE Transactions on Circuits and Systems I: Regular Papers* [online]. IEEE, 2011, 58(7), 1511-1517 [cit. 2020-10-11]. ISSN 1549-8328. DOI:10.1109/TCSI.2011.2158490
- [45] CHEN, Poki, Chun-chi CHEN, Jia-chi ZHENG and You-sheng SHEN. A PVT Insensitive Vernier-Based Time-to-Digital Converter With Extended Input Range and High Accuracy. *IEEE Transactions on Nuclear Science* [online]. IEEE, 2007, 54(2), 294-302 [cit. 2020-10-11]. ISSN 0018-9499. DOI:10.1109/TNS.2007.892944
- [46] RAYMOND, Mina, Maged GHONEIMA and Yeha ISMAIL. A programmable multi-step cyclic Vernier time-to-digital converter. *International Journal of Circuits and Architecture Design* [online]. Inderscience, 2013, 1(1), 41-61 [cit. 2020-10-11]. DOI: 10.1504/IJCAD.2013.057454
- [47] PARK, Young Jun and Fei YUAN. Two-step pulse-shrinking time-to-digital converter. *Microelectronics Journal* [online]. Elsevier, 2017, 60, 45-54 [cit. 2020-10-11]. ISSN 0026-2692. DOI:10.1016/j.mejo.2016.11.015

- [48] SEO, Young-hun, Jun-seok KIM, Hong-june PARK and Jae-yoon SIM. A 0.63 ps resolution, 11b pipeline TDC in 0.13 μm CMOS. In: *2011 Symposium on VLSI Circuits - Digest of Technical Papers* [online]. IEEE, 2011, s. 152-153 [cit. 2020-10-11]. ISBN 9781612841755.
- [49] JEE, Dong-woo, Young-hun SEO, Hong-june PARK a Jae-yoon SIM. A 2 GHz Fractional-N Digital PLL with 1b Noise Shaping $\Delta\Sigma$ TDC. *IEEE Journal of Solid-State Circuits* [online]. IEEE, 2012, 47(4), 875-883 [cit. 2020-10-11]. ISSN 0018-9200. DOI:10.1109/JSSC.2012.2185190
- [50] FRIEDLAND, Bernard. Introduction to “Stabilized Feed-Back Amplifiers”. *Proceedings of the IEEE* [online]. 1999, 87(2), 376-378 [cit. 2020-10-17]. ISSN 0018-9219. Available at: <<http://search.proquest.com/docview/27072128/>>
- [51] BALAZ, Igor, Zdenko BREZOVIC, Marian MINARIK et al. Barkhausen criterion and another necessary condition for steady state oscillations existence. In: *2013 23rd International Conference Radioelektronika (RADIOELEKTRONIKA)* [online]. IEEE, 2013, s. 151-155 [cit. 2020-10-17]. ISBN 9781467355162. DOI:10.1109/RadioElek.2013.6530906
- [52] ADLER, Robert. A Study of Locking Phenomena in Oscillators. *Proceedings of the IRE* [online]. IEEE, 1946, 34(6), 351-357 [cit. 2020-10-17]. ISSN 0096-8390. DOI:10.1109/JRPROC.1946.229930
- [53] RAZAVI, Behzad. A study of injection locking and pulling in oscillators. *IEEE Journal of Solid-State Circuits* [online]. IEEE, 2004, 39(9), 1415-1424 [cit. 2020-10-17]. ISSN 0018-9200. DOI:10.1109/JSSC.2004.831608
- [54] PACIOREK, L.J. Injection locking of oscillators. *Proceedings of the IEEE* [online]. IEEE, 1965, 53(11), 1723-1727 [cit. 2020-10-17]. ISSN 0018-9219. DOI:10.1109/PROC.1965.4345
- [55] TOSO, Stefano Dal. *Analysis and Design of Injection-Locked Building Blocks for RF Frequency Generation in Ultra-Scaled CMOS Technologies* [online]. Padua, 2010-07-10. Doctoral thesis. University of Padua, Department of Information Engineering. [cit. 2020-10-17]. Available at: <http://paduaresearch.cab.unipd.it/3151/1/tesi_completa.pdf>
- [56] LI, Mingze. *A Fully Synthesized Injection Locked Ring Oscillator Based on a Pulse Injection Locking Technique* [online]. Ontario, Ottawa, 2010-07-10. Master’s thesis. Carleton University, Department of Electronics. [cit. 2020-10-17]. DOI:10.22215/etd/2017-12225

- [57] PERROTT, Michael H. *High Speed Communication Circuits and Systems, Lecture 14: Voltage Controlled Oscillators* [lecture presentation]. Massachusetts Institute of Technology, 2005-03-29. Michael H. Perrot: © 2005. [cit. 2020-10-17]. Available at: <<https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-776-high-speed-communication-circuits-spring-2005/lecture-notes/lec14.pdf>>
- [58] ALI, Iqbal, B. N BISWAS a Sudhabindu RAY. Improved Closed Form Large Injection Perturbation Analytical Model on the Output Spectrum of Unlocked Driven Oscillator-Part I: Phase Perturbation. *IEEE Transactions on Circuits and Systems I: Regular Papers* [online]. IEEE, 2014, 61(1), 106-119 [cit. 2020-10-18]. ISSN 1549-8328. DOI:10.1109/TCSI.2013.2268196
- [59] CHIEN, Jun-chau and Liang-hung LU. Analysis and Design of Wideband Injection-Locked Ring Oscillators With Multiple-Input Injection. *IEEE Journal of Solid-State Circuits* [online]. IEEE, 2007, 42(9), 1906-1915 [cit. 2020-10-25]. ISSN 0018-9200. DOI:10.1109/JSSC.2007.903058
- [60] GANGASANI, Gautam Reddy and Peter R. KINGET. Time-domain model for injection locking in nonharmonic oscillators. *IEEE Transactions on Circuits and Systems I: Regular Papers* [online]. IEEE, 2008, 55(6), 1648-1658 [cit. 2020-10-25]. ISSN 1549-8328. DOI:10.1109/TCSI.2008.916605
- [61] BAE, Woorham. CMOS Inverter as Analog Circuit: An Overview. *Journal of low power electronics and applications* [online]. MDPI, 2019, 9(3), 26 [cit. 2020-11-22]. DOI:10.3390/jlpea9030026

SYMBOLS AND ABBREVIATIONS

ADAS	Advanced Driver Assistance System
ADC	Analog to Digital Converter
APD	Avalanche Photodiode
BSI	Backside Illumination
CC	Constant Current
CCCS	Current Controlled Current Source
CCDL	Current Controlled Delay Line
CCDLU	Current Controlled Delay Line Unit
CMOS	Complementary Metal Oxide Semiconductor
CTDC	Coarse Time to Digital Converter
dSiPM	Digital Silicon Photomultiplier
DC	Direct Current
DFT	Discrete Fourier Transform
DLL	Delay Locked Loop
DLU	Delay Line Unit
DSP	Digital Signal Processing
DToF	Direct Time of Flight
EEL	Edge Emitting Laser
ESR	Equivalent Series Resistance
FIR	Finite Impulse Response
FPA	Focal Plane Array
FPD	Frequency Phase Detector
FPGA	Field Programmable Gate Array
FoV	Field of View

FTDC	Fine Time to Digital Converter
FWHM	Full Width at Half Maximum
GAPD	Geiger-mode Avalanche Photodiode
IC	Integrated Circuit
ILO	Injection Locked Oscillator
ILRO	Injection Locked Ring Oscillator
IR	Infrared
LIDAR	Light Detection and Ranging
LSB	Least Significant Bit
MEMS	Micro-electro-mechanical System
MOS	Metal Oxide Semiconductor
MPE	Maximum Permissible Exposure
MSB	Most Significant Bit
NIR	Near Infrared
OPA	Optical Phased Array
PDK	Process Design Kit
PLL	Phase Locked Loop
PQAR	Passive Quenching Active Recharge
PVT	Process Voltage Temperature
QVGA	Quarter Video Graphics Array
RF	Radio Frequency
RO	Ring Oscillator
SFTDC	Sub-Fine Time to Digital Converter
SiPM	Silicon Photomultiplier
SOA	Safe Operating Area

SPAD	Single Photon Avalanche Diode
SNR	Signal to Noise Ratio
TCSPC	Time Correlated Single Photon Counting
TB	Thermometric to Binary
TDC	Time to Digital Converter
TIA	Trans-impedance Amplifier
ToF	Time of Flight
VT	Voltage Temperature
VCO	Voltage Controlled Oscillator
VCR	Voltage Controlled Resistor
VCDLU	Voltage Controlled Delay Line Unit
VCSEL	Vertical Cavity Surface Emitting Laser
V-to-I	Voltage-to-Current
α	absorption coefficient
δ	symbol for relative change
δf_{LR}	locking range (relative to f_0)
$\delta\omega_{LR}$	locking range (relative to ω_0)
λ	wavelength
ϕ	RLC tank phase shift; generic symbol for angle
ρ	reflection constant
τ	RC circuit time constant
θ	incident angle
ϑ	injection angle
ω	angular frequency

ω_0	free-running angular frequency
ω_{inj}	injection angular frequency
Δ	symbol for absolute change; injection waveform time shift
$\Delta\omega_{LR}$	locking range (in radians)
Δf_{LR}	locking range (in Hz)
c	speed of light
d	zero crossing delay due to injection
f	frequency
f_0	free-running frequency
f_{inj}	injection frequency
$f_{inj,max}$	upper locking range limit
$f_{inj,min}$	lower locking range limit
g_m	transconductance
k	generic symbol for integer; permittivity
t	time
t_d	time delay, propagation delay
A_d	detector optical receiving area
AP	avalanche probability
B	number of bits
C	capacitance
C_C	compensation capacitance
DC	duty cycle

DCR	dark count rate
DNL	differential non-linearity
FF	fill factor
GBW	gain bandwidth product
$E_{e,\lambda}$	spectral irradiance in wavelength
$H(s)$	system transfer function (in Laplace domain)
I	electric current
I_{inj}	injection current
I_{osc}	oscillation current
I_Q	latching current
INL	integral non-linearity
K	injection ratio
\mathcal{L}	phase noise power
L	inductance; transistor channel length
N	number of ILRO stages; generic symbol for integer
M	generic symbol for integer
$MTBF$	mean time between failure
P_c	collected power
P_t	transmitted power
PDE	photon detection efficiency
PDP	photon detection probability
PSD	power spectral density
$PSRR$	power supply rejection ratio
Q	quality factor; charge
Q_{inj}	injected charge

QE	quantum efficiency
R	distance to the reflector; resistance; regression coefficient
RMS	Root Mean Square
T	period
T_0	free-running period
TCR	thermal coefficient of resistance
T_{inj}	injection period
TS	time stamp
$TS0$	time stamp zero
V	voltage
V_{th}	threshold voltage
V_{BD}	SPAD breakdown voltage
V_{CPO}	charge pump output voltage
V_{DD}	supply voltage
V_E	SPAD excess bias voltage
V_{GS}	gate-source voltage
W	transistor channel width

LIST OF APPENDICES

A	Mathematical derivations of ILRO time domain model	188
A.1	Zero cross delay derivation for $\Delta > 0$	188
A.2	Zero cross delay derivation for $\Delta < 0$	189
A.3	Maximum zero cross delay	190
A.4	Minimum zero cross delay	191
B	Eight stage ILRO MATLAB model	193
C	Digital appendix	194

A MATHEMATICAL DERIVATIONS OF ILRO TIME DOMAIN MODEL

A.1 Zero cross delay derivation for $\Delta > 0$

The goal is to find the analytical expression for d , which is defined as

$$d(\Delta) = t_{zc}\{v_{\text{sum}}(t), \Delta \neq 0\} - t_{zc}\{v_{\text{sum}}(t), \Delta = 0\} \quad (\text{A.1})$$

where

$$v_{\text{sum}}(t) = v_{\text{osc}}(t) + v_{\text{inj}}(t) = -V_{\text{osc,max}} + (V_{\text{osc,max}} + V_{\text{osc}}) \cdot \exp\left(-\frac{t}{\tau}\right) - V_{\text{inj,max}} + (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{-t + \Delta}{\tau}\right) \quad (\text{A.2})$$

We can start from

$$v_{\text{sum}}(t_{zc}) = 0 \quad (\text{A.3})$$

and the analytical expression for $t_{zc}(\Delta)$ can be derived as follows.

$$\begin{aligned} & -V_{\text{osc,max}} + (V_{\text{osc,max}} + V_{\text{osc}}) \cdot \exp\left(-\frac{t_{zc}}{\tau}\right) \\ & -V_{\text{inj,max}} + (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{-t_{zc} + \Delta}{\tau}\right) = 0 \end{aligned} \quad (\text{A.4})$$

$$\begin{aligned} \exp\left(-\frac{t_{zc}}{\tau}\right) \cdot \left[V_{\text{osc,max}} + V_{\text{osc}} + (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{\Delta}{\tau}\right) \right] \\ = V_{\text{osc,max}} + V_{\text{inj,max}} \end{aligned} \quad (\text{A.5})$$

$$\exp\left(-\frac{t_{zc}}{\tau}\right) = \frac{V_{\text{osc,max}} + V_{\text{inj,max}}}{V_{\text{osc,max}} + V_{\text{osc}} + (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{\Delta}{\tau}\right)} \quad (\text{A.6})$$

$$\ln\left[\exp\left(-\frac{t_{zc}}{\tau}\right)\right] = \ln\left[\frac{V_{\text{osc,max}} + V_{\text{inj,max}}}{V_{\text{osc,max}} + V_{\text{osc}} + (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{\Delta}{\tau}\right)}\right] \quad (\text{A.7})$$

$$-\frac{t_{zc}}{\tau} = \ln\left[\frac{V_{\text{osc,max}} + V_{\text{inj,max}}}{V_{\text{osc,max}} + V_{\text{osc}} + (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{\Delta}{\tau}\right)}\right] \quad (\text{A.8})$$

$$t_{zc}(\Delta) = -\tau \ln\left[\frac{V_{\text{osc,max}} + V_{\text{inj,max}}}{V_{\text{osc,max}} + V_{\text{osc}} + (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{\Delta}{\tau}\right)}\right] \quad (\text{A.9})$$

We can easily express $t_{zc}\{v_{\text{sum}}(t), \Delta = 0\}$ as

$$t_{zc}(\Delta = 0) = -\tau \ln\left(\frac{V_{\text{osc,max}} + V_{\text{inj,max}}}{V_{\text{osc,max}} + V_{\text{osc}} + V_{\text{inj,max}} + V_{\text{inj}}}\right) \quad (\text{A.10})$$

and subtracting it from Equation A.9 (per Equation A.1) finally yields

$$d(\Delta)|_{\Delta>0} = \tau \ln\left[\frac{V_{\text{osc,max}} + V_{\text{osc}} + (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{\Delta}{\tau}\right)}{V_{\text{osc,max}} + V_{\text{osc}} + V_{\text{inj,max}} + V_{\text{inj}}}\right] \quad (\text{A.11})$$

A.2 Zero cross delay derivation for $\Delta < 0$

This derivation is nearly identical to the one discussed in section A.1. The only difference is that instead of using

$$v_{\text{inj}}(t) = -V_{\text{inj,max}} + (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{-t + \Delta}{\tau}\right) \quad (\text{A.12})$$

we use

$$v_{\text{inj}}(t) = V_{\text{inj,max}} - (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{-t + \Delta}{\tau}\right) \quad (\text{A.13})$$

This is because for negative Δ , we are interested in the interaction of the *rising part* of the injection waveform with the falling part of the oscillation waveform. The derivation then continues as shown previously.

The goal is to find the analytical expression for d , which is defined as

$$d(\Delta) = t_{\text{zc}}\{v_{\text{sum}}(t), \Delta \neq 0\} - t_{\text{zc}}\{v_{\text{sum}}(t), \Delta = 0\} \quad (\text{A.14})$$

where

$$\begin{aligned} v_{\text{sum}}(t) = v_{\text{osc}}(t) + v_{\text{inj}}(t) = & -V_{\text{osc,max}} + (V_{\text{osc,max}} + V_{\text{osc}}) \cdot \exp\left(-\frac{t}{\tau}\right) \\ & + V_{\text{inj,max}} - (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{-t + \Delta}{\tau}\right) \end{aligned} \quad (\text{A.15})$$

We can start from

$$v_{\text{sum}}(t_{\text{zc}}) = 0 \quad (\text{A.16})$$

and the analytical expression for $t_{\text{zc}}(\Delta)$ can be derived as follows.

$$\begin{aligned} & -V_{\text{osc,max}} + (V_{\text{osc,max}} + V_{\text{osc}}) \cdot \exp\left(-\frac{t_{\text{zc}}}{\tau}\right) \\ & + V_{\text{inj,max}} - (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{-t_{\text{zc}} + \Delta}{\tau}\right) = 0 \end{aligned} \quad (\text{A.17})$$

$$\begin{aligned} \exp\left(-\frac{t_{\text{zc}}}{\tau}\right) \cdot \left[V_{\text{osc,max}} + V_{\text{osc}} - (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{\Delta}{\tau}\right) \right] \\ = V_{\text{osc,max}} - V_{\text{inj,max}} \end{aligned} \quad (\text{A.18})$$

$$\exp\left(-\frac{t_{\text{zc}}}{\tau}\right) = \frac{V_{\text{osc,max}} - V_{\text{inj,max}}}{V_{\text{osc,max}} + V_{\text{osc}} - (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{\Delta}{\tau}\right)} \quad (\text{A.19})$$

$$\ln\left[\exp\left(-\frac{t_{\text{zc}}}{\tau}\right)\right] = \ln\left[\frac{V_{\text{osc,max}} - V_{\text{inj,max}}}{V_{\text{osc,max}} + V_{\text{osc}} - (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{\Delta}{\tau}\right)}\right] \quad (\text{A.20})$$

$$-\frac{t_{\text{zc}}}{\tau} = \ln\left[\frac{V_{\text{osc,max}} - V_{\text{inj,max}}}{V_{\text{osc,max}} + V_{\text{osc}} - (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{\Delta}{\tau}\right)}\right] \quad (\text{A.21})$$

$$t_{\text{zc}}(\Delta) = -\tau \ln\left[\frac{V_{\text{osc,max}} - V_{\text{inj,max}}}{V_{\text{osc,max}} + V_{\text{osc}} - (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{\Delta}{\tau}\right)}\right] \quad (\text{A.22})$$

We can easily express $t_{zc}\{v_{\text{sum}}(t), \Delta = 0\}$ as

$$t_{zc}(\Delta = 0) = -\tau \ln \left(\frac{V_{\text{osc,max}} - V_{\text{inj,max}}}{V_{\text{osc,max}} + V_{\text{osc}} - V_{\text{inj,max}} - V_{\text{inj}}} \right) \quad (\text{A.23})$$

and subtracting it from Equation A.22 finally yields

$$d(\Delta)|_{\Delta < 0} = \tau \ln \left[\frac{V_{\text{osc,max}} + V_{\text{osc}} - (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{\Delta}{\tau}\right)}{V_{\text{osc,max}} + V_{\text{osc}} - V_{\text{inj,max}} - V_{\text{inj}}} \right] \quad (\text{A.24})$$

A.3 Maximum zero cross delay

As discussed in section 3.3, $v_{\text{osc}}(\Delta_{\text{max}}) = -V_{\text{inj}}$. Therefore

$$-V_{\text{osc,max}} + (V_{\text{osc,max}} + V_{\text{osc}}) \cdot \exp\left(-\frac{\Delta_{\text{max}}}{\tau}\right) = -V_{\text{inj}} \quad (\text{A.25})$$

$$\exp\left(-\frac{\Delta_{\text{max}}}{\tau}\right) = \frac{V_{\text{osc,max}} - V_{\text{inj}}}{V_{\text{osc,max}} + V_{\text{osc}}} \quad (\text{A.26})$$

$$\Delta_{\text{max}} = -\tau \ln \left(\frac{V_{\text{osc,max}} - V_{\text{inj}}}{V_{\text{osc,max}} + V_{\text{osc}}} \right) \quad (\text{A.27})$$

We can input this equation into Equation A.11 and start rearranging.

$$\begin{aligned} d_{\text{max}} &= \tau \ln \left[\frac{V_{\text{osc,max}} + V_{\text{osc}} + (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{\Delta_{\text{max}}}{\tau}\right)}{V_{\text{osc,max}} + V_{\text{osc}} + V_{\text{inj,max}} + V_{\text{inj}}} \right] \\ &= \tau \ln \left\{ \frac{V_{\text{osc,max}} + V_{\text{osc}} + (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left[\frac{-\tau}{\tau} \cdot \ln\left(\frac{V_{\text{osc,max}} - V_{\text{inj}}}{V_{\text{osc,max}} + V_{\text{osc}}}\right)\right]}{V_{\text{osc,max}} + V_{\text{osc}} + V_{\text{inj,max}} + V_{\text{inj}}} \right\} \\ &= \tau \ln \left[\frac{V_{\text{osc,max}} + V_{\text{osc}} + (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \frac{V_{\text{osc,max}} + V_{\text{osc}}}{V_{\text{osc,max}} - V_{\text{inj}}}}{V_{\text{osc,max}} + V_{\text{osc}} + V_{\text{inj,max}} + V_{\text{inj}}} \right] \end{aligned} \quad (\text{A.28})$$

After multiplying and further rearranging we obtain

$$d_{\text{max}} = \tau \ln \left(\frac{V_{\text{osc,max}}^2 + V_{\text{osc,max}}V_{\text{osc}} + V_{\text{osc,max}}V_{\text{inj,max}} + V_{\text{inj,max}}V_{\text{osc}}}{V_{\text{osc,max}}^2 + V_{\text{osc,max}}V_{\text{osc}} + V_{\text{osc,max}}V_{\text{inj,max}} - V_{\text{osc}}V_{\text{inj}} - V_{\text{inj}}V_{\text{inj,max}} - V_{\text{inj}}^2} \right) \quad (\text{A.29})$$

The expression for injection ratio from Equation 3.55 allows further rearranging of Equation A.29

$$d_{\text{max}} = \tau \ln \left(\frac{V_{\text{osc,max}}^2 + V_{\text{osc,max}}V_{\text{osc}} + KV_{\text{osc,max}}^2 + KV_{\text{osc}}^2}{V_{\text{osc,max}}^2 + V_{\text{osc,max}}V_{\text{osc}} + KV_{\text{osc,max}}^2 - KV_{\text{osc}}^2 - V_{\text{inj}}V_{\text{inj,max}} - V_{\text{inj}}^2} \right) \quad (\text{A.30})$$

This is the most simplified analytic expression for d_{\max} applicable to ILROs of any number of stages. It is however quite complicated and not very practical. Thankfully, for ILROs whose number of stages is greater or equal to 5, we can write $V_{\text{osc}} \approx V_{\text{osc,max}}$ and $V_{\text{inj}} \approx V_{\text{inj,max}}$ (as was proven in section 3.3), and these two equations can simplify the expression for d_{\max} even further.

$$\begin{aligned}
d_{\max} &= \tau \ln \left(\frac{V_{\text{osc}}^2 + V_{\text{osc}}^2 + KV_{\text{osc}}^2 + KV_{\text{osc}}^2}{V_{\text{osc}}^2 + V_{\text{osc}}^2 + KV_{\text{osc}}^2 - KV_{\text{osc}}^2 - V_{\text{inj}}^2 - V_{\text{inj}}^2} \right) \\
&= \tau \ln \left(\frac{2V_{\text{osc}}^2 + 2KV_{\text{osc}}^2}{2V_{\text{osc}}^2 - 2V_{\text{inj}}^2} \right) \\
&= \tau \ln \left(\frac{V_{\text{osc}}^2 + KV_{\text{osc}}^2}{V_{\text{osc}}^2 - V_{\text{inj}}^2} \right)
\end{aligned} \tag{A.31}$$

Since $V_{\text{inj}} = KV_{\text{osc}}$, we can continue in simplifying.

$$\begin{aligned}
d_{\max} &= \tau \ln \left[\frac{V_{\text{osc}}^2 (1 + K)}{V_{\text{osc}}^2 - (KV_{\text{osc}})^2} \right] \\
&= \tau \ln \left[\frac{V_{\text{osc}}^2 (1 + K)}{V_{\text{osc}}^2 (1 - K^2)} \right] \\
&= \tau \ln \left[\frac{1 + K}{(1 - K^2)} \right] \\
&= \tau \ln \left[\frac{1 + K}{(1 - K)(1 + K)} \right] \\
&= \tau \ln \left(\frac{1}{1 - K} \right)
\end{aligned} \tag{A.32}$$

A.4 Minimum zero cross delay

Analogous derivation to the one shown in section A.3 can be done for the minimum zero cross delay d_{\min} .

As discussed in section 3.3, $v_{\text{osc}}(\Delta_{\min}) = V_{\text{inj}}$. Therefore we can write

$$-V_{\text{osc,max}} + (V_{\text{osc,max}} + V_{\text{osc}}) \cdot \exp \left(-\frac{\Delta_{\min}}{\tau} \right) = V_{\text{inj}} \tag{A.33}$$

$$\exp \left(-\frac{\Delta_{\min}}{\tau} \right) = \frac{V_{\text{osc,max}} + V_{\text{inj}}}{V_{\text{osc,max}} + V_{\text{osc}}} \tag{A.34}$$

$$\Delta_{\min} = -\tau \ln \left(\frac{V_{\text{osc,max}} + V_{\text{inj}}}{V_{\text{osc,max}} + V_{\text{osc}}} \right) \tag{A.35}$$

We can input this equation into Equation A.24 and start rearranging.

$$\begin{aligned}
d_{\min} &= \tau \ln \left[\frac{V_{\text{osc,max}} + V_{\text{osc}} - (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left(\frac{\Delta_{\min}}{\tau}\right)}{V_{\text{osc,max}} + V_{\text{osc}} - V_{\text{inj,max}} - V_{\text{inj}}} \right] \\
&= \tau \ln \left\{ \frac{V_{\text{osc,max}} + V_{\text{osc}} - (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \exp\left[\frac{-\tau}{\tau} \cdot \ln\left(\frac{V_{\text{osc,max}} + V_{\text{inj}}}{V_{\text{osc,max}} + V_{\text{osc}}}\right)\right]}{V_{\text{osc,max}} + V_{\text{osc}} - V_{\text{inj,max}} - V_{\text{inj}}} \right\} \\
&= \tau \ln \left[\frac{V_{\text{osc,max}} + V_{\text{osc}} - (V_{\text{inj,max}} + V_{\text{inj}}) \cdot \frac{V_{\text{osc,max}} + V_{\text{osc}}}{V_{\text{osc,max}} + V_{\text{inj}}}}{V_{\text{osc,max}} + V_{\text{osc}} - V_{\text{inj,max}} - V_{\text{inj}}} \right]
\end{aligned} \tag{A.36}$$

After multiplying and further rearranging we obtain

$$d_{\min} = \tau \ln \left(\frac{V_{\text{osc,max}}^2 + V_{\text{osc,max}} V_{\text{osc}} - V_{\text{osc,max}} V_{\text{inj,max}} - V_{\text{inj,max}} V_{\text{osc}}}{V_{\text{osc,max}}^2 + V_{\text{osc,max}} V_{\text{osc}} - V_{\text{osc,max}} V_{\text{inj,max}} + V_{\text{osc}} V_{\text{inj}} - V_{\text{inj}} V_{\text{inj,max}} - V_{\text{inj}}^2} \right) \tag{A.37}$$

The expression for injection ratio from Equation 3.55 allows further rearranging of Equation A.37

$$d_{\min} = \tau \ln \left(\frac{V_{\text{osc,max}}^2 + V_{\text{osc,max}} V_{\text{osc}} - K V_{\text{osc,max}}^2 - V_{\text{inj,max}} V_{\text{osc}}}{V_{\text{osc,max}}^2 + V_{\text{osc,max}} V_{\text{osc}} - V_{\text{inj}} V_{\text{inj,max}} - V_{\text{inj}}^2} \right) \tag{A.38}$$

This is the most simplified analytic expression for d_{\min} applicable to ILROs of any number of stages. For ILROs whose number of stages is greater or equal to 5, we can write $V_{\text{osc}} \approx V_{\text{osc,max}}$ and $V_{\text{inj}} \approx V_{\text{inj,max}}$ (as was proven in section 3.3), and these two equations can simplify the expression for d_{\min} even further.

$$\begin{aligned}
d_{\min} &= \tau \ln \left(\frac{V_{\text{osc}}^2 + V_{\text{osc}}^2 - K V_{\text{osc}}^2 - K V_{\text{osc}}^2}{V_{\text{osc}}^2 + V_{\text{osc}}^2 - V_{\text{inj}}^2 - V_{\text{inj}}^2} \right) \\
&= \tau \ln \left(\frac{2V_{\text{osc}}^2 - 2K^2 V_{\text{osc}}^2}{2V_{\text{osc}}^2 - 2V_{\text{inj}}^2} \right) \\
&= \tau \ln \left(\frac{V_{\text{osc}}^2 - K^2 V_{\text{osc}}^2}{V_{\text{osc}}^2 - V_{\text{inj}}^2} \right)
\end{aligned} \tag{A.39}$$

Since $V_{\text{inj}} = K V_{\text{osc}}$, we can continue in simplifying.

$$\begin{aligned}
d_{\min} &= \tau \ln \left[\frac{V_{\text{osc}}^2 (1 - K)}{V_{\text{osc}}^2 - (K V_{\text{osc}})^2} \right] \\
&= \tau \ln \left[\frac{V_{\text{osc}}^2 (1 - K)}{V_{\text{osc}}^2 (1 - K^2)} \right] \\
&= \tau \ln \left[\frac{1 - K}{(1 - K^2)} \right] \\
&= \tau \ln \left[\frac{1 - K}{(1 - K)(1 + K)} \right] \\
&= \tau \ln \left(\frac{1}{1 + K} \right)
\end{aligned} \tag{A.40}$$

C DIGITAL APPENDIX

Below is the directory tree of the digital appendix. The digital appendix is split into several levels of folders. Due to size constraints, only some of the contents of the directory will be listed, but the names of the folders should be self-explanatory and should allow the reader to find the desired files.

```
digital_appendix/ .....root folder
├── latex/
│   ├── chapters/ ..... LATEX files of all chapters and sections
│   │   └── ...
│   ├── diplomova_prace.tex ..... main LATEX file
│   └── ...
├── figures/ .....all figures made in Ipe 7.2.20
│   └── ...
├── charts_and_data/ .....PGFplots charts and source .csv and .tex files
│   └── ...
├── matlab/
│   ├── data/ ..... MATLAB-generated data used for charts
│   │   └── ...
│   ├── scripts/ ..... .m files for calculations and/or plotting
│   │   └── ...
│   ├── schematics/ .....exported Simulink schematics in PDF
│   │   └── ...
│   ├── simscape_lib/ .....Simscape inverter model and library
│   │   └── ...
│   ├── simulink/ ..... Simulink testbenches
│   │   └── ...
│   └── workspaces/ ..... .mat workspaces (variable definitions)
│       └── ...
├── cadence/
│   ├── cadence_data/ ..... Cadence simulation outputs used for charts
│   │   └── ...
│   └── schematics.pdf ..... Cadence Virtuoso designed schematics
```