RING OSCILLATOR BASED RANDOM NUMBER GENERATORS

by

Ülkühan Güler

B.S., Electronics and Telecommunication Engineering, ITU, 1999

M.S., Electrical and Electronics Engineering, The University of Tokyo, 2003

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Doctor of Philosophy

Graduate Program in Electrical and Electronics Engineering

Boğaziçi University

2014

RING OSCILLATOR BASED RANDOM NUMBER GENERATORS

APPROVED BY:

Prof. Günhan Dündar (Thesis Supervisor)	.....
Assoc. Prof. Erkay Savaş	.....
Assoc. Prof. Berk Sunar	.....
Assist. Prof. İsmail Faik Başkaya	.....
Assist. Prof. Ali Emre Pusane	.....

DATE OF APPROVAL: 16.06.2014

ACKNOWLEDGEMENTS

I am very proud to state that it would not be possible to complete this work if I was not inspired, leaded, and supported many people from my family, company, and university. I would like to mention some of them here.

First and foremost, I'd like to express my deepest appreciation and gratitude to my supervisor Prof. Günhan Dündar. It has been an honor to be his Ph.D. student. I would like to thank him for encouraging and mentoring me during my research. His advice, experience, and deep knowledge has always led my studies for a better and productive research.

It was also a fortunate for me to work with my co-advisor Asist. Prof. Ali Emre Pusane. I feel very lucky to be his student as an ex-classmate and doing the research with his wise advises and inspirations.

I would like to sincerely thank to my thesis committee Assoc. Prof. Erkay Savaş, Assist. Prof. İsmail Faik Başkaya, and Prof. Berk Sunar for kindly taking part and sharing their constructive ideas on my research which were very important for the development of this thesis.

I am very thankful to my company TÜBİTAK-BİLGEM for their motivating policy for academic career and funding research studies. I would like to thank my managers Aziz Ulvi Çalışkan and Yaman Özelçi in YİTAL, for their encouraging and supportive attitude to complete my research.

I am also grateful to Salih Ergün who convinced me to study on random number generators in TÜBİTAK-BİLGEM. His supports was so valuable at the beginning of my research. I learnt so many things in a very short time from him, it was a real short cut for me.

The support of my colleagues from my department was very helpful and motivating. I specially thank to Sedat Soydan, Umut Güvenç, and Giray Kömürcü for the cheerful working environment and for lightening the pressure of work and study. I would like to also acknowledge Mustafa Ufuk Demirci for his supports during academic writings.

Also I owe much to Betül Soysal, Muhammet Şahinoğlu, and Elif Büyükkaya from Cyber Security department for their valuable cooperation. I also want to express my thanks to my friend, colleague Gaye Aktürk for listening me and my problems. Her psychological support was great.

Finally, I'd like to thank to my family for all means of support for all the times. Words cannot express how grateful I am to my mother and father for all of the sacrifices that they have made on my behalf. My mother was always with me taking care of children. Finally my deepest thanks go to my husband and my children for their love and patience. It was impossible to complete the research, without their support.

To my husband Arslan, my children İhsan and Bahar and my niece Ecehan

ABSTRACT

RING OSCILLATOR BASED RANDOM NUMBER GENERATORS

System-on-a-chip solutions require hardware based integrated circuit (IC) random number generator (RNG) for trustworthy transmission of information. Fully digital gate based IC RNGs became popular because of their uncomplicated integration to digital platforms. RNGs based on ring oscillators (ROs) are the most well-known and widely used type among various digital gate based RNGs. Existing works on generating random bits by ROs mostly do not have detailed analysis on phase noise and jitter which are the entropy source of an RNG. This thesis analyzes the suitability of existing ROs for random number generation and possible improvements in order to increase the randomness of an RO. Three main contributions are presented. The first main contribution of this thesis to RNG is investigating the possibility of weak inversion operation of complementary metal oxide semiconductor (CMOS) transistors in order to maximize the randomness of CMOS RO used in RNG. In order to predict the weak inversion noise performance of an RO, phase noise and jitter models of a CMOS RO in weak inversion operating region are obtained. These models fill a missing point in the literature and introduce the phase noise and jitter models of RO in weak inversion. This is the second main contribution of this thesis. Despite the consensus on the positive effects of white noise for random number generation, the effects of flicker noise on randomness are insufficiently investigated. Finally, the third main contribution of this thesis in terms of RNG is investigating the effects of flicker noise in order to address its usefulness for random number generation.

ÖZET

HALKA OSİLATÖR TABANLI RASGELE SAYI ÜRETEÇLERİ

Kırmık üzerindeki sistem çözümleri, bilginin güvenli transferi için donanım tabanalı entegre devre (IC) rasgele sayı üreteçlerine (RNG) gereksinim duymaktadırlar. Tamamen sayısal IC RNG'ler sayısal platformlar ile kolayca entegre olabildikleri için sıkça tercih edilmektedirler. Çeşitli sayısal RNG'ler içerisinde halka osilatör (RO) tabanlı RNG'ler en çok bilinen ve kulanılan RNG çeşitidir. Hali hazırdaki RO tabanlı RNG'lerin çoğunun entropi kaynağı olan faz gürültüsü- nün ve saat seğirmesinin analizleri detaylı bir şekilde yapılmamıştır. Bu tezde, hali hazırda kullanılan RO'ların rasgele sayı üretimi için uygun olup olmadıkları ve bir RO'nun rasgeleliğini artıtmak için mümkün olan iyileştirmeler araştırılmaktadır. Rasgele sayı üretimi açısından iki ana katkı ve ROnun faz gürültüsü ve seğirme analizi açısından da bir ana katkı sağlanmıstır. RNG'de kullanılan bütünleyici metal oksit yarı iletken (CMOS) RO'nun rasgeleliğini artırmak için CMOS transistörlerin zayıf evirtim bölgesinde çalıştırılıp çalıştırılamayacağının araştırılması RNG açısından ilk ana katkıdır. RO'nun zayıf evirtimdeki gürültü performansını tahmin edebilmek için CMOS RO'nun zayıf evirtimdeki faz gürültüsü ve seğirme modellerini çıkartmak gerekti. Bu ikinci ana katkı literatürde var olan bir eksikliği gidermiş ve zayıf evirtimde çalışan bir RO'nun faz gürültüsü ve seğirme modellerini takdim etmiştir. Beyaz gürültünün rasgele sayı üretimine olumlu katkılarının herkes tarafından kabul edilmesine rağmen titrek gürültünün rasgelelik üzerinde etkileri konusunda akade- mik çevrelerde bir birliktelik yoktur. Üçüncü ana katkı ise titrek gürültü etkisinin rasgele sayı üretimi üzerindeki faydasını tesbit etmektir.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS . . . iii
ABSTRACT . . . vi
ÖZET . . . vii
LIST OF FIGURES . . . xii
LIST OF TABLES . . . xix
LIST OF SYMBOLS . . . xxi
LIST OF ACRONYMS/ABBREVIATIONS . . . xxiv
1. INTRODUCTION . . . 1
1.1. Motivation . . . 1
1.2. Contributions . . . 2
1.3. Outline of the Dissertation and Publications . . . 5
2. OVERVIEW ON RANDOM NUMBER GENERATORS . . . 8
2.1. RNG Types . . . 8
2.2. RNG Requirements . . . 9
2.3. RNG Applications . . . 10
2.4. RNG Design Techniques . . . 11
2.5. State of the Art . . . 12
2.5.1. The Intel RNG . . . 12
2.5.2. Intel's Digital RNG . . . 13
2.5.3. A Noise-based IC RNG for Applications in Cryptography . . . 15
2.5.4. The Infineon RNG . . . 16
2.5.5. A High-Speed IC Random Number Source for SmartCard Microcontrollers . . . 17
2.5.6. A High-Speed Oscillator-Based Truly Random Number Source for Cryptographic Applications on a SmartCard IC . . . 18
2.5.7. The Nehemiah RNG . . . 19
2.5.8. The RNG for RFID Tag . . . 20
2.5.9. FIGARO . . . 21

2.6. RO-based RNG 23
2.6.1. Model of RO-based RNG 25
2.6.2. Design Details of RO-based RNG 26
2.6.2.1. Ring Oscillators 26
2.6.2.2. XOR Tree Technique 27
2.6.3. Debates on RO-based RNG 31
2.7. Conclusion 33
3. RANDOMNESS SOURCE 35
3.1. Jitter as a Randomness Source 35
3.1.1. Jitter Definition 36
3.1.2. Sources of Timing Jitter 38
3.1.2.1. Intrinsic Noise Sources 40
3.1.2.2. Non-Intrinsic Noise Sources 40
3.1.3. Jitter Types 42
3.1.4. Jitter Accumulation 47
3.1.5. Jitter Decomposition 50
3.1.5.1. Random jitter 51
3.1.5.2. Deterministic jitter 51
3.1.6. A Practical Jitter Measurement 54
3.1.6.1. Determining The Inherent Jitter for Improving the Accuracy 55
3.1.6.2. Period Jitter Measurements 56
3.1.6.3. Cycle to Cycle Jitter Measurements 58
3.1.6.4. Long-Term Jitter Measurements 58
3.2. Flicker Noise as a Randomness Source 59
3.3. Randomness Source in Weak Inversion Region 63
3.4. T-Entropy as an RNG Evaluation Measure 63
3.5. A Pre-Evaluation Method for RNG 64
3.6. Conclusion 66
4. ASIC RNG IMPLEMENTATIONS 67

4.1. A High Speed IC Random Number Generator Based on Phase Noise in Ring Oscillators . . . 68
4.1.1. Implementation Details . . . 68
4.1.2. Measurement and Statistical Test Results . . . 69
4.2. A Digital IC Random Number Generator with Logic Gates Only . . . 73
4.2.1. Fibonacci and Galois Ring Oscillators . . . 74
4.2.2. Implementation Details, Measurement and Statistical Test Results 77
4.3. A High Speed, Fully Digital IC Random Number Generator . . . 82
4.3.1. Implementation Details . . . 83
4.3.2. Measurement and Statistical Test Results . . . 84
4.4. Conclusions . . . 92
5. PHASE NOISE AND JITTER MODELS OF CMOS RO IN STRONG INVERSION REGION . . . 94
5.1. Link between White Noise induced Jitter and Phase Noise . . . 94
5.2. Link between Flicker Noise induced Jitter and Phase Noise . . . 95
5.3. Inverter-based Ring Oscillator . . . 96
5.3.1. Phase Noise and Jitter due to White Noise . . . 97
5.3.2. Phase Noise and Jitter due to Flicker Noise . . . 98
5.4. Differential Ring Oscillator . . . 100
5.4.1. Phase Noise and Jitter due to White Noise . . . 100
5.4.2. Phase Noise and Jitter due to Flicker Noise . . . 102
5.5. Proof of the Correction of Flicker Noise Component of Phase Noise Equation for IbRO . . . 103
5.6. Proof of the Correction of Flicker Noise Component of Phase Noise Equation for DRO . . . 104
5.7. Experimental Verification . . . 106
5.8. Conclusion . . . 108
6. PHASE NOISE AND JITTER MODELS OF CMOS RO IN WEAK INVERSION REGION . . . 115
6.1. Inverter-Based Ring Oscillator . . . 116
6.1.1. Phase Noise and Jitter due to White Noise . . . 116

6.1.2. Phase Noise and Jitter due to Flicker Noise 117
6.2. Differential Ring Oscillator 117
6.2.1. Phase Noise and Jitter due to White Noise 117
6.2.2. Phase Noise and Jitter due to Flicker Noise 119
6.3. Experimental Validation 120
6.4. Conclusion 122
7. DERIVATION OF RANDOMNESS EQUATIONS 126
7.1. Randomness in Strong Inversion Region 127
7.1.1. Inverter-Based Ring Oscillator 127
7.1.2. Differential Ring Oscillator 128
7.2. Randomness in Weak Inversion Region 130
7.2.1. Inverter-Based Ring Oscillator 130
7.2.2. Differential Ring Oscillator 131
7.3. Performance of RO as RNG 131
7.4. Conclusion 136
8. FLICKER NOISE EFFECT ON RANDOMNESS OF CMOS RO 138
8.1. Folding Effect of Flicker Noise on the Sampling Frequency 138
8.2. Entropy Analysis of Noise Sources of an RO 141
8.3. Conclusion 146
9. DESIGN OF EFFICIENT CMOS RO-Based RNG 147
9.1. RO-based RNG Design 147
9.2. Discussion 151
9.3. Experiments 153
9.4. Conclusion 159
10. CONCLUSION 160
10.1. Future Work 161
APPENDIX A: LIST OF PUBLICATIONS 163
A.1. JOURNAL PUBLICATIONS 163
A.2. CONFERENCE PUBLICATIONS 163
REFERENCES 165

LIST OF FIGURES

Figure 1.1. Dark Boxes: PN and jitter models for SI proposed in [1]; Striped Boxes: PN and jitter models for WI proposed in this thesis; Light Boxes: FN induced jitter models proposed in this thesis. 3

Figure 2.1. Classification of random number generators. 9

Figure 2.2. Block diagram of the Intel RNG. 13

Figure 2.3. All-digital TRNG circuit. 14

Figure 2.4. Prototype RNG system architecture of [2]. 15

Figure 2.5. Block diagram of the Infineon RNG. 16

Figure 2.6. RNG architecture of [3]. 17

Figure 2.7. RNG architecture of [4]. 18

Figure 2.8. Nehemiah entropy source design. 19

Figure 2.9. RNG schematic for UWB RFID. 20

Figure 2.10. The transistor level implementation of a single inverter. 21

Figure 2.11. Fibonacci ring oscillator. 22

Figure 2.12. Galois ring oscillator. 22

Figure 2.13. Combination of Firo and Garo. 22

Figure 2.14. Circuit structure based on ring oscillators. 27

Figure 2.15. (a)Inverter-based ring oscillator (b)Differential ring oscillator. . . 28

Figure 2.16. (a)Inverter-based delay cell (b)Differential delay cell. 28

Figure 2.17. XOR tree technique. 29

Figure 2.18. Probability distribution of two Ni after XOR function without phase drift. 30

Figure 2.19. Probability distribution of after XOR gate with phase shift. 31

Figure 3.1. Jittered signal. 36

Figure 3.2. Gaussian Distribution. 37

Figure 3.3. Jitter sources. 39

Figure 3.4. Wander vs jitter. 42

Figure 3.5. Jitter types. 44

Figure 3.6. Relation between jitter types. 46

Figure 3.7. Illustration of phase shift. 48

Figure 3.8. Jitter accumulation with varying time. 48

Figure 3.9. RMS jitter versus measurement time. 49

Figure 3.10. Components of jitter. 50

Figure 3.11. Random jitter. 51

Figure 3.12. Periodic jitter. 52

Figure 3.13. Crosstalk. 53

Figure 3.14. Inter symbol interference. 53

Figure 3.15. Duty cycle distortion. 54

Figure 3.16. Inherent jitter measurement setup. 56

Figure 3.17. Inherent jitter measurement. 57

Figure 3.18. Period jitter measurement. 57

Figure 3.19. Cycle-to-cycle jitter measurement. 58

Figure 3.20. TIE measurement with 500ns accumulation. 59

Figure 3.21. Time interval error measurement with 500ns accumulation. 60

Figure 3.22. Time interval error measurement with 500ns accumulation. 60

Figure 3.23. Accumulated jitter versus measurement time. 62

Figure 4.1. Layout of rings design. 68

Figure 4.2. Chip photo. 70

Figure 4.3. Measured output of one of the ring oscillator. 71

Figure 4.4. Consumed current versus supply voltage. 73

Figure 4.5. Fibonacci ring oscillator. 75

Figure 4.6. Galois ring oscillator. 75

Figure 4.7. Layout of FIGARO. 76

Figure 4.8. Chip photo. 78

Figure 4.9. Proposed configuration. 78

Figure 4.10. Measured output of FIGARO. 82

Figure 4.11. Layout of ring oscillators design. 83

Figure 4.12. Chip photo. 84

Figure 4.13. Measured output of XOR tree of 256 ROs. 85

Figure 4.14. Measured output of XOR tree of 256 ROs with histogram. 86

Figure 5.1. Inverter-based ring oscillator. 96

Figure 5.2. Inverter symbol. 97

Figure 5.3. Transistor based architecture of inverter. 97

Figure 5.4. Differential ring oscillator. 100

Figure 5.5. Transistor based architecture. 101

Figure 5.6. Fabricated die photo. 108

Figure 5.7. Measurement set-up. 109

Figure 5.8. Measurement environment. 109

Figure 5.9. Frequency-domain measurement of IbRO in strong inversion. . . . 110

Figure 5.10. White noise component of phase noise for IbRO in strong inversion region. 111

Figure 5.11. Flicker noise component of phase noise for IbRO in strong inversion region. 112

Figure 5.12. White noise component of phase noise for DRO in strong inversion region. 113

Figure 5.13. Flicker noise component of phase noise for DRO in strong inversion region. 114

Figure 6.1. Frequency-domain measurement of IbRO in weak inversion. 121

Figure 6.2. White noise component of phase noise for IbRO in weak inversion region. 122

Figure 6.3. Flicker noise component of phase noise for IbRO in weak inversion region. 123

Figure 6.4. IbRO's phase noise due to white noise. 124

Figure 6.5. Flicker noise component of phase noise for DRO in weak inversion region. 125

Figure 7.1. Randomness vs energy for IbRO. 133

Figure 7.2. Randomness vs energy delay product for IbRO. 133

Figure 7.3. Supply voltage vs randomness behavior for a given frequency. . . . 135

Figure 8.1. PSD of white noise with varying fs. 139

Figure 8.2. PSD of flicker noise with varying fs. 140

Figure 8.3. Aliasing in flicker noise model used in analysis. 141

Figure 8.4. Accumulated jitter vs time for an RO with 23 inverters in strong inversion. 142

Figure 8.5. T-Entropy values of an RO in strong inversion with different noise sources induced bit streams. 143

Figure 8.6. Accumulated jitter vs time for an RO with 23 inverters in weak inversion. 144

Figure 8.7. T-Entropy values of an RO in weak inversion with different noise sources induced bit streams. 145

Figure 9.1. Entropy comparison of 23-inverter RO in strong and weak inversion. 148

Figure 9.2. T-Entropy values of XORed ROs in strong inversion. 149

Figure 9.3. T-Entropy values of XORed ROs in strong inversion with scattered frequencies. 150

Figure 9.4. T-Entropy values of XORed ROs in weak inversion. 151

Figure 9.5. T-Entropy values of XORed ROs in weak inversion with scattered frequencies. 152

Figure 9.6. T-Entropy per energy versus throughput comparison of RNGs in both region. 153

Figure 9.7. Chip photo. 154

Figure 9.8. Measurement Setup. 155

LIST OF TABLES

Table 2.1. An example estimation for *N=100* urns, the number r of ring oscillators necessary to fill at least $f \times N$ of the urns with probability at least p. 26

Table 2.2. Comparison table. 34

Table 3.1. Standard deviation amount versus window coverage of all measurements. 38

Table 3.2. RNG Pre-Evaluation Check List. 65

Table 4.1. Statistical test results of RO-based RNG with 114 ROs. 72

Table 4.2. Summarized performance parameters of RO-based RNG with 114 ROs. 74

Table 4.3. Number of required FIGARO and corresponding sampling frequency. 79

Table 4.4. Statistical test results of seven XORed FIGARO with processed data. 80

Table 4.5. Statistical test results of eight XORed FIGARO with raw data. . . 81

Table 4.6. Summarized performance parameters of FIGARO. 81

Table 4.7. Statistical test results of RO-based RNG with 256 ROs. 87

Table 4.8. Statistical test results of the RO-based RNG with 512 ROs 88

Table 4.9. Comparison of the ASIC implemented RNGs with some well-known RNGs in the literature. 90

Table 4.10. Summarized performance parameters of 256 XORed ROs. 91

Table 5.1. Units of terms. 105

Table 5.2. Verification of corrections for phase noise due to flicker noise in IbRO. 106

Table 5.3. Verification of corrections for phase noise due to flicker noise in IbRO. 107

Table 7.1. Transistor aspect ratios of DRO. 135

Table 7.2. Comparison of performance parameters for both RO ($f_0 \approx 10\,MHz$). 137

Table 9.1. Statistical test results of RNG with 12 XORed ring oscillators operating in strong inversion. 156

Table 9.2. Statistical test results of RNG with three XORed ring oscillators operating in weak inversion. 157

Table 9.3. Performance comparison. 158

LIST OF SYMBOLS

A	Current mirror ratio
C	Load capacitance
C_{ox}	Oxide capacitance of transistor
E	Energy
f	Fill rate
f_0	Nominal frequency which an oscillator orbits in steady-state
f_c	Corner frequency
f_{osc}	Oscillation frequency of an oscillator
g_m	Transconductance of transistor
I	Current
I_{dN}	Current of NMOS device
I_{dP}	Current of PMOS device
I_{sat}	Saturation current
k	Boltzman coefficient
k'	Product of mobility (μ) and C_{ox}
K_I	Current sensitivity
K_f	Flicker noise coefficient
K_{fN}	Flicker noise coefficient of NMOS device
K_{fP}	Flicker noise coefficient of PMOS device
L	Length of transistor
$L(f)$	Single Side Band Lorentzian Phase Noise
$L^{1/f}(f)$	Single Side Band Lorentzian Phase Noise due to flicker noise
$L(f)_w$	Single Side Band Lorentzian Phase Noise due to white noise
M	Number of stages in an RO
n	Subthreshold slope factor
N	Number of urns
N_1	Number one bit stream
N_2	Number two bit stream

$N_{(XOR)}$	XORed bit stream
p	Level of confidence
P	Power Consumption
r	Minimum number of ring oscillators
q	Coulomb coefficient
R	Resistance
R_{eqload}	Equivalent resistance at load
$S_I(f)$	Power spectral density of current
$S_{iN}(f)$	Power spectral density of current of NMOS device
$S_{iP}(f)$	Power spectral density of current of PMOS device
$S_V(f)$	Power spectral density of voltage
$S_{v_n}(f)$	Power spectral density of voltage of NMOS device
$S_{v_p}(f)$	Power spectral density of voltage of PMOS device
$S_\tau(f)$	Power spectral density of the period
$S_\phi(f)$	Power spectral density of phase
T	Temperature coefficient
t_d	Propagation delay
t_{dN}	Propagation delay of NMOS device
t_{dP}	Propagation delay of PMOS device
U_t	Thermal voltage
V_{DD}	Supply voltage
V_{ds}	Drain source voltage
V_{dst}	Drain source voltage of transistor at the tail
V_{dsd}	Drain source voltage of transistor at the differential input
V_{effd}	Effective overdrive voltage of input transistors
V_{efft}	Effective overdrive voltage of tail transistor
V_{gs}	Gate source voltage
V_n	Noise voltage
V_{nR}	Noise voltage on loads
V_{op}	Differential peak output voltage swing
V_t	Threshold voltage

W	Width of transistor
$\frac{1}{f^{\alpha}}$	Flicker noise spectrum
ΔT	Time difference between two certain points
γ	Noise coefficient
γ_N	Noise coefficient of NMOS transistor
γ_P	Noise coefficient of PMOS transistor
μ	Mean value of N_1
μ_N	Mobility of NMOS device
μ_P	Mobility of PMOS device
μ	Mean value of N_1
ν	Mean value of N_2
ϕ	Phase of an oscillator
ψ	Mean value of $N_{(XOR)}$
$\mathcal{R}$	Randomness
ρ	Correlation coefficient
σ_τ	Period jitter is defined as standard deviation of τ
$\sigma_\tau^{1/f}$	Flicker noise induced period jitter is defined as standard deviation of τ
σ_τ^w	White noise induced period jitter is defined as standard deviation of τ
σ_τ^2	Mean-square value of jitter
σ_τ^{1/f^2}	Mean-square value of flicker noise induced jitter
$\sigma^{2^w}_\tau$	Mean-square value of white noise induced jitter
τ	Mean-period

LIST OF ACRONYMS/ABBREVIATIONS

3D	Three Dimensional
ABUJ	Aperiodic Bounded Uncorrelated Jitter
A/D	Analog Digital Converter
ADC	Analog Digital Converter
AES	Advanced Encryption Standard
AM	Amplitude Modulation
ASIC	Application Specific Integrated Circuit
AVA	Assurance of Vulnerability Assessment
BUJ	Bounded Uncorrelated Jitter
CC	Common Criteria
CCO	Current Controlled Oscillator
CMOS	Complementary Metal Oxide Semiconductor
DC	Direct Current
DCD	Duty Cycle Distortion
DDJ	Data Dependent Jitter
DES	Data Encryption Standard
DFF	D-type Flip Flop
DRO	Differential Ring Oscillator
DUT	Design Under Test
EMI	Electromagnetic Interference
FET	Field Effect Transistor
FF	Flip Flop
FIFO	First In First Out
FIPS	Federal Information Processing Standards
FM	Frequency Modulation
FN	Flicker Noise
FPGA	Field Programmable Gate Array
FSM	Finite-state Machine

FIRO	Fibonacci Ring Oscillator
HHNEC	Hua Hong Nippon Electric Company
GARO	Galois Ring Oscillator
IbRO	Inverter based Ring Oscillator
IC	Integrated Circuit
I/O	Input Output
ISI	Inter Symbol Interference
ITU	International Telecommunication Union
LFSR	Linear Feedback Shift Register
MOSFET	Metal Oxide Semiconductor Field Effect Transistor
NFC	Near Field Communication
NFET	N-channel Field Effect Transistor
NIST	National Institute of Standards and Technology
NMOS	N-channel Metal Oxide Semiconductor
PN	Phase Noise
PCI	Peripheral Component Interconnect
PFET	P-channel Field Effect Transistor
PLL	Phase Locked Loop
PM	Phase Modulation
PMOS	P-channel Metal Oxide Semiconductor
PRNG	Pseudo Random Number Generator
PSD	Power Spectral Density
RF	Radio Frequency
RFID	Radio Frequency Identification
RIPE	RACE Integrity Primitives Evaluation
RIS	Random Interleaved Sampling
RISC	Reduced Instruction Set Computing
RMS	Root Mean Square
RNG	Random Number Generator
RO	Ring Oscillator
SC	Smart Card

SC	Switch Capacitor
S/H	Sample and Hold
SDA	Serial Data Analysis
SI	Strong Inversion
SSB	Single Side Band
SSC	Spread Spectrum Clock
SHA	Secure Hashing Algorithm
TFF	T-type Flip Flop
TIE	Time Interval Error
TRNG	True Random Number Generator
UWB	Ultra Wide Band
UI	Unit Interval
VAN	Vulnerability Analysis
VCO	Voltage Controlled Oscillator
VLSI	Very Large Scale Integration
WI	Weak Inversion
WN	White Noise
WSN	Wireless Sensor Network
XOR	Exclusive OR

1. INTRODUCTION

In the information age, random number generators (RNGs) are indispensable in achieving secure communication. The increasing demand for random bits in the digitized world for secure communication makes RNGs widely used not only in cryptographic applications but also in typical communication applications as secret keys, masking and blinding values, padding bits, nonces, and seeds.

Most of the typical communication equipments are implemented using digital gates; therefore, an RNG, which is implemented by digital gates, is preferred by designers, because it provides the opportunity of implementing the whole circuit as an FPGA or as an ASIC. However, ASIC implementation of RNGs is a relatively limited research area, and the reports are still insufficient on RNG designs and their analysis.

Among the different digital gate based RNG types, ring oscillator (RO)-based RNGs have become very popular due to their simple design, easy implementation, relatively compact area, and high speed compared to other RNG design methods. However, in spite of the fact that VLSI circuits are designed very energy-consciously in this era, most of the oscillator-based RNGs are power hungry circuits, which is apparently a drawback.

1.1. Motivation

In [5, 6], Sunar *et al.* proposed an RNG based on jittered ring oscillators with a model that estimates the amount of entropy. The amount of entropy is extracted from the design parameters, such as the number of inverters in a chain and the number of ring oscillators in the circuit.

Sunar's design suggestion increases the number of ROs to gain sufficient entropy. It is obvious that increasing the number of ROs causes a higher area occupation and power consumption. Moreover, the applicability of RNG becomes more limited due to

its huge power consumption.

The drawbacks of higher area occupation and power consumption motivate better RNG designs. To alleviate these drawbacks, the best solution seems to be to increase the entropy of a single RO and decrease the number of ROs used in an RNG. Increasing the randomness at the source leads to an entropy increase in RNG as well. Therefore, this has a vital significance, especially in terms of preserving resources.

1.2. Contributions

In this dissertation, we have investigated better RNG design solutions to minimize and conserve the resources. Thus, the need for a large variety of RNG application areas would be met with new RO-based RNGs. The main contributions of this dissertations can be listed as follows:

- This dissertation proposes a unified framework by filling the gaps in the literature regarding phase noise (PN) and jitter models, and correcting some errors. PN and jitter models which are induced from white noise (WN) and flicker noise (FN), are explored for both strong inversion (SI) and weak inversion (WI) regions. Figure 1.1 presents all of the cases covered in the thesis briefly. The phase noise and jitter derivation approach [1] is verified and validated by the measurements with our design technology as shown with the dark boxes in the figure.

  Jitter due to flicker noise, not included in [1], is obtained by the help of the approximation given in [7]. The light boxes indicate these cases in the figure. Furthermore, errors in flicker noise induced phase noise, published in [1], are corrected for both inverter-based ring oscillator (IbRO) and differential ring oscillator (DRO).

  The phase noise analysis is extended to weak inversion region to derive phase noise and jitter models of a CMOS RO operating in the sub-threshold region which are shown with the striped boxes. Both cases of IbRO and DRO are in-

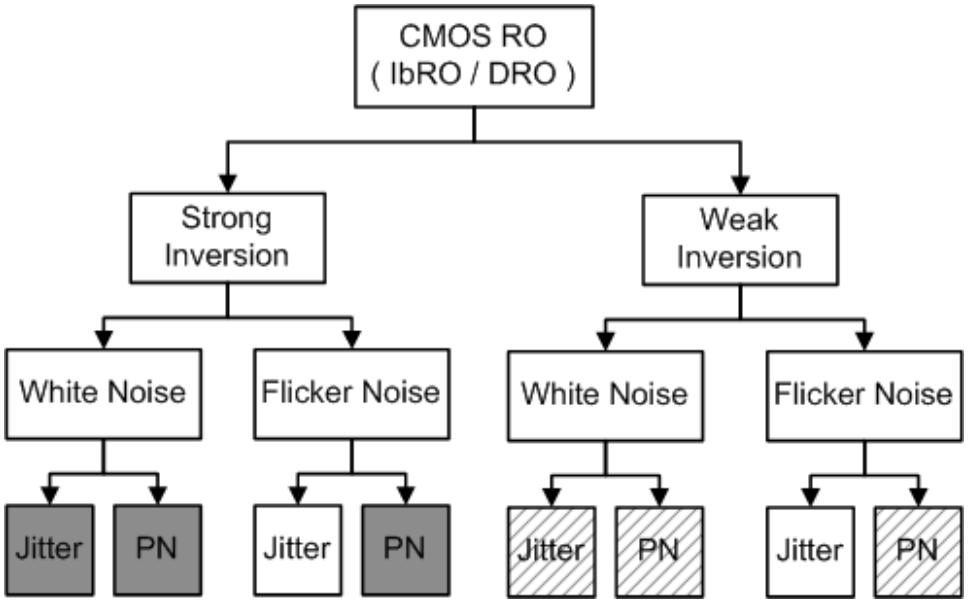

Figure 1.1. Dark Boxes: PN and jitter models for SI proposed in [1]; Striped Boxes: PN and jitter models for WI proposed in this thesis; Light Boxes: FN induced jitter models proposed in this thesis.

cluded. In this dissertation, all of these models are verified with both simulations and measurements.

- A well-defined quality metric for an RNG was missing in the literature. In this dissertation, a figure of merit is proposed to compare the performance of RNGs. It defines the amount of uncertain zones in a period as *Randomness* ($\mathcal{R}$). We have formulated it as the jitter to period ratio, given as

$$\mathcal{R} = \frac{\sigma_\tau}{\tau} = \sigma_\tau \mathbf{f_0}. \tag{1.1}$$

  After the introduction of the randomness metric, randomness equations are obtained from phase noise and jitter equations. The strong inversion and weak inversion counterparts of randomness have been derived and compared. It is proved that ROs in weak inversion exhibit more randomness than ROs in strong inversion.

- By using the advantage of more randomness and less power consumption in weak inversion region, we have proposed to operate RO-based RNGs in weak inversion. Operating ROs in weak inversion region provides a significant improvement in power and energy consumption. Furthermore, the required level of randomness is achieved with fewer ROs which are operating in weak inversion.

- The effect of flicker noise on randomness was almost an untouched research area in random number generation. Analysis showed that flicker noise can be used in a favorable way to provide more randomness to RNG.

- Although [6] uses many ROs when forming an RNG, we demonstrate in this thesis that fewer ROs may be enough to acquire the required level of randomness. We analyze the minimum necessary number of ROs to attain a specified randomness level.

By taking into account all these contributions listed above, a hardware-based, low power, true random number generation method with a moderate speed is proposed. Some of the possible application areas are given below.

- RFID Systems: A real random number generator plays an important role in tag collision protocol of radio frequency identification (RFID) system [8]. In addition, the problem of information security becomes more and more critical with the extensive use of RFID systems [9]. Therefore, implementing a low power RNG in a small area is an important issue in RFID system design.

- Wireless Sensor Networks: Wireless Sensors Networks (WSN) are networked constrained devices using radio communication and providing sensing services such as surveillance of a restricted area or sensing of environment. Many new security protocols are being designed. Most of them rely on cryptography, therefore, often require a good random number generator [10]. Some WSN systems require an ultra-low power sensor network platform such as i-Bean Network [11]. A re-

liable RNG in WSN systems is necessary for various purposes, such as random backoffs, random transmission delays, and random packet sequence numbers.

- Low-power Micro-controllers and Low-power Embedded Applications: Some multi purpose micro-controllers, such as MSP430 [12], have aggressive low energy properties. They are designed for very low power specifications. Since these are low power micro-controllers, it is obvious that such micro-controllers should save as much energy as possible at all modules. For this reason, RNGs in such devices should also be low power. A low power RNG is proposed for the MSP430 micro-controller family in [13].

- Secure Micro-controllers: There are many contact and contactless secure micro-controller solutions in the market today [14]. Depending on the application requirements, some low-power low-cost secure micro-controllers with reduced instruction set computing (RISC) [14] also available. All secure micro-controllers have various cryptographic engines. RNGs supply random keys to these engines. In the case of low-power low-cost micro-controllers, security and power are both critically important. Hence, RNGs also need to meet the low-power, low cost requirements and need to produce highly qualified bits.

1.3. Outline of the Dissertation and Publications

This thesis is organized in the following manner:

In Chapter 2, RNG types, requirements, and design techniques are briefly summarized. The current state of RNG designs is discussed. An introduction of RO-based RNG, which is the main topic of this dissertation, is presented. Moreover, the primary blocks in RO-based RNG, such as ring oscillators and exclusive-or (XOR) tree, are explained.

In Chapter 3, detailed information is provided about jitter; namely, sources of jitter, types of jitter, and decomposition of jitter are summarized. Furthermore, a

practical jitter measurement is presented. In the long-term behavior of jitter, the effect of flicker noise is emphasized. Methods where by randomness can be increased are investigated. A brief information is given about the noise level of CMOS transistors in weak inversion region. T-Entropy measure is introduced also in this chapter. In addition, a practical pre-evaluation method for the output waveform of an RNG is proposed [15].

The first ASIC implementation examples [16–18] of RO-based RNG and Fibonacci and Galois ring oscillator (FIGARO) are presented in Chapter 4.

Chapter 5 briefly summarizes Abidi's approach published in [1]. Furthermore, the missing link between flicker noise induced phase noise and jitter is established. After that, flicker noise induced jitter equation is derived for both type of ROs [19].

Phase noise and jitter equations of CMOS ROs are derived for weak inversion region [19, 20] in Chapter 6.

In Chapter 7, a randomness metric is defined as the ratio of jitter to mean-period, and randomness equations are obtained for both strong inversion and weak inversion regions [19–21]. The corresponding estimation, simulation, and measurement results are discussed. Moreover, performance of ROs are investigated in terms of RNG usage. In the context of randomness behavior, appropriateness of each case to RNG is explored. In the light of the findings, how maximum randomness could be achieved is discussed.

Chapter 8 questions the flicker noise effect on randomness [22, 23]. Synthetic bit streams are produced in order to compare the effects of the noise sources individually. However, the realistic long-term jitter values are embedded into the flicker and white noise models. In addition, sampling effect on randomness is examined.

In Chapter 9, design of efficient RO-based RNG is explored. T-Entropy measure is used to determine the randomness levels of bit streams. As a performance measure,

entropy per energy is introduced. The explored design parameters for an efficient design are the number of inverters in an RO and the number of ROs in an RNG [23]. According to the analysis, some RNG configurations are selected and measured. Measurements are coherent with the analysis.

Conclusions of the thesis and prospective future works are given in Chapter 10.

2. OVERVIEW ON RANDOM NUMBER GENERATORS

Random numbers are defined in [24] as "a sequence of independent numbers with a specified distribution and specified probability of falling in any given range of values." Random numbers are generated with physical or computational devices which are called random number generators.

2.1. RNG Types

Random Number Generators can be divided into two sub-categories namely pseudo random or true random, which can be further categorized based on their implementation as software or hardware. Hardware RNGs can also be separated into two classes; fully digital blocks and analog blocks, where each have various implementation methods. Classification of random number generators is depicted in Figure 2.1. Complexity or simplicity, security issues, and cost are the main concerns to determine the type of RNG. Although it is relatively simple to produce random bits by software based Pseudo Random Number Generators (PRNGs), which use key strokes, mouse movements [25], computer system clock [26], hard disk drive turbulence [27] as the seed. The random number generation method should be kept as secret in this type of RNG generation methods. Hardware based PRNGs may produce random bits also in a deterministic way by implementing a polynomial or an algorithm. However, by properly seeding a PRNG from a reliable source [28], sufficient entropy could be obtained for generating truly random numbers. On the other hand, TRNGs produce random streams from a nondeterministic natural source; electronic noise and radioactive decay are two examples of usable processes [3].

Since all the RNGs designed and analyzed in this dissertation are hardware based RNGs and use random source, TRNG is simply called RNG in the rest of the dissertation. Moreover, the reader should keep in the mind that the term *randomness* means *true randomness* in this context.

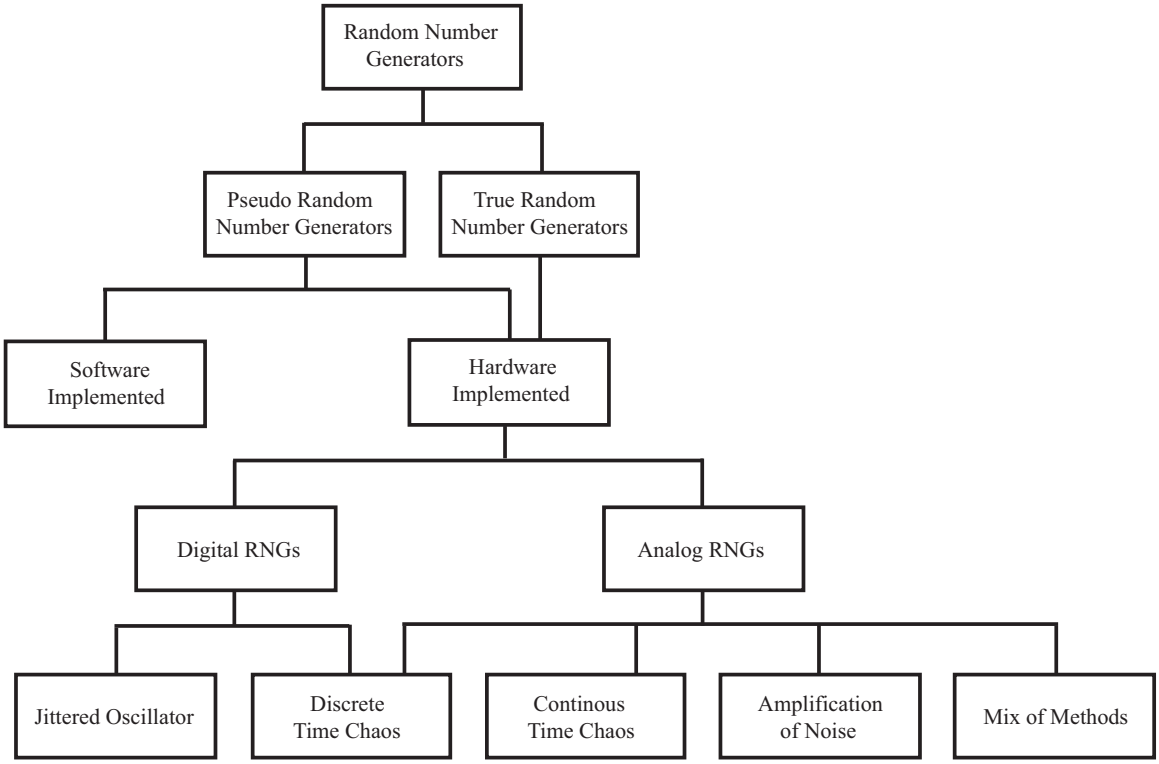

Figure 2.1. Classification of random number generators.

2.2. RNG Requirements

Security protocols rely on the unpredictability of the keys they use; therefore, RNGs must meet stringent requirements [28]. Even the designer of the RNG should not be able to acquire any useful information about the random sequence. There are three important criteria that RNGs should meet for true randomness:

(i) The next random bit must be unpredictable.
(ii) Random bit stream must pass all statistical tests of randomness e.g. NIST 140-2 Test Suite [29], NIST 800-22 Test Suite [30], DIEHard Battery Test [31], and Mourer's Universial Statistical Test [26].
(iii) The same random bit stream must not be able to be reproduced [26].

Since RNGs are designed for cryptographic purposes, robustness against possible attacks is crucial. From this point of view, the integrated circuit implementation of

an RNG has primary importance. Electronic noise and timing jitter, which is actually the result of the noise in a circuit, are usually the only stochastic phenomena that are suitable in integrated implementations [32]. It is pointed out in [26] using the natural randomness of real world is the best way to generate truly random numbers.

Besides quality, unpredictability, and unreproducability requirements, the generator must be robust against aging effects and intentional or unintentional environmental variations, such as temperature, power supply, electromagnetic radiation, etc. [33]

2.3. RNG Applications

RNGs are utilized in a variety of application areas. Some of them will be listed below.

- Cryptography: Random numbers are at the heart of cryptology. Since algorithms may be open to public, the strength of cryptography is achieved by highly qualified random numbers.
- Simulation and Testing: Random sequences are used in simulations of complex scientific and financial models. In addition, in testing of algorithms and programs, software developers frequently use random numbers.
- Digital Signatures: Enhanced network and computer security are very important in e-Business and e-Government applications, hence random numbers are necessary.
- Equation Solving: To solve complex equations, mathematicians use random bits.
- Secure Communication: Secure data and data transmission require unpredictable keys
- Entertainment: Random numbers are used, especially, in lottery and gambling machine algorithms.
- Music and Graphics Composition: In image processing, 3D techniques for computer graphics, and electronic music, random bits are needed.

2.4. RNG Design Techniques

RNG design techniques rely on three essential methods:

(i) The first method is the amplification of a white noise (WN) source such as thermal and shot noise [2, 4, 28]. However, noise on power supplies may be the dominant noise source in the circuit, which will then affect the quality of generated random bits. Moreover, it is an open door for attackers to manipulate the output random sequences. Hence, a careful work is necessary to eliminate the power supply noise.

(ii) The second method is jittered oscillator sampling [34,35]. Although this method is generally used in combination with other methods, there are some designs implemented with purely digital blocks [5, 36]. Since these circuits are purely digital, they either usually do not ensure enough entropy source to generate truly random bits or a large number of replicated structures have to be used to get sufficient entropy levels.

(iii) The third method is continuous [37–39] and discrete time chaos [40, 41]. Having a period, even with too long periods, such as a couple of years, is the drawback of discrete time chaos. On the other hand, continuous time chaotic circuits are faster than discrete ones. In spite of demonstration of their numeric and experimental verification, their mathematical verification has not been performed yet.

There is some research in the literature that combines some of the aforementioned methods [2, 39, 42]. Amplification of noise, jittered oscillator, and discrete time chaos techniques are utilized together in [2]. Similar to this approach, an RNG design that uses the dual oscillator architecture with continuous-time chaotic oscillator was proposed in [42]. Numerical model of that work, which allows the estimation of the output entropy and bias as a function of the design parameter, was further presented in [38].

The advantage of its easier integration with digital micro-processors and capability of its implementation on reconfigurable platforms makes digital gate based RNGs attractive for designers who work on purely digital environments. Moreover, digital gate based RNGs may provide higher throughput than analog ones [16]. On the other hand, the designer has to pay special attention to the quality of the entropy source, because the limited entropy source problem is a known fact in digital gate based RNGs.

Recently, ROs have been widely used for random number generation in security applications due to their simple design, easy implementation, compact area, and relatively high speed compared to other RNG design methods. By sampling the jittered oscillator, the oscillator jitter is utilized as the entropy source to gather randomness. Despite the uncertainty on quality and amount of the entropy of this method, it is widely used in generation of random bits in combination with other methods [4,32,34] or with purely digital blocks [5,36,43].

2.5. State of the Art

In this section, some of the well-known RNGs in the literature will be briefly summarized, and their specifications will be compared.

2.5.1. The Intel RNG

The Intel RNG [28] primarily samples thermal noise by amplifying the voltage across un-driven resistors. The amplified noise-based voltage drives a voltage controlled oscillator (VCO), which samples a high speed oscillator. The output sequence is post processed using the von Neumann corrector and then hashed using Secure Hashing Algorithm (SHA-1). The block diagram of the Intel RNG is seen in Figure 2.2.

The output bit rate is $75kbps$. There is no given information about the design technology, design details and the test result applied to raw data before post processing. The processed data passes the chosen tests from NIST Test Suite of FIPS 140-1 [44].

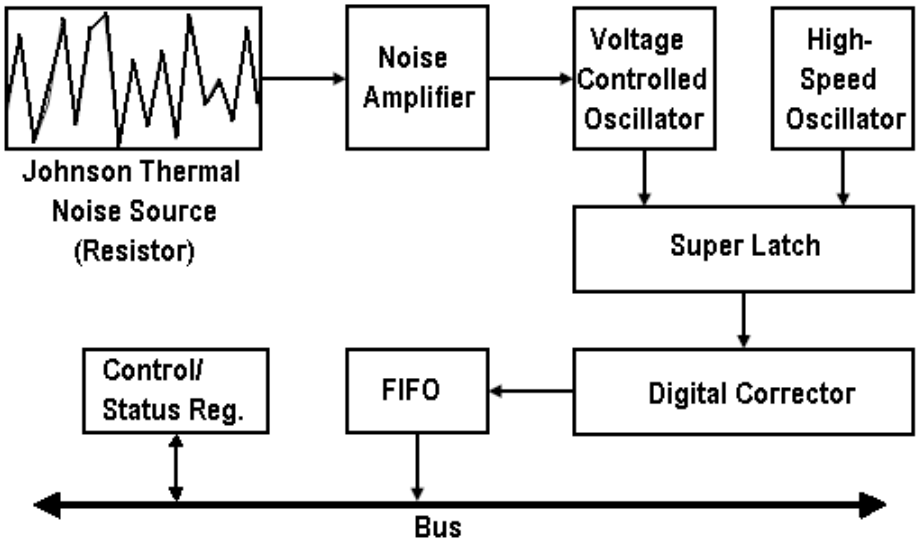

Figure 2.2. Block diagram of the Intel RNG.

2.5.2. Intel's Digital RNG

The all-digital True Random Number Generator [45] is fabricated in $45nm$ CMOS high-K/metal gate technology. The TRNG has $2.4Gbps$ random bit throughput and consumes a total power of $7mW$. Two-step coarse/fine-grained tuning with a self-calibrating feedback loop enables robust operation in the presence of 20% process variation while providing immunity to run-time voltage and temperature fluctuations. The 100% digital design enables a compact layout occupying $4004\mu m^2$ with measured entropy of 0.999965, and scalable operation down to $280mV$, while passing all NIST RNG tests.

The all-digital TRNG, shown in Figure 2.3, harvests entropy from differential thermal-noise at the diffusion nodes of a cross-coupled inverter pair to resolve out of meta-stability, generating one random bit/cycle. The cross-coupled inverter is forced into metastability by pre-charging output nodes a and b while cutting off short-circuit current path using a shared NMOS footer device. During the evaluation phase, both output node voltages discharge towards the intersection of inverter, and one of the nodes a or b pulls back up to V_{CC} under the influence of random differential thermal

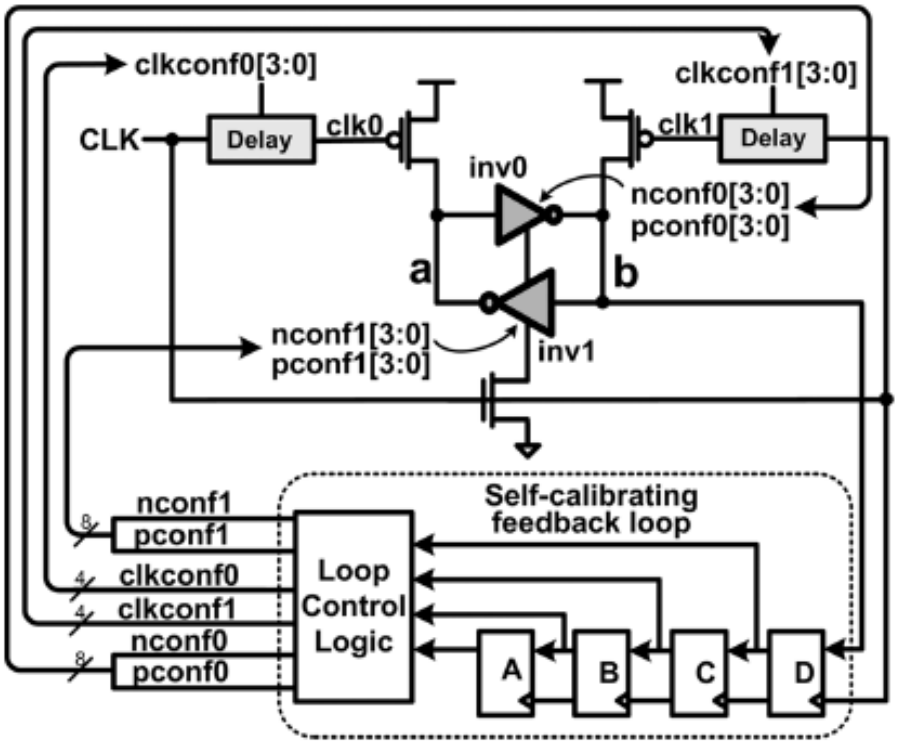

Figure 2.3. All-digital TRNG circuit.

noise. Process, voltage, and temperature (PVT) variations, mismatches or coupled noise can disrupt this ideal behavior by introducing biases that favor resolution towards one state. Three digital mechanisms are provided to overcome biases:

(i) Coarse-grained tuning using configuration bits pconf[3:0], nconf[3:0] that turn on/off parallel P/N legs to modulate inverter device strengths.

(ii) Fine-grained tuning using programmable clock delays to control relative skew of pre-charge release times.

(iii) Self-calibrating feedback loop controlled by a finite-state machine (FSM) that monitors the output bit and updates one of six 4b counters to keep the TRNG continuously biased in the high-entropy zone.

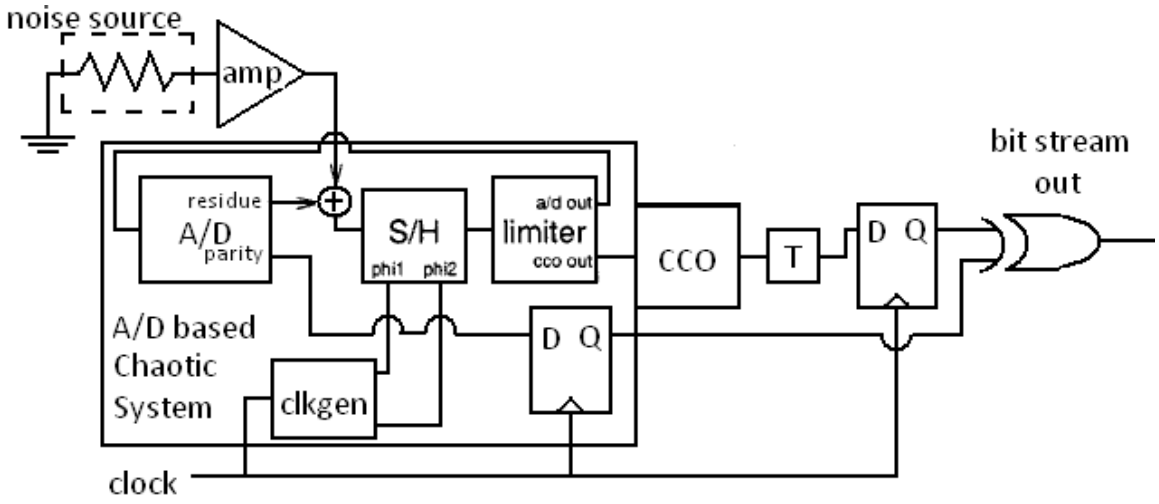

Figure 2.4. Prototype RNG system architecture of [2].

2.5.3. A Noise-based IC RNG for Applications in Cryptography

Three different RNG design techniques are combined in [2]. Amplified thermal noise is summed into the analog to digital converter (A/D)-based chaotic system, which is used to drive a current controlled oscillator (CCO). The CCO output is sampled at a lower, user defined clock frequency using a D flip-flop. A limiter circuit is added to limit the A/D input signal in the range (0, *Iref*) in order to prevent the chaotic loop from getting stuck in a steady-state saturated condition. The prototype RNG system architecture is shown in Figure 2.4.

The prototype RNG was fabricated in a $2\mu m$ n-well CMOS process. The circuit area excluding pads was $850\mu m \times 1750\mu m \approx 1.5mm^2$. The RNG system was powered with a single 3V supply and dissipated $3.9mW$ of power when clocked at $1MHz$. The system passed the chosen tests from NIST Test Suite of FIPS 140-1 [44] up to $1.4MHz$ clock frequency.

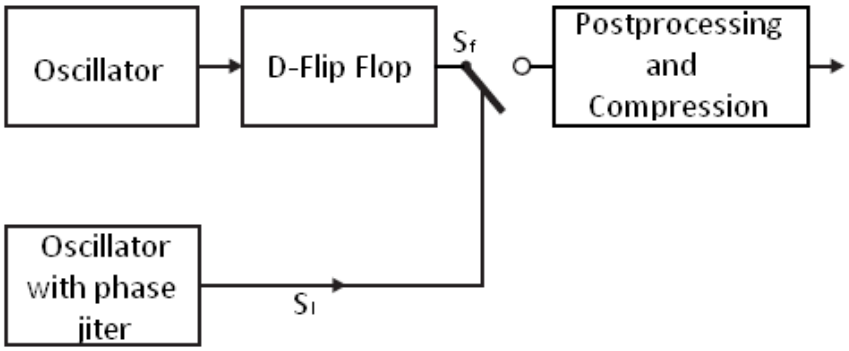

Figure 2.5. Block diagram of the Infineon RNG.

2.5.4. The Infineon RNG

The Infineon RNG [35], shown in Figure 2.5, samples a high frequency signal S_f by a low frequency signal S_l that has considerable phase jitter. The period deviation of the jittered-oscillator was about 1% of the period length. The frequency of S_f was chosen to be more than a hundred times of S_l. This ensures that at the time the next edge of S_l triggers a new sampling of S_f, the state of the oscillator producing S_f is independent of the state at the previous sampling. A D flip-flop was used to divide the period of the fast oscillator period to its half value in order to balance the duty-cycle of S_f, which improves the bias of the sampled bit stream.

In the Infineon hardware RNG, a linear feedback shift register is used for post processing of the random data. The sum of the feedback bits and the bit resulting from the sampling process are fed into the first stage of the shift register. The linear feedback shift register (LFSR) is followed by a compression step. The RNG uses 16 bits of data generated by the process described above and combines them to an eight bit random number.

There is no detailed information about the architecture of the oscillators, process technology, and throughput of the RNG. The quality of the Infineon RNG is evaluated by using the RIPE test suite. RIPE (RACE Integrity Primitives Evaluation) is a project funded by the European Union in the years 1989 through 1992. Moreover,

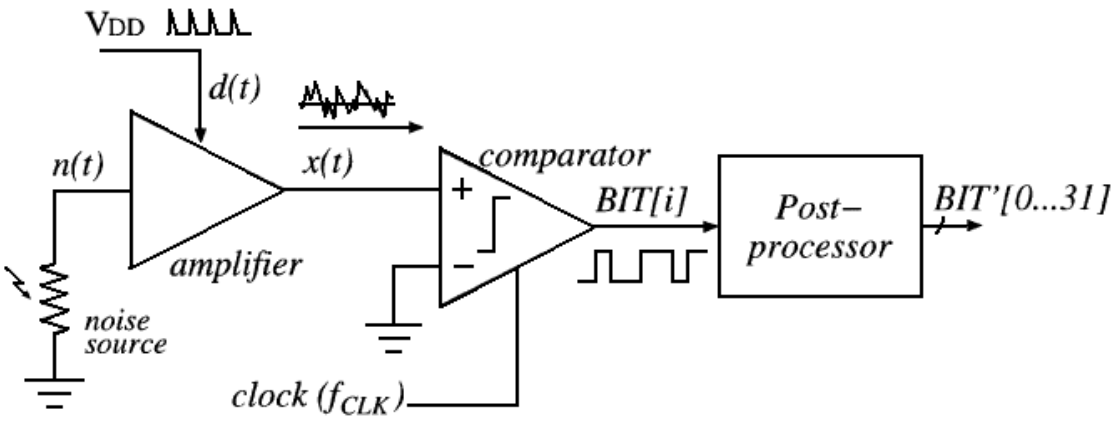

Figure 2.6. RNG architecture of [3].

NIST Test Suite of FIPS 140-1 [44] was used for RNG evaluation. The requirements of the test are easily met by the Infineon RNG with one exception for processed data: the raw data exhibited bias, which caused most tests to indicate a deviation from the ideal RNG.

2.5.5. A High-Speed IC Random Number Source for SmartCard Microcontrollers

The RNG in [3] is based on the amplification of thermal noise generated by integrated resistors. The amplified noise is compared to a reference voltage by a clocked comparator whose output is a random bit stream. In Figure 2.6 the RNG architecture is demonstrated. Since no adequate shielding is available and the RNG is integrated on a common silicon substrate with other circuitry, substrate and power supply, and external interferences are the main concern in Smart Card applications. Therefore, a well-designed decorrelating algorithm is used as post processor to remove the effect of nonidealities.

The prototype RNG was fabricated in a $0.18\mu m$ CMOS process. The RNG macro cell area, excluding pads, was $220\mu m \times 116\mu m \approx 0.025mm^2$. A $3.3V$ supply was used for analog circuits and $1.8V$ was used for digital circuits. The power consumption was

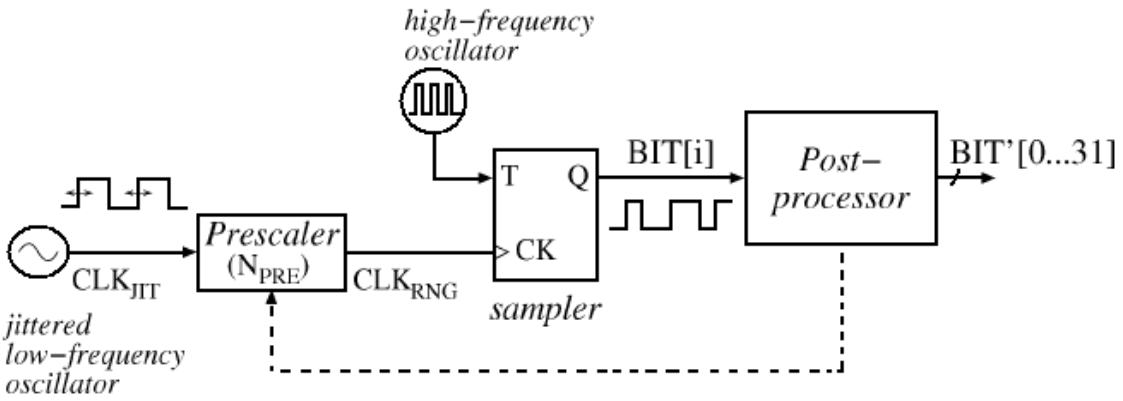

Figure 2.7. RNG architecture of [4].

$3.6mW$ when clocked at $10MHz$.

The results of the performed tests on the raw bit stream before the de-correlator were not successful from the NIST Test Suite of FIPS 140-1 [44] at $10MHz$ clock. On the other hand, processed data passed the tests for the clock frequencies up to $80MHz$ with reduced throughput by a factor of two.

2.5.6. A High-Speed Oscillator-Based Truly Random Number Source for Cryptographic Applications on a SmartCard IC

This is a fully digital implementation that employs CMOS standard-cell ring oscillators having a jitter-to-mean period ratio lower than 10^{-4} for a $0.18\mu m$ process. Therefore, a full-custom oscillator, which yields a standard deviation of about 10% of the period length, with an amplified noise source was used in the RNG of [4] to achieve faster bit rates. The clock period of the jittered oscillator is about $107ns$ and the jitter amount is around $9ns$. Furthermore, the jittered oscillator requires a $450ns$ start-up time. The high speed oscillator has been implemented with a 10-stage CMOS ring oscillator that typically oscillates at $1GHz$. Furthermore, in order to remove the biasing of the output bit sequence due to an unbalanced duty cycle from the ring oscillator, a T flip-flop (TFF) was used as a sampling circuit. A programmable

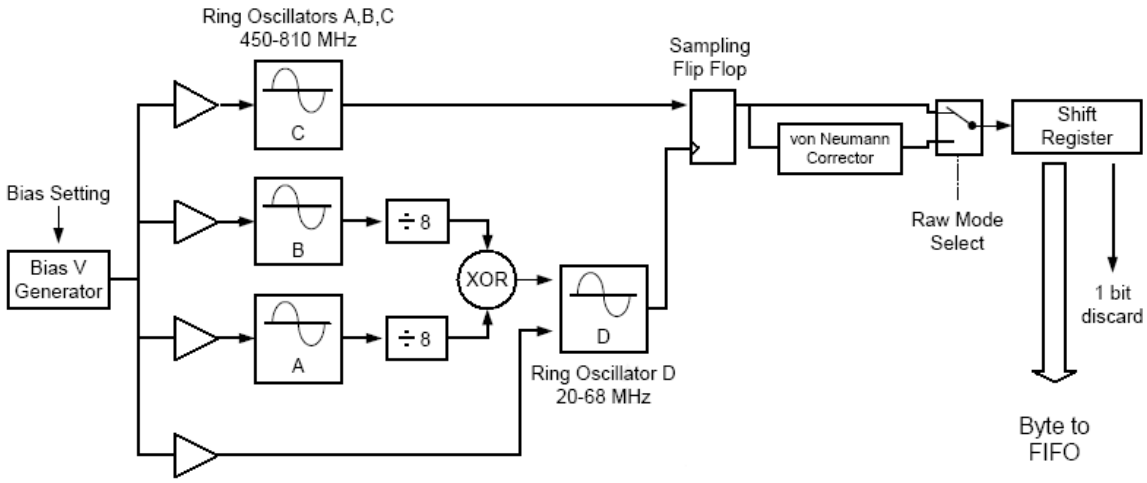

Figure 2.8. Nehemiah entropy source design.

prescaler is also present at the output the low frequency oscillator. Scaling factors from one to 128 are provided in order to experiment with different jitter to mean frequency ratios. The RNG architecture of [4] is presented in Figure 2.7.

The prototype RNG was fabricated in a $0.18\mu m$ CMOS process. The RNG macro cell area, excluding pads, was $184\mu m \times 86\mu m \approx 0.0016mm^2$. A $3.3V$ supply was used for analog circuits and $1.8V$ was used for digital circuits. The power consumption was $2.3mW$.

NIST Test Suite of FIPS 140-1 [44] is performed on raw data, which has shown very good random behavior by passing the statistical tests.

2.5.7. The Nehemiah RNG

A collection of freewheeling oscillators serves as an entropy source in [46]. A slow freewheeling oscillator with frequency $\sim 30MHz$ is utilized to sample the output of a fast freewheeling oscillator with frequency $\sim 600MHz$. The jitter of the slow oscillator is further increased by adjusting the oscillator's bias voltage with output from two additional fast oscillators. The Nehemiah entropy source design is shown in Figure 2.8.

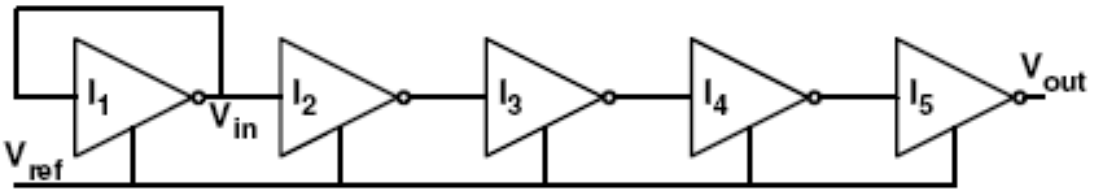

Figure 2.9. RNG schematic for UWB RFID.

Pairs of bits in the sampled output are passed through a von Neumann whitener, which reduces (but does not completely remove) single bit biases from the sampled data. Whitened bits are then accumulated in a FIFO containing four 8-byte buffers (32 bytes total). To save power, the raw source oscillators are automatically shut down when the FIFO is full.

There is no information whether randomness tests have been applied to Nehemiah RNG; however there is entropy estimation on raw and processed data. The raw bit generator (with the whitener disabled) was estimated to yield 0.78 to 0.99 bits of entropy per raw output bit. Operation with the whitener enabled is believed to provide 0.99 bits (typical) of entropy per output bit.

2.5.8. The RNG for RFID Tag

An ultra low power RNG for an ultra wide band (UWB) RFID is proposed in [47]. It amplifies noise by using a chain of identical inverters to generate a serial output stream of random bits. The schematic of RNG is demonstrated in Figure 2.9. This RNG generates a random signal through direct amplification of noise. Each of the inverters amplifies the signal of the previous stage until a clipped, digital signal is generated. Random bits are extracted from this signal. The transistor level implementation of a single inverter is depicted in Figure 2.10.

This circuit has been designed and fabricated with a $130nm$ CMOS process. The

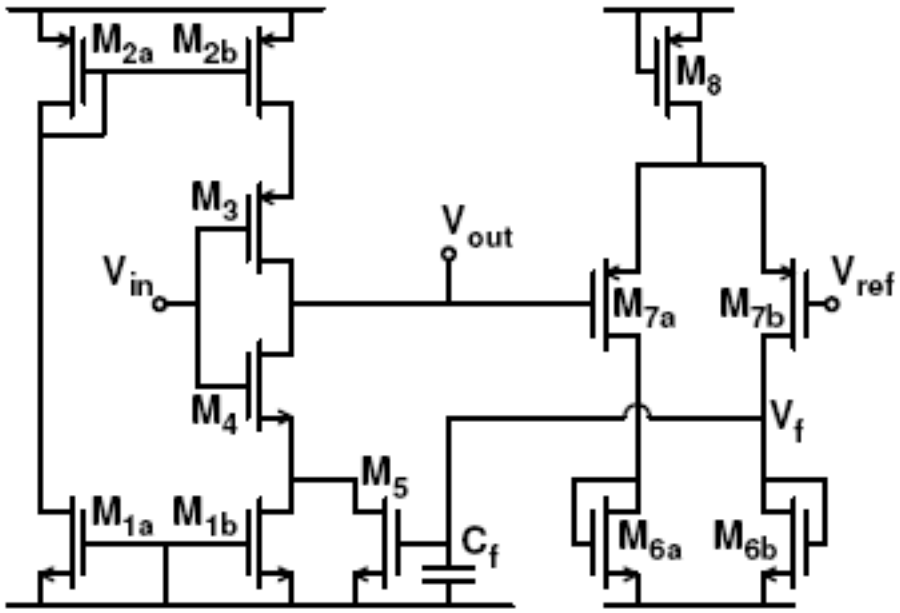

Figure 2.10. The transistor level implementation of a single inverter.

RNG occupies $290\mu m \times 850\mu m \approx 0.246mm^2$ of area and operates from a supply voltage of only $V_{DD} = 500mV$ and consumes a supply current of $I_{DD} = 1.3nA$. The observed frequency of the RNG output is $10MHz$; however, in order to produce random bits, the output is sampled with up to $25Hz$. The generated raw data can not pass the statistical tests of NIST. After subjecting the raw data to a 6-bit LFSR based post processor, the processed data can pass the NIST test.

2.5.9. FIGARO

Using Fibonacci ring oscillator (FIRO) and Galois ring oscillator (GARO) as a random number generation method is a novel approach, which is influenced by Fibonacci and Galois LFSRs where delay cells (D-type flip flops) are replaced by the inverters [48]. FIRO-based and GARO-based circuit structures are given in Figure 2.11 and Figure 2.12, FIRO and GARO are defined by an associated feedback polynomial. The feedback connections are chosen to make sure that there are no fixed points in the corresponding state transition function. Therefore, the $f(x)$ polynomial should be

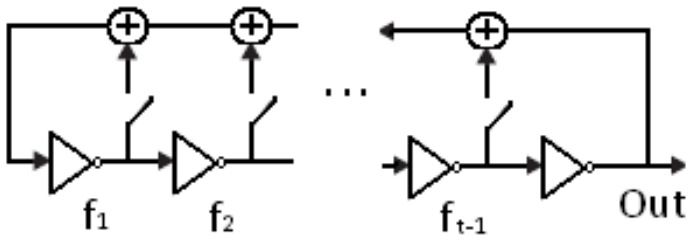

Figure 2.11. Fibonacci ring oscillator.

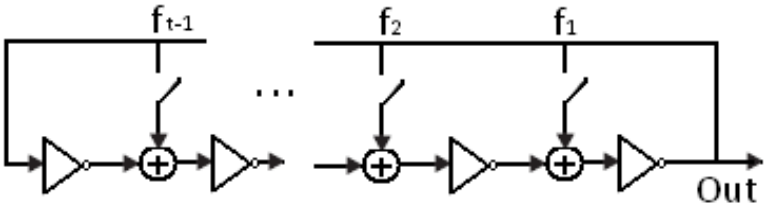

Figure 2.12. Galois ring oscillator.

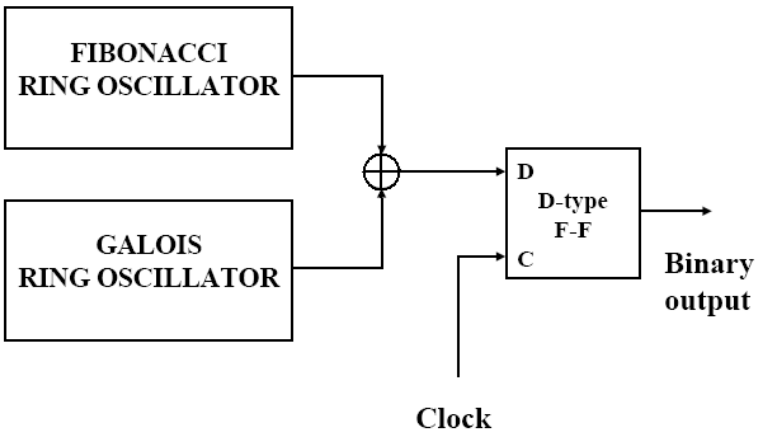

Figure 2.13. Combination of Firo and Garo.

chosen as $f(x) = (1 + x)h(x)$, where $h(x)$ is a primitive polynomial and should meet the condition of $h(1) = 1$. An FPGA implementation of the proposed architecture for RNG application is reported in [36].

The FIROs and GAROs were experimentally tested by using a Xilinx Spartan-3 Starter Kit board based on the Xilinx FPGA XC3S200-4FT256C. Dichlt *et al.* pro-

posed to use FIRO and GARO both in continuous mode and restart mode. Statistically independent bits can be achieved due to restarting, since there will be no residual long-term statistical dependencies. In restart mode, FIRO is sampled with T flip-flop, where it has a throughput of 6.25 $Mbps$; on the other hand, it has a throughput of 7.14 $Mbps$ when it is sampled D flip-flop. They can also be used in a combined FIGARO configuration, as shown in Figure 2.13. When it is sampled both directly and with the intermediate T flip-flop, the raw data rate is doubled to $14.28Mbps$. There is no information about the statistical test results.

2.6. RO-based RNG

Before Sunar *et al.*, RNG designs were validated by running the DIEHARD [31] or NIST Test Suite of FIPS 140-1 [44]. Typically, the optimal sampling frequency was determined by trial and error: the sampling frequency was decreased until the output sequence starts to pass the NIST or DIEHARD tests. Sunar *et al.* was inspired by these observations and developed a model that makes it possible to estimate the amount of entropy, and consequently the design parameters, such as the number of inverters in a chain and the number of rings in the circuit, before starting to realize the proposed digital RNG design based on ring oscillators. A sample design consists of 114 rings with 13 inverters and was developed on a Xilinx Virtex II FPGA.

A resilient function is recommended in the same study to eliminate the non-random components in the post-processing unit. A simple technique is proposed to obtain suitable resilient functions from error correcting codes (linear cyclic code). The resilient function is based on a cyclic [256, 16,113]-code [5]. This post processing has a compression factor of 256/16 = 16. The output after post processing has almost $2.5Mbps$ bit rate and a sampling frequency of around $40MHz$.

RO-based RNGs became very popular for cryptographic applications after their proposal. The main reason of this popularity is the low implementation complexity in both ASIC and FPGA styles and the good quality of the produced random bits on the contrary of the common opinion about the low entropy level of digital gate based

RNGs.

Since then, this RNG structure has been scrutinized in the literature. An FPGA implementation of that model is reported in [49]. The ROs in this implementation contain only three inverters in a ring. Furthermore, 210 ring oscillators are used to build the RNG. This RNG features a throughput of more than $2Mbps$. There is no exact information about the randomness tests, although the authors have declared that they have checked their design with standard tests (NIST and DIEHARD) and confirmed that the statistical properties of the produced random numbers are fine.

The first ASIC implementation of [5] was presented in [16], while the first ASIC implementation of [49] was presented in [17]. Fulfilled test results were achieved from NIST 140-2 test suite after a von Neumann corrector with $16.5Mbps$ and $18.5Mbps$ throughput, respectively. On the other hand, Dichtl *et al.* pointed out some serious criticisms in [36]. While Sunar was responding to these criticisms in [50], some enhancements were proposed to Sunar's RNG by taking the Dichtl's criticism into account in [51, 52]. For example, the FPGA implementation was enhanced by using D-type Flip Flop just before the XOR tree for every ring oscillator in [51].

From a security point of view, researchers were curious about strength and robustness of RO-based RNGs. For this reason, some studies were conducted in [53–56] to examine possible security issues. In [53], frequency injection was applied to RO-based RNG over the power supply. This injection was able to lock the frequencies of ROs and reduce the entropy. In [54], an active electromagnetic attack was proposed. The signal was injected by the RF channel and thus the RNG was forced to produce biased sequences. In order to increase the effectiveness of the proposed attack in [54], electromagnetic emission of the RNG was observed and some useful information about the RNG was derived from the RF leakage in [55]. On the other side, a detection mechanism for active nonintrusive attacks was proposed with randomized bit-cell concept in [56].

The underlying entropy source has become another interest area for researchers.

For example, jitter was observed and analyzed in [57]. An improved version against attacks was proposed in [58] based on the analysis in [57]. In [59], randomness was broken down into its components as true and pseudo randomness. In order to maximize the jitter amount in ROs, a study was proposed in [21]. In order to increase the randomness and entropy of ROs, [20] recommended forcing ROs to operate in weak inversion region by using the advantage of higher noise in this operating region.

2.6.1. Model of RO-based RNG

According to the model in [5], an interval, I is divided into k equal subintervals which is called an urn. If a transition occurs during an urn, this is called filling the urn. It is also mentioned in the same study that two ring oscillators with an equal number of inverters will exhibit a great deal of overlap in their transition zones. However, non-deterministic elements, such as initial delay and phase drift, make it highly likely that two such ring oscillators will have no overlap in their contributions to the jitter.

A detailed analysis of theoretical entropy estimation, which gives the name of the paper as "Provably Secure" due to fill rates of urns and Coupon Collector's Problem, is given in [5]. The minimum number of ring oscillators $r = M(N, f, p)$ can be determined via the theorem, where N stands for number of urns, $0 < f < 1$ stands for fill rate of an urn and $0 < p < 1$ stands for level of confidence. In the case of $f = 1$, this becomes to the Coupon Collector's Problem. However, achieving a fully filled urn with a reasonable p is very hard for practical implementation of jittered-ring oscillator application, hence p should be taken very small. Instead of this, the authors prefer to keep the confidence level of p close to one while decreasing the fill rate of f. The problem of $M(N, f, p)$, which is not well-studied, is converted to $P(N, r, f)$, which is the probability that at least fN out of N urns are filled with exactly r ring oscillators. The relationship between $M(N, f, p)$ and $P(N, r, f)$ is given as

$$\mathbf{M(N, f, p) = min\{r : P(N, r, f)\}}. \tag{2.1}$$

Table 2.1 presents an example of this estimation, shown in(2.1), for $N = 100$ urns.

Table 2.1. An example estimation for *N=100* urns, the number r of ring oscillators necessary to fill at least $f \times N$ of the urns with probability at least p.

p	**f**				
0.5	0.5	0.6	0.7	0.8	0.9
0.6	70	93	122	163	234
0.7	72	95	125	167	241
0.8	75	100	131	177	258
0.9	78	104	137	185	271
0.99	86	114	151	205	307

Further analysis can be found in [5].

2.6.2. Design Details of RO-based RNG

Figure 2.14 presents the circuit structure of the sample design given in [5]. This sample design has 114 ring oscillators and each RO has 13 inverters. There is a brief information in the following subsections about the basic circuit blocks used in an RO-based RNG.

2.6.2.1. Ring Oscillators. Ring oscillators are among the most frequently used and fabricated circuits in the VLSI industry. Both analog and digital designers are familiar with ROs,which are the most common type of oscillators. ROs consist of delay stages that are connected to each other sequentially and forming a loop with a feedback connection. The delay stages are connected to each other in a ring configuration with a feedback shown as in Figure 2.15. The output of any of these inverters is a square wave signal with a period that is actually the sum of the delay of each inverter. The most well-known delay stages are single-ended (or called as inverter based) and differential delay stages illustrated in Figure 2.16.

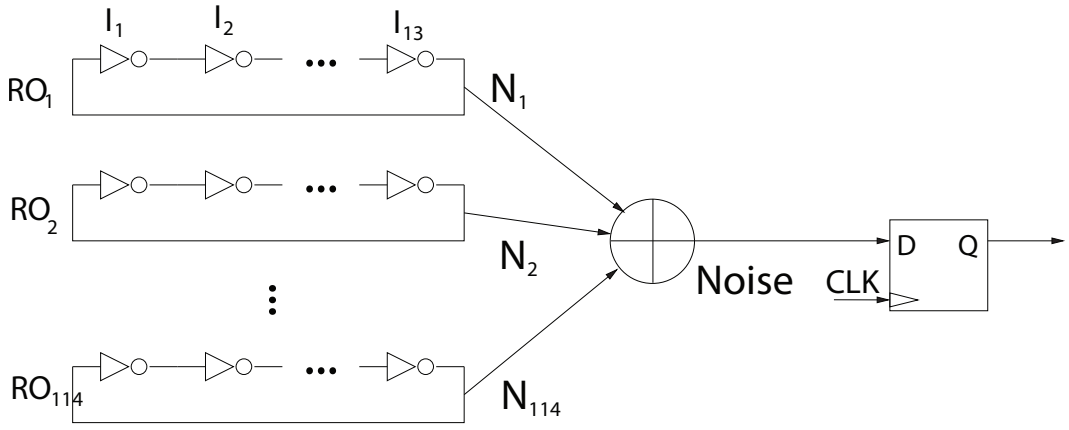

Figure 2.14. Circuit structure based on ring oscillators.

ROs can only oscillate when Barkhausen's oscillation criterion is met. According to this criterion, a phase shift of 2π and unity voltage gain at the oscillation frequency is necessary. In order to meet Barkhausen's oscillation criterion, an inverter-based RO should consist of an odd number of inverter stages. On the other hand, DRO should consist of more than one delay stage with even or odd count; however, it is difficult to meet Barkhausen's oscillation criterion of 2π phase shift with only two delay stages. Therefore, generally more than two delay stages are used in the designs.

2.6.2.2. XOR Tree Technique. XOR Tree Technique technique, shown in Figure 2.17, is used to combine oscillator rings by XOR gates. This technique is used as one way of increasing entropy in random number generation. Following this direction, new binary data is generated according to

$$\mathbf{N_{(xor)i}} = \mathbf{N_1} \oplus \mathbf{N_2}. \tag{2.2}$$

The mean value, ψ, of the binary sequence $\mathbf{N_{(xor)i}}$ can be calculated using t

$$\psi = \frac{\mathbf{1}}{\mathbf{2}} - \mathbf{2}(\mu - \frac{\mathbf{1}}{\mathbf{2}})(\nu - \frac{\mathbf{1}}{\mathbf{2}}), \tag{2.3}$$

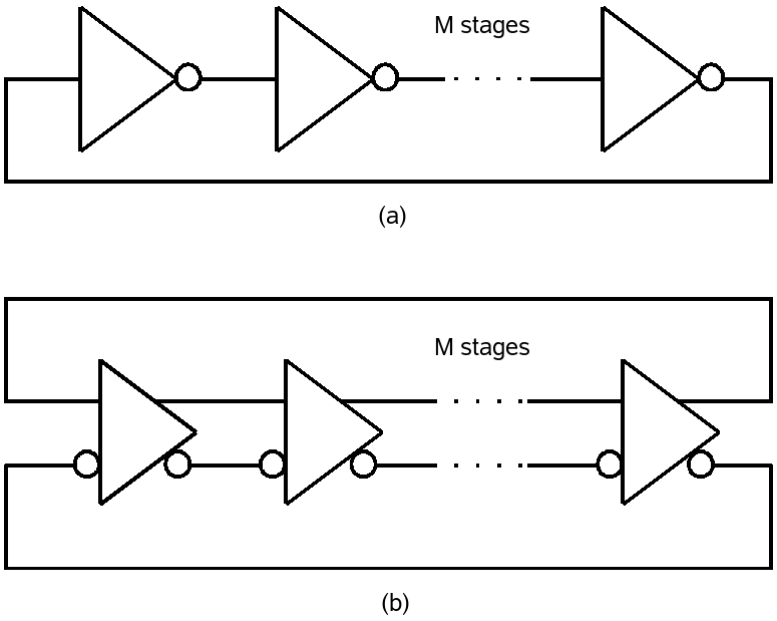

Figure 2.15. (a)Inverter-based ring oscillator (b)Differential ring oscillator.

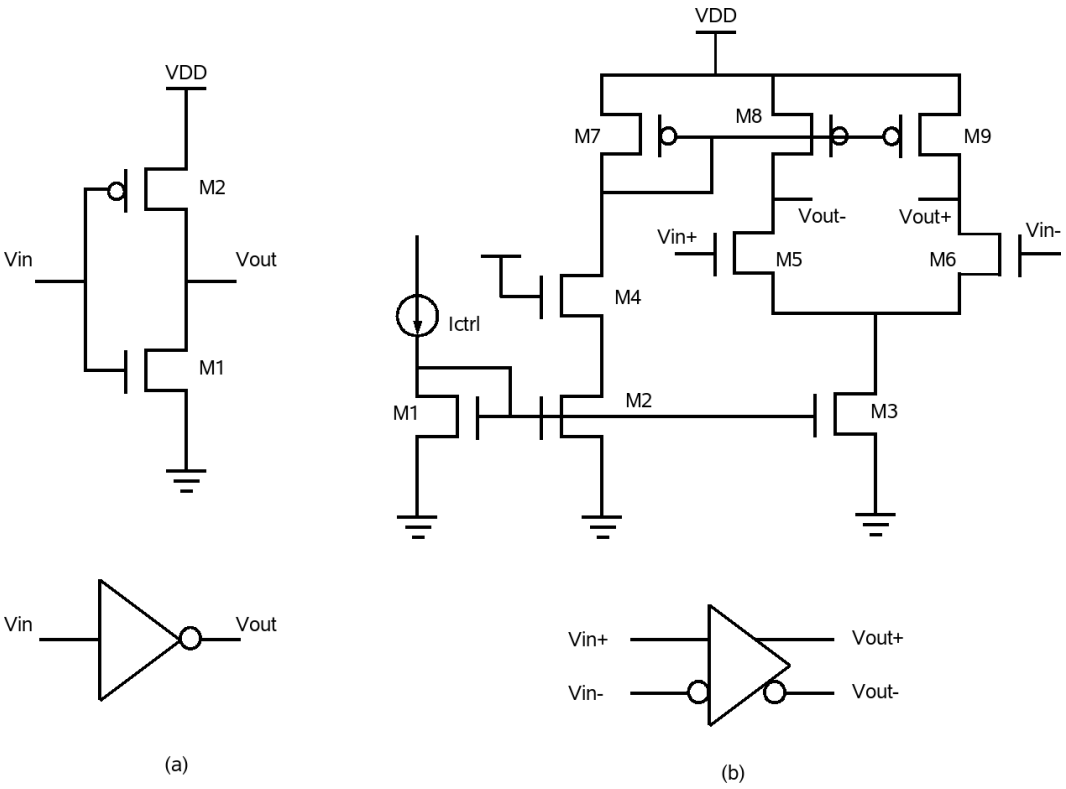

Figure 2.16. (a)Inverter-based delay cell (b)Differential delay cell.

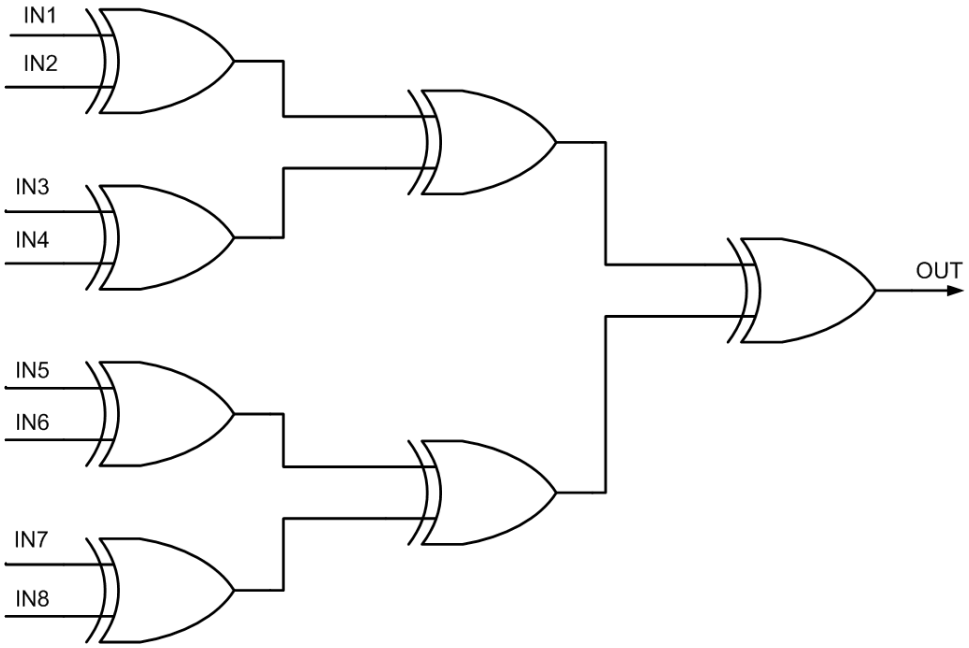

Figure 2.17. XOR tree technique.

where the mean value of $\mathbf{N_1}$ is μ and the mean value of $\mathbf{N_2}$ is ν. Thus, when μ and ν approach $\frac{1}{2}$, ψ becomes very close to $\frac{1}{2}$.

However, there is a potential problem with the XOR method : a small amount of correlation between the inputs of XOR will add significant bias to the output [26]. When correlation is involved in the equation, it becomes

$$\psi = \frac{\mathbf{1}}{\mathbf{2}} - \mathbf{2}(\mu - \frac{\mathbf{1}}{\mathbf{2}})(\nu - \frac{\mathbf{1}}{\mathbf{2}}) - \frac{\mathbf{1}}{\mathbf{2}}\rho, \tag{2.4}$$

where ρ stands for the correlation coefficient.

At this point, correlation significance becomes more important rather than correlation itself, since the number of bits in a bitstream is another parameter that should be taken into account. Considering this, correlation coefficient should be calculated first. Then, with the help of the correlation significance table [60], correlation significance should be determined between at least two of the adjacent rings to ensure the XOR tree technique can be used. If the correlation significance is less than 0.0073

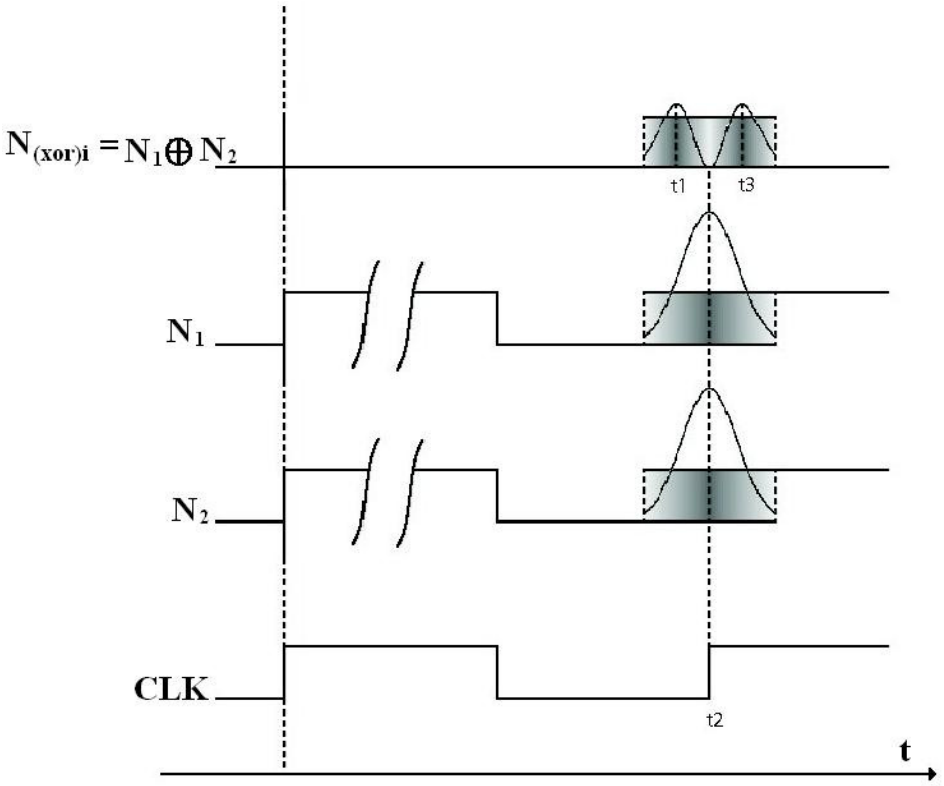

Figure 2.18. Probability distribution of two Ni after XOR function without phase drift.

for 32,000 bit sequences, then these two rings can be determined as uncorrelated, and hence, XORing technique can be applied.

The probability distribution of two signals, having Gaussian distribution, are investigated when they are XORed. As illustrated in Figure 2.18, the distribution can be predicted. It is clear from the figure that XORing improves the bias significantly in the jittered area, because it increases the probability of transition numbers, which yields an increase in the number of filled urns. In Figure 2.19, the phase drift between the input signals of an XOR gate is considered. In the first figure, there are two overlapped Gaussian signals. Later on, these two signals are separated from each other with a phase drift. As seen in the figure, the probability of transitions increases with increasing phase drift.

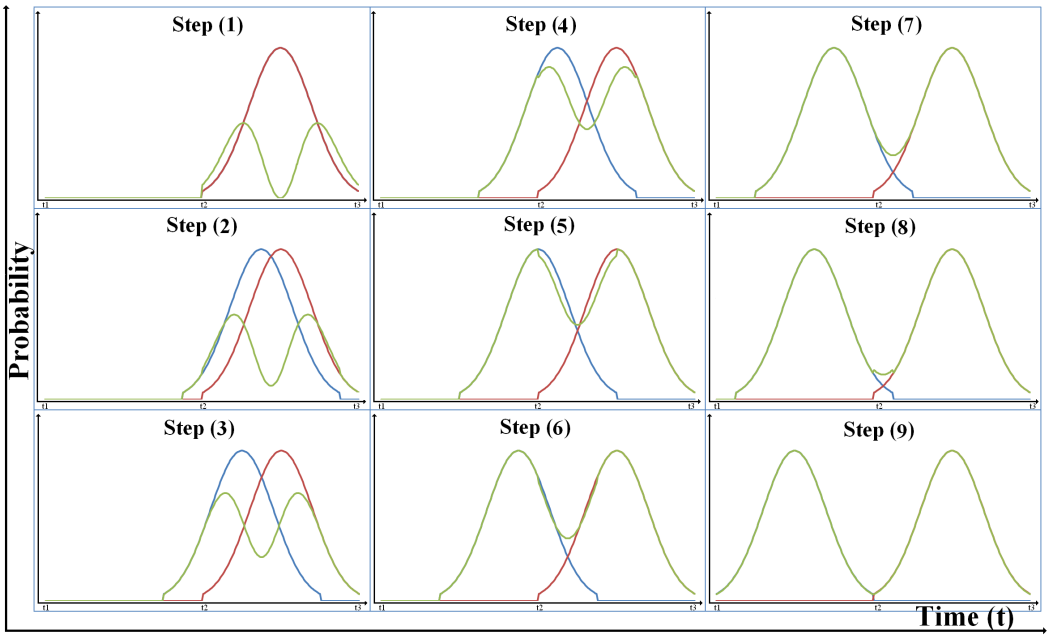

Figure 2.19. Probability distribution of after XOR gate with phase shift.

2.6.3. Debates on RO-based RNG

A few years ago, there was a hot debate about the true randomness of [5] between Sunar *et al.* and Dichtl *et al.* Dichlt mentioned in [36] that security proofs in [5] are based on highly unrealistic assumptions; furthermore, the presented test results in [49] may be due to pseudo randomness.

In [36], Dichtl has summed up his criticisms to [5] under four headings. As an answer to these criticisms, Sunar published his response in [50]. We will briefly touch on these criticisms and the related responses in this subsection. Moreover, we will add our comments and describe our experiences when necessary.

(i) Dichtl mentioned that the urn model stands for an unrealistic assumption on jitter, since it is assumed that ROs act as perfect built-in clocks and that jitter only occurs around the transition. In fact, in time, there will be phase jitter, and transitions will be shifted from the original point.

On the other hand, Sunar responded by stating that, for the independence of the samples taken from the urns, which may be filled by the same ring, incremental component of the jitter is considered. Sunar emphasizes that this criticism is a misunderstanding of the jitter model used in [5].

Actually, we think that this omission is done for the sake of simplicity; however, it should be mentioned in [5] by Sunar.

(ii) In the second criticism of Dichtl, he claimed that in the urn model, Sunar ignored the coupling between ROs and assumed that the transitions in individual RO signals are uniformly distributed among the chosen 100 time slots the period T is divided into. He implemented 3-inverter connected ring oscillators on the same FPGA as an example of how they are interacting.

However, Sunar already mentioned that ring oscillators should be placed properly and not close to each other in order to decrease or eliminate the interaction. Moreover, he said that I/O paths may also introduce correlation if they share common signals, and the bandwidth of I/O paths may be limited.

A successful work has been achieved in terms of reducing the correlation in the ASIC implementations of RO-based RNG in [16] and [17]. It was emphasized in these works that placement of ring oscillators and routings were primary concern of the designers. Despite the cost of increasing area, all the ring oscillators were placed into different wells, and routings were carefully done to prevent long running parallel wires which may cause correlation.

(iii) Unrealistic speed of transitions at the output of XOR gate is the third criticism of Dichtl. In the case of 114 ROs of length 13, 8.77 transitions should occur per gate delay which does not seem feasible. 210 ROs of length three is an even more severe case.

Sunar's response to this criticism is sampling at a more reasonable rate, which sounds rational to us. He also mentioned about the narrow signal rejection which was studied in [52]. They propose a new model for the rejection to effect pulses as narrow as the urn width. Figure six in [52] shows the worst case reduction in the fill rate due to narrow pulse rejection. Furthermore, they claimed that some transitions may come up to the same point and cancel each other. If an odd number of transition occur in an urn, the urn is filled.

(iv) As a fourth criticism, Dichtl claims that set-up and hold time violations of flip-flop may corrupt the security proof contention.

Sunar tells that this is something very common in many RNG designs. Also this criticism is irrelevant to the model. This should also be considered when designing a digital circuit that will reliably generate deterministic bits.

It is exhibited in [51] that [5] shows true randomness characteristic with a restart test as in [36]. A restart test was repeated for 1,000 times and standard deviation for all traces was calculated. It was seen that the traces deviate from each other which is the proof of true randomness. Moreover, the authors proposed another work with an improved design [52], which has better performance, and is robust against the weaknesses which were noticed by the authors with the original design.

2.7. Conclusion

In this chapter, RNG types, requirements, and design techniques are briefly summarized. Moreover, some of the well-known RNG designs are discussed. In Table 2.2, comparison of these RNGs as well as summary is given. RO-based RNG with some design details and primary blocks is also expressed.

Table 2.2. Comparison table.

	[28]	**[45]**	**[2]**	**[35]**	**[3]**	**[4]**	**[46]**	**[5]**	**[36]**	**[47]**
Organization	Intel Corp.	Intel Corp.	Georgia Tech.	Infineon	Gemplus Infineon Uni. of Rome	Gemplus Infineon Uni. of Rome	Via Tech.	Worcester Polytechnic University	Siemens Telecom Italia	Katholieke Universiteit Leuven
Year	1999	2012	2000	2000	2003	2003	2003	2007	2007	2010
Technology	N/A	$45nm$ CMOS	2μm CMOS 3V	N/A	0.18μm CMOS 1.8V	0.18μm CMOS 3.3V	N/A	Xilinx FPGA XC2VP30	Xilinx FPGA XC3S200	$130nm$ CMOS
Speed	N/A	2.4 GHz	$1.4MHz$	N/A	$80MHz$	$10MHz$	20-68 MHz	$40MHz$	$14.8MHz$	$10MHz$
Area	N/A	$4004\mu m^2$	$1.5mm^2$	N/A	$0.025mm^2$	$0.016mm^2$	N/A	973 slices	N/A	$0.246mm^2$
Power Consumption	N/A	$7mW$ @2.4GHz	$3.9mW$ @1MHz	N/A	$3.6mW$ @10MHz	$2.3mW$	N/A	N/A	N/A	$0.65nW$
Entropy Source	Thermal Noise	Metastability	Thermal Noise	Phase Jitter	Thermal Noise	Thermal Noise	Phase Jitter	Phase Jitter	Phase Jitter	Noise Amplification
Raw Data Pass?	No	N/A	Almost	No	No	Yes	N/A	N/A	N/A	No
Processed Data Pass?	Yes	Yes	N/A	Almost	Yes	Yes	N/A	N/A	N/A	Yes
Final Throughput	$75kbps$	$2.4Gbps$	$1.4Mbps$	N/A	$40Mbps$	$10Mbps$	4-9 $Mbps$	2.5 $Mbps$	$14.28Mbps$	$25bps$

3. RANDOMNESS SOURCE

Uncertainties at zero crossings observed at the output of the RO occur in a random manner, since they are the result of thermal and shot noises in the circuit itself. In fact, it is very difficult to obtain perfect square waves that do not have variations in zero crossings or so-called jitter. In contrast to low jitter oscillators, jitter is desirable in random number generation, since it is the source of randomness.

Recently, the utilization of jitter of CMOS ROs as a randomness source has become more common in security applications such as RNG [5,43,61] and PUF (Physically Unclonable Function) [62, 63]. The amount of jitter in a period is a significant metric in RNG applications, as the ratio of jitter to period determines the randomness in bits. Since oscillation jitter is used to provide randomness for RNG, maximizing the phase noise and jitter in ROs will also increase the entropy. In [5], Sunar assumed that jitter amount in a period is around 2%, and then all the calculations were done based on this assumption. However, it is mentioned in [4] that jitter to mean-period ratio of a ring oscillator with 0.18μm standard CMOS process is lower than 10^{-4}. Actually, the reason of this confusion is that accumulated long-term jitter is considered in [5] whereas short-term jitter is mentioned in [4].

Despite the fact that the only entropy source of an oscillator is jitter, there is limited research providing a detailed phase noise and jitter analysis for ring oscillator-based RNGs. A sample study is performed in [21], which examines the suitability of existing ring oscillators. Although there is some research in the literature on how to decrease jitter and get pure clock signals [64, 65], studies like [21] are not enough.

3.1. Jitter as a Randomness Source

Before discussing the randomness issues and randomness in RNGs, the jitter phenomenon should be well understood. In order to achieve this, the following subsections will provide some definitions about jitter as well as explore the source and types

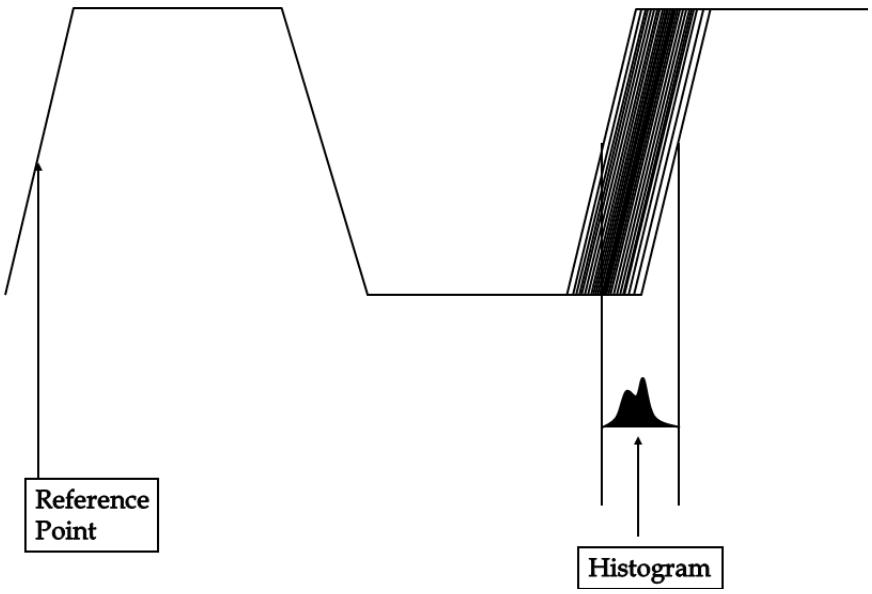

Figure 3.1. Jittered signal.

of the jitter.

3.1.1. Jitter Definition

In an ideal clock, successive cycles of a noise free waveform will have exactly the same period. However, in an actual clock, noise sources will cause variation in the clock period. This variation is called jitter and is illustrated in Figure 3.1.

In fact, jitter has two main components: random and deterministic jitter [66]. Random jitter is characterized with a Gaussian distribution. This Gaussian distribution will have a mean value, a standard deviation, a peak-to-peak value, and population measures, which are illustrated in Figure 3.2 and defined as follows:

- Mean Value is the average value of the measured clock periods.
- Standard Deviation is the average amount by which a measurement deviates from its mean value.
- Maximum, Minimum Values refer to values actually observed during a measurement interval. Maximum deviations from the mean value in both sides give the

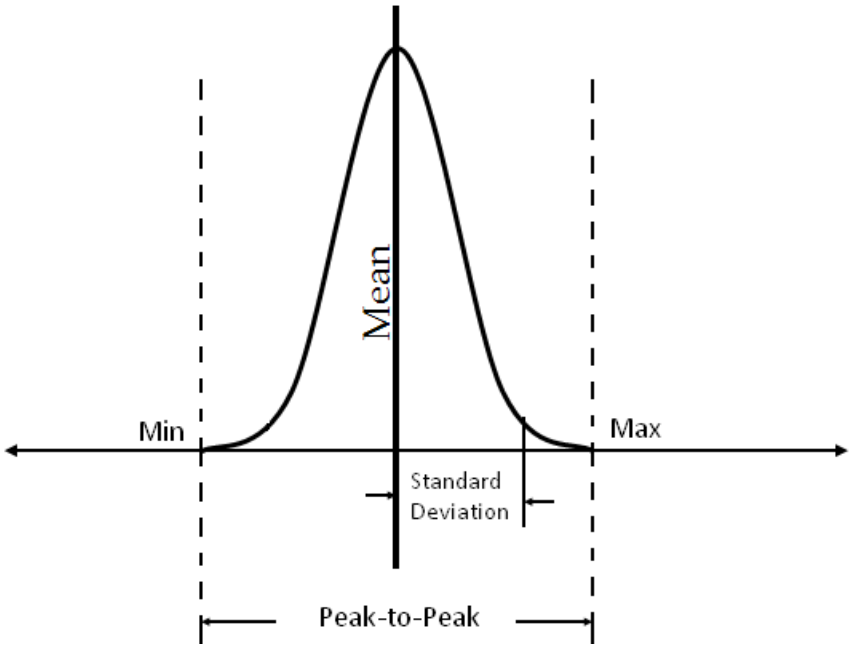

Figure 3.2. Gaussian Distribution.

maximum and minimum values.

- Peak-to-Peak Value is the amplitude of jitter which only uses the extremes of the measurements.
- The population is the number of individual observations included in a statistical data set. A high population for random processes gives greater confidence that the measurement results are repeatable.

For a deterministic signal, these values may equal the estimated values even after a relatively short measurement interval. However, for a random signal, there is theoretically no limit on the max and min values. The observed peak-to-peak value will grow over time. For this reason, the peak-to-peak values should be used in conjunction with the population size and some knowledge of the type of the distribution [67].

Standard deviation is used to predict the occurrence of deviation from the mean. Table 3.1 [68] shows the amount of standard deviations with the corresponding percentage of all the measurements contained in that amount standard deviation.

Table 3.1. Standard deviation amount versus window coverage of all measurements.

Standard Deviation	**Window Coverage Percentage**
$\pm 1\sigma$	68.26%
$\pm 2\sigma$	95.4%
$\pm 3\sigma$	99.73%
$\pm 4\sigma$	99.99366%
$\pm 6\sigma$	$(100 - 1.973 \times 10^{-7}\%$
$\pm 7\sigma$	$(100 - 1 \times 10^{-12}\%$
$\pm 8\sigma$	$(100 - 1.244 \times 10^{-13}\%$
$\pm 10\sigma$	$(100 - 1.973 \times 10^{-21}\%$

On the other hand, existence of deterministic part of jitter changes the pure Gaussian distribution form and multiple peaks can be seen in the distribution. Power supply ripple, which modulates the clock or the power, may cause deterministic jitter. The combination of random and deterministic jitter is called total jitter.

3.1.2. Sources of Timing Jitter

It is possible to classify the sources of timing into two groups: intrinsic and non-intrinsic [66]. It is known that the reason of the intrinsic noise source is the electron and hole activities inside the semiconductor devices, where non-intrinsic noise sources are caused by design related issues and a careful design may eliminate or minimize this jitter component on the other hand. Intrinsic and non-intrinsic jitter sources are categorized as in Figure 3.3. Brief definitions of all jitter sources will be given in the coming subsections.

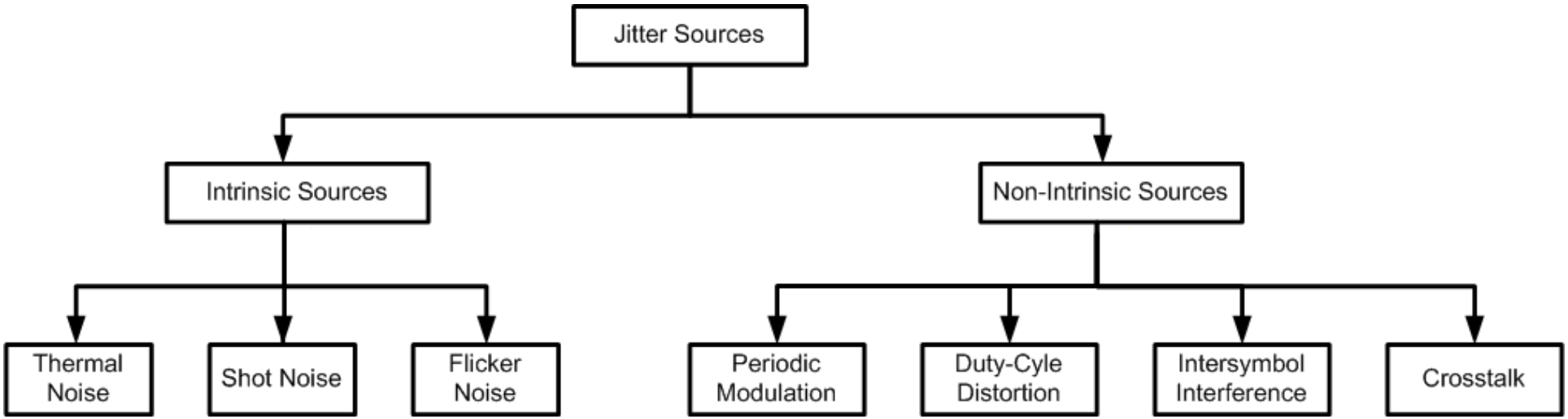

Figure 3.3. Jitter sources.

3.1.2.1. Intrinsic Noise Sources. Intrinsic noise is caused by the movements of electrons and holes in electronic, optical, or semiconductor devices. It is not possible to remove this noise source totally; however there are available noise minimization techniques. Thermal, shot, and flicker noise are the main sources of intrinsic noise [66].

Thermal Noise: Thermal noise is produced by the the random motion of charge carriers under the conditions of thermal equilibrium. The power spectrum density (PSD) of the thermal noise is white and additionally proportional to its temperature.

Shot Noise: Individual quantized carrier flow (current) in a potential barrier causes shot noise with a random generation time or spatial distribution. In other words, shot noise is basically due to random flow fluctuation. Shot noise is directly proportional to DC bias current, as well as the charge of the carrier. The PSD of the shot noise is also white.

Flicker Noise: Flicker noise is defined as a phenomenon which is found to have a noise power spectrum inversely proportional to the frequency over a wide range of frequencies. DC current is required to produce flicker noise. There is no universal theory as accepted to be explaining the cause and mechanism of flicker noise, unlike the causes of thermal and shot noise. It can be said that, the quantitative study of flicker noise is mostly empirical. It has been found that the PSD of flicker noise is proportional to 1/f. Because of this reason, flicker noise is called also as 1/f noise.

3.1.2.2. Non-Intrinsic Noise Sources. Non-intrinsic jitter has deviations related to design. To put it in another way, those types of jitter are controllable or fixable with appropriate design improvements. Commonly encountered non-ideal design-related noise and jitter include periodic modulation (phase, amplitude, and frequency), duty cycle distortion (DCD), inter symbol interference (ISI), crosstalk, undesired interference such as electromagnetic interference (EMI) as a result of radiation, and reflection caused by unmatched media [66].

Periodic Noise and Jitter: Periodic noise or jitter is a type of signal which repeats every time period. Various modulation mechanisms may cause periodic jitter, such as amplitude modulation (AM), frequency modulation (FM), and phase modulation (PM). It is evident that a periodic amplitude noise causes period timing jitter, with the amplitude inversely proportional to the slope or slew rate of the edge transition. Period noise/jitter can be caused by switching power supply, spread-spectrum clock (SSC), and period EMI sources in the electronics environment.

Duty Cycle Distortion: DCD is defined as the deviation of the duty cycle from its normal value. A duty cycle is the ratio of pulse width to its period for a clock signal. DCD may be caused by pulse width deviation, period deviation, or both. Furthermore, pulse width deviation can be caused by the deviation of reference signal level. Additionally, it may be caused by the propagation delay if the clock is formed from rising and falling edges of two half-rate clocks and those two half-rate clocks undergo different propagation delays.

Inter Symbol Interference: Although ISI is related to data signal, a clock signal does not have ISI by definition. Similar to the clock signal, a data signal is known as a generic digital signal form that does not have to have an edge transition in every unit interval or bit period. Unsimilar to the clock signal, the data signal can be kept at the same amplitude level for many unit intervals without an edge transition. In a lossy medium, the previous bits can cause errors both in transition timing and amplitude level from the ideal values. Each transition has a finite charge or discharge time, due to the capacitive effect. If the transition happens such that the next transition occurs when the previous transition has not reached the designated level, deviation of both time and level occurs for the current bit.

Crosstalk: Crosstalk is known as an interference phenomenon. Crosstalk is generally involved in a parallel channel system in which signals are propagated concurrently and affect each other. Capacitive coupling is the dominant mechanism for the integrated circuits where the geometry and space between interconnects is relatively small. When a signal transition happens in one channel, some of its energy leaks to the adja-

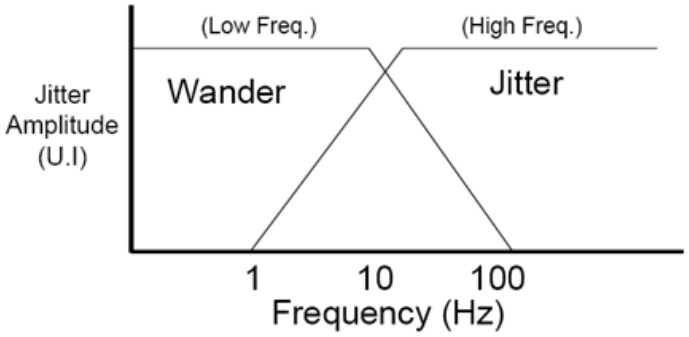

Figure 3.4. Wander vs jitter.

cent channel through charge flow due to capacitive coupling, resulting the signal level in that channel to fluctuate.

3.1.3. Jitter Types

Jitter is a generic term which needs to be defined. There are many definitions about jitter according to the type of the application. For example, jitter is specified as period or cycle-to-cycle jitter in digital data path synchronization applications. In other type, communication applications, it is specified as root mean square (RMS) jitter. In RF applications, it is specified as phase noise. In some cases different names are used even for the same definitions.

First thing to do is clarifying and distinguishing wander from jitter. Timing variations that occur slowly are called wander. Jitter describes the timing variations that occur more rapidly. The threshold between wander and jitter, illustrated in Figure 3.4, is defined to be $10Hz$ according to the International Telecommunication Union (ITU).

The ways in which jitter may be measured on a single waveform are period jitter, cycle-cycle jitter, time interval error (TIE), and long term jitter. It is important to understand how these measurements relate to each other and what they reveal [67]. Figure 3.5, represents all these jitter types in a single diagram.

Timing Jitter: Timing jitter measures the earliness or lateness of the transitions of a signal at any given time in its history. Timing jitter is defined as the difference in time between the actual occurrence of a transition and the time it should have occurred referred to an ideal clock. Timing jitter is commonly referred to as time-interval errors in the transition times of signal [69].

Time Interval Error (TIE): It measures how far each active edge of the clock varies from its ideal position. Some sample TIEs are shown in Figure 3.5 by the measurements from TIE1 through TIE4. For this measurement to be performed, the ideal edges must be known or estimated. For this reason, it is difficult to observe TIE directly with an oscilloscope, unless some means of clock recovery or post-processing is available [67].

In sampling oscilloscopes, and in equivalent time modes of real-time oscilloscopes, generally a clock recovered from the signal under test is the reference clock. A circuit monitors the signal and locks to the edges it finds. It is common that it is a PLL, a phase interpolator, an oversampling circuit, or even calculated from a real-time record of edge crossings. The measurement tool measures the time difference between the reference edge and the signal edge. After first subtracting the ideal clock period from each measured period, it is possible to obtain the TIE also by integrating the period jitter.

After measuring a significant number of time intervals and their related timing errors, it is possible to compute the standard deviation and peak values. This statistical information is the TIE jitter. When the reference is built by summing up the periods according to the amount of the first period, the mean TIE should be zero and the standard deviation and peak values are of interest. When the reference is calculated to emulate a PLL or is an electrically recovered clock, the mean value is rarely zero, but should be close to zero. The importance of TIE comes from showing the cumulative effect that even a small amount of period jitter can have over time. From the record that maintains a record of error versus time, accumulated phase error measurements become possible.

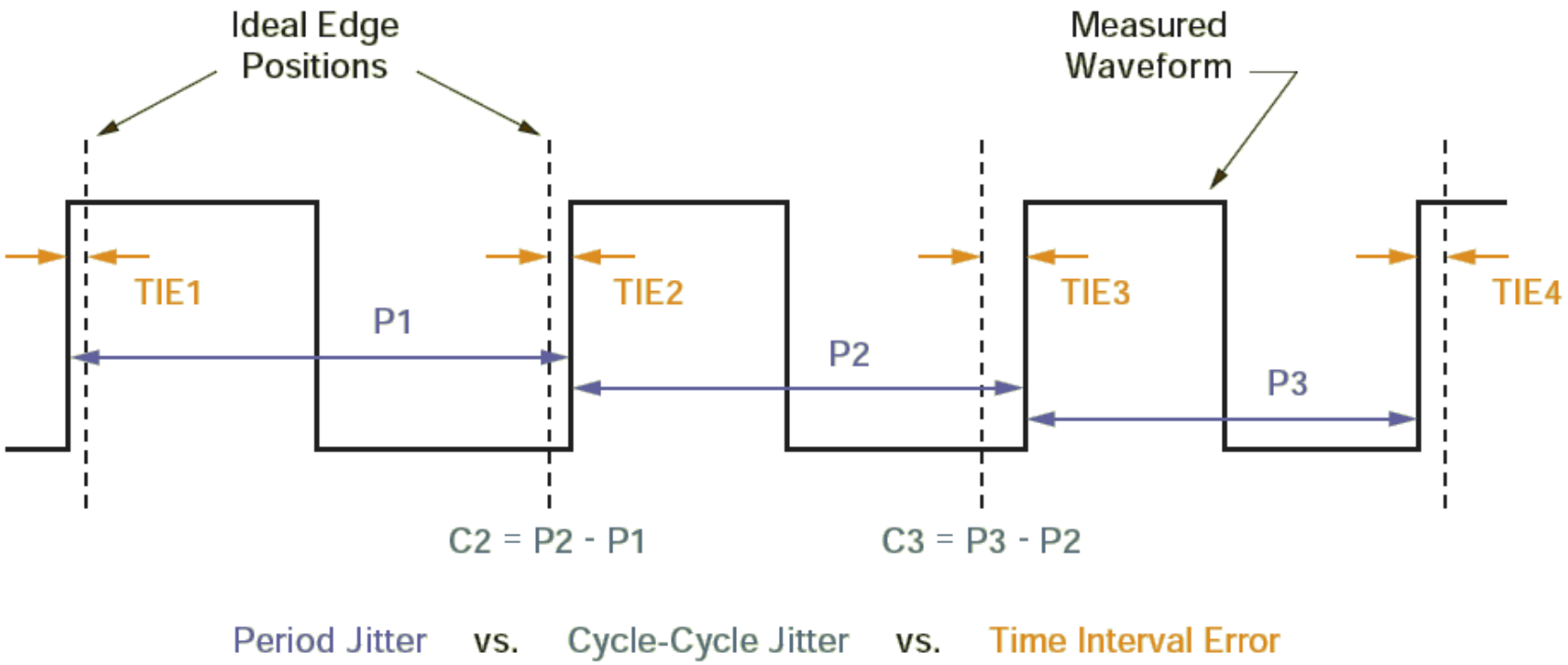

Figure 3.5. Jitter types.

Period Jitter: The period jitter, indicated by the measurements P1, P2, and P3 in Figure 3.5, basically measures the period of each clock cycle in the waveform. In other means, period jitter refers to the distribution of periods. In a waveform, the difference of the $(k+1)^{th}$ cycle timing jitter from the $(k)^{th}$ cycle timing jitter gives the period jitter of the $(k)^{th}$ cycle; therefore, it can be said that period jitter is the first-differentiation of the timing jitter [69].

Cycle-to-cycle Jitter: The cycle-to-cycle jitter measures the clock period change amount between any two adjacent cycles, which is indicated as C2 and C3 shown in Figure 3.5. It can be found by applying a first-order differentiation operation to the period jitter. Actually, second-order differentiation of timing jitter gives also cycle-to-cycle jitter. Figure 3.6 explains visually the relation between jitter types [69].

Cycle-to-cycle jitter measurement is preferred to be used to illustrate the stability of spread spectrum clocks as it is not as sensitive as period jitter to the frequency spreading feature. No knowledge of the ideal edge locations of the reference clock is required to calculate either the period jitter or the cycle-cycle jitter.

N-Cycle and N-Period Jitter: A grouping of N periods is measured, instead of measuring the difference between adjacent periods. After taking a significant sample of periods and measuring the differences in the multiple groupings, the standard deviation and peak values can be resolved. There are two permutations of N-Cycle jitter. The first reports statistics from the difference between adjacent groupings (best called N-Cycle jitter) where the second one reports statistics from the variation of the multiple periods (best called N-Period jitter) [70].

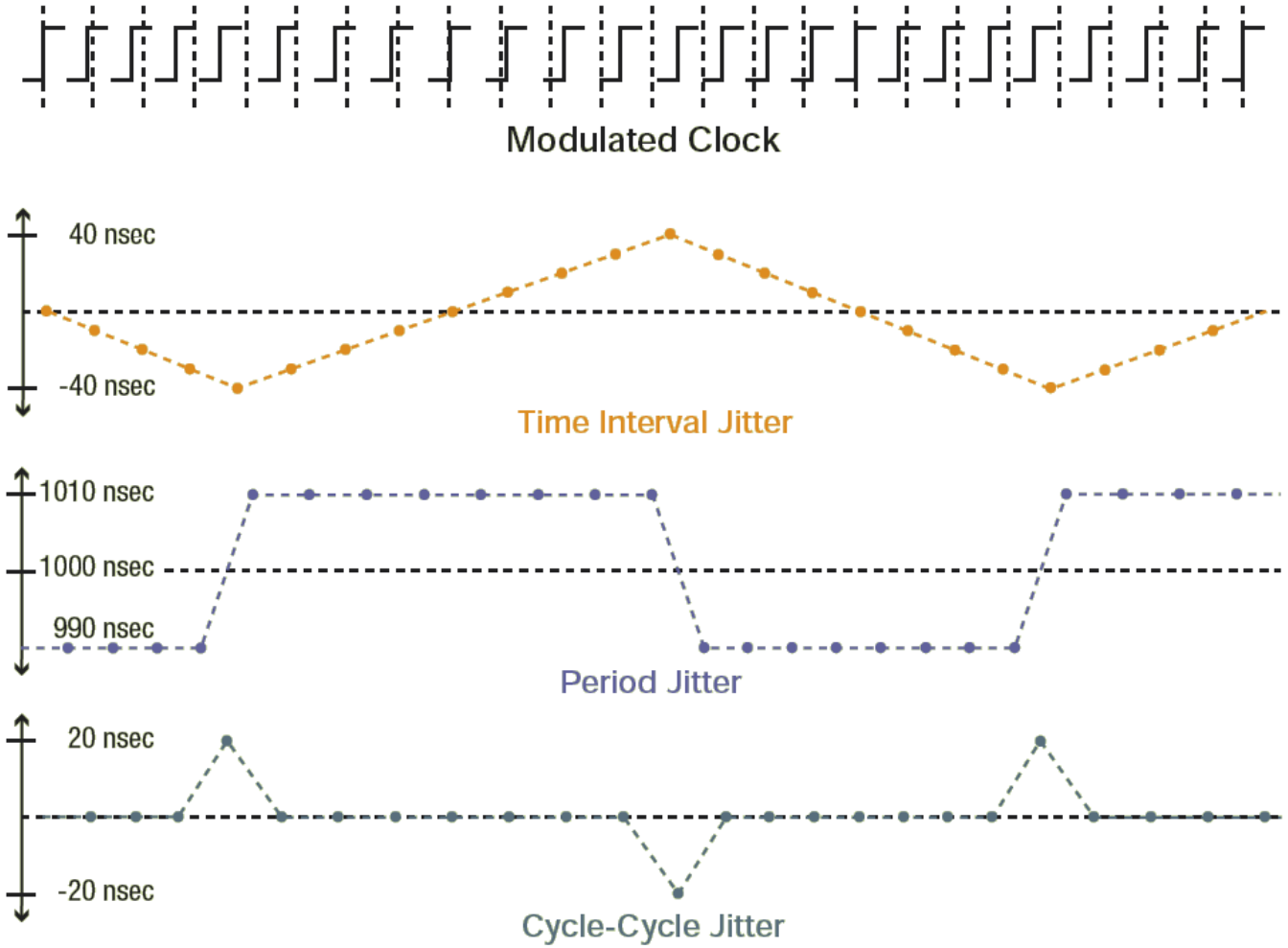

Figure 3.6. Relation between jitter types.

Long-term Jitter (Accumulated Jitter): It measures the change in a clock's output from the ideal position, during several consecutive cycles. The actual number of cycles used in the measurement is application dependent. Long term jitter is different from period jitter and cycle-to-cycle jitter because it represents the cumulative effect of jitter on a continuous stream of clock cycles over a long time interval. This is the reason why long-term jitter is sometimes referred to as the accumulated jitter [71].

There is one shortcoming, if only looked at cycle-to-cycle jitter: only high frequency jitter is visible. Jitter over one cycle can have very high frequency content relative to the fundamental frequency of the circuit being tested. Observing adjacent cycles filters out any longer term effects that are happening within the signal. Utilizing sufficient number of samples enables determining the long-term characteristics of a signal in certain statistical profiles.

An important risk when a long-term measurement is made that if the count of cycles is not precise, the signal may have more noise than the measured values. Therefore, when measuring accumulated jitter relative to different triggers, it is absolutely necessary to ensure a constant number of edges so that jitter greater than one unit interval (UI) is properly measured. Another method of measuring accumulated jitter is to measure the signal relative to reference signal. This allows a direct comparative view of the relationships between the reference and the signal; both over a cycle-to-cycle basis or over the entire measurement period.

3.1.4. Jitter Accumulation

One of the main concerns in clock circuits is the stability of the generated waveforms. Therefore, in order to get an accurate clock, any kind of noise source is unwanted and should be suppressed or minimized [72]. Due to this reason, long-term behavior of jitter was studied in [73] with a measurement-based approach.

Based on that study, random phase shift, which can be also defined as jitter, occurs in oscillators when the phase noise does not exist. Once a phase shift occurs,

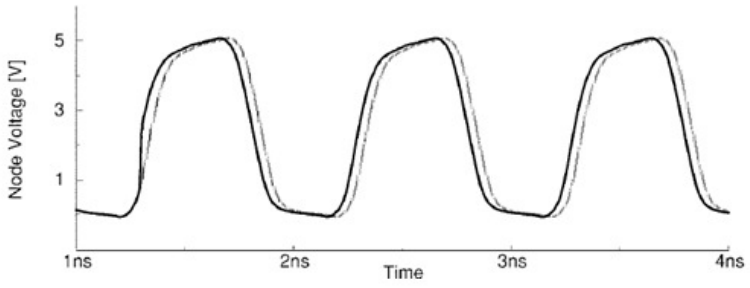

Figure 3.7. Illustration of phase shift.

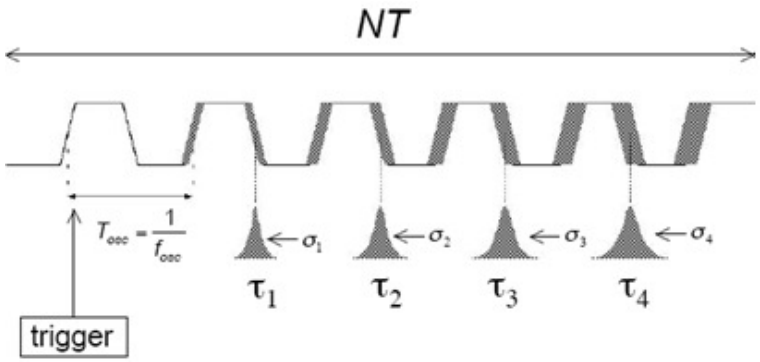

Figure 3.8. Jitter accumulation with varying time.

it persists for an indefinite period of time, as illustrated in Figure 3.7. Therefore, subsequent transitions accumulate the phase shifts in time. Figure 3.8 shows the jitter accumulation of an oscillator with varying time. Hajimiri *et al.* describe "jitter accumulation" in [73] as "Jitter accumulation occurs because any uncertainty in an earlier transition affects all the following transitions, and its effect persists indefinitely. Therefore, the timing uncertainty when ΔT seconds have elapsed is the sum of the uncertainties associated with each transition."

Timing jitter may be affected by different noise influences in different times associated with an oscillator. Two cases are studied in the literature: correlated and uncorrelated noises. In semiconductor devices, the effects of thermal and shot noises are modeled as uncorrelated, while the effect of low frequency $\frac{1}{f}$ flicker noise is modeled as correlated noises.

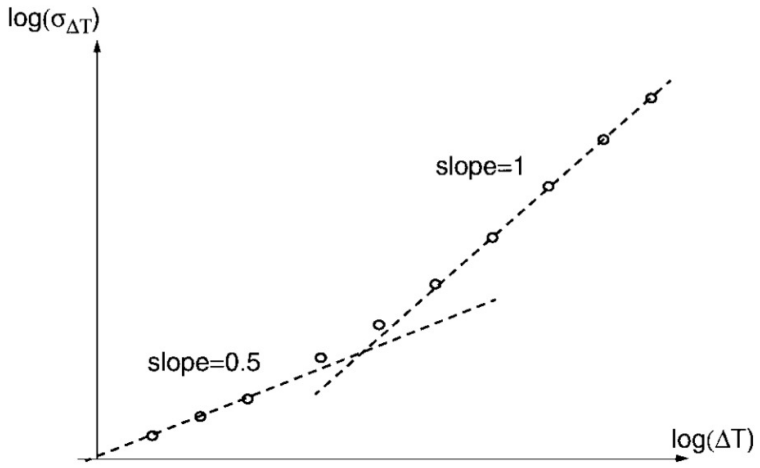

Figure 3.9. RMS jitter versus measurement time.

For a ring oscillator, both cases should be examined. Let us start from the uncorrelated white noise case; identical stages contribute to jitter by the same amount via independent noise sources. The jitter inserted by each stage is independent of the jitter inserted by other stages. Therefore, the total variance of the jitter is the sum of variances of each stage. Furthermore, since an effect persists indefinitely as phase shift, m transitions later (during ΔT time) each stage has jitter as $\sigma^2_{\Delta T} = m\sigma^2_S$, where σ^2_S is the jitter of each stage during one transition. Since m transitions are proportional to ΔT, in [73], the jitter associated with time is formulated as

$$\sigma_{\Delta T} = \kappa\sqrt{\Delta T}, \tag{3.1}$$

where κ is a constant determined by the circuit parameters.

In the case of the correlated low frequency $\frac{1}{f}$ flicker noise sources, correlated jitter is observed over multiple transitions. In [73], Hajimiri adds the standard deviations rather than variance. Therefore, the standard deviation of the jitter after ΔT seconds is proportional to ΔT as

$$\sigma_{\Delta T} = \zeta\Delta T, \tag{3.2}$$

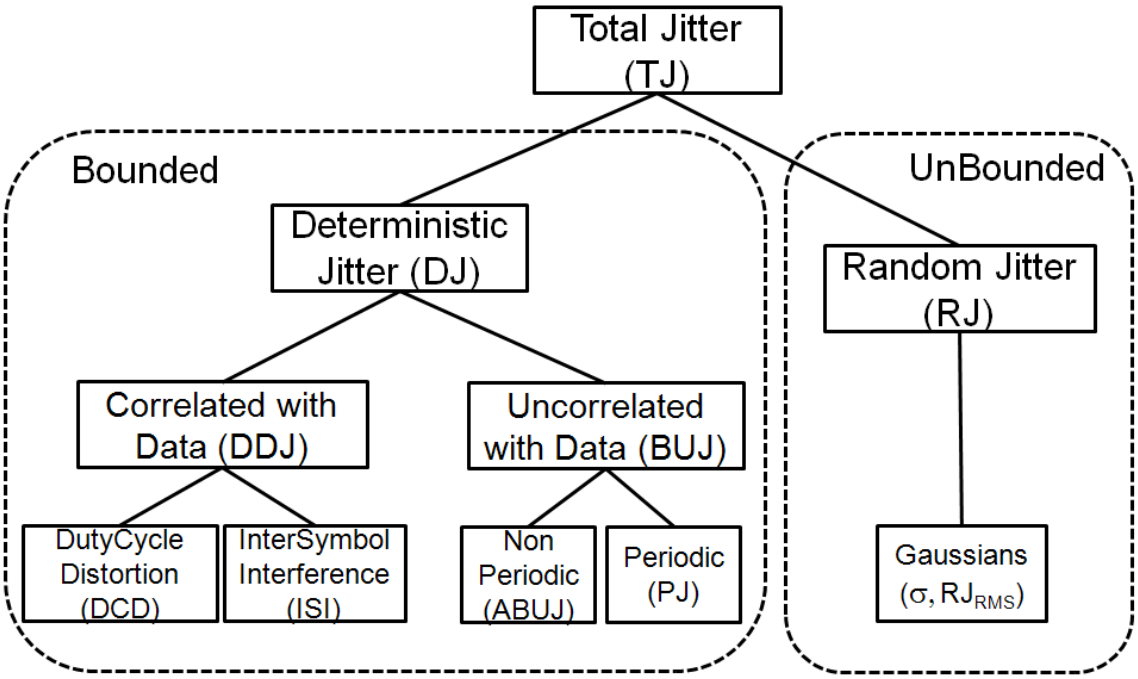

Figure 3.10. Components of jitter.

where ζ is another proportionality constant.

The delay measurements of a ring oscillator demonstrate the effects of both correlated and uncorrelated noise sources. If timing jitter $\sigma_{\Delta T}$ is plotted vs measurement time ΔT in log-log scale, the effect of uncorrelated noise source is displayed with a slope of $\frac{1}{2}$ and the effect of correlated noise is displayed with a slope 1, as illustrated in Figure 3.9. Similar results are also presented in [74] and [75].

3.1.5. Jitter Decomposition

In order to understand the noise sources associated with a signal, knowledge of the distribution of jitter is very important. Finally, dealing with jitter becomes more comprehensible. First order classification of jitter is done as random and deterministic jitter. Random jitter?s peak-to-peak value increases with observation time. On the other hand, deterministic jitter?s peak-to-peak value is limited which makes it bounded. Further classifications can be seen in Figure 3.10.



Figure 3.11. Random jitter.

3.1.5.1. Random jitter. Random jitter is timing noise that has no distinguishable pattern and cannot be predicted. It is assumed that it has Gaussian distribution for the purpose of the jitter model, shown in Figure 3.11. One reason for this is that the primary source of random noise in many electrical circuits is thermal noise or shot noise, which is known to have a Gaussian distribution. Another and more fundamental reason is that the composite effect of many uncorrelated noise sources, irrespective of the distributions of the individual sources, approaches a Gaussian distribution [67].

3.1.5.2. Deterministic jitter. It is the timing jitter that is repeatable and predictable. Due to this stable characteristic, the peak-to-peak value of this jitter is bounded, and the bounds can usually be observed or predicted with high confidence even based on a reasonably low number of observations [67]. This category of jitter is further subdivided to whether it has any dependency on the data or not.

(i) Uncorrelated with Data

In the previous classifications of jitter, "Bounded Uncorrelated Jitter" (BUJ) is defined as Periodic Jitter only. However, in more recent classifications, an aperiodic component is also added to BUJ.

- Periodic Jitter: Jitter that repeats itself in a cyclic fashion is called peri-

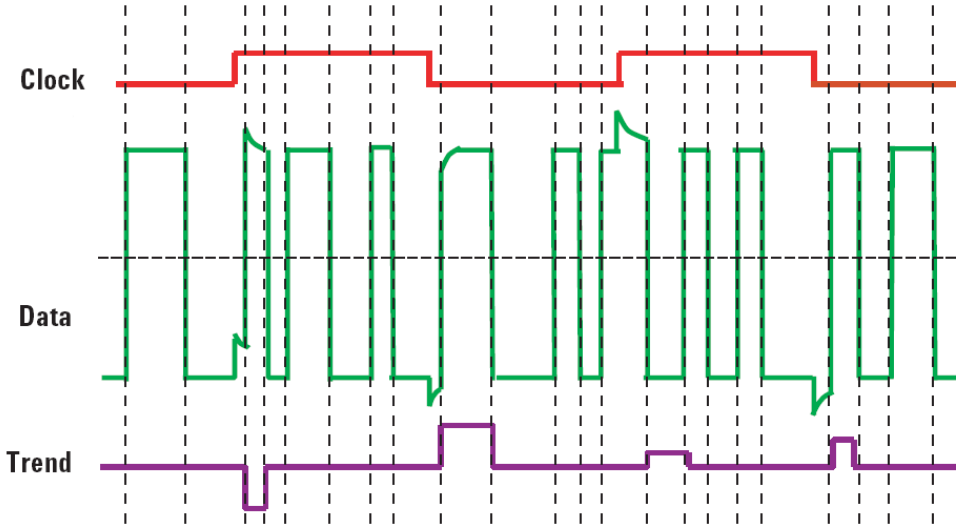

Figure 3.12. Periodic jitter.

odic jitter. Periodic jitter is typically caused by external deterministic noise sources coupling into a system, such as switching power-supply noise or a strong local RF carrier. It may also be caused by an unstable clock-recovery PLL. A sample illustration is demonstrated in Figure 3.12. By convention, periodic jitter is uncorrelated with any periodically repeating patterns in a data stream [67].

- Aperiodic BUJ (ABUJ) Jitter: Generally, crosstalk or other mechanisms that are not periodic in the system are the reasons of the ABUJ, as shown in Figure 3.13.

(ii) Data Dependent Jitter

Any jitter that is correlated with the bit sequence in a data stream is called Data-Dependent Jitter (DDJ). This group is classified into two: Inter Symbol Interference (ISI) and Duty Cycle Distortion (DCD).

- Inter Symbol Interference: ISI means that preceding bits affect the timing

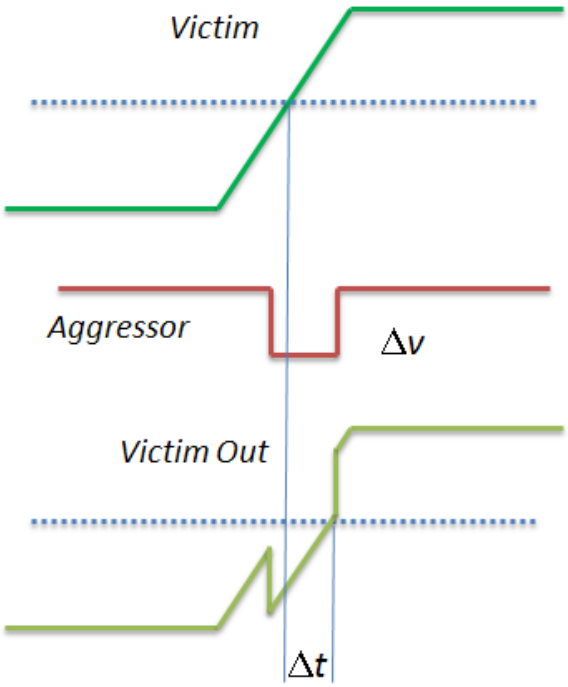

Figure 3.13. Crosstalk.

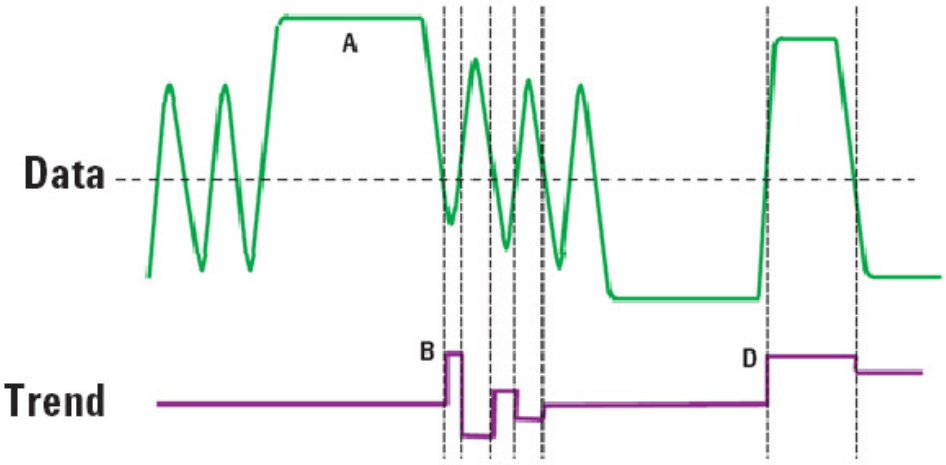

Figure 3.14. Inter symbol interference.

of the current bit. ISI is generally caused by the frequency response of a cable or device. Because of the filtering, unless there are several bits in a row of the same polarity [67], the waveform doesn't reach a full HIGH or LOW state.

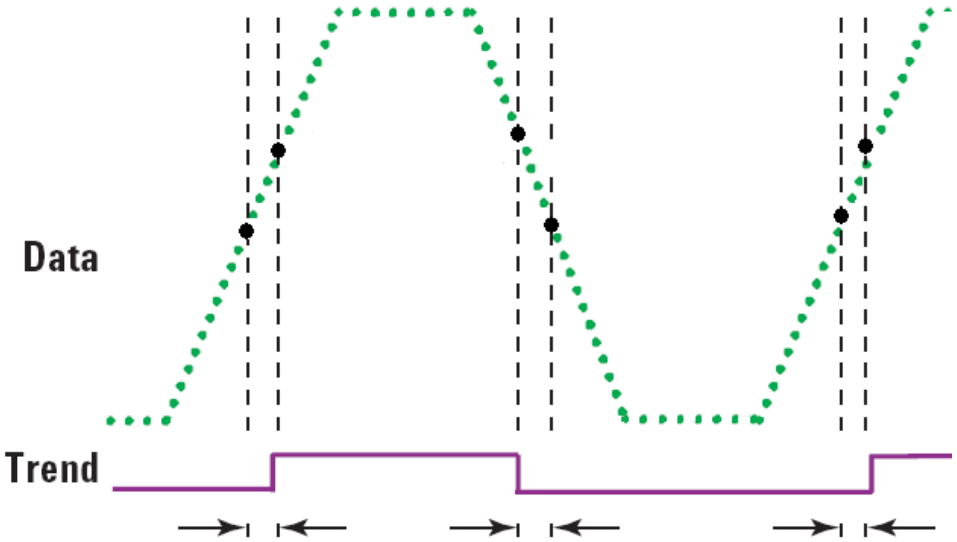

Figure 3.15. Duty cycle distortion.

As another example, consider a waveform superimposed on an offset version of itself. An earlier threshold crossing occurs. Since this timing shift is predictable and is related to the particular data preceding the transition, it is an example of ISI [67]. A sample illustration can be found in Figure 3.14.

- Duty Cycle Distortion (DCD): DCD caused jitter may be predicted based on whether the associated edge is rising or falling. There are two common causes of DCD: First one is that the slew rate for the rising edges differs from that of the falling edges, and the second one is that the decision of threshold for a waveform is higher or lower than it should be. A demonstrative illustration is exhibited in Figur 3.15.

3.1.6. A Practical Jitter Measurement

An inverter based ring oscillator with 23 inverters in a ring is utilized as the design under test (DUT). It has an oscillation frequency of 302 MHz at 2.5 V supply voltage. The details of the circuitry are given in Section 2.6.2. Jitter measurements

will be performed on this design.

A Teledyne LeCroy Wavemaster 8 Zi type real time oscilloscope, which has 16 GHz bandwidth and 40 GS/s sampling rate capability, is used for the measurements. However, the sampling rate can be increased until 200 GS/s with random interleaved sampling (RIS) mode. In RIS mode, the real time oscilloscope acts as a sampling oscilloscope. RIS is an acquisition technique that allows effective sampling rates higher than the maximum single-shot sampling rate. It is used on repetitive waveforms with a stable trigger. The maximum effective RIS sampling rate is achieved by making multiple single-shot acquisitions at maximum real-time sample rate. Thus the acquired signals are positioned approximately five ps (200 GS/s) apart [76].

The oscilloscope has SDAIII-CompleteLinQ [77] software which performs jitter and crosstalk analysis; moreover, it has has an extensive capability of jitter decomposition, and eye diagramming and analysis.

3.1.6.1. Determining The Inherent Jitter for Improving the Accuracy. Ignoring the intrinsic noise of the measurement instrument causes errors in jitter measurements. The intrinsic noise of the instrument, used in the experiment, should be determined experimentally before starting jitter measurements. The following experiment aims to determine the inherent jitter of the instrument. If the inherent jitter is much less than the measured jitter of the signal, intrinsic noise can be ignored and directly measured jitter can be used in the further analysis.

The inherent jitter of the oscilloscope may originate from trigger jitter, time base stability, and delay jitter. Generally the inherent jitter, which is the internal jitter of oscilloscope, is uncorrelated with the jitter of device under test [78] . Therefore, the instrument jitter can be subtracted from the total jitter using quadrature subtraction, shown as

$$t_{DUT} = \sqrt{t_{meas}^2 - t_{inst}^2}, \tag{3.3}$$

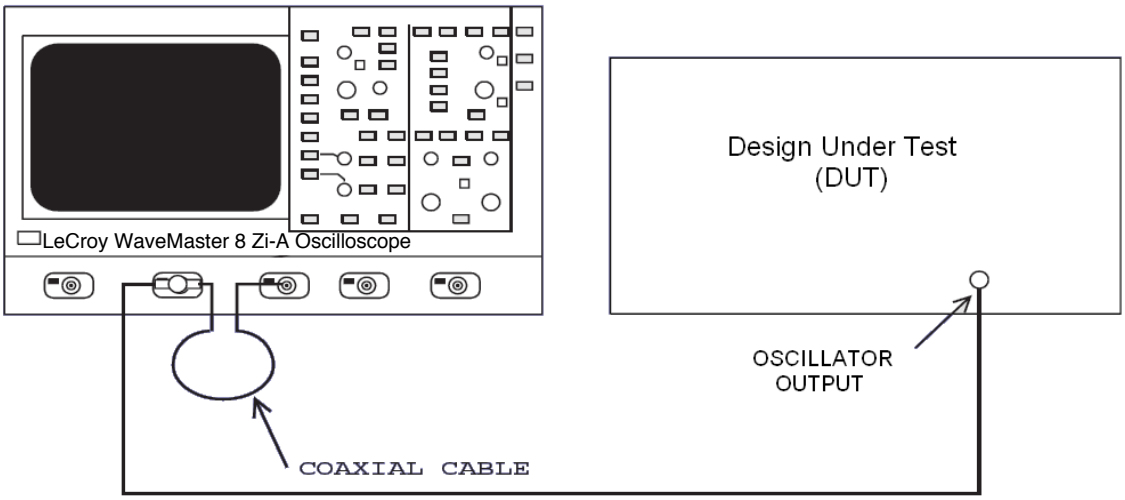

Figure 3.16. Inherent jitter measurement setup.

where, t_{DUT} stands for jitter of device under test , t_{meas} stands for total measured jitter, and t_{inst} stands for jitter due to instrument. The jitter due to instrument should be measured under the specific conditions of the test being performed. The pictorial representation of the test set-up is given in Figure 3.16.

For this experiment, RIS acquisition mode of the real-time oscilloscope is used. The oscilloscope is triggered externally using the output of DUT. Delay between the signals is produced by the cable used in measurement. The oscillator is set to a sampling rate of 200GS/s with RIS acquisition mode. The time delay between the Channel-1 and Channel-2 waveforms was measured. The cable delay is constant and any variation in the delay is due to the oscilloscope. This experiment allows testing of the accuracy of the measurement system [78]. Measurement results are shown in Figure 3.17. The inherent jitter of the LeCroy 8Zi is around $678fs$ in this experiments.

3.1.6.2. Period Jitter Measurements. As defined previously, period jitter refers to the distribution of periods. We simply measure the period of each signal cycle in the waveform and find the distribution. For the period jitter measurement, RIS acquisition mode of the real-time oscilloscope is also used as in the inherent jitter measurement. A sampling rate of 200GS/s is achieved. The measured period jitter is $1.89ps$ which

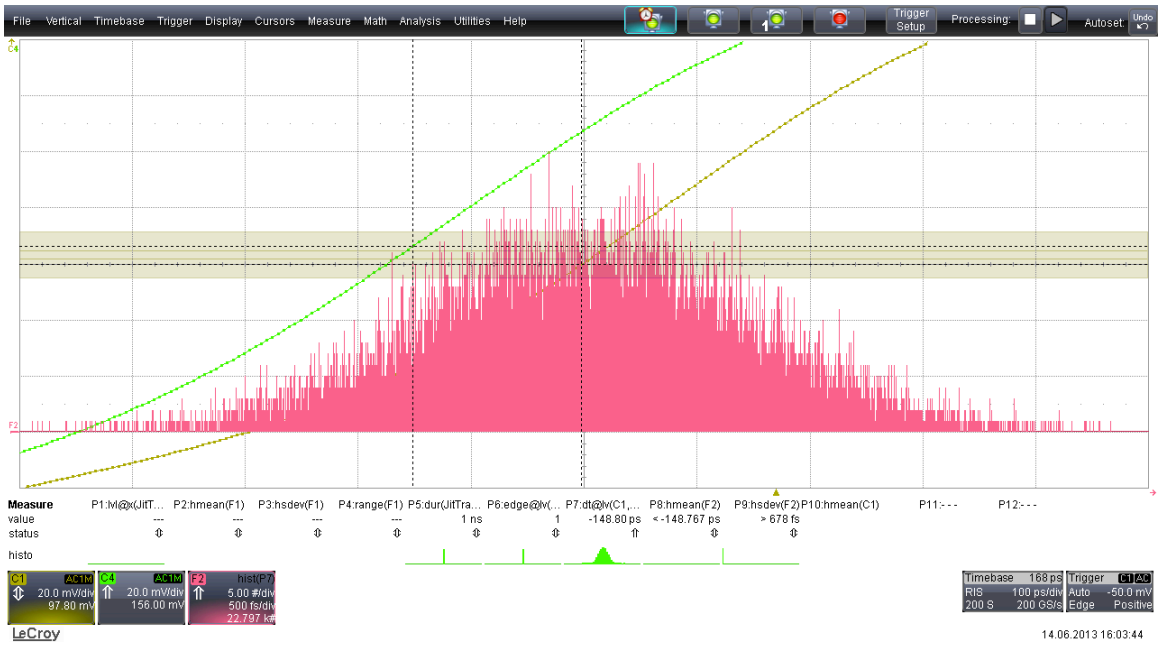

Figure 3.17. Inherent jitter measurement.

can be seen from Figure 3.18.

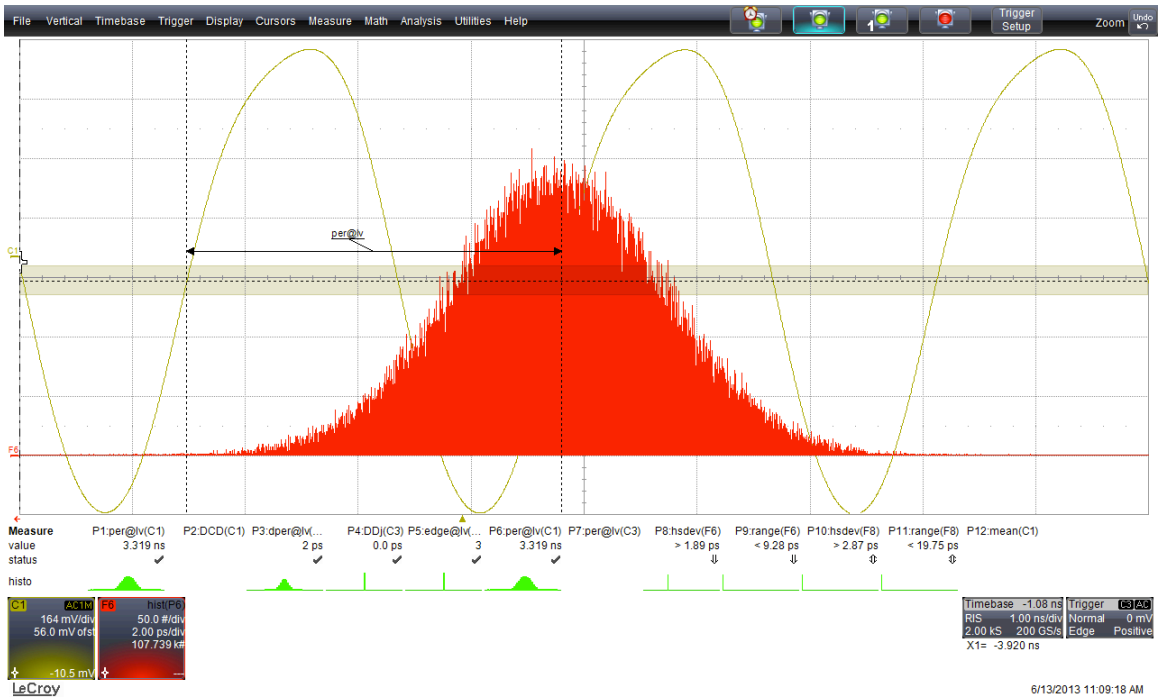

Figure 3.18. Period jitter measurement.

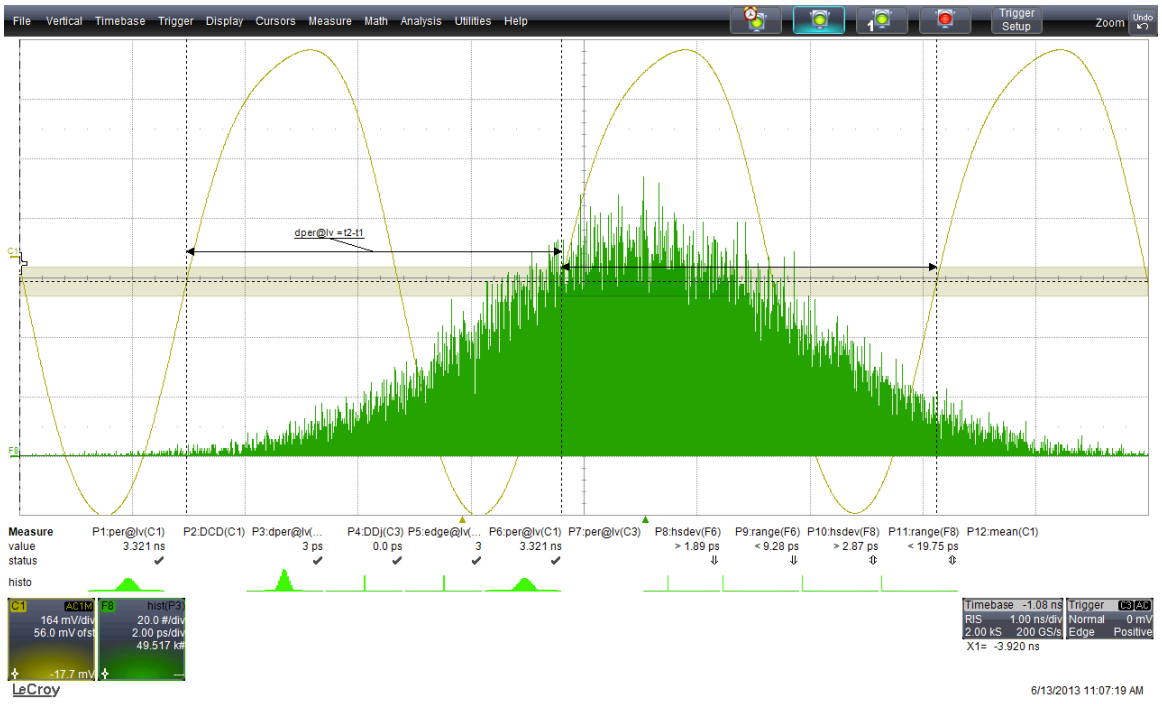

Figure 3.19. Cycle-to-cycle jitter measurement.

3.1.6.3. Cycle to Cycle Jitter Measurements. The changes in the signal period between any two adjacent cycles is called as cycle to cycle jitter. Therefore, the seen from Figure 3.19, the changes between two adjacent cycles are plotted as a histogram. Standard deviation of this histogram gives cycle to cycle jitter as $2.87ps$, which is higher than period jitter as expected.

3.1.6.4. Long-Term Jitter Measurements. The change in the output of the signal from the ideal position over consecutive cycles gives the long-term jitter in that duration. Long-term jitter measurements, which accumulates the jitter, need a reference signal, and the measurements are performed relative to this reference signal. Time interval error between the signal and the reference clock gives the long-term value of the jitter. LeCroy MasterWave 8 Zi oscilloscope has a golden clock which is generated from a software PLL [77].

Jitter accumulation can be seen from the following figures. Figure 3.20 shows the TIE histogram after $500ns$ which corresponds around 60 periods of the RO. A

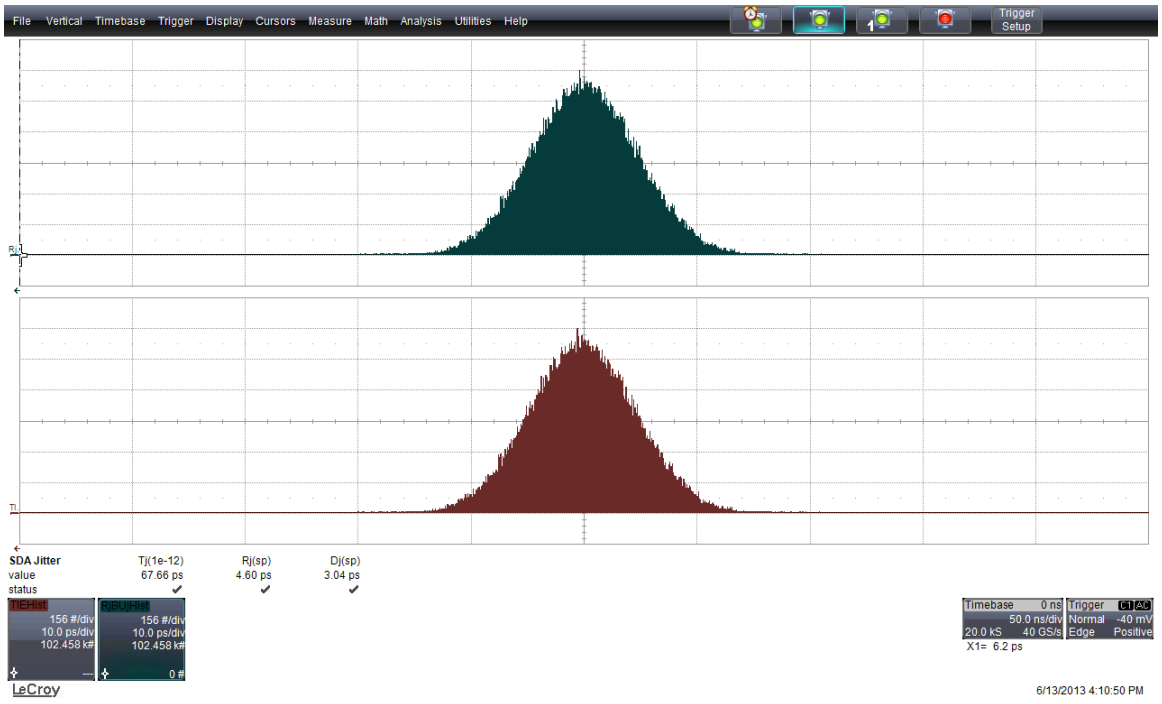

Figure 3.20. TIE measurement with 500ns accumulation.

longer measurement can be seen in Figure 3.21, where one *ms* accumulation time is performed. Figure 3.22 shows the measurement with two *ms* accumulation time. After this length of accumulation time, deterministic jitter starts to appear for this RO. Since jitter accumulates over time, higher jitters can be observed within longer acquisition times.

3.2. Flicker Noise as a Randomness Source

In 1925, J.B. Johnson performed an experiment [79] to test the shot noise in vacuum tubes (Schottky's theory [80]). However, he saw that the noise in low frequencies was not white. In 1926, Schottky described that the observed noise was flicker noise [81]. Until Bell, it was thought that the reason of flicker noise was physical properties; however, Bell claimed that it was a cooperative phenomenon arising also from the statistics of electrons' queues [82]. In 1974, Caloyannides measured flicker noise in semiconductors from $10^{-6.3}$ Hz to one Hz [83].

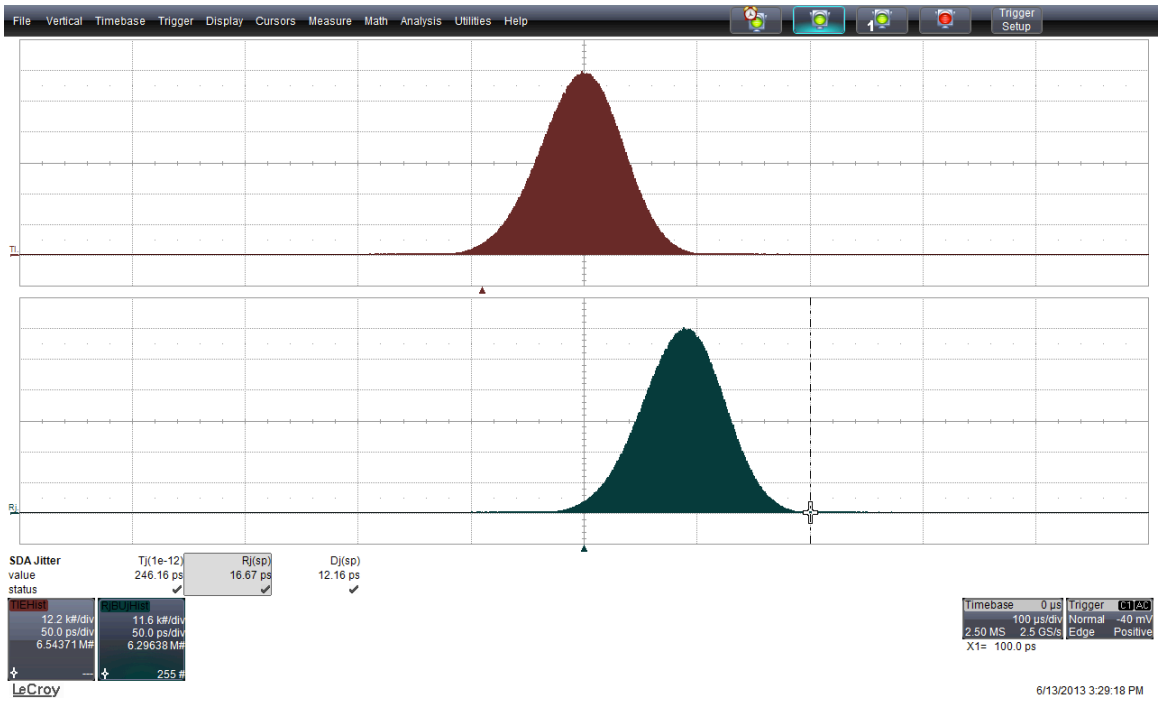

Figure 3.21. Time interval error measurement with 500ns accumulation.

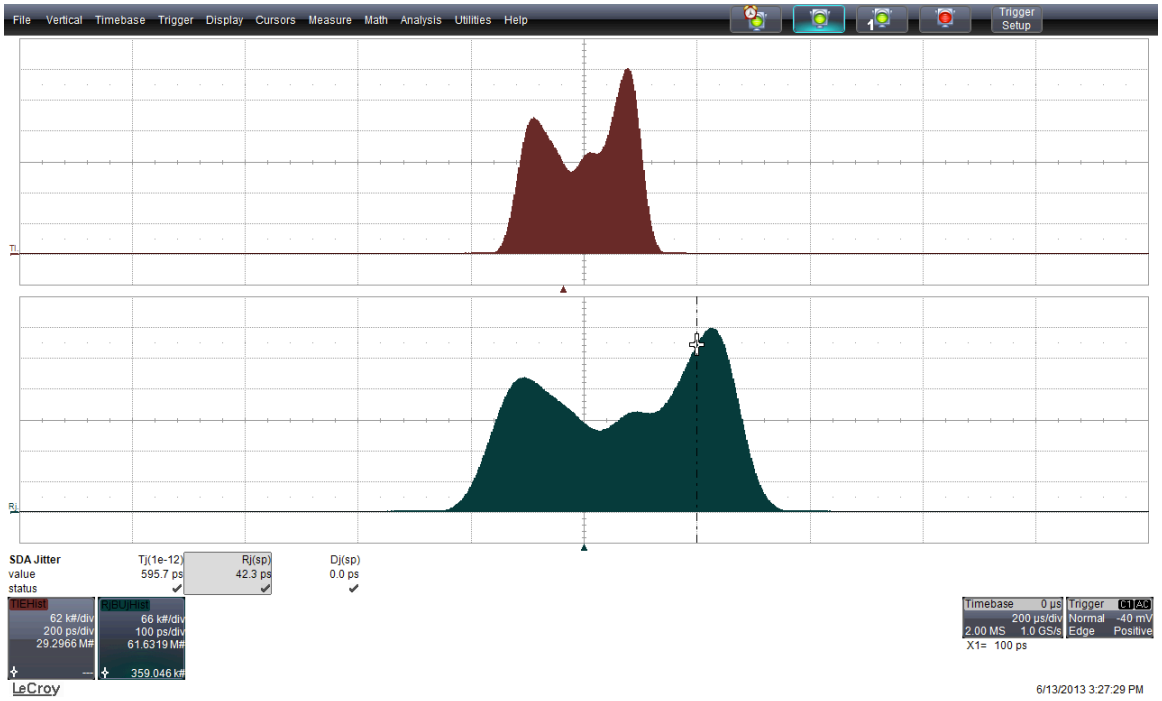

Figure 3.22. Time interval error measurement with 500ns accumulation.

Later on, it was noticed that this unexpected noise existed in many different systems, including the human heart, the brain, the stock market, etc. Flicker noise has a Gaussian distribution and appears at low frequencies. In different systems, flicker noise has the PSD of $\frac{1}{f^{\alpha}}$, where α is between 0.5 and 1.5. Noise sources can be classified into two categories as colored and white noise sources. Flicker noise is called as colored noise, because it is dominant in low frequencies.

In spite of the fact that flicker noise affects phase noise and jitter of oscillators, most studies have modeled phase noise and jitter by ignoring flicker noise [74,84] until the late 90s. Phase noise and jitter models with the presence of flicker noise started to appear in the literature around 2000s [1, 19, 73, 85]. In [1], contribution of noise sources to phase noise and jitter is examined. Phase noise is formulated with flicker and white noise sources; however, jitter is formulated without flicker noise.

Since oscillation jitter is used to provide randomness for RNG, understanding the phase noise and jitter mechanisms in ROs is vital in order to increase the entropy. In contrast to clock circuits, random number generators require noise in order to produce random bits with high entropy. There is a general agreement that white noise is necessary for random number generation; however, the same statement is not valid for flicker noise. Studies about flicker effect on randomness are still not sufficient.

In [34], it is expressed that colored PSD of flicker noise may generate correlated bits in the output of RNG. Therefore, some researchers tried to suppress flicker noise [3]; however, most researchers simply ignored it [4, 28, 48].The relation between autocorrelation and sampling period was studied in [86] without clearly addressing the underlying theory. According to that study, autocorrelation of adjacent bits is reduced from 62% to 7% by increasing the sampling period 20 times. In [22] the flicker noise effect is investigated with unrealistic jitter values and without the confirmation from measurement results.

According to the study [73], mentioned in Section 3.1.4, the rms jitter originating from flicker noise accumulates over time with a slope of 1, while the rms jitter

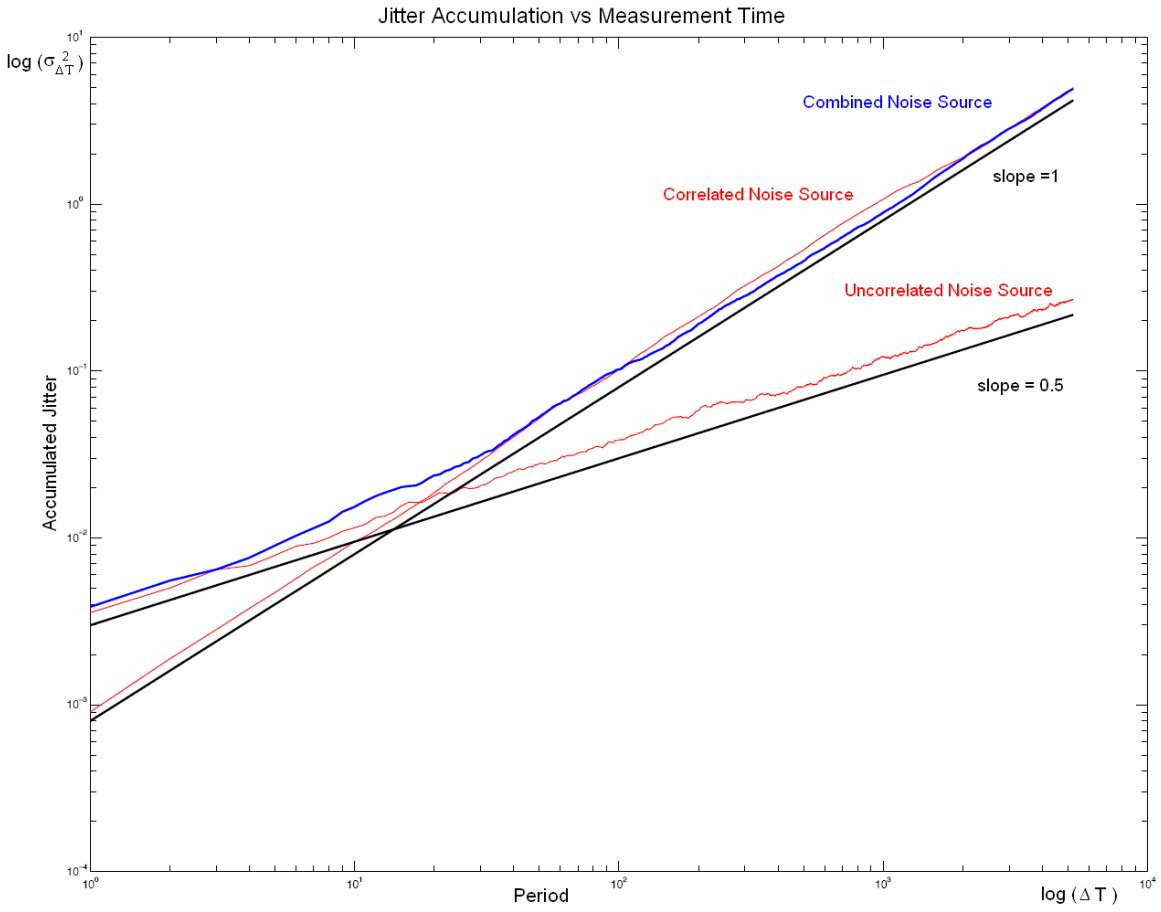

Figure 3.23. Accumulated jitter versus measurement time.

originating from white noise accumulates over time with a slope of 0.5 in log-log plot of the PSD. On the bases of long-term jitter explanations and studies, randomness of a ring oscillator at the mth cycle can be predicted. In order to estimate the jitter amount after m periods, a MATLAB based analysis can be performed. The measured phase noise data can be inserted to the model to estimate the accurate jitter statistics. We aim to develop such an analysis to decide the specifications of an oscillator for a desired randomness from the beginning of design process.

A sample long-term jitter study is shown Figure 3.23 which demonstrates the accumulated jitter of correlated, uncorrelated, and combined noise sources with varying measurement time. The proposed MATLAB model produces white and colored random noise for 100 trials, because jitter estimation depends on statistical data.

Thereafter, the variance of jitter is estimated according to these trials. Although, arbitrary noise parameters are utilized in the present MATLAB code, accurate phase noise data can be included after measurements.

3.3. Randomness Source in Weak Inversion Region

In addition to suitability analysis, maximizing the RO jitter for security applications is also investigated in [21]. As a consequence of this study, it is understood that decreasing V_{DD} caused an increase in randomness. The lowest supply voltages can be achieved when the devices are in the sub-threshold region. Moreover, it is a known fact that the amount of jitter variance depends on the operating regime of CMOS transistors. [87] shows that CMOS devices have more current noise in sub-threshold region.

Actually, using CMOS transistors in weak inversion region is not popular. It is stated in [88] that, the absolute values of the currents and of the transconductances become so small that the noise becomes exceedingly large; therefore, designers do not want to use the weak inversion current region. In the contrary, maximum randomness can be obtained, if only the supply voltage of RO is lowered down to its minimum value where it still has oscillator behavior.

3.4. T-Entropy as an RNG Evaluation Measure

NIST Statistical Test Suites of 140-2 [29], 800-22 [30], and DIE Hard Test Suite [31] are used most commonly in order to evaluate RNGs. These statistical test suites report their results as fail or pass scores. However, most of the time, small changes in the entropy levels cannot be distinguished with these test suites. Therefore, in order to analyze the randomness of ROs in different cases, a different measure of entropy is needed.

Shannon entropy is the most well-known and reliable entropy estimation method for infinite sequences [89]. T-entropy [90, 91] is a successful approximation of the

Shannon entropy for finite bit streams. T-entropy measure has started to be used in RNG applications [92] as well. We also utilize the T-entropy measure to calculate the randomness of the ROs in our analysis.

3.5. A Pre-Evaluation Method for RNG

Random number generators are the heart of cryptographic blocks. If the RNG in the system does not produce random bits with a certain quality, even the most secure algorithms will present weakness. Therefore, special interest should be given to evaluation of RNGs. Traditional RNG evaluation is done with some statistical test suites [29–31] which are based on complex mathematical evaluation methods. Furthermore, these tests are applied to RNG during evaluation stage which is performed when RNG design and fabrication completed. A pre-evaluation method is necessary to get some feedback and improve the design before the final stage. In this section, a new pre-evaluation method for RNG is proposed [15]. This pre-evaluation method can be also used in Common Criteria (CC) evaluations to evaluate the RNGs used in Smart Cards (SC), RFID (Radio Frequency Identification), and NFC (Near Field Communication) tags.

Beside the design, the placement and the interaction with the other elements of the RNG in the whole system is very important in terms of protecting the quality of the bits. In order to do this, jitter decomposition method is proposed for use in the AVA_VAN (AVA: Assurance of Vulnerability Assessment, VAN: Vulnerability Analysis) evaluations of RNGs. By using this method, jitter in the random bits can be broken down to its components. The first level classification starts with random and deterministic jitter. Actually, random jitter is the source of entropy of RNG. On the other hand, deterministic jitter should be investigated deeply, because it gives the clues of vulnerability during RNG attacks. Further separation can be done under deterministic jitter as periodic jitter and data dependent jitter. Periodic jitter shows how sensitive the random bit wiring is to crosstalk, power supply switching, and some other EMIs in the integrated circuit. Thus, periodic jitter may give clues of such vulnerabilities. Beside that, data dependent jitter shows how much random bits

Table 3.2. RNG Pre-Evaluation Check List.

Jitter Type	Noise Source
Random Jitter	Thermal Noise Shot Noise Flicker Noise
Duty Cycle Distortion	Threshold level of transmitter Asymmetry in rise and fall time of waveform
Inter Symbol Interference	Bandwidth limitations Reflection due to impedance mismatches
Non-Periodic Jitter (ABUJ)	Crosstalk Ground bounce
Periodic Jitter	Clock correlation Switching effect of power supply

are sensitive to data patterns contained in the random bits. This information about deterministic jitter has significance during RNG evaluation in terms of defining RNG attacks during AVA_VAN evaluation.

RNG evaluation can be processed by examining all these jitter components. Advanced oscilloscopes have jitter analyzing tools. They also give the amount of each jitter element. The weakness of the RNG can be determined by interpreting the analysis results. RNG attack scenarios can be developed after the interpretation of RNG analysis results; on the other hand, a developer can eliminate the dominant noise source and improve the design in an earlier stage. Table 3.2 presents the RNG evaluations steps in this method.

This RNG evaluation method is developed from a practical point of view that does not rely on complex math. It is a rapid method which examines the waveform instead of bit streams. Investigating the long-term behavior of waveform and breaking down the jitter of waveform to its components gives an intuitive feel about the

dominant noise.

3.6. Conclusion

In this chapter, detailed information is provided about jitter, namely, sources of jitter, types of jitter, and decomposition of jitter. After that a practical jitter measurement is given. The effect of flicker noise on randomness and the effect of weak inversion region on randomness are questioned. In order to understand the small changes in entropy, T-Entropy measure is introduced for evaluations. In addition, a measurement-based, pre-evaluation method for the output waveform of an RNG is proposed.

4. ASIC RNG IMPLEMENTATIONS

Under grant 106G007, the Scientific and Technological Research Council of Turkey (TÜBİTAK) supported the RNGs research in the content of the Turkish e-ID Card project. ASIC implementations of some of the well-known digital gate based RNGs are also supported under this grant. This chapter expresses the design methodologies, implementation details, and measurement results of the ASIC implemented RNGs.

Three ASIC RNG implementations are performed based on two different type of oscillators. The first type of oscillator is a ring oscillator and two of the implementations are based on the models given in [5]. The differences between these two implemented circuits are the number of ROs and the number of inverters in a ring. The other type of implementation is based on FIGARO, modeled in [48]. FPGA implementations of these RNGs are presented in [5, 36, 49].

Implementing the RNG as an IC gives the designer the flexibility to reduce the correlation between adjacent oscillators. Thus, XORing the oscillators becomes more effective because it improves the quality of the bit stream. Moreover, the design becomes more robust against attacks by implementing RNG as an IC.

Measurement results of RO-based ASIC implemented RNGs show that more than 6.5 times higher throughput is achieved than the previous FPGA implemented RNG results in [5] and [49]. Furthermore, the complex post-processing stage, which reduces the throughput 16 times, as expressed in [5, 49], has not been used in ASIC. This is because the correlation and coupling are reduced and jitter is increased by ASIC implementation.

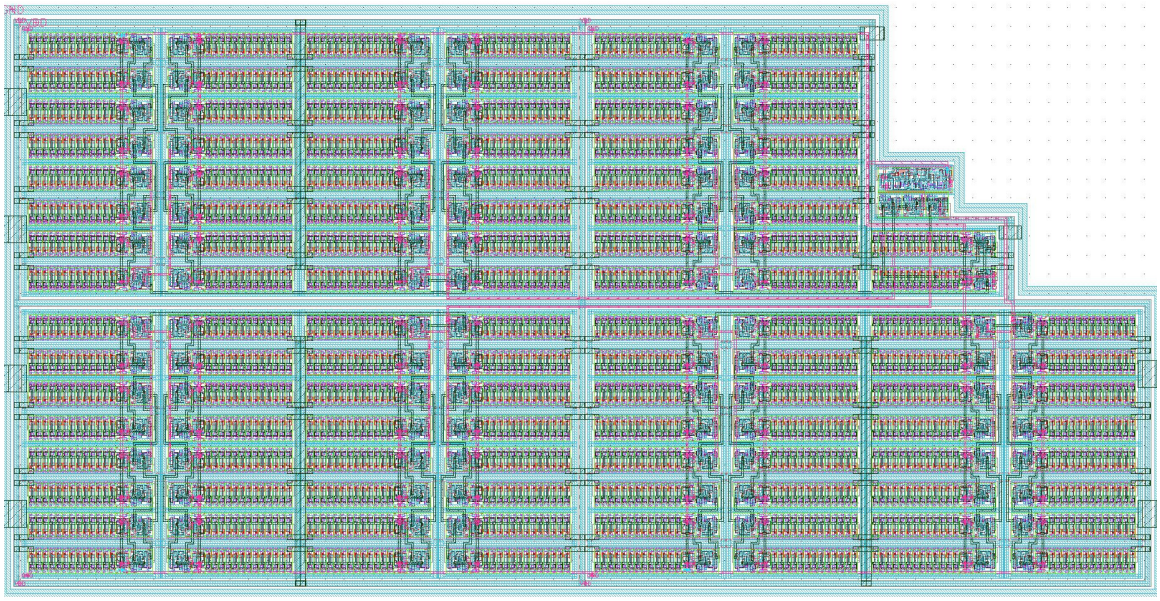

Figure 4.1. Layout of rings design.

4.1. A High Speed IC Random Number Generator Based on Phase Noise in Ring Oscillators

In this work, fulfilled test results are obtained from the NIST 140-2 test suite after a simple corrector (von Neumann). In fact, it was noticed that the post processing method of resilient functions used in [5] and [49] results in a fourfold rate reduction in comparison with the von Neumann corrector. Beside these, further analysis has been performed to find out how to reduce the current consumption. By decreasing supply voltage, statistically unaffected raw data can be derived down to $1.6V$. This reduces power consumption approximately four times.

In Section 4.1.1, implementation details can be found. In Section 4.1.2, measurement results, analysis of results, and the statistical test results are presented.

4.1.1. Implementation Details

HHNEC's 0.25μm eFlash process with a supply voltage of $2.5V$ is used for implementation and fabrication of the prototypes. In terms of layout design techniques,

every ring oscillator is located in a separate n-well. Separate n-well technique is a must, in order to prevent coupling and correlation between adjacent ring oscillators as much as possible. Furthermore, this separate n-well technique decreases the leakage between substrate junctions and substrate noise, which improves the mutual effects of grounds between rings. An AND gate is added to the rings to switch them off in necessary conditions. Of course, this causes a small frequency reduction of rings due to the additional AND gate delay. All these additions enlarge the layout by almost twice, as seen in Figure 4.1. In addition, XOR gates were placed very near ring oscillators which were connected to them. Thus, routings causing coupling in the case of long adjacent paths would be very short. Paths from XOR gates to ROs are drawn in slightly different orders to differentiate the path delays.

Designed circuitry consists of 114 ring oscillators, which have 13 inverters in a ring, a sampler, and a post processor after sampler. Post processing circuitry is not implemented as hardware. It is planned to be implemented as software for necessary conditions. During measurements and tests, it is noticed that instead of the processing, it is sufficient just to apply the von Neumann corrector, which does not reduce the throughput as much as post processing applied in [5] and [49] do. Furthermore, to implement the von Neumann correction is much easier than post processing, since von Neumann correction can be realized just with a few logic gates. However, in this design, even the von Neumann corrector is not implemented as hardware.

4.1.2. Measurement and Statistical Test Results

The total circuit area of the implemented circuit excluding pads is 0.043 mm^2. On-chip guard rings are used between ring oscillators and other prototypes. Fabricated chip photo can be seen in Figure 4.2. An FPGA based hardware, which has a PCI interface, was designed to upload the binary data to computer. Maximum data storage rate of FPGA based hardware is two $Gbps$. Up to 250 MHz of sampling frequency can be used during data acquisition.

The output of a ring oscillator is observed in order to understand how it works as

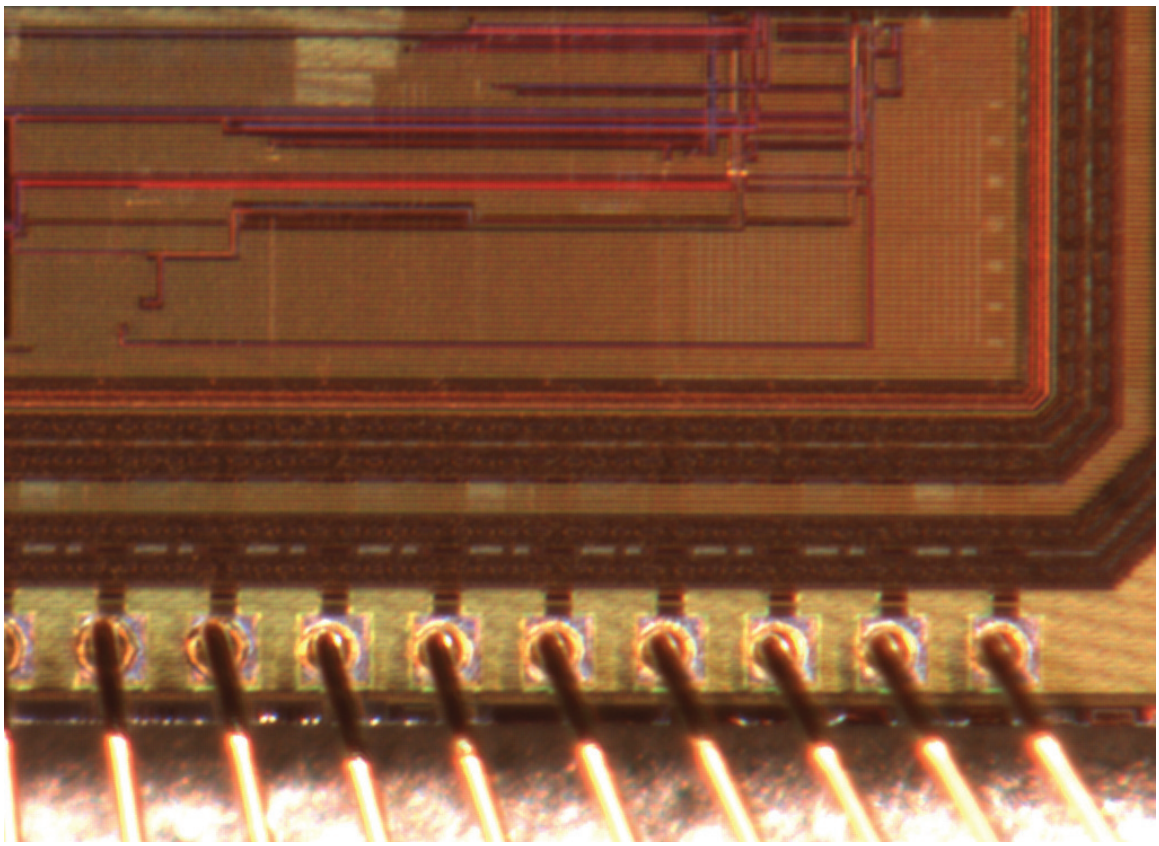

Figure 4.2. Chip photo.

an integrated circuit. The oscilloscope screenshot shows the measured ring oscillator output in Figure 4.3. As can be seen from the figure, considerable amount of jitter ($\sim$ 8%) is observed even at the output of one ring oscillator. In [5], all the calculations were done based on the assumption of 2% of jitter. This explains how fulfilled test results can be achieved without strong resilient functions. The output of two XOR trees with 32 ring oscillators is also observed. Correlation coefficient is estimated by using output of these ring oscillators. The correlation coefficient is estimated as 0.0073 between two 32 XORed ROs outputs of 32,000 bits. Moreover, maximum frequency at the output of 114 rings is $256MHz$.

A bit stream of length 160 $MBytes$ was acquired through the PCI interface of the FPGA based hardware without von Neumann processing. The obtained bits are subjected to full the NIST 140-2 test suite [29]. After applying the von Neumann

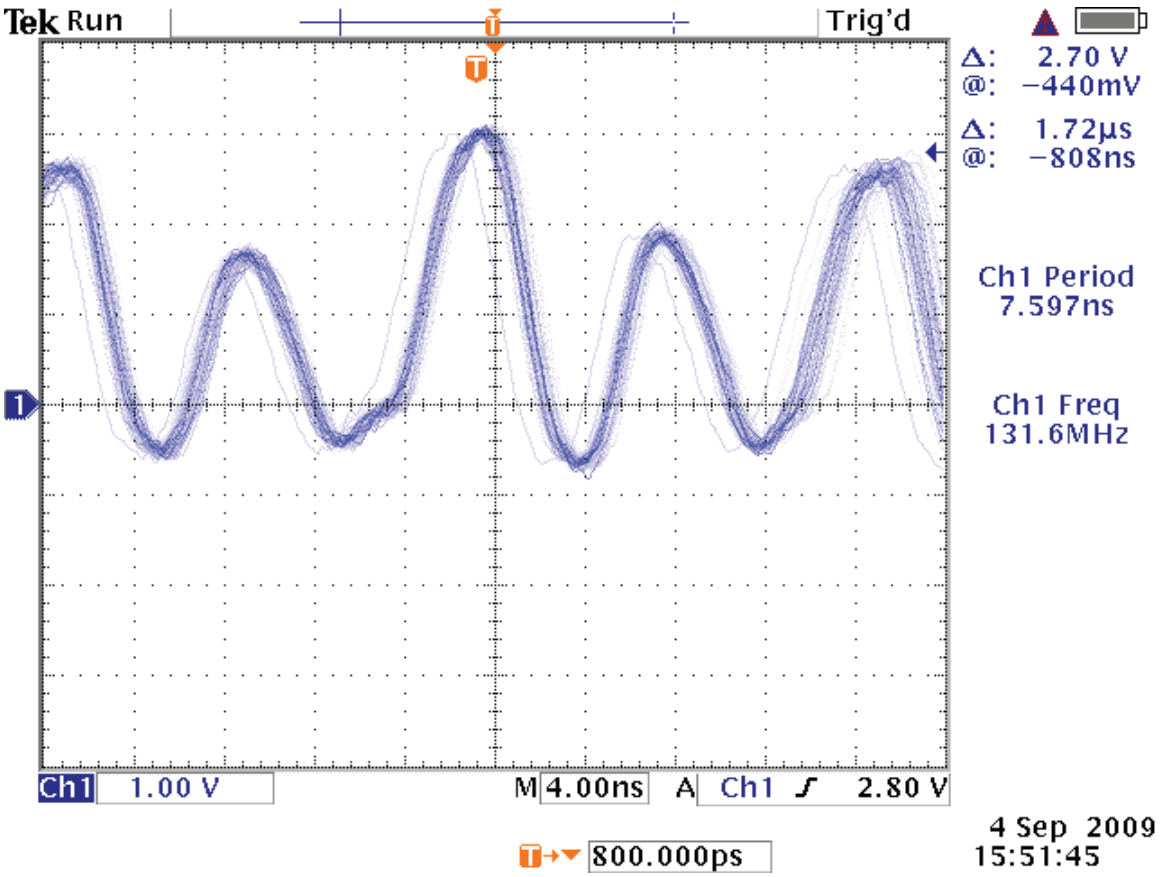

Figure 4.3. Measured output of one of the ring oscillator.

corrector to the binary data obtained from these ring oscillators in software, processed binary data pass the tests of the full NIST random number test suite. Von Neumann's de-skewing technique [93] is used in order to eliminate the existing bias in the output sequence. In the bit stream, 00 and 11 are discarded, 01 turns into zero and 10 turns into one. Throughput approximately decreases by a factor of four by generating one bit from four bits. Results of the tests are given in Table 4.1 with pass rates and p-values. The minimum pass rate for each statistical test with the exception of the random excursion (variant) test is approximately 0.973618 for a sample size of $332 \times 1MBit$ binary sequences. More than 20 randomly selected chips are subjected to the full NIST 140-2 test suite, and all of them pass. This shows that the design technique is stable; it does not vary from chip to chip, and does not show any dependency on process parameters, which is very important in terms of the reliability of RNG.

Table 4.1. Statistical test results of RO-based RNG with 114 ROs.

Statistical Test	**P Values**	**Pass Rates**
Frequency	0.474403	0.9819
Block Frequency	0.090107	0.9970
Cumulative Runs	0.564949	0.9880
Runs	0.206562	0.9729
Longest Run	0.102341	0.9880
Rank	0.326515	0.9880
FFT	0.116003	0.9940
Nonperiodic Templates	0.992390	1.0000
Overlapping Templates	0.678058	0.9880
Universal	0.001552	0.9880
Apen	0.862344	0.9910
Random Excursions	0.124210	1.0000
Random Excursion Variants	0.507859	1.0000
Serial	0.745968	0.9940
Linear Complexity	0.090107	0.9849

Additionally, Figure 4.4 shows current consumption of the RNG when the supply voltage is varied from $2.5V$ to $1.5V$. Statistically unaffected raw data can be obtained until the supply voltage is dropped to $1.6V$. The output bit stream also passes the statistical test. Power consumption decreases from $40mW$ to $11mW$ while supply voltage is decreased from $2.5V$ to $1.6V$. Therefore, supply voltage should be adjusted to its minimum level if the power consumption is a critical design constraint. However, reducing supply voltage in favor of current consumption will yield a reduction in speed, too. In our RNG, output frequency decreases to $50MHz$ at $1.6V$. A summary of performance parameters is also presented in Table 4.2.

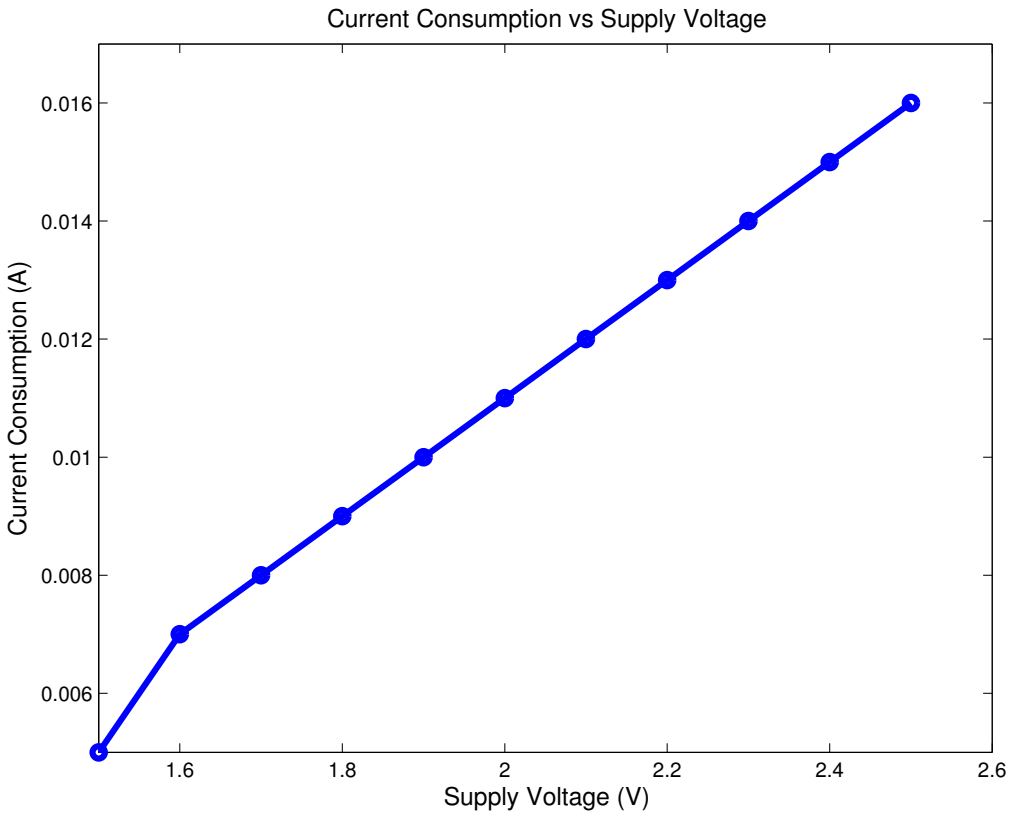

Figure 4.4. Consumed current versus supply voltage.

4.2. A Digital IC Random Number Generator with Logic Gates Only

In this section, the ASIC implementation of [48] is presented. FIRO and GARO are implemented and fabricated by using HHNEC's 0.25μm eFlash process with a supply voltage of 2.5V. An XORed combination of these two oscillators is also implemented. A new structure, which utilizes more than one FIGARO in an XORed configuration, is proposed to build an RNG. Increasing the number of FIGAROs utilized in the circuit produces faster output bit stream and improves the quality of output bit stream. In the following section, FIRO and GARO structures and their properties are summarized. Implementation details, measurement results, and the statistical test results are presented in Section 4.2.2.

Table 4.2. Summarized performance parameters of RO-based RNG with 114 ROs.

Parameters	**Values**
Maximum Speed of Output Bit Stream	256 MHz
Maximum Clock Speed for Passing Tests	66 MHz
Maximum Speed of Throughput	16.5 Mbps
Typical 1/0 Ratio of Raw Data	1.22929
Typical 1/0 Ratio after von Neumann	1.00029
Correlation Coefficient between two 32-ROs of 32,000 bits	0.0073
Jitter/Period	~ 0.08
Size	0.043 mm^2
Power Dissipation @ 2.5V	40 mW
Power Dissipation @ 1.6V	11 mW
Supply Voltage	2.5V
Fabrication Technology	HHNEC's 0.25 μm eFLash

4.2.1. Fibonacci and Galois Ring Oscillators

Using FIRO and GARO as random number generation methods is a novel approach that is influenced by Fibonacci and Galois LFSRs, where basically delay cells (D-type flip flops) are replaced by inverters. In FIRO, the number of inverters, r, can be chosen either even or odd; however, it is necessary that $r \neq 2$. On the other hand, in GARO, r should be odd. Actually, these inverters can consist of an odd number of elementary inverter logic gates, k.

FIRO-based and GARO-based circuit structures are given in Figures 4.5 and 4.6, respectively. FIRO and GARO are defined by an associated feedback polynomial. The feedback connections are chosen to make sure that there are no fixed points in the corresponding state-transition function. Therefore, the $f(x)$ polynomial should be chosen as $f(x) = (1 + x)h(x)$, where $h(x)$ is a primitive polynomial [94], and should

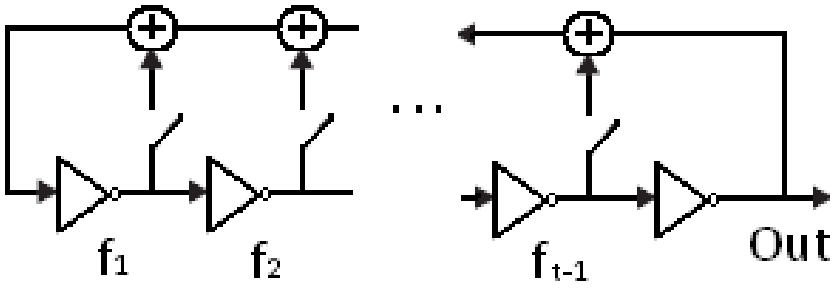

Figure 4.5. Fibonacci ring oscillator.

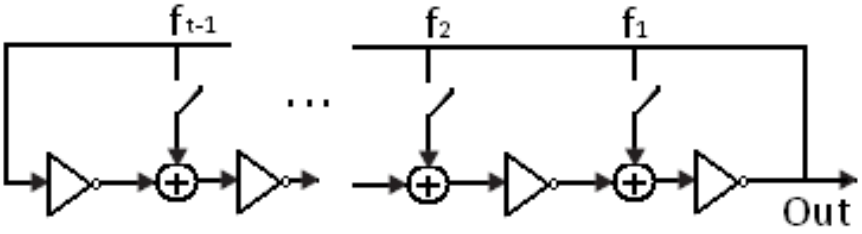

Figure 4.6. Galois ring oscillator.

meet the condition of $h(1) = 1$. In addition, the feedback coefficients of f_0 and f_r should meet the condition of $f_0 = f_r = 1$.

Although the output of GARO is taken from the last inverter, in FIRO, it can be taken from any inverter in the cascade. For both configurations, the output oscillating signal is irregular and comprises both pseudo randomness and true randomness. True randomness is due to the random delay and transition times of all the logic gates in the circuit, which are the result of thermal and shot noises in the circuit itself. In order to increase the randomness and robustness, another configuration that combines the properties of both structures can be formed by XORing FIRO and GARO. It is recommended in [48] that the lengths of the two oscillators should preferably be mutually prime. This maximizes the period of the random sequence and minimizes the interlocking and coupling effect.

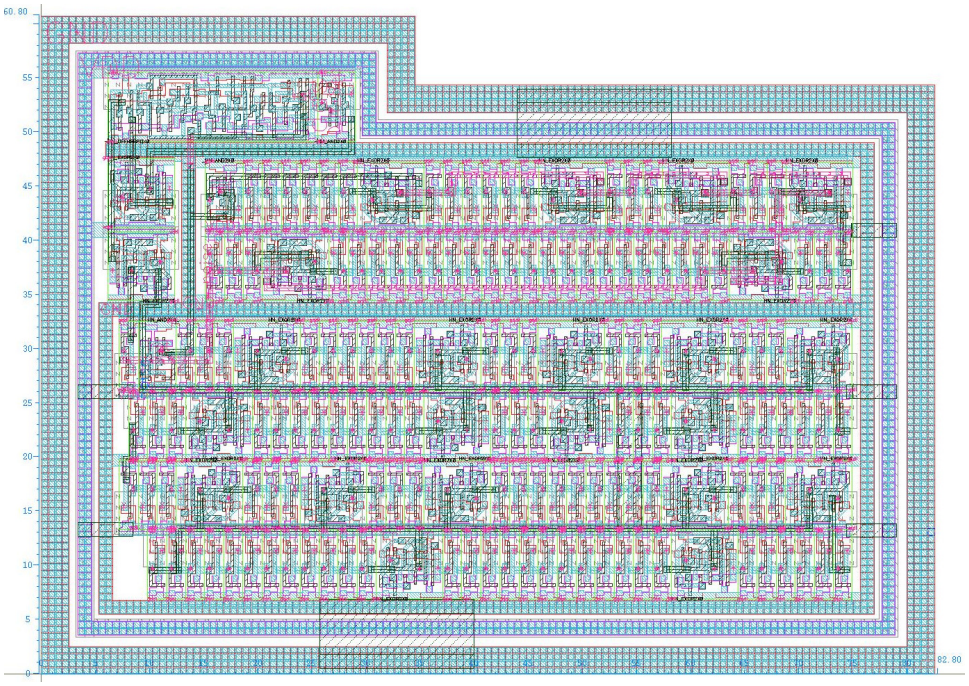

Figure 4.7. Layout of FIGARO.

The generated output signal will be a high-speed, noise-like irregular signal that combines true and pseudo randomness. Therefore, it is mentioned in [48] that mutual coupling will be decreased, and consequently, primary randomness will be increased. Moreover, due to quick propagation of jitter and its transformation through feedback, sensitivity to jitter significantly increases, which is desirable for generation of random bit streams. Binary sequences can be obtained by using a DFF or an intermediate TFF, which reduces the bias.

Distinguishing true and pseudo randomness is an important issue in digital random number generation. In order to do this, a restart test was applied in [35] 1000 times and standard deviation for all traces was calculated. It is observed that the traces deviate from each other, which is the proof of true randomness. In addition, autocorrelation, entropy estimations, and data acquisition rates with proper bias are presented in [35].

4.2.2. Implementation Details, Measurement and Statistical Test Results

HHNEC's 0.25μm eFlash process with a supply voltage of 2.5V is used for implementation and fabrication of the prototypes. In order to prevent coupling and correlation between adjacent FIROs and GAROs as much as possible, isolated n-well technique is applied. Regarding layout design techniques, isolated n-well technique reduces the leakage between substrate junctions, and substrate noise, which improves the mutual effects of grounds between oscillators. The layout of the implemented FIGARO circuit is seen in Figure 4.7. The total circuit area of the implemented circuit excluding pads is $0.0048mm^2$. On-chip guard rings are used between oscillators and other prototypes. Fabricated chip photo can be seen in Figure 4.8.

The designed circuitry consists of a FIRO that realizes the feedback polynomial of $x^{15} + x^{14} + x^7 + x^6 + x^5 + x^4 + x^2 + 1$ and a GARO which realizes the feedback polynomial of $x^{31}+x^{27}+x^{23}+x^{21}+x^{20}+x^{17}+x^{16}+x^{15}+x^{13}+x^{10}+x^9+x^8+x^6+x^5+x^4+x^3+x+1$. Every inverter is essentially formed by three elementary inverter logic gates. A sampler and a post processor after the sampler are the other components of the circuitry. Post processing circuitry is not implemented as hardware. It is planned to be implemented as software in required cases. Due to its easier implementation and simple functionality, von Neumann corrector is utilized in this design as a post processor.

A bit stream, which has a length of $100MBytes$, was acquired through the PCI interface of the FPGA based hardware without von Neumann post processing. The acquired bit stream, which exhibits considerable amount of bias even at low frequencies such as 100 kHz, is subjected to the full NIST 140-2 test suite [29]. However, it does not pass from the statistical tests. After applying the von Neumann corrector in software to the binary data obtained from FIGARO at $100kHz$, processed binary data passes the tests of the full NIST random number test suite except for the runs test. Von Neumann's de-skewing technique [93] is used in order to eliminate the existing bias in the output sequence.

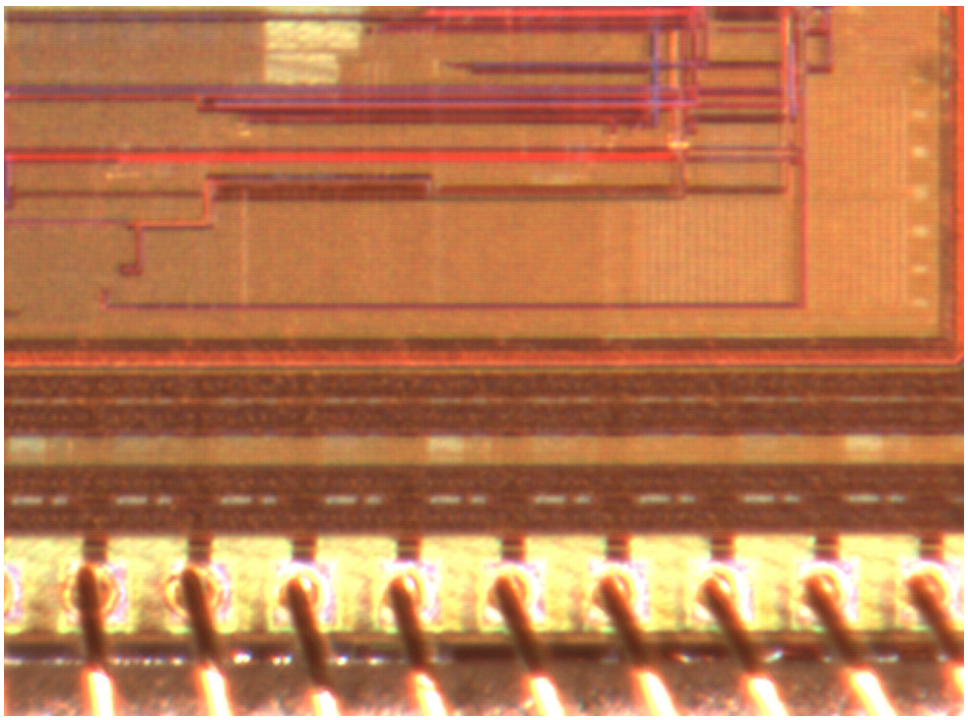

Figure 4.8. Chip photo.

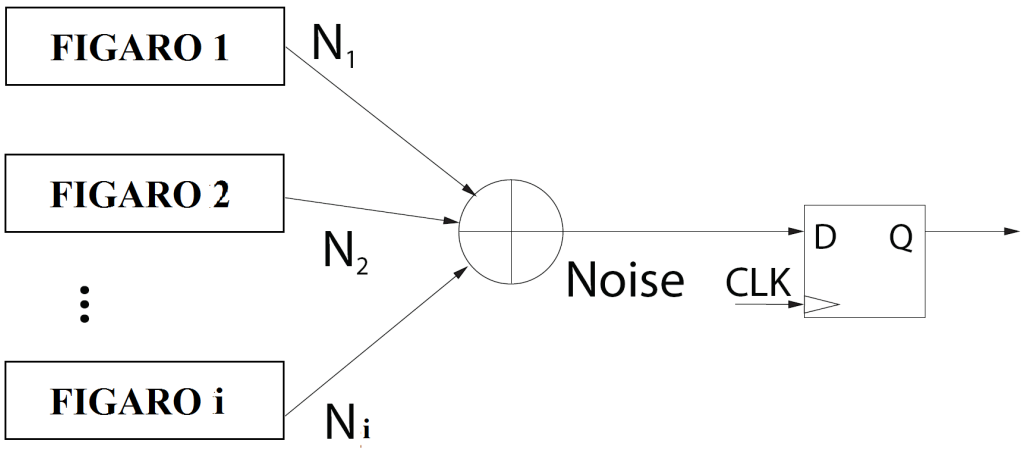

Figure 4.9. Proposed configuration.

Using more than one FIGARO in XORed configuration is proposed in order to improve the quality of randomness and increase the frequency (hence, the throughput) of the output bit stream. The proposed configuration is demonstrated in Figure

Table 4.3. Number of required FIGARO and corresponding sampling frequency.

Frequency (MHz)	**Number of FIGARO**	
	Processed Data	**Raw Data**
125	7	8
62.5	6	7
31.25	5	6
12	4	5
5	3	4
1	2	3

4.9. Table 4.3 gives the required number of XORed FIGARO and the corresponding sampling frequency for passing the tests of NIST 140-2 for both raw data and processed data with the von Neumann corrector. It is observed that XORing seven FIGARO provides a bitstream which passes from NIST 140-2 test suite after a simple corrector (von Neumann) with a sampling frequency of $125MHz$. This will supply a $31.25Mbps$ throughput rate. Results of the tests are given in Table 4.4 with pass rates and p-values. The minimum pass rate for each statistical test with the exception of the random excursion variant test is approximately 0.968840 for a sample size of $199\times1Mbit$ binary sequences. The minimum pass rate for the random excursion variant test is approximately 0.962751 for a sample size of $120\times1Mbit$ binary sequences.

Moreover, it is derived that if the number of FIGAROs is further increased to 8, fulfilled test results are also obtained from NIST 140-2 test suite without any post processing stage with a sampling frequency of $125MHz$. Since no postprocessing is needed, no throughput reduction will occur. This throughput rate is 62.5 times higher than the previous design results shown in [5] and [49]; additionally, no post-processing stage is required. Results of the tests are given in Table 4.5 with pass rates and p-values. The minimum pass rate for each statistical test with the exception of the random excursion variant test is approximately 0.972766 for a sample size of 300×1Mbit binary sequences. The minimum pass rate for the random excursion variant test is approximately 0.968113 for a sample size of 186×1Mbit binary sequences. The oscilloscope screen shot shows the measured output of FIGARO sampled at $125MHz$ of

Table 4.4. Statistical test results of seven XORed FIGARO with processed data.

Statistical Test	P Values	Pass Rates
Frequency	0.860922	0.9899
Block Frequency	0.210358	0.9849
Cumulative Runs	0.815583	1.0000
Runs	0.786230	0.9799
Longest Run	0.869361	1.0000
Rank	0.261554	0.9950
FFT	0.539459	0.9950
Nonperiodic Templates	0.986590	0.9849
Overlapping Templates	0.528851	1.0000
Universal	0.626161	0.9950
Apen	0.950449	1.0000
Random Excursions	0.834308	0.9917
Random Excursion Variants	0.941144	0.9833
Serial	0.713319	1.0000
Lempel Ziv	0.024313	0.9799
Linear Complexity	0.518316	0.9899

clock frequency in Figure 4.10.

In fact, XORing FIGAROs gives the flexibility of choosing the configuration in terms of area, power, and speed. One FIGARO block consumes approximately $1mA$ of current. Therefore, dissipated power can be estimated according to the chosen configuration given in Table 4.3. For example, a configuration, with a sampling frequency of $125MHz$, needs eight FIGAROs to be XORed and consumes $8mA$ of current. A digital RNG based on ring oscillators and implemented with the same size of process technology is reported in [16]. This RNG, which has a sampling frequency of $66MHz$ and a von Neumann post processor, dissipates $40mW$ of power and occupies $0.043mm^2$ of area. Although the proposed RNG occupies 1.13 times less area ($0.038mm^2$), it has a 7.5 times faster throughput and dissipates less power ($20mW$). A summary of performance parameters for eight XORed FIGAROs is also presented in Table 4.6.

Table 4.5. Statistical test results of eight XORed FIGARO with raw data.

Statistical Test	P Values	Pass Rates
Frequency	0.561227	0.9800
Block Frequency	0.000839	0.9700
Cumulative Runs	0.840081	0.9833
Runs	0.083867	0.9633
Longest Run	0.037566	0.9800
Rank	0.822534	0.9967
FFT	0.035174	0.9967
Nonperiodic Templates	0.992478	0.9933
Overlapping Templates	0.487885	0.9867
Universal	0.862344	0.9900
Apen	0.969347	0.9933
Random Excursions	0.964295	0.9785
Random Excursion Variants	0.911413	1.0000
Serial	0.810470	0.9967
Lempel Ziv	0.494392	0.9900
Linear Complexity	0.449672	0.9900

Table 4.6. Summarized performance parameters of FIGARO.

Parameters	Values
Maximum Clock Speed for Passing Tests	125 MHz
Maximum Speed of Throughput	125 Mbps
Typical 1/0 Ratio of Raw Data	1.000117
Size	$0.038mm^2$
Power Dissipation @ 2.5V	$20mW$
Supply Voltage	2.5V
Fabrication Technology	0.25μm eFlash

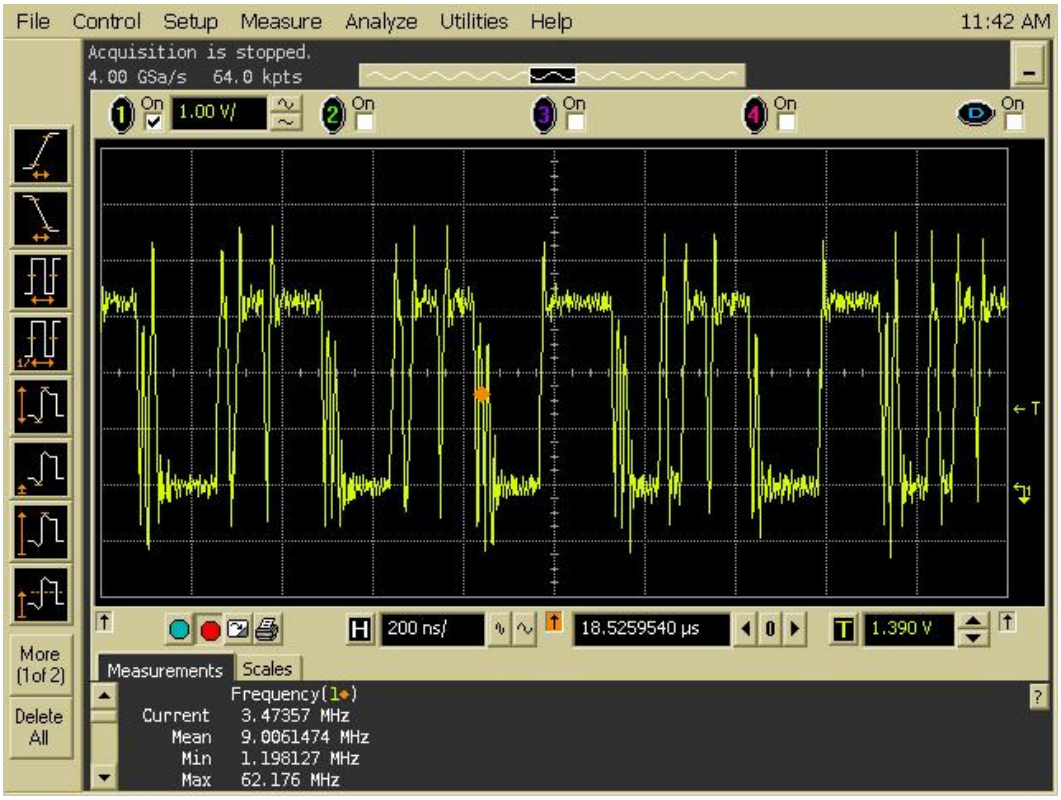

Figure 4.10. Measured output of FIGARO.

4.3. A High Speed, Fully Digital IC Random Number Generator

Measurement results of the ASIC implementation of [49] show that over $18.5 Mbps$ throughput with fulfilled test results of NIST 140-2 test suite can be achieved after a simple corrector (von Neumann corrector) and without the complex post-processing stage. Furthermore, doubling the number of ring oscillators produces output bit stream with $125 Mbps$ throughput which passes the full NIST 140-2 tests without any post processing.

Implementation details are given in Section 4.3.1. Section 4.3.2 exhibits measurement results, analysis of results, and the statistical test results.

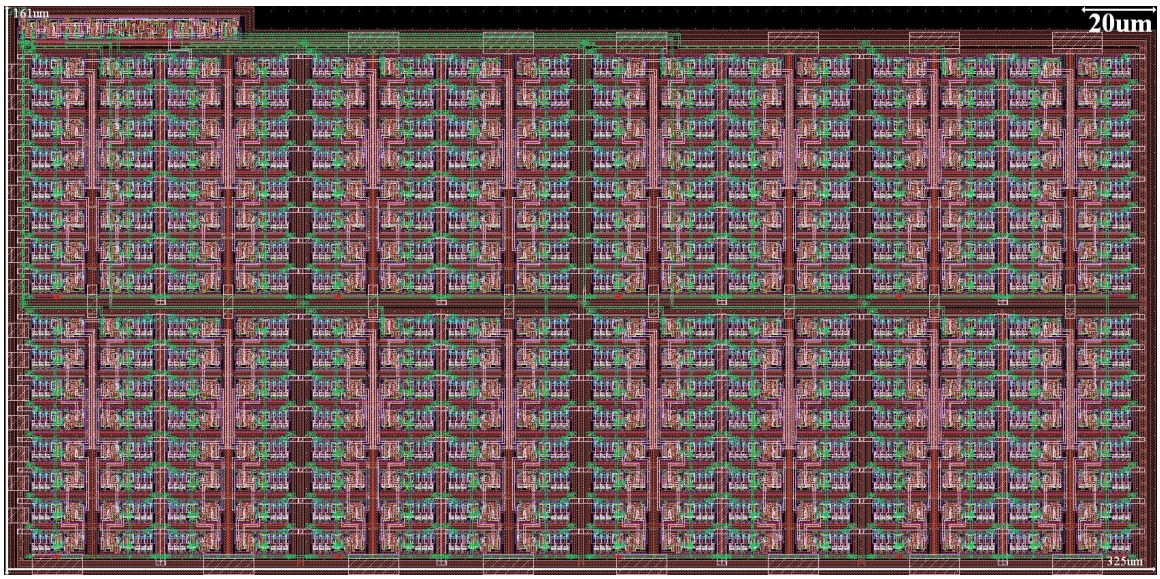

Figure 4.11. Layout of ring oscillators design.

4.3.1. Implementation Details

HHNEC's 0.25 μm eFlash process with a supply voltage of $2.5V$ technology is used for implementation and fabrication of the prototypes. All of the layout design techniques mentioned in Subsection 4.1.1 are applied in this design as well. The layout of the design can be seen in Figure 4.11. The designed circuitry consists of 256 ring oscillators which have three inverters per ring, a sampler, and a post-processor after the sampler. Post processing circuitry is not implemented as hardware.

XOR gates were placed very close to the ring oscillators which were connected to them. This minimizes the coupling of long running adjacent interconnects. To differentiate the path delays, the routing of paths from ROs to XOR gates are drawn slightly different. Changing the wire length will yield a difference between the loading capacitances of ring oscillators which goes to different inputs of XOR gate. This will also lead to a delay between the adjacent signals. Even if the ring oscillators couple to each other, this delay will vary the arrival times of signals to inputs of XOR gate. Hence, extraction of bits is increased by the help of the delay which not only leads a decreased correlation between input pairs of XOR but also produces an increase in

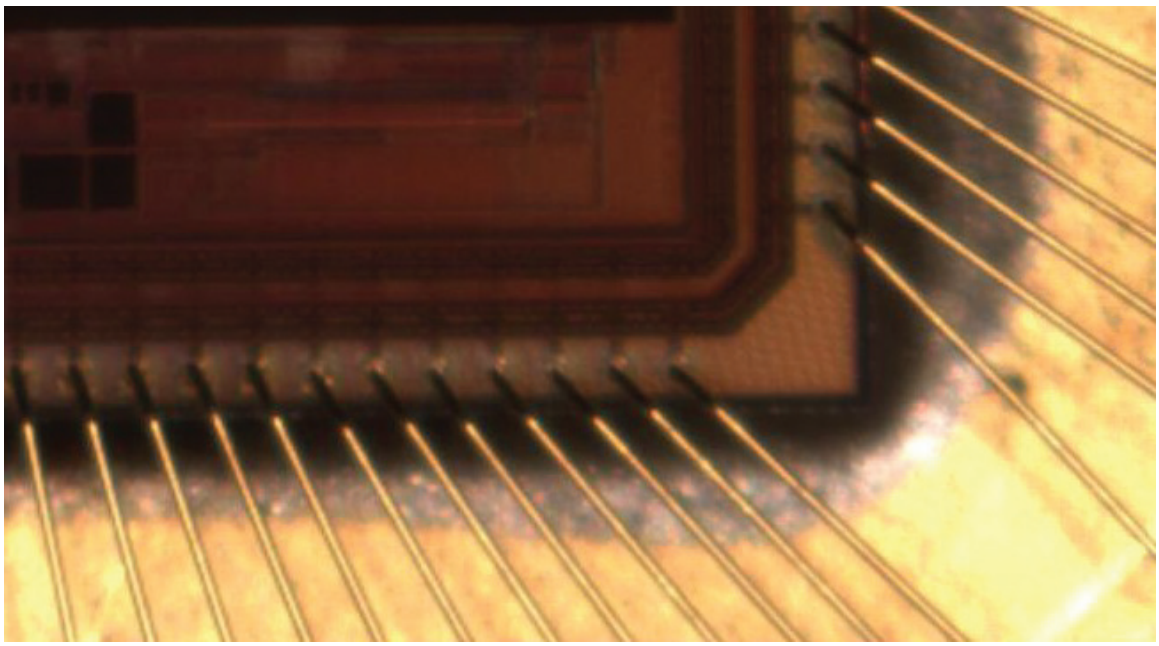

Figure 4.12. Chip photo.

entropy extraction.

4.3.2. Measurement and Statistical Test Results

The total circuit area of design excluding pads is $0.052mm^2$. On-chip guard rings are used between ring oscillators and other prototypes. Fabricated chip photo can be seen in Figure 4.12. The same data acquisition board used in Subsection 4.1.2 is also utilized in this measurement.

In this design, the ring oscillators consist of three inverters. 3-inverter based ring oscillator used in [17] will have more jitter although it consumes more current than 13-inverter based ring oscillator used in [16]. The detailed analysis on jitter is performed in the next chapters.

In addition, if we consider the effect of process technology on jitter, roughly we can say that the attained frequency increases with the shrinking technologies. As a matter of fact, newer technologies have more jitter than the others. Since the proposed

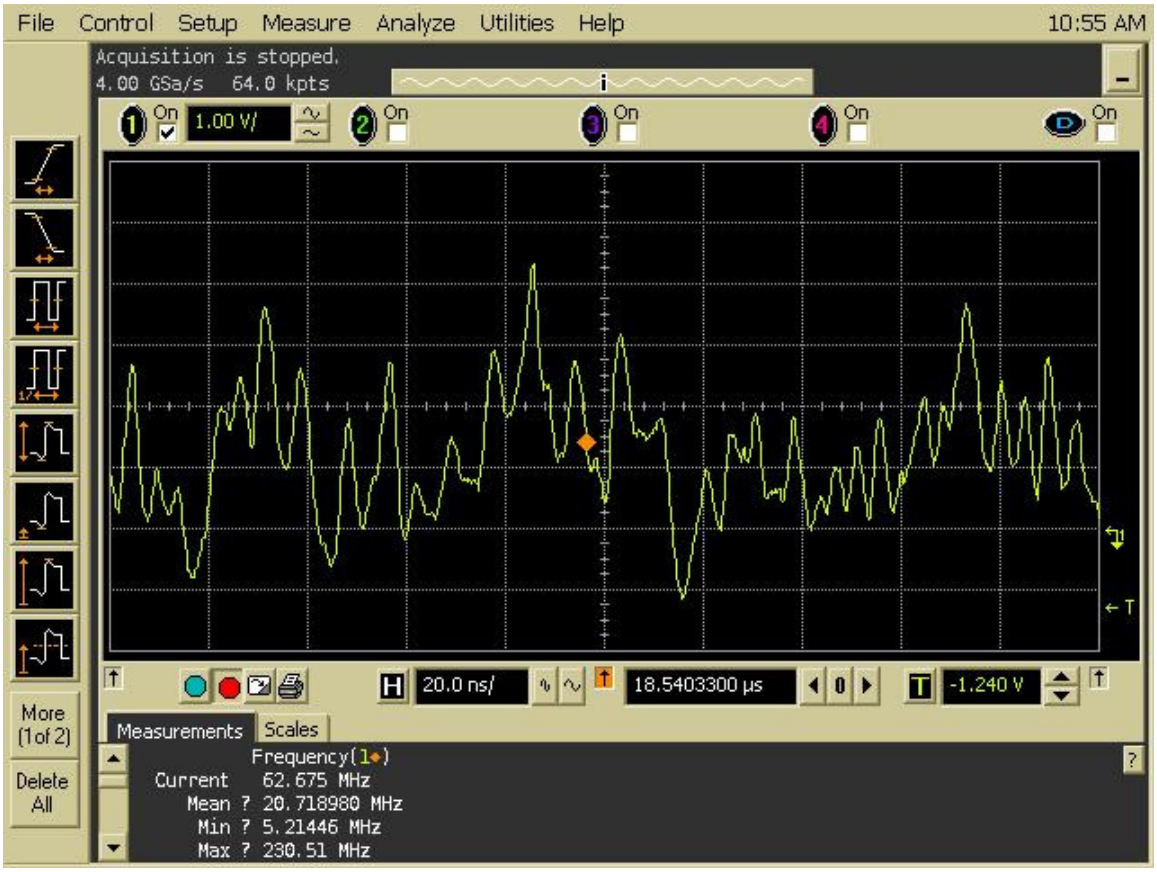

Figure 4.13. Measured output of XOR tree of 256 ROs.

RNG is designed with 0.25μm technology, which can be counted as current technology, the obtained jitter is at average level.

A ring oscillator with three inverters in a ring will exhibit a frequency around $750MHz$ which makes it very tough to observe. As a consequence of higher amount of jitter, fulfilled test results without strong resilient functions could be achieved. The output of 256 XORed ring oscillators can be seen in Figure 4.13. According to the histogram of output voltage of 256 XORed ring oscillators, which is given in Figure 4.14, the output voltage exhibits a proper distribution in terms of bias.

A bit stream of length $160MBytes$ was acquired through the PCI interface of the FPGA based hardware without von Neumann processing. The obtained bits were processed by the full NIST 140-2 test suite [29]. After applying the von Neumann

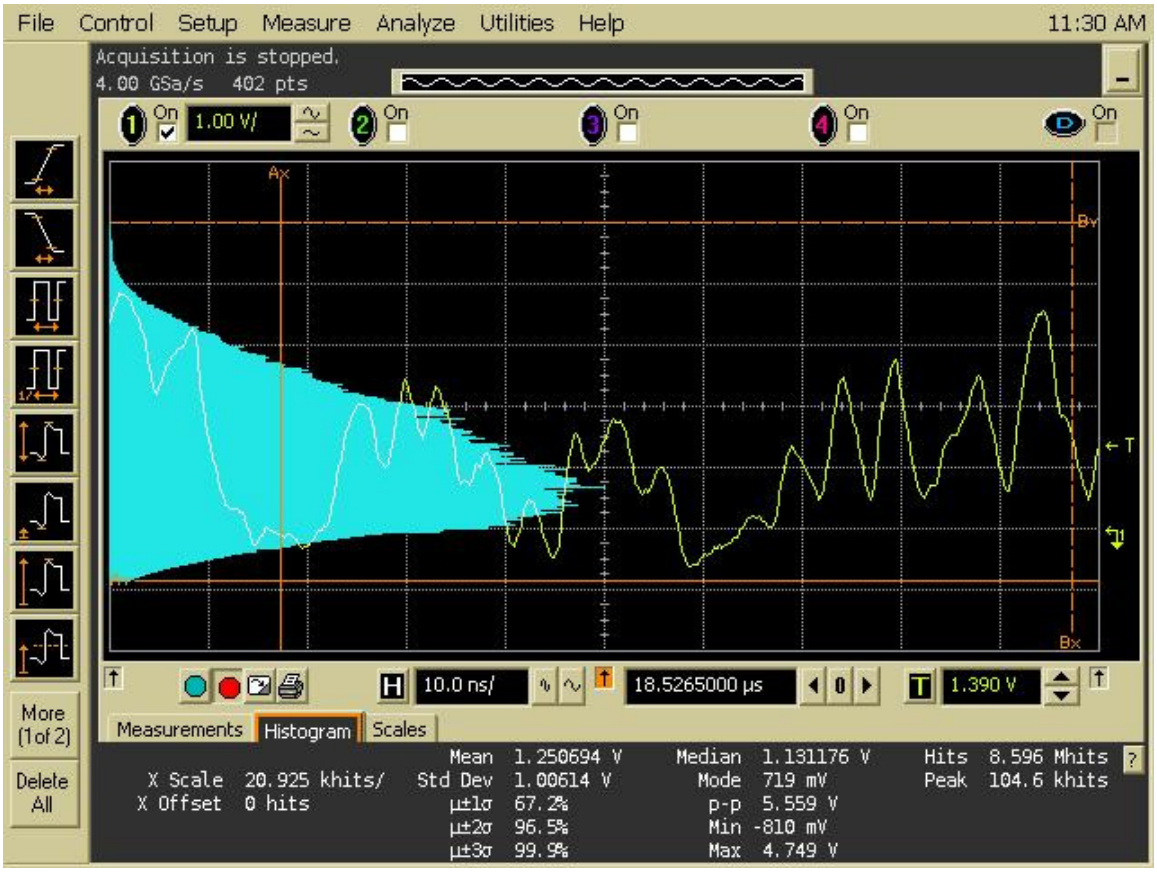

Figure 4.14. Measured output of XOR tree of 256 ROs with histogram.

corrector to the binary data obtained from these ring oscillators in the software, the processed binary data passed the tests of full NIST random number test suite.

Table 4.7 presents the test results with pass rates and p-values. The minimum pass rate for each statistical test with the exception of the random excursion (variant) test is approximately 0.973314 for a sample size of 320×1MBit binary sequences. The minimum pass rate for the random excursion (variant) test is approximately 0.969050 for a sample size of 203×1MBit binary sequences. More than 10 randomly selected chips are subjected to full NIST 140-2 test suite. All of them pass full NIST tests.

Moreover, two random number generators with 256 ring oscillators and three inverters in a ring were XORed and the output bit stream was subjected to full NIST 140-2 test suite. It is seen that the raw data passed the tests of the full NIST ran-

Table 4.7. Statistical test results of RO-based RNG with 256 ROs.

Statistical Test	P Values	Pass Rates
Frequency	0.288780	0.9812
Block Frequency	0.870659	1.0000
Cumulative Runs	0.330628	0.9812
Runs	0.017156	0.9812
Longest Run	0.180322	0.9875
Rank	0.758528	0.9906
FFT	0.578763	0.9875
NonOverlapping Template	0.986869	0.9906
Overlapping Templates	0.239112	0.9812
Universal	0.572333	0.9906
Approximate Entropy	0.496841	0.9938
Random Excursions	0.908091	0.9901
Random Excursion Variants	0.863690	0.9951
Serial	0.302291	0.9969
Linear Complexity	0.643627	0.9938

dom number test suite without any post processing. The test results are presented in Table 4.8. The minimum pass rate for each statistical test with the exception of the random excursion (variant) test is approximately 0.973314 for a sample size of 320×1MBit binary sequences. The minimum pass rate for the random excursion (variant) test is approximately 0.968569 for a sample size of 194×1MBit binary sequences. In order to determine the quality of random source, it is important for a configuration to pass the statistical tests without any post processing. This shows that the random source has enough entropy to be used as a true RNG. In other words, if the circuit would be designed with 512 ring oscillators instead of 256 ring oscillators, it would pass the statistical tests without any post processing. The design has successfully achieved a maximum throughput of $125Mbps$, since there was no need for post processing of the raw data, sampled at a frequency of $125MHz$, would yield as $125Mbps$ of throughput. Of course, these improvements have been accomplished at the cost of two fold increase

Table 4.8. Statistical test results of the RO-based RNG with 512 ROs

Statistical Test	P Values	Pass Rates
Frequency	0.875539	0.9844
Block Frequency	0.559523	0.9938
Cumulative Runs	0.392456	0.9906
Runs	0.782780	0.9969
Longest Run	0.239112	0.9906
Rank	0.293235	0.9906
FFT	0.752361	0.9906
NonOverlapping Template	0.993405	0.9933
Overlapping Template	0.042290	0.9812
Universal	0.177264	0.9906
Approximate Entropy	0.460664	0.9938
Random Excursions	0.729339	0.9897
Random Excursion Variants	0.952942	0.9897
Serial	0.823278	0.9844
Linear Complexity	0.302291	0.9938

in current and device area.

In most RNG applications, a hash function such as SHA-1 or a cryptographic algorithm such as Advanced Encryption Standard (AES) or Data Encryption Standard (DES) is applied to a truly random seed in order to obtain high speed random sequences. However, random numbers used in cryptographic applications should satisfy strict randomness criteria such as being produced truly in a non-deterministic fashion. Recently, communication speed among crypto-based IPs has reached the GHz range, for example an IPSec device given in [95] utilizes a true random number generator with a $200Mbps$ throughput. Therefore, in order to fulfill the requirements of high speed secure IP communication equipment, RNG should offer high speed, truly random keys. Since, the proposed design achieves an increased throughput of $125Mbps$; it is suitable to be used in high speed secure communication, where power consumption is not a

critical issue.

Furthermore, in Table 4.9, there is a comparison between well-known RNGs in the literature in terms of power consumption, area, and throughput. In addition, a comparison is conducted to provide the same throughput from each RNG. In order to achieve the same throughput, given circuits are used in parallel structures. Therefore, some of comparison results are presented according to the replicated structures.

Table 4.9. Comparison of the ASIC implemented RNGs with some well-known RNGs in the literature.

	[3]	**[4]**	**[2]**	**RO-based RNG with 512 ROs (ASIC of [49])**	**RO-based RNG with 114 ROs (ASIC of [5])**	**FIGARO -based RNG (ASIC of [36])**
Technology	0.18μm CMOS 1.8V	0.18μm CMOS 3.3V	2μm CMOS 3V	0.25μm eFlash 2.5V	0.25μm eFlash 2.5V	0.25μm eFlash 2.5V
Speed	10MHz	10MHz	1.4MHz	N/A	256MHz	N/A
Area	0.025mm^2	0.016mm^2	1.5mm^2	0.104mm^2	0.043mm^2	0.034mm^2
Power Consumption	3.6mW	2.3mW	3.9mW	190mW	40mW	20mW
Post-Processing	Yes (XOR decorrelating)	No	No	No	Yes	No
Final Throughput Rate	10$Mbps$	10$Mbps$	1.4$Mbps$	125$Mbps$	16.5$Mbps$	125$Mbps$
Replication Number	13	13	90	1	8	1
Power Consumption of Replicated Circuit	46.8mW	29.9mW	351mW	190mW	320mW	20mW
Area of Replicated Circuit	0.325mm^2	0.208mm^2	135mm^2	0.104mm^2	0.344mm^2	0.034mm^2

Table 4.10. Summarized performance parameters of 256 XORed ROs.

Parameters	**Values**
Maximum Speed of Output Bit Stream	310 MHz
Maximum Clock Speed for Passing Tests	74 MHz
Maximum Speed of Throughput	18.5 Mbps
Typical 1/0 Ratio of Raw Data	1.01716
Typical 1/0 Ratio after von Neumann Corrector	1.00013
Size	0.052 mm^2
Power Dissipation @ 2.5V	95 mW
Power Dissipation @ 1.6V	27 mW
Supply Voltage	2.5V
Fabrication Technology	HHNEC's 0.25 μm eFLash process

In comparison to [3], the proposed design has three times area advantage; however, it has a disadvantage of four fold power consumption. Moreover, [3] needs post processing in order to pass the statistical test, which is not necessary for the proposed design. [4] has 6.4 times superior behavior than this work in terms of power consumption; however, its area is two times greater. The proposed design is almost 1300 times smaller than [2]. Of course, one should take the technology differences into account in these designs; however, it should also be considered that [2] has big analog modules such as analog to digital converter (ADC), and sample and hold circuitry which will occupy relatively big area compared to the other RNGs given above in the same technology. [2] also has the disadvantage of two times higher power consumption than the proposed circuitry.

Additionally, statistically unaffected raw data can be obtained until the supply voltage reaches 1.6 V. Power consumption decreases from 95mW to 27mW while supply voltage is decreased from 2.5V to 1.6V. Therefore, supply voltage should be

adjusted to its minimum level if the power consumption is a critical design constraint. However, reducing supply voltage in favor of current consumption will yield to a reduction in speed, too. A summary of performance parameters is also presented in Table 4.10.

4.4. Conclusions

Three high speed, fully digital different IC RNGs have been implemented and fabricated with HHNEC's 0.25μm eFlash process with a supply voltage of $2.5V$. Two of the implemented RNG circuitries are based on ring oscillators modeled in [5], one of them is based on Fibonacci and Galois sing oscillators modeled in [36]. The circuit given in [17] occupies $0.052mm^2$ and dissipates $95mW$ of power. On the other hand, the circuit given in [16] occupies 0.043 mm^2 and dissipates 40 mW of power at $2.5V$ supply voltage. If lower frequencies such as $50MHz$ is enough for the application that the RNG is used for, it is suggested that the use of smaller supply voltages.

The raw bit streams generated with the circuit [17] behave nearly randomly before post processing for up to $74MHz$ clock frequencies and for up to 66 MHz clock frequencies with the circuit in [16]. It is proposed to use von Neumann corrector instead of resilient function, thus four times faster throughput can be obtained. After applying the von Neumann corrector, the circuit in [17] fulfills randomness test of NIST 140-2 with $18.5Mbps$ throughput and the circuit in [16] with 16.5 $Mbps$ throughput.

The correlation between the ring oscillators in [16] was investigated and the correlation coefficient was calculated as 0.0073 between two 32 XORed ROs outputs of 32,000 bits. The measurement results showed that the suggested design parameters in [5], number of inverters in an RO and number of ring oscillators in an RNG, are satisfies the highly qualified random bits without strong post processing. In addition, it is recommended utilizing the ring oscillators with a doubled number [17]. Fulfilled test results can be obtained via this configuration without any post processing. A $125MHz$ throughput is attained without any post processing by using doubled ring oscillators.

In [18] a new configuration, which utilizes FIGAROs in an XORed structure, is proposed. This configuration gives a flexibility of choosing the RNG according to the design constraints. Fulfilled test results of NIST 140-2 are accomplished with a eight XORed FIGARO structures without any further post processing. This structure provides a throughput of $125Mbps$. Furthermore, it consumes 20 mW of power at $2.5V$ supply voltage and occupies 0.038 mm^2 area. FIGARO based RNG consumes less power and occupies less area compared to the RNGs implemented in [16] and [17] which have also successful test results among the digital gate based RNGs.

5. PHASE NOISE AND JITTER MODELS OF CMOS RO IN STRONG INVERSION REGION

Timing jitter and phase noise studies of oscillators were initially performed and developed until the beginning of 1980s [96–98] and rapidly became very accurate and complex towards the mid 1990s [72, 74, 99]. However, a simple and accurate phase noise and timing jitter model of ROs became a serious requirement due to their wide usage. Finally, a simple and coherent model was published in 2006 by Abidi in [1].

Phase noise indicates random accelerations and decelerations in phase (ϕ) as an oscillator orbits at a nominally constant frequency (f_0) in steady-state. Jitter arises from sampling the orbit at certain points [1]. If DC period is defined as τ, then period jitter is the standard deviation (σ_τ) around its mean value.

In this chapter, phase noise and jitter models for strong inversion region will be summarized as given in [1]; furthermore, jitter due to flicker noise, which is not included in [1], will be derived for both IbRO and DRO. In order to derive flicker noise induced jitter, a link is defined between flicker noise induced phase noise and jitter similar to the link given in Equation 5.2 for white noise induced jitter. Finally, phase noise and jitter measurements are given.

5.1. Link between White Noise induced Jitter and Phase Noise

The link between white noise induced *jitter* and *phase noise* is given as

$$\sigma^2{}_\tau = \int_0^\infty \mathbf{S}_\tau(\mathbf{f})\mathbf{df} = \int_0^\infty \mathbf{S}_\phi(\mathbf{f})\frac{\sin^2(\frac{\pi \mathbf{f}}{\mathbf{f_0}})}{(\pi \mathbf{f_0})^2}\mathbf{df} \tag{5.1}$$

by using Wiener-Khinchine theorem [100], where $\sigma^2{}_\tau$ is the mean-square value of jitter, S_τ is the power spectral density of the period, and $S_\phi(f)$ is power spectral density of phase.

Single Side Band (SSB) phase noise power spectral density a.k.a Lorentzian spectrum is simplified as

$$\mathbf{L(f)} = \sigma^{\mathbf{2}}{}_{\tau}\frac{\mathbf{f_0^3}}{\mathbf{f^2}} \tag{5.2}$$

under the assumption of white noise sources alone [101].

5.2. Link between Flicker Noise induced Jitter and Phase Noise

It was mentioned in [1] that the integral in Equation 5.1 is not so easy to solve; hence, jitter due to flicker noise is not included. However, it is indicated in [7] that an approximate solution is available. Based on these two studies, jitter equations due to flicker noise are derived.

According to [7], jitter due to flicker noise is

$$\sigma^{\mathbf{2}}{}_{\tau}^{\mathbf{1/f}} = \int_{\mathbf{0}}^{\infty} \mathbf{S}_{\phi}^{\mathbf{1/f}}(\mathbf{f})\mathbf{df} = \int_{\sim\mathbf{0}}^{\infty} \frac{\mathbf{2N_1 f_c}}{\mathbf{f^3}} \frac{\sin^{\mathbf{2}}(\pi \mathbf{f \Delta T})}{(\pi \mathbf{f_0})^{\mathbf{2}}}\mathbf{df}. \tag{5.3}$$

Since we are dealing for one period, $\Delta T = \frac{1}{f_0}$. f_c is the corner frequency, where flicker noise and white noise meet. In [7] $\frac{N_1 f_c}{f^3}$ corresponds to $-30dB/dec$ slope in the phase noise plot.

Lorentzian phase noise is given in [1] as

$$\mathbf{L(f)} = \frac{\mathbf{S_\phi(f)}}{\mathbf{2}}. \tag{5.4}$$

Hence, Lorentzian phase noise due to flicker noise becomes

$$\mathbf{L^{1/f}(f)} = \frac{\mathbf{S_\phi^{1/f}(f)}}{\mathbf{2}} = \frac{\mathbf{N_1 f_c}}{\mathbf{f^3}}. \tag{5.5}$$

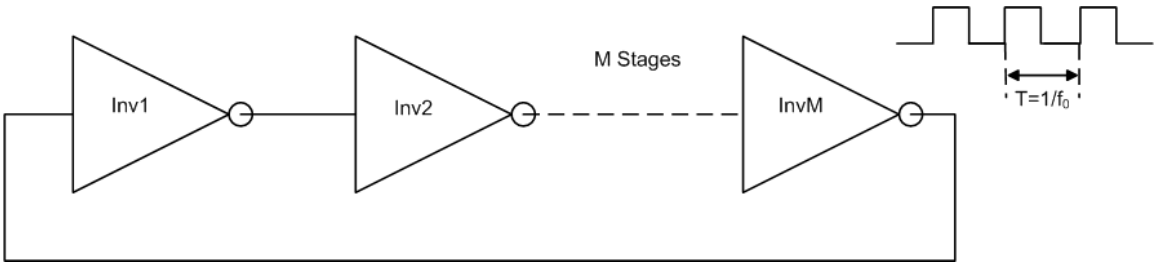

Figure 5.1. Inverter-based ring oscillator.

In [7], the standard deviation of jitter due to flicker noise is given as

$$\sigma_{\tau}^{\mathbf{1/f}} = \mathbf{5}\kappa\sqrt{\mathbf{f_c}}\mathbf{\Delta T}, \tag{5.6}$$

where ΔT is $\frac{1}{f_0}$ for one period and $\kappa = \frac{\sqrt{N_1}}{f_0}$. Rearranging terms and taking the square, jitter due to flicker noise is obtained as

$$\sigma^{\mathbf{2}}_{\tau}{}^{\mathbf{1/f}} = \mathbf{25}\frac{\mathbf{N_1 f_c}}{\mathbf{f_0^4}}. \tag{5.7}$$

When Equations 5.5 and 5.7 are combined, link between flicker noise induced phase noise and jitter is obtained as

$$\mathbf{L^{1/f}(f)} = \mathbf{25}\sigma^{\mathbf{2}}_{\tau}{}^{\mathbf{1/f}}\frac{\mathbf{f_0^4}}{\mathbf{f^3}}. \tag{5.8}$$

5.3. Inverter-based Ring Oscillator

Figure 5.1 demonstrates the inverter delay cell based ring oscillator. IbRO should consist of an odd number of inverter stages. The detailed circuitry of an inverter and its functionality are given in Figure 5.3 and Figure 5.2, respectively.

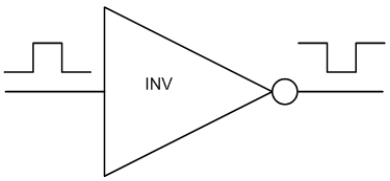

Figure 5.2. Inverter symbol.

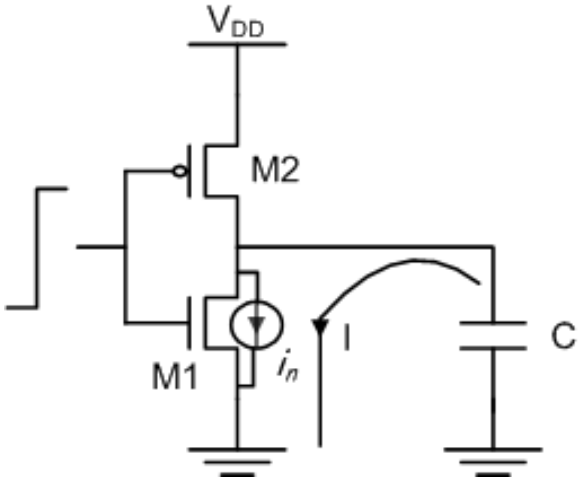

Figure 5.3. Transistor based architecture of inverter.

5.3.1. Phase Noise and Jitter due to White Noise

Three approximations are used for simplifying the nominal oscillation frequency (f_0) of an inverter-based ring oscillator:

(i) Propagation delays of NMOS and PMOS devices are assumed to be equal ($t_{dN} = t_{dP} = t_d$).

(ii) Pull-down and pull-up currents are assumed to be equal ($I_N = I_P = I$).

(iii) Switching point of inverter is supposed to occur at $\frac{V_{DD}}{2}$.

After simplification, f_0 becomes

$$\mathbf{f_0 = \frac{1}{M(t_{dN} + t_{dP})} \simeq \frac{2}{MCV_{DD}}(\frac{1}{I_N} + \frac{1}{I_P})^{-1} \simeq \frac{I}{CMV_{DD}}}, \tag{5.9}$$

where V_{DD} is the supply voltage, M is the number of stages, and C is the load capacitance. For the sake of simplicity, it is assumed that the load capacitance of C consists of only gate capacitance.

Noise in the pull-up and pull-down processes adds jitter to propagation delay in every stage. Since noise events are uncorrelated, the mean-squares of their variances can be added as

$$\sigma_{\tau}^{\mathbf{2}} = \mathbf{M}(\sigma_{\mathbf{tdN}}^{\mathbf{2}} + \sigma_{\mathbf{tdP}}^{\mathbf{2}}). \tag{5.10}$$

If we assume that threshold voltages are equal for both PMOS and NMOS transistors ($V_{tN} = V_{tP} = V_t$), the variance of period jitter is obtained as

$$\sigma_{\tau}^{\mathbf{2}} = \frac{\mathbf{2kT}}{\mathbf{If_0}}\left(\frac{\mathbf{1}}{\mathbf{V_{DD} - V_t}}(\gamma_{\mathbf{N}} + \gamma_{\mathbf{P}}) + \frac{\mathbf{1}}{\mathbf{V_{DD}}}\right), \tag{5.11}$$

where γ_N and γ_P are the noise coefficients of NMOS and PMOS transistors.

By using Equations 5.11 and 5.2, SSB phase noise due to white noise is obtained as

$$\mathbf{L(f)} = \frac{\mathbf{2kT}}{\mathbf{I}}\left(\frac{\mathbf{1}}{\mathbf{V_{DD} - V_t}}(\gamma_{\mathbf{N}} + \gamma_{\mathbf{P}}) + \frac{\mathbf{1}}{\mathbf{V_{DD}}}\right)\frac{\mathbf{f_0^2}}{\mathbf{f^2}}. \tag{5.12}$$

5.3.2. Phase Noise and Jitter due to Flicker Noise

Noise on the frequency is closely related with current flowing on transistors; therefore, the current sensitivity K_I ($\frac{f_0}{I}$) on the nominal frequency of f_0 is used to

derive the flicker noise component of phase noise as

$$\mathbf{L^{1/f}(f) = \frac{K_I^2}{4f^2} S_I(f)}, \tag{5.13}$$

$$\mathbf{L^{1/f}(f) = \frac{1}{16MI^2}(S_{iN}^{1/f} + S_{iP}^{1/f})(\frac{f_0}{f})^2}, \tag{5.14}$$

where M is the number of stages of the RO. Flicker noise in the drain current is equal to

$$\mathbf{S_{i_n}^{1/f} = g_m^2 S_{v_n}^{1/f}}, \tag{5.15}$$

where $S_{v_n}^{1/f}$ is the flicker noise PSD in nMOS referred to the FET gate as a voltage $V_n^{1/f}$ and given by

$$\mathbf{S_{v_n}^{1/f} = \frac{K_{fN}}{WLC_{ox}f}}. \tag{5.16}$$

If Equations 5.15 and 5.16 are used in Equation 5.14, phase noise due to flicker noise can be written as

$$\mathbf{L^{1/f}(f) = \frac{1}{8MI}(\frac{\mu_N K_{fN}}{L_N^2} + \frac{\mu_P K_{fP}}{L_P^2})\frac{f_0^2}{f^3}}. \tag{5.17}$$

In fact, the original equation of flicker noise induced phase noise in [1] has an additional C_{ox} term which should not be included. The proof of the claim is given in Section 5.5.

By using the link given in Equation 5.8, final equation of jitter due to flicker noise in strong inversion region becomes

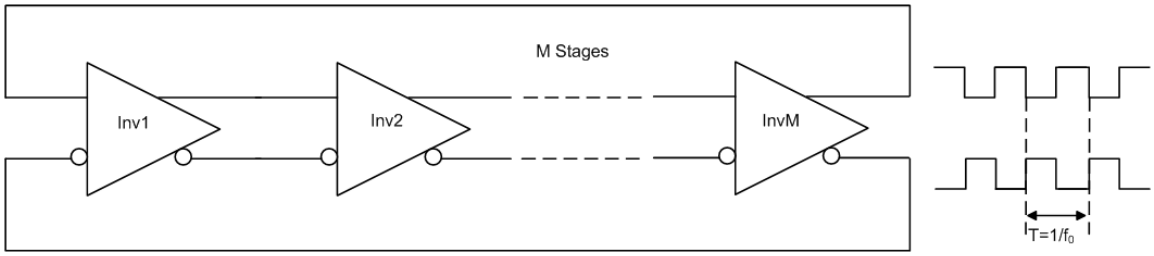

Figure 5.4. Differential ring oscillator.

$$\sigma_{\tau}^{2\,1/f} = \frac{25}{8MI}\frac{1}{f_0^2}\left(\frac{\mu_N K_{fN}}{L_N^2} + \frac{\mu_P K_{fP}}{L_P^2}\right). \tag{5.18}$$

5.4. Differential Ring Oscillator

Figure 5.4 shows the differential delay cell based ring oscillator. DRO should consist of more than one delay stage with even or odd count; however, it is difficult to meet Barkhausen's oscillation criterion of 2π phase shift with two delay stages. Because, the required level of amplification may not supplied with two delay stages and Barkhausen's oscillation criterion can not be met. Therefore, more than two delay stages are used in the designs. The detailed circuitry of a differential delay stage is given in Figure 5.5.

5.4.1. Phase Noise and Jitter due to White Noise

Propagation delay

$$t_d = \frac{V_{op} C \ln 2}{I} \tag{5.19}$$

is the time between an input step and the zero crossing of the output in the differential waveform.

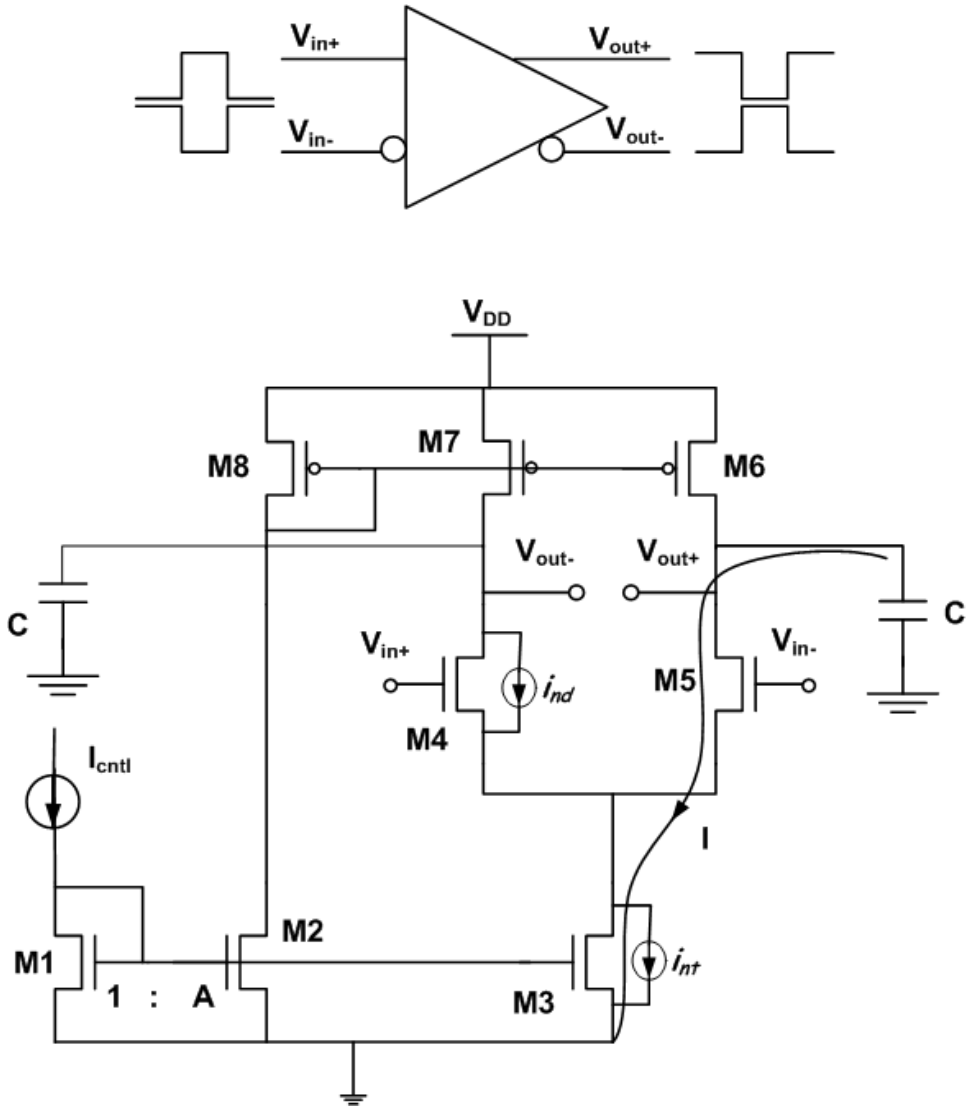

Figure 5.5. Transistor based architecture.

Nominal oscillating frequency of a DRO is

$$\mathbf{f_0 = \frac{1}{2Mt_d} = \frac{I}{2MV_{op}Cln2}}, \tag{5.20}$$

where C is the load capacitance, M is the number of stages, V_{op} is the differential peak output voltage swing, and I is the average current which charges and discharges the load capacitance.

The period jitter of a DRO is calculated by the addition of three uncorrelated

noise voltages.

(i) Noise source due to the load transistors which is continuously coupled into the load capacitors

(ii) The differential noise voltage across the loads due to the tail current in steady state

(iii) The differential input pair transistor's current noise on the differential load during transition

When all these noise sources are added, jitter due to white noise is written as

$$\sigma^{2}{}_{\tau} = \mathbf{2M}\sigma^{2}_{\mathbf{td}} = \frac{\mathbf{2kT}}{\mathbf{If_0 ln2}}[\gamma(\frac{\mathbf{3}}{\mathbf{4V_{effd}}} + \frac{\mathbf{1}}{\mathbf{V_{efft}}}) + \frac{\mathbf{1}}{\mathbf{V_{op}}}], \tag{5.21}$$

where V_{effd} and V_{efft} are the effective overdrive voltage on input transistors and tail transistor, respectively.

By using Equations 5.21 and 5.2x, Lorentzian phase noise due to white noise is obtained as

$$\mathbf{L(f)} = \frac{\mathbf{2kT}}{\mathbf{Iln2}}[\gamma(\frac{\mathbf{3}}{\mathbf{4V_{effd}}} + \frac{\mathbf{1}}{\mathbf{V_{efft}}}) + \frac{\mathbf{1}}{\mathbf{V_{op}}}]\frac{\mathbf{f_0^2}}{\mathbf{f^2}}. \tag{5.22}$$

5.4.2. Phase Noise and Jitter due to Flicker Noise

It was mentioned in [1] that flicker noise in the tail current has stronger effect on the delay rather than flicker noise in the differential pair. Phase noise equation was derived with the formula given in Equation 5.13 where K_I is the frequency sensitivity to tail current ($\frac{f_0}{I}$). The final equation is

$$\mathbf{L^{1/f}(f)} = \mathbf{A}\frac{\mathbf{K_f}}{\mathbf{WLC_{ox}}}(\frac{\mathbf{1}}{\mathbf{V^2_{efft}}})\frac{\mathbf{f_0^2}}{\mathbf{f^3}}, \tag{5.23}$$

where A is the current mirror ratio, K_f is the process dependent flicker noise coefficient.

In Section 5.6, it is shown that the original equation has an additional term of $\frac{1}{f}$ which should not be included.

For the same reason given in the derivation of the jitter due to flicker noise for IbRO, the jitter due to flicker noise for DRO was not included in [1]. Here the same approach expressed in Equation 5.8 is used for obtaining the flicker noise induced jitter for DRO as well.

$$\sigma_{\tau}^{2\,1/f} = 25A\frac{1}{f_0^2}\frac{K_f}{WLC_{ox}}\left(\frac{1}{V_{efft}^2}\right) \tag{5.24}$$

5.5. Proof of the Correction of Flicker Noise Component of Phase Noise Equation for IbRO

The expression of flicker noise component of phase noise in IbRO, given in [1] and cited with the equation number (45), has a small error. In order to prove it, the following analysis is done.

Equation Equation 5.14, which corresponds to (41) in [1], is used to derive SSB phase noise induced flicker noise. Therefore, we will continue our analysis from that equation. By using Equation 5.15 and Equation 5.16, spectral density of flicker noise as current in the drain of NFET is derived as

$$S_{i_n}^{1/f} = g_m^2 S_{v_n}^{1/f} = \left(\frac{2I}{V_{DD}-V_t}\right)^2 \frac{K_{fN}}{WLC_{ox}f}, \tag{5.25}$$

which corresponds to equation (43) in [1]. Equation 5.25 and its counterpart in PFET drain are combined in Equation 5.14 in order to derive

$$L^{1/f}(f) = \frac{1}{16MI^2}\left(\frac{4I^2K_{fN}}{C_{ox}W_NL_N(V_{DD}-V_t)^2} + \frac{4I^2K_{fP}}{C_{ox}W_PL_P(V_{DD}-V_t)^2}\right)\frac{f_0^2}{f^3}. \tag{5.26}$$

After some simplifications, SSB phase noise induced from flicker noise becomes

$$\mathbf{L^{1/f}(f) = \frac{1}{4M(V_{DD} - V_t)^2} \frac{1}{C_{ox}} (\frac{K_{fN}}{W_N L_N} + \frac{K_{fP}}{W_P L_P}) \frac{f_0^2}{f^3}.} \tag{5.27}$$

However, in the original equation, published in [1] with the equation number of (44), there is a missing a term of $\frac{1}{C_{ox}}$. Obviously, this will lead to a mistake in the final equation as well.

Further simplification to Equation 5.27 can be done by using the MOSFET's current equation

$$\mathbf{I = \frac{1}{2} C_{ox} \mu \frac{W}{L} (V_{DD} - V_t)^2.} \tag{5.28}$$

Finally, flicker noise induced SSB phase noise becomes

$$\mathbf{L^{1/f}(f) = \frac{1}{8MI} (\frac{\mu_N K_{fN}}{L_N^2} + \frac{\mu_P K_{fP}}{L_P^2}) \frac{f_0^2}{f^3}.} \tag{5.29}$$

Furthermore, a comparative dimensional analysis has been performed in order to demonstrate the disagreement between phase noise unit and the given equation of (45) in [1]. For this reason, the units of the terms given in equations are provided in Table 5.1.

The final flicker noise induced SSB phase noise in [1] with the equation number of (45) has an additional C_{ox} which should not be included.

5.6. Proof of the Correction of Flicker Noise Component of Phase Noise Equation for DRO

In [1], the equation of phase noise due to flicker noise for DROs, cited equation (64), also has a small error. Confirmation of the claim will be done with the following

Table 5.1. Units of terms.

Term Name	**Symbol**	**Unit**
Flicker Noise Induced SSB Phase Noise	$L^{1/f}(f)$	s
Mobility	μ	$\frac{m^2}{Vs}$
Oxide Capacitance	C_{ox}	$\frac{F}{m^2}$
Flicker Noise Coefficient	K_f	V^2F
Number of Stages	M	$unitless$

analysis.

The main equation used for deriving flicker noise induced phase noise is

$$\mathbf{L^{1/f}(f) = \frac{{K_I}^2}{4f_2} S_I^{1/f}(f) = \frac{1}{4I^2}(\frac{f_0}{f})^2 S_I^{1/f}(f)}, \tag{5.30}$$

which is given with equation number of (62) in [1]. Power spectral density of current noise arising from the diode-connected FET at the tail of differential inverter is

$$\mathbf{S_I^{1/f}(f) = A(\frac{2I}{V_{efft}^2})^2 \frac{K_f}{WLC_{ox}f}} \tag{5.31}$$

with the citation number of (63) in [1]. When PSD of tail current noise is written in Equation 5.30, flicker noise induced phase noise for DRO will become

$$\mathbf{L(^{1/f}f) = A\frac{K_f}{WLC_{ox}}(\frac{1}{V_{efft}^2})\frac{f_0^2}{f^3}}, \tag{5.32}$$

where it differs with the absence of the term of $\frac{1}{f}$ from the original equation, cited with the equation number of (64) in [1].

Despite the known fact that the phase noise spectrum of an oscillator drops with a -30dB slope, if the original equation, (64) in [1], is considered, it should have a slope

Table 5.2. Verification of corrections for phase noise due to flicker noise in IbRO.

PN due to FN for IbRO in [1]	**PN due to FN for IbRO in [19]**
$\frac{C_{ox}}{8MI}\left(\frac{\mu_N K_{fN}}{L_N^2}+\frac{\mu_P K_{fP}}{L_P^2}\right)\frac{f_0^2}{f^3}$	$\frac{1}{8MI}\left(\frac{\mu_N K_{fN}}{L_N^2}+\frac{\mu_P K_{fP}}{L_P^2}\right)\frac{f_0^2}{f^3}$
$\frac{C_{ox}}{8MI}\left(\frac{\mu K_f}{L^2}\right)\frac{f_0^2}{f^3}$	$\frac{1}{8MI}\left(\frac{\mu K_f}{L^2}\right)\frac{f_0^2}{f^3}$
$\frac{F}{m^2}\frac{1}{A}\left(\frac{m^2}{Vs}\frac{V^2F}{m^2}\right)\frac{1}{Hz}$	$\frac{1}{A}\left(\frac{m^2}{Vs}\frac{V^2F}{m^2}\right)\frac{1}{Hz}$
$\frac{F^2V}{m^2A}=\frac{F}{m^2}s$	$\frac{FV}{A}=s$
Not verified	Verified

of -40dB due to the $\frac{1}{f^4}$ term.

Furthermore, a comparative dimensional analysis has also been performed for DRO in order to demonstrate the disagreement between phase noise unit and the equation (64) in [1]. The term of A is the current mirror ratio between the diode-connected FET and tail FET, therefore; it is unitless.

5.7. Experimental Verification

For the experimental validation of the equations above, $0.25\mu m$ standard CMOS process has been used for the design and fabrication of ROs. A die photo of the fabricated chip is depicted in Figure 5.6. For IbRO and DRO, a supply voltage of $2.5V$ is utilized for strong inversion measurements. White and flicker noise components of phase noise equations in strong inversion, derived in [1], are verified and validated with this design technology. Figure 5.7 exhibits the measurement set-up used to prove the derivations and Figure 5.8 shows the measurement environment.

Table 5.3. Verification of corrections for phase noise due to flicker noise in IbRO.

PN due to FN for DRO in [1]	**PN due to FN for DRO in [19]**
$A\frac{K_f}{WLC_{ox}L}(\frac{1}{V_{efft}^2})\frac{f_0^2}{f^3}$	$A\frac{K_f}{WLC_{ox}}(\frac{1}{V_{efft}^2})\frac{f_0^2}{f^3}$
$\frac{1}{m^2}V^2F\frac{1}{\frac{F}{m^2}}\frac{1}{Hz}(\frac{1}{V^2})\frac{1}{Hz}$	$\frac{1}{m^2}V^2F\frac{1}{\frac{F}{m^2}}(\frac{1}{V^2})\frac{1}{Hz}$
$\frac{1}{Hz^2} = s^2$	$\frac{1}{Hz} = s$
Not verified	Verified

The measured IbRO consists of 23 inverters and a buffer stage. Similarly, the measured DRO consists of 7 delay stages and a buffer stage. Figure 5.9 is the phase noise snapshot of the frequency-domain measurements taken with a spectrum analyzer.[1]

Phase noise measurement results of IbRO are given for strong inversion region in the following figures. In Figure 5.10, measurement results of the white noise component of phase noise are presented together with simulation and estimation results based on Equation 5.12. Similarly, Figure 5.11 shows the measured and simulated flicker noise components of the phase noise as well as estimations performed using Equation 5.17.

As seen from the figures, measurement, simulation, and estimation results present good matching in strong inversion region for flicker and white noise components of phase noise.

Results of DRO measurement, simulation, and estimation are displayed in Figures 5.12, and 5.13 for white and flicker noise components in strong. Estimations are

[1]Rohde and Schwarz FSU 20Hz - 3.66GHz model spectrum analyzer is used for the frequency domain measurements.

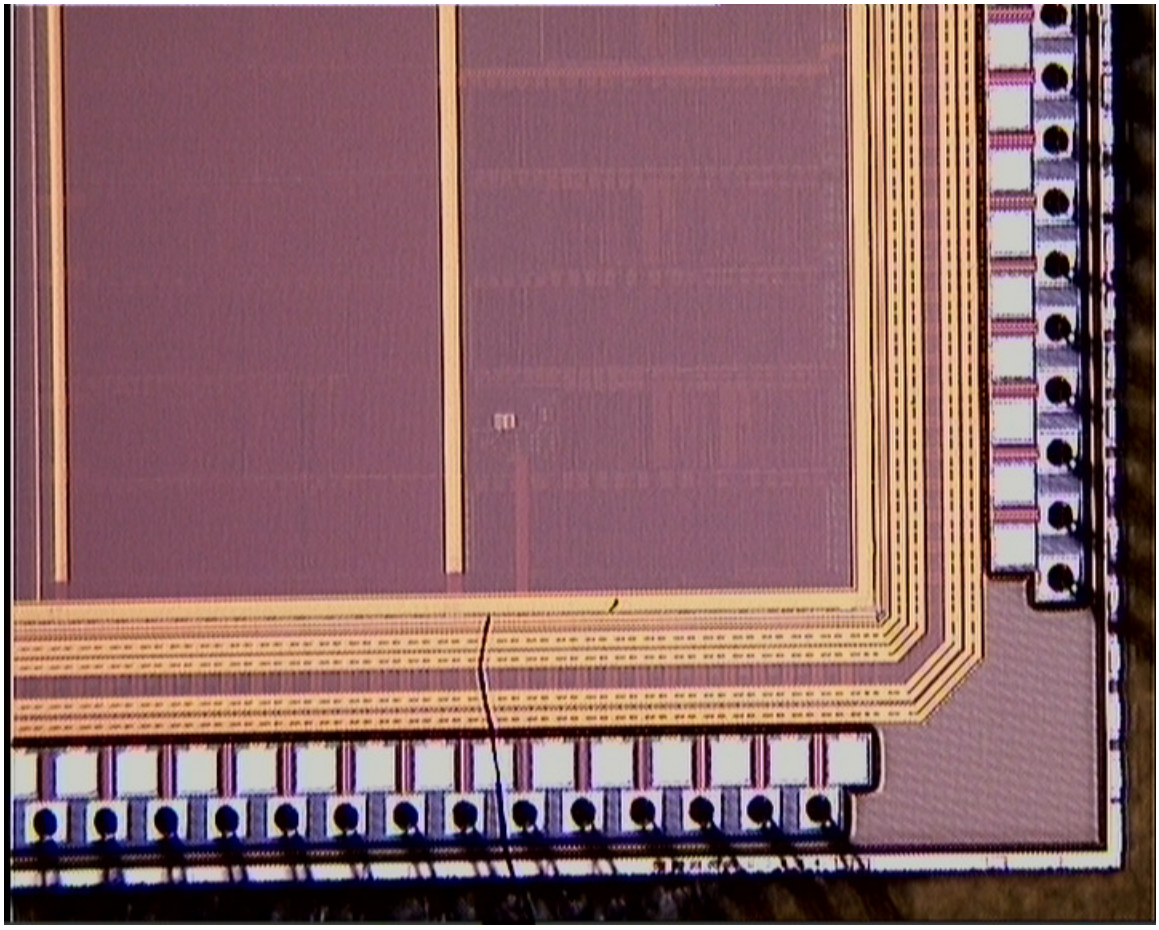

Figure 5.6. Fabricated die photo.

done according to the formulas given in Equations 5.22 and 5.23.

5.8. Conclusion

0.25μm standard CMOS process has been used for design and analysis for two types of ROs. Existing phase noise and jitter models of ROs for strong inversion region have been reviewed. A good match is attained between analyses and measurements.

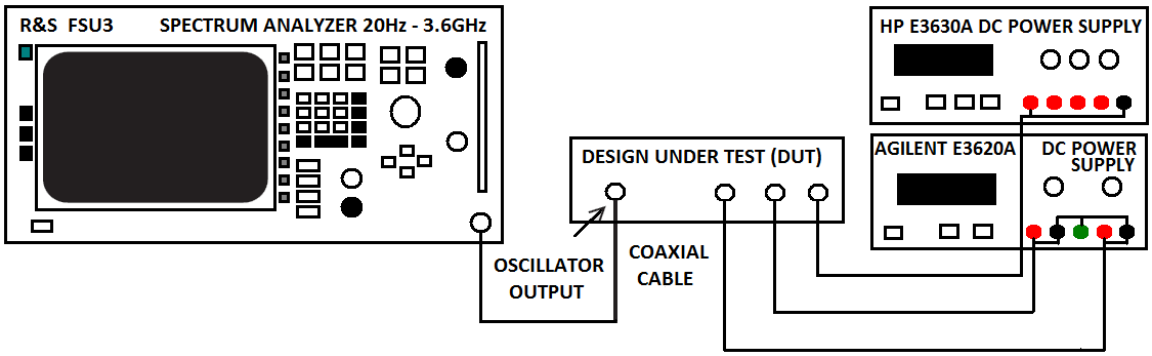

Figure 5.7. Measurement set-up.

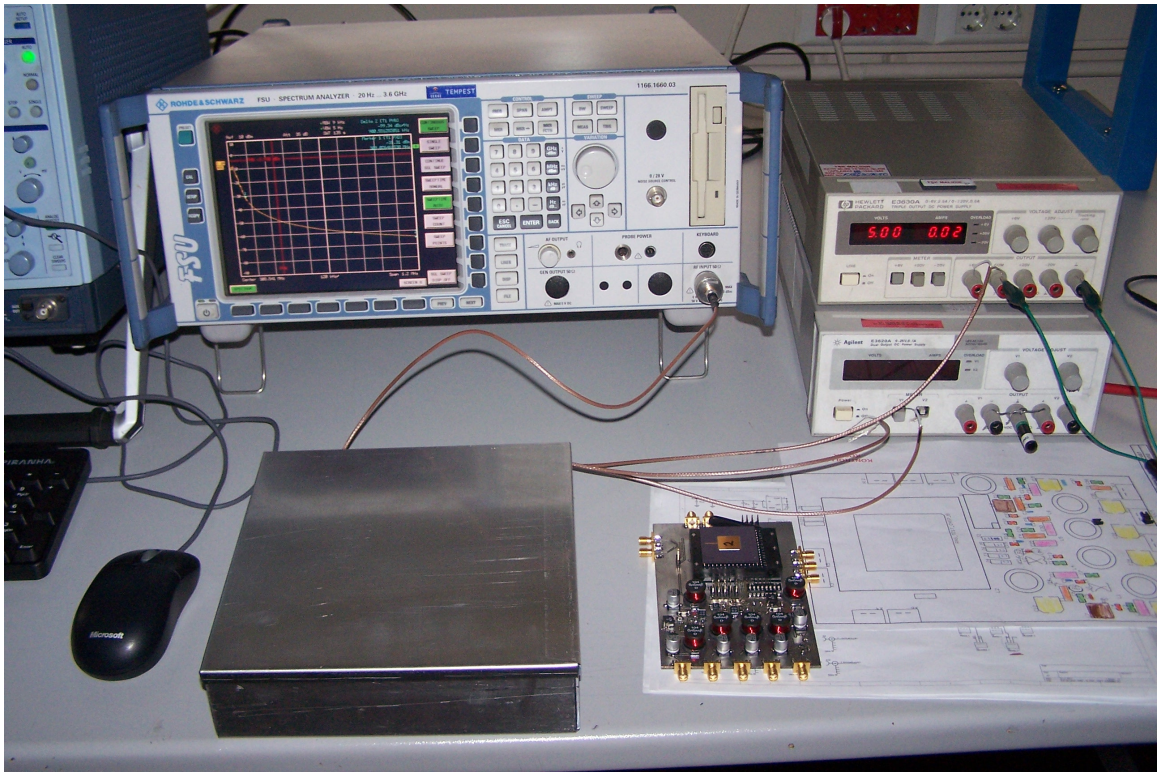

Figure 5.8. Measurement environment.

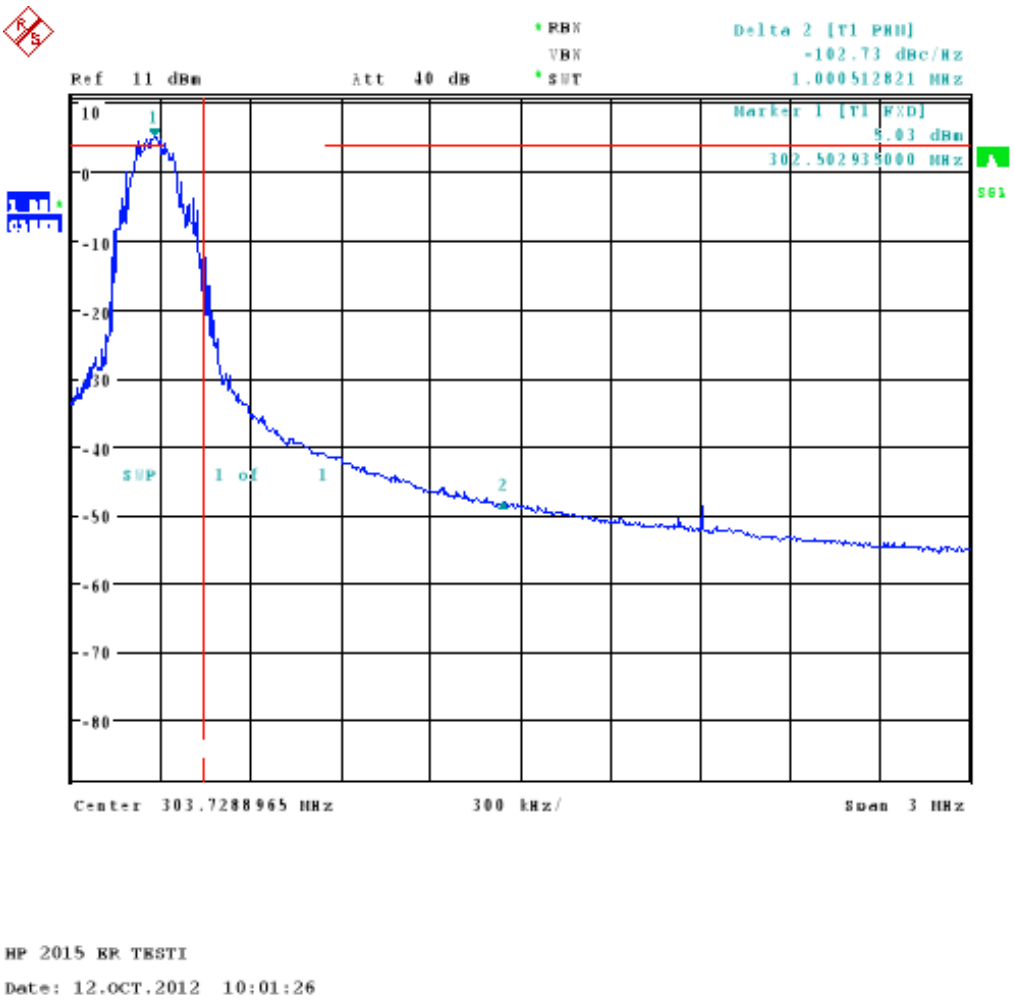

Figure 5.9. Frequency-domain measurement of IbRO in strong inversion.

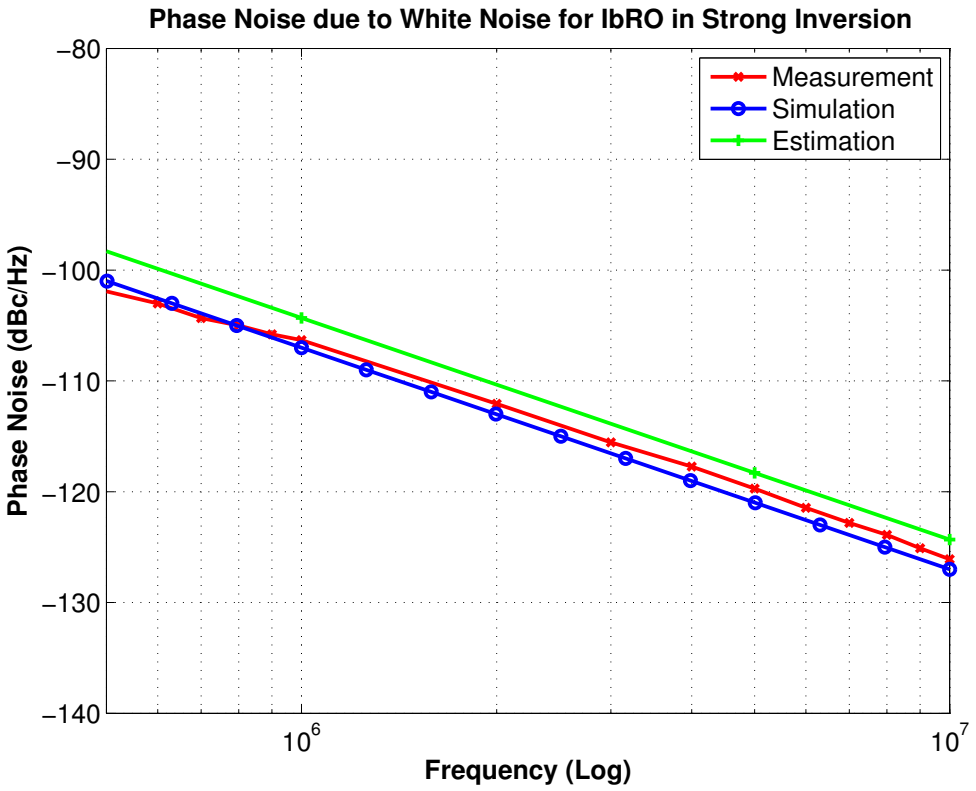

Figure 5.10. White noise component of phase noise for IbRO in strong inversion region.

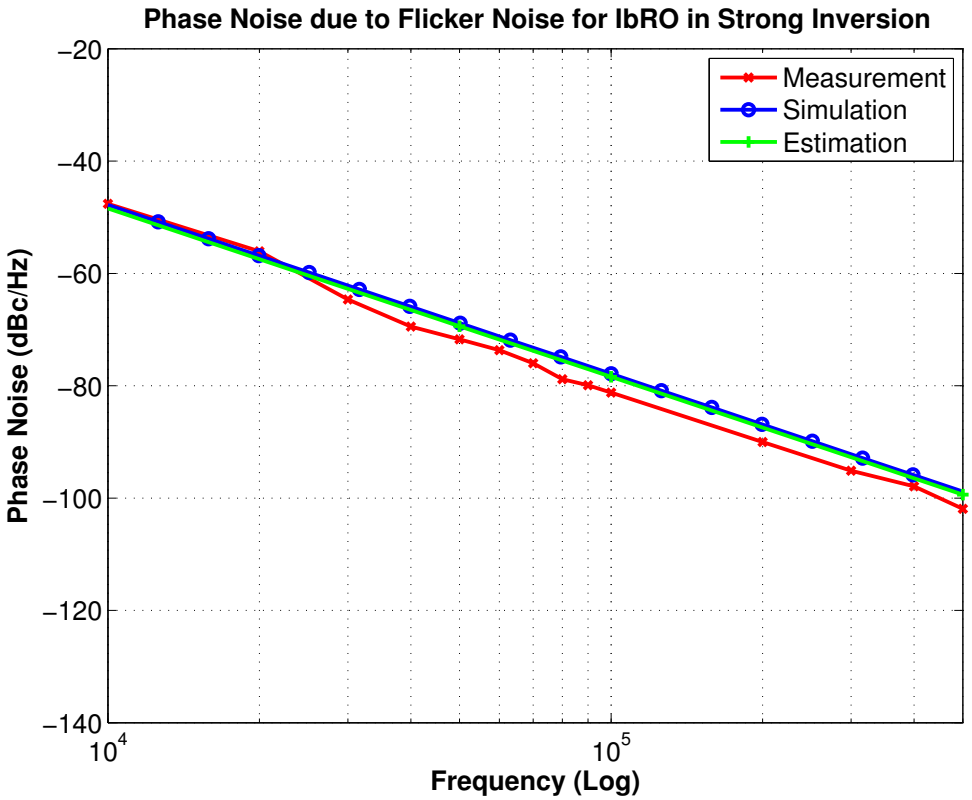

Figure 5.11. Flicker noise component of phase noise for IbRO in strong inversion region.

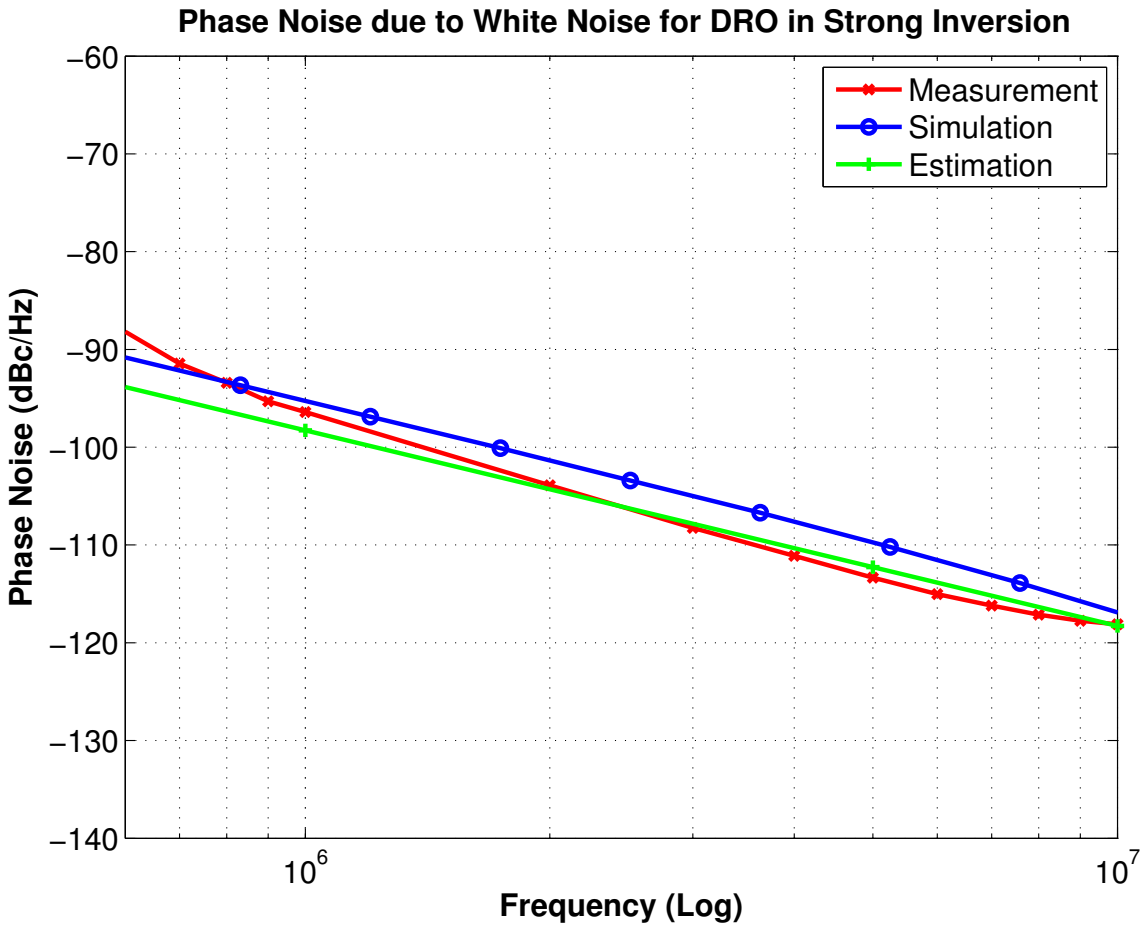

Figure 5.12. White noise component of phase noise for DRO in strong inversion region.

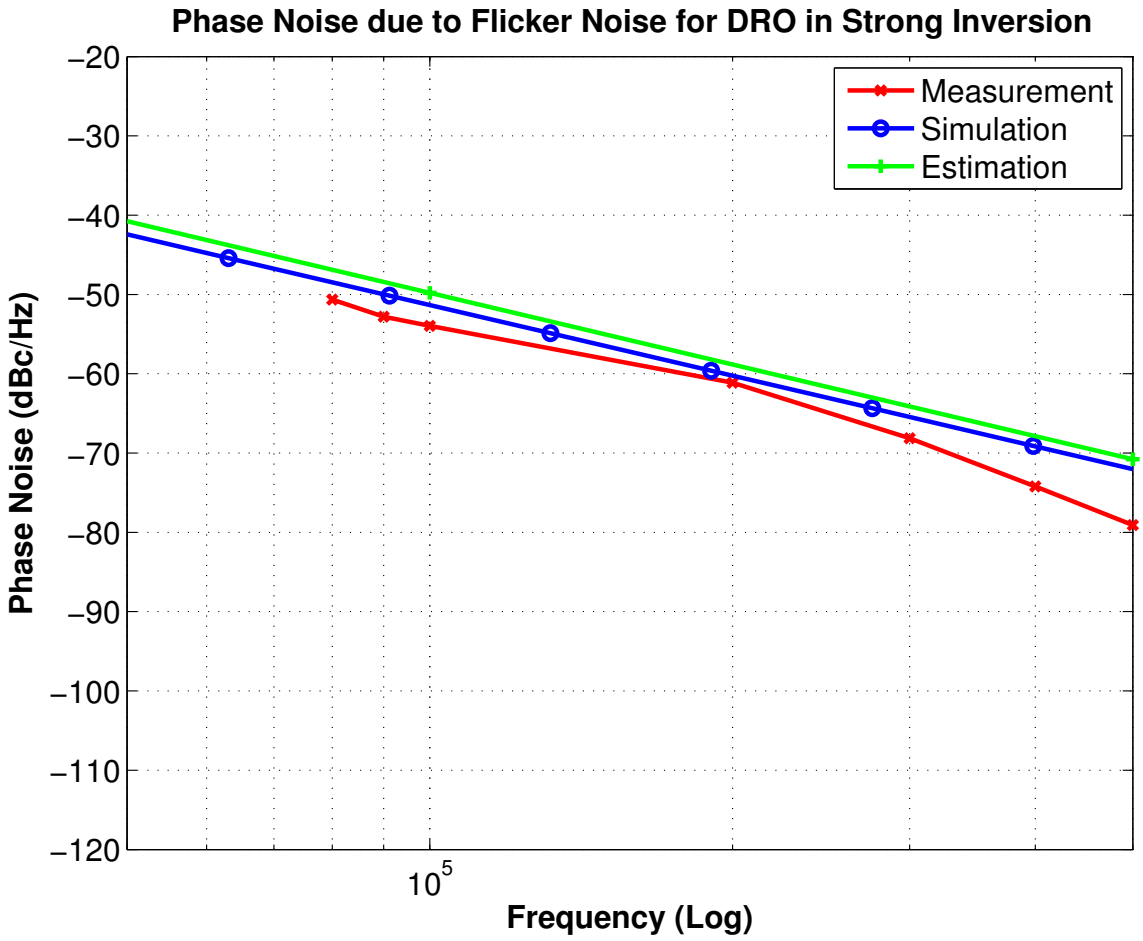

Figure 5.13. Flicker noise component of phase noise for DRO in strong inversion region.

6. PHASE NOISE AND JITTER MODELS OF CMOS RO IN WEAK INVERSION REGION

[87] claims that CMOS devices exhibit more current noise in the sub-threshold region, which will obviously lead to more jitter in RO and more randomness in RNG. For this reason, we aim to operate the ROs in sub-threshold region to gather more randomness.

Despite the extensive research on phase noise and jitter of CMOS RO in strong inversion, there is limited information in the literature related to their weak inversion counterparts. In [20] phase noise and jitter of CMOS ROs was partially modeled; however, not validated with measurement results. In this chapter, we model and derive the equations of phase noise and jitter of CMOS RO in weak inversion. Flicker and white noise components of phase noise and jitter are all included for IbRO and DRO architectures.

In [1], equations for phase noise and jitter of a CMOS RO, due to flicker and white noise, are derived for strong inversion region and verified with the simulations and measurements. The sections below follow the same methodology to obtain the equations in weak inversion region for the cases of IbRO and DRO. Even though the contribution of flicker noise to randomness is assumed to be limited and not calculated in [21], its exact effect is an open research topic. Therefore, phase noise due to flicker noise is calculated.

6.1. Inverter-Based Ring Oscillator

6.1.1. Phase Noise and Jitter due to White Noise

The current and the power spectrum of current noise in weak inversion are different from the equations given in [1] . The current in weak inversion is given as

$$\mathbf{I = I_{sat}(1 - e^{-V_{ds}/U_t})} \tag{6.1}$$

and the power spectrum of current noise in weak inversion is presented as

$$\mathbf{S_{in} = 2qI_{sat}(1 + e^{-V_{ds}/U_t})}, \tag{6.2}$$

where q is the Coulomb constant, I_{sat} is the saturation current of the transistor, U_t is the thermal voltage given as $\frac{kT}{q}$, and V_{ds} is the drain to source voltage.

Please note that channel resistance due to the strong inversion of channel, included in [1], will not be included in weak inversion equations.

If all these changes are applied to Equation 5.10 and the equations are reorganized, jitter equation becomes

$$\sigma_\tau^2 = \frac{\mathbf{q}}{\mathbf{If_0}}\mathbf{(1 + e^{-V_{DD}/2U_t})}, \tag{6.3}$$

where, V_{DD} is the supply voltage, and f_0 is the oscillation frequency.

From the link Equation 5.2 between two directly measurable quantities *jitter* (σ_τ^2) and *phase noise L(f)*, the phase noise equation becomes

$$\mathbf{L(f) = \frac{q}{I}(1 + e^{-V_{DD}/2U_t})(\frac{f_0}{f})^2}. \tag{6.4}$$

6.1.2. Phase Noise and Jitter due to Flicker Noise

Phase noise equations given in Equations 5.13 and 5.14 are still valid in weak inversion region; however, g_m, the transconductance of the FET, is different compared to that in strong inversion and formulated in sub-threshold region as

$$\mathbf{g_m} = \frac{\mathbf{I_{sat}}}{\mathbf{nU_T}}. \tag{6.5}$$

Using Equations 5.14 - 5.16 and 6.5, the phase noise equation in weak inversion will be obtained as

$$\mathbf{L^{1/f}(f)} = \frac{\mathbf{1}}{\mathbf{16M(1+e^{-V_{ds}/U_t})^2}} \frac{\mathbf{1}}{\mathbf{C_{ox}(nU_T)^2}} (\frac{\mathbf{K_{fN}}}{\mathbf{W_nL_n}} + \frac{\mathbf{K_{fP}}}{\mathbf{W_pL_p}}) \frac{\mathbf{f_0^2}}{\mathbf{f^3}}, \tag{6.6}$$

where W and L are the width and length of the transistor, K_f is the flicker noise parameter, C_{ox} is the oxide capacitance, and n is the sub-threshold slope.

The expression provided in Equation 5.8, is used to obtain the jitter due to flicker noise in weak inversion for IbRO as

$$\sigma_{\tau}^{\mathbf{2^{1/f}}} = \mathbf{25} \frac{\mathbf{1}}{\mathbf{f_0^2}} \frac{\mathbf{1}}{\mathbf{16M(1+e^{-V_{ds}/U_t})^2}} \frac{\mathbf{1}}{\mathbf{C_{ox}(nU_T)^2}} (\frac{\mathbf{K_{fN}}}{\mathbf{W_nL_n}} + \frac{\mathbf{K_{fP}}}{\mathbf{W_pL_p}}). \tag{6.7}$$

6.2. Differential Ring Oscillator

6.2.1. Phase Noise and Jitter due to White Noise

The period jitter for DRO is calculated by adding the three uncorrelated noise voltages below as mentioned in the previous sections.

(i) Noise source due to the load transistors which is continuously coupled into the

load capacitors

$$< \mathbf{V_{nR}^2} >= \frac{\mathbf{2kT}}{\mathbf{C}}. \tag{6.8}$$

It is shown in [87] that $kT = qIR$, by using this equation, Equation 6.8 can be reorganized as

$$< \mathbf{V_{nR}^2} >= \frac{\mathbf{2qIR}}{\mathbf{C}}. \tag{6.9}$$

(ii) Noise source due to the differential noise voltage across the loads due to the tail current in steady state. These noise sources can be rearranged by using Equations 6.1 and 6.2 as

$$< \mathbf{V_n^2} > (\mathbf{left}) + < \mathbf{V_n^2} > (\mathbf{right}) = \frac{\mathbf{qIR}}{\mathbf{2C}}(\mathbf{1} + \mathbf{e}^{-\mathbf{V_{dst}/U_t}}). \tag{6.10}$$

(iii) Noise source due to the differential input pair transistor's current noise on the differential load during transition. The mean-square differential voltage can be written as

$$< \mathbf{V_{ndiff}^2} >= \frac{\mathbf{3}}{\mathbf{8C}}\mathbf{2qIR}(\mathbf{1} + \mathbf{e}^{-\mathbf{V_{dsd}/U_t}}), \tag{6.11}$$

where C is the load capacitance, R is the load resistance, V_{dst} and V_{dsd} are the drain to source voltages of tail and differential transistors.

Finally, if all three uncorrelated noise voltages in Equations 6.9, 6.10, and 6.11 are summed, the period jitter in weak inversion for DRO becomes

$$\sigma_\tau^2 = \frac{\mathbf{q}}{\mathbf{If_0 ln2}}[\frac{\mathbf{3}}{\mathbf{4}}(\mathbf{1} + \mathbf{e}^{-\mathbf{V_{dsd}/U_t}}) + \frac{\mathbf{1}}{\mathbf{2}}(\mathbf{1} + \mathbf{e}^{-\mathbf{V_{dst}/U_t}}) + \mathbf{2}]. \tag{6.12}$$

Furthermore, from Equations 6.12 and 5.2, phase noise equation is derived as

$$\mathbf{L(f) = \frac{q}{Iln2}[\frac{3}{4}(1 + e^{-V_{dsd}/U_t}) + \frac{1}{2}(1 + e^{-V_{dst}/U_t}) + 2](\frac{f_0}{f})^2.} \tag{6.13}$$

6.2.2. Phase Noise and Jitter due to Flicker Noise

Flicker noise in the differential pair does not cause phase noise; however, flicker noise in the tail current directly modulates the delay. Flicker noise induced phase noise can be written as

$$\mathbf{L^{1/f}(f) = \frac{1}{4I^2}(\frac{f_0}{f})^2 S_I^{1/f}.} \tag{6.14}$$

In a VCO, the width of the diode-connected FET is $1/A$ times the width of the tail FETs in the delay stages. Therefore, the noise current arising from the diode-connected FET at the output mirror has a factor of $1/A$ and it is written as

$$\mathbf{S_I^{1/f} = A g_m^2 S_V^{1/f}.} \tag{6.15}$$

The noise current of a DRO operating in sub-threshold region becomes

$$\mathbf{S_I^{1/f} = A(\frac{I_{sat}}{nU_T})^2 \frac{K_f}{WLC_{ox}f}.} \tag{6.16}$$

Afterwards, phase noise of a DRO operating in sub-threshold region is reorganized as

$$\mathbf{L^{1/f}(f) = \frac{A}{4(1 + e^{-V_{ds}/U_t})^2}(\frac{1}{nU_T})^2 \frac{K_f}{WLC_{ox}} \frac{f_0^2}{f^3}.} \tag{6.17}$$

By the help of Equation 5.8, jitter due to flicker noise in weak inversion for DRO

is obtained as

$$\sigma_{\tau}^{2\,1/f} = 25\frac{1}{f_0^2}\frac{A}{4(1+e^{-V_{ds}/U_t})^2}\left(\frac{1}{nU_T}\right)^2\frac{K_f}{WLC_{ox}}. \tag{6.18}$$

6.3. Experimental Validation

For the experimental validation of the equations above, $0.25\mu m$ standard CMOS process has been used for the design and fabrication of ROs, as in the strong inversion case. The die photo of the fabricated chip can be seen in Figure 5.6. For IbRO, a supply voltage of $0.5V$ is used for weak inversion measurements. For DRO, $0.7V$ supply voltage is used for weak inversion region measurements. White and flicker noise components of phase noise equations in weak inversion are confirmed by measurements. The measurement set-up, used to prove the derivations, and measurement environment can be also seen in Figures 5.7 and 5.8.

The measured IbRO consists of 23 inverters and a buffer stage. Similarly, the measured DRO consists of 7 delay stages and a buffer stage. Both are the same circuits used in the strong inversion case. Figure 6.1 is the phase noise snapshots of the frequency-domain measurements in the weak inversion taken with a spectrum analyzer[2] as in the strong inversion case.

Phase noise measurement results of IbRO are given for weak inversion regions in the following figures. In Figure 6.2, measurement results of the white noise component of phase noise are presented together with simulation and estimation results based on Equation 6.4. Similarly, Figure 6.3 shows the measured and simulated flicker noise components of the phase noise as well as estimations performed using Equation 6.6.

As seen from the figures, white and flicker noise components of the phase noise in sub-threshold match quite well between measurement, simulation, and estimation

[2]Rohde and Schwarz FSU 20Hz - 3.66GHz model spectrum analyzer is used for the frequency domain measurements

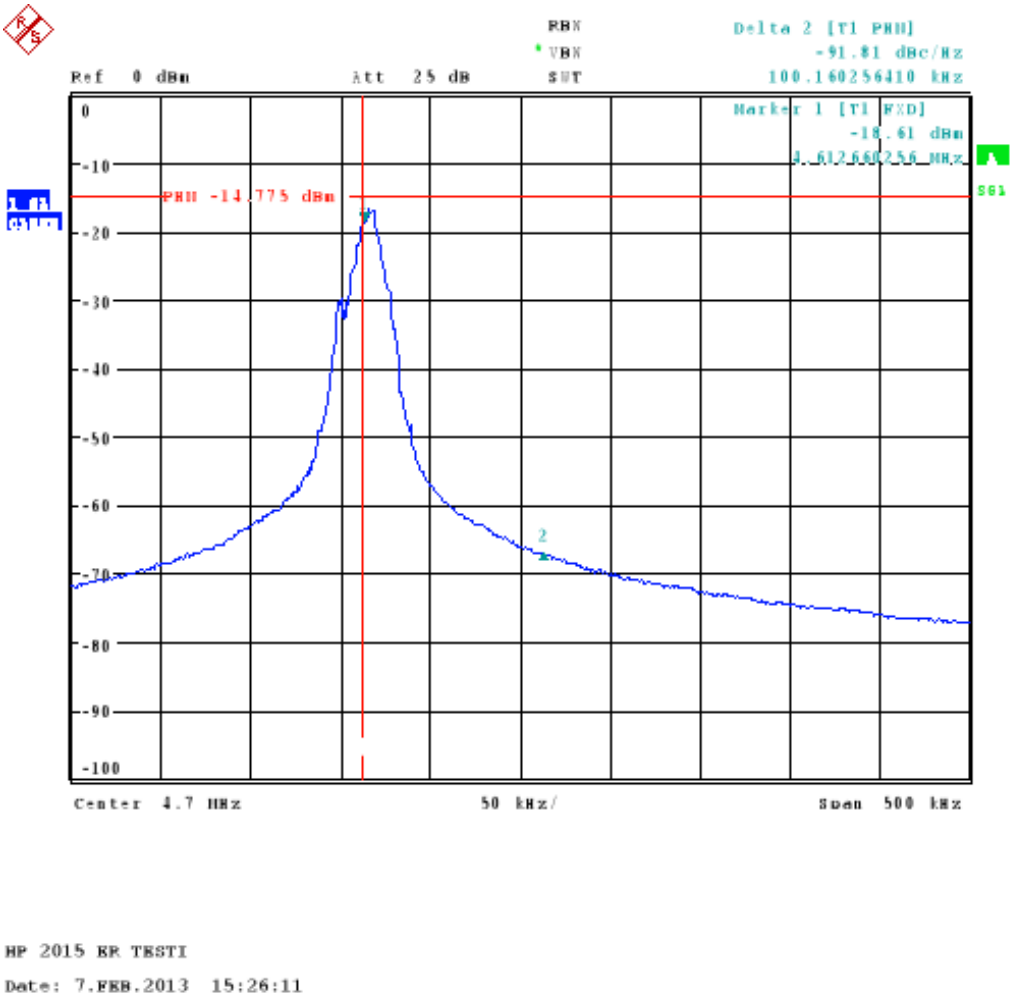

Figure 6.1. Frequency-domain measurement of IbRO in weak inversion.

results. Additionally, one should pay special attention to Figure 6.4, because this figure shows that white noise induced phase noise follows the same line in both inversion regions. In other words, phase noise due to white noise does not depend on oscillation frequency and supply voltage, but depends mainly on the architecture. Of course, observability and measurability of phase noise value in a specific offset frequency differs according to the flicker noise effect, noise floor of the ring oscillator, and spectrum analyzer.

Results of DRO measurement, simulation, and estimation are displayed in Figure 6.5 for flicker noise components in weak inversion, similar to the IbRO case, except

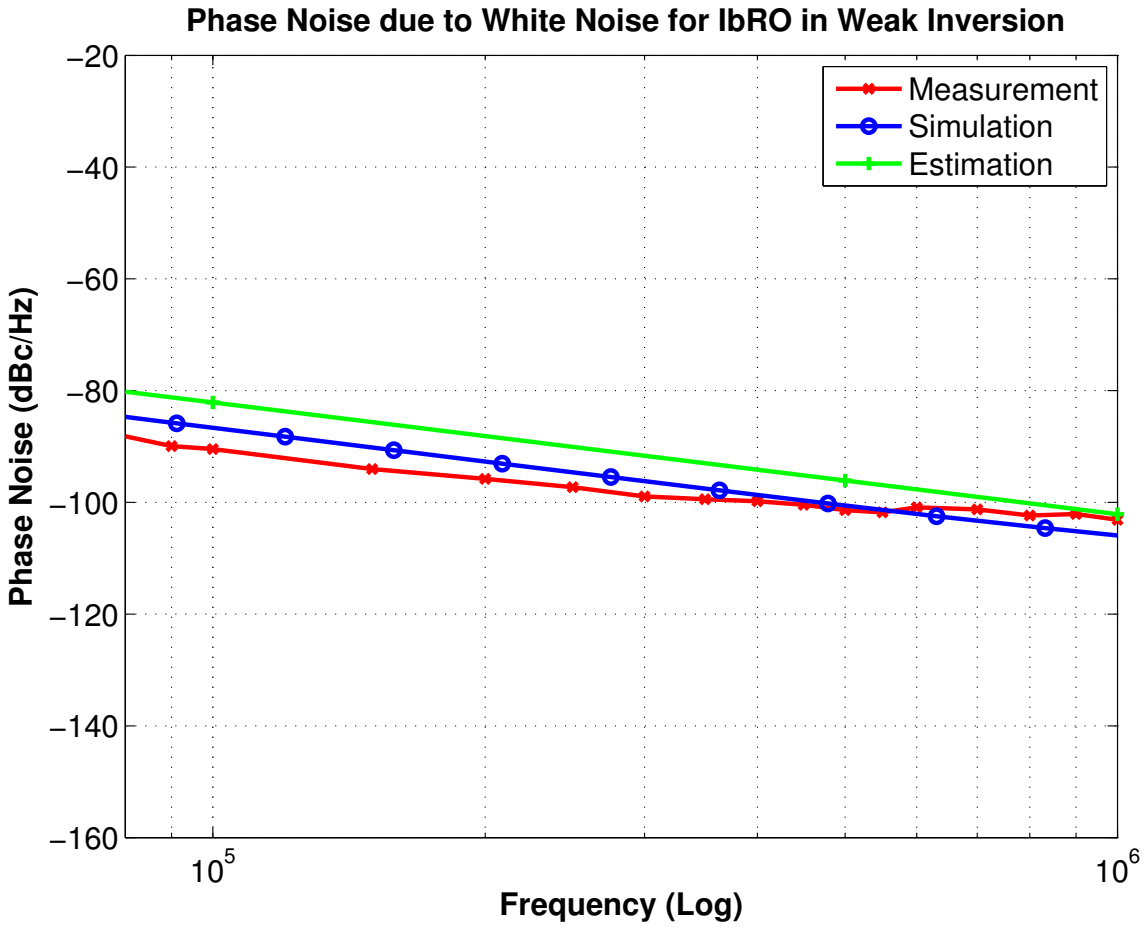

Figure 6.2. White noise component of phase noise for IbRO in weak inversion region.

for the white noise component of DRO in weak inversion. It is observed from weak inversion region measurement results of DRO that flicker noise component of phase noise dominates all the spectrum until it reaches the phase noise floor of oscillator. Therefore, phase noise due to white noise can not be observed for DRO in weak inversion measurement. Estimations are done according to the equations given in Equations 6.13 and 6.17.

6.4. Conclusion

0.25μm standard CMOS process has been used for design and analysis for two types of ROs. Phase noise and jitter models of ROs are derived for weak inversion region. A good match is attained between analyses and measurements.

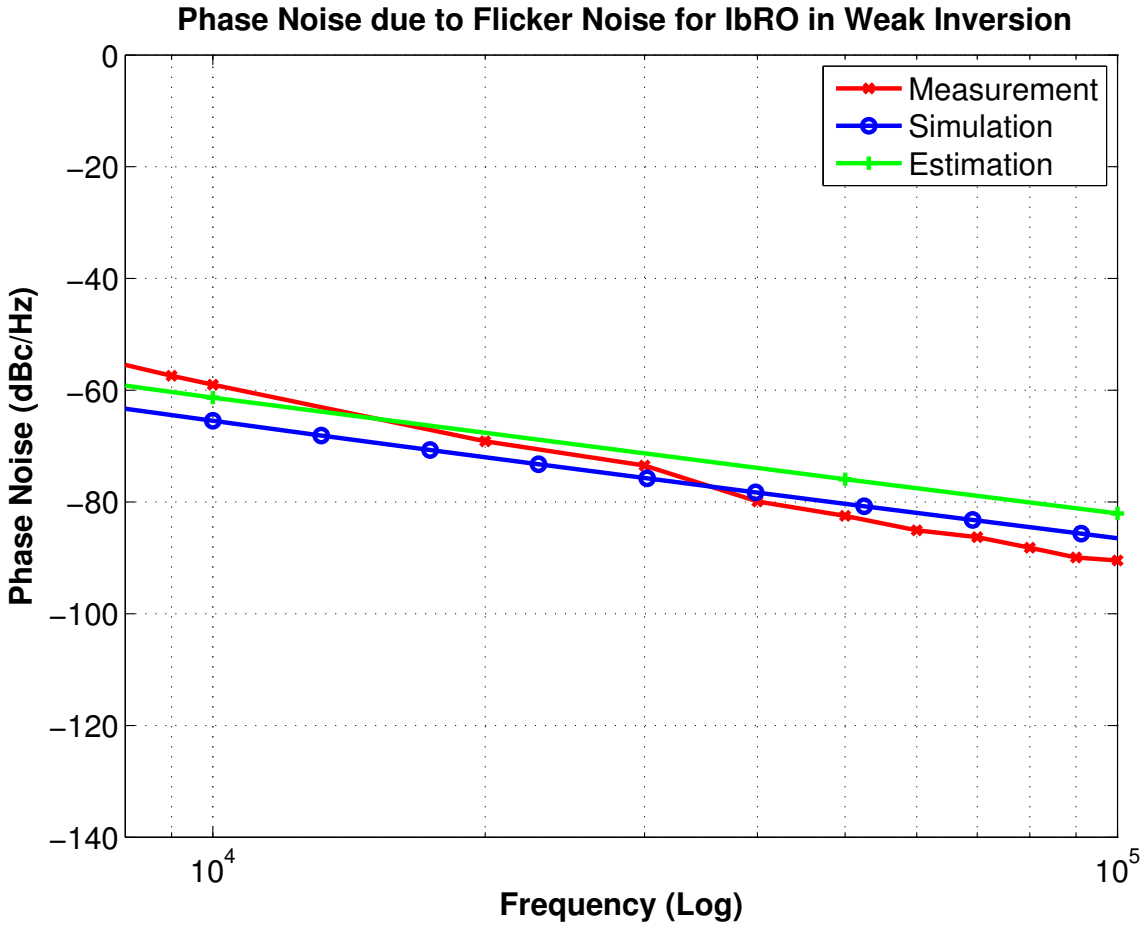

Figure 6.3. Flicker noise component of phase noise for IbRO in weak inversion region.

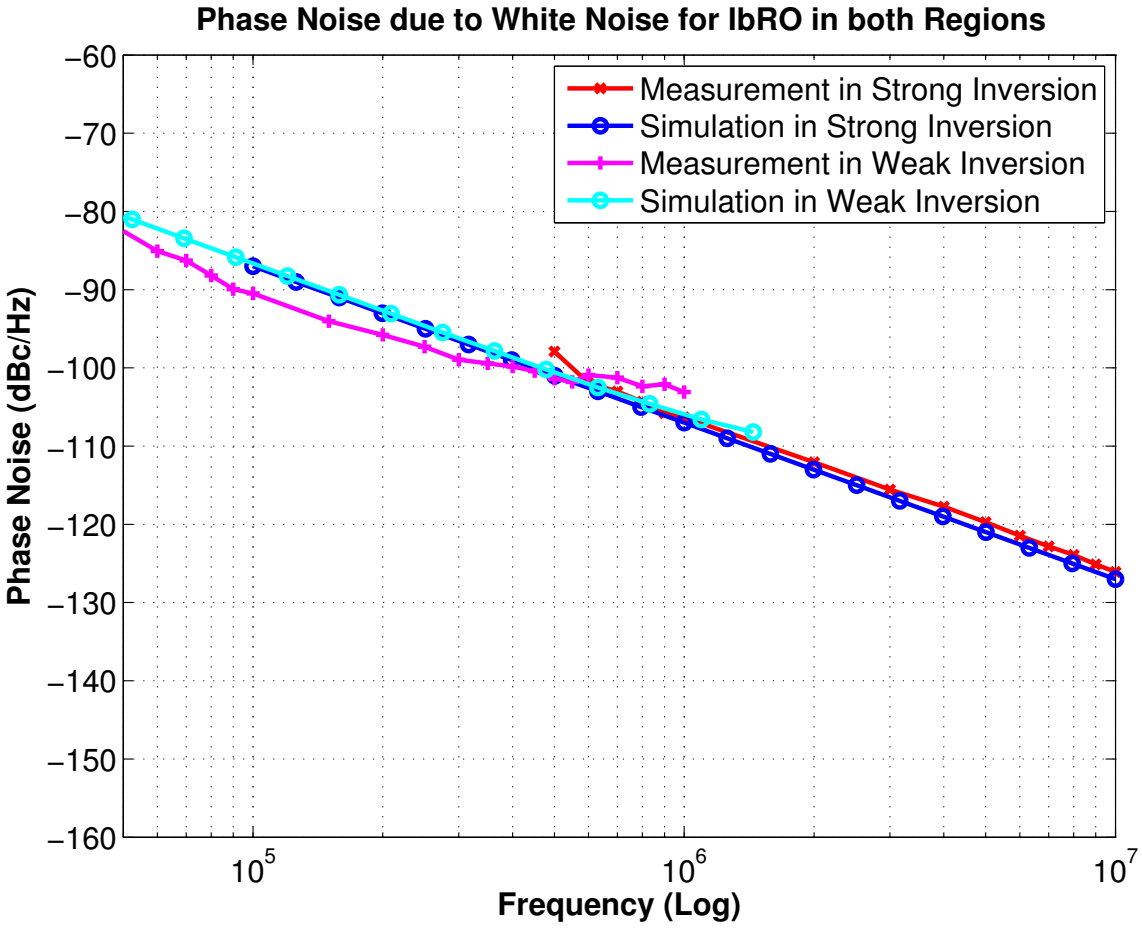

Figure 6.4. IbRO's phase noise due to white noise.

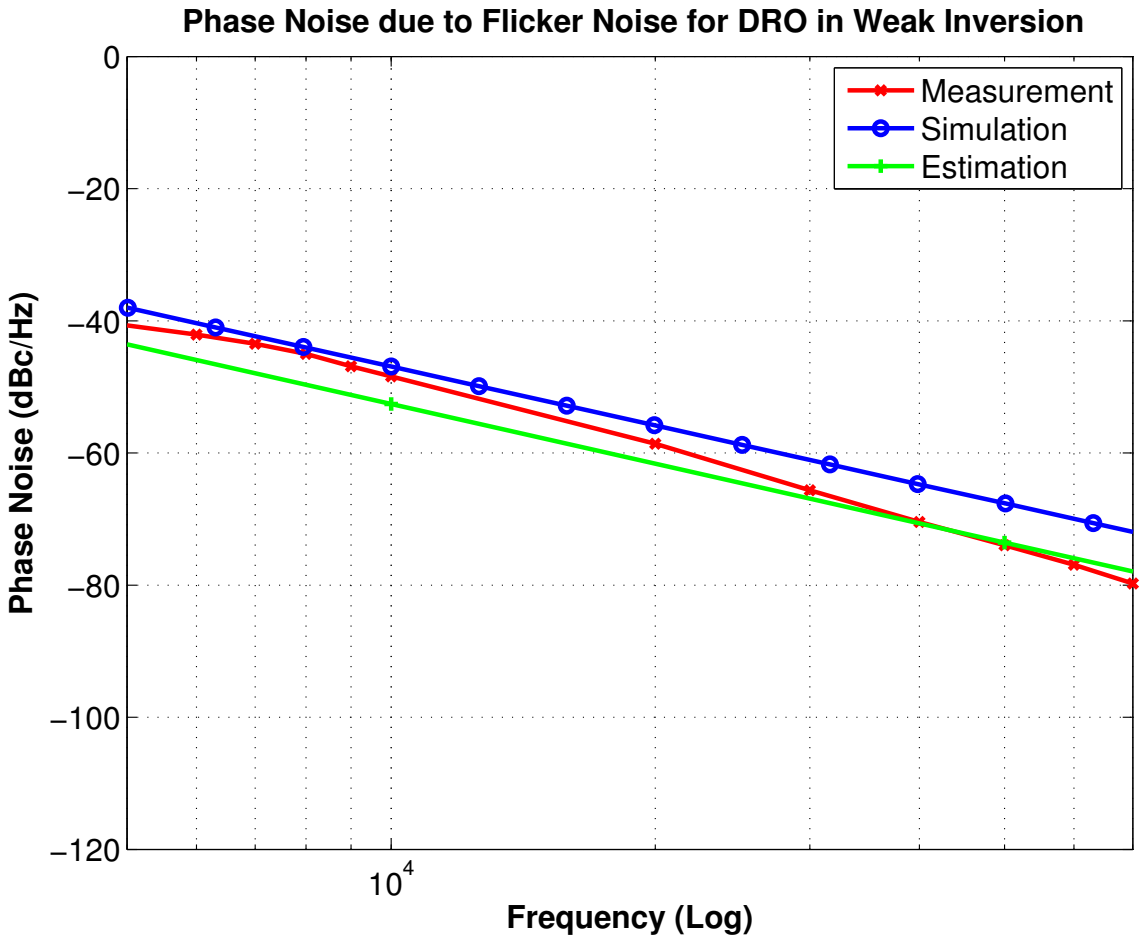

Figure 6.5. Flicker noise component of phase noise for DRO in weak inversion region.

7. DERIVATION OF RANDOMNESS EQUATIONS

In security applications, determining the quality of a RNG is not a well-defined process. In this chapter, a figure of merit is proposed to compare the performance of RNGs. Since RNGs use entropy sources originating from randomness, we define the amount of uncertain zones in a period as *Randomness* ($\mathcal{R}$) and formulate it as the jitter to period ratio

$$\mathcal{R} = \frac{\sigma_\tau}{\tau} = \sigma_\tau \mathbf{f_0}. \tag{7.1}$$

In the previous studies, the contribution of flicker noise to randomness was assumed to be limited and was not calculated [21] for strong inversion region. In weak inversion, current levels of CMOS transistors are much smaller compared to strong inversion and it is assumed that white noise dominates the spectrum. Therefore, in [20] the contribution of flicker noise to jitter was supposed to be limited and was not included in randomness equations either. Even though the exact effect of flicker noise on jitter is not formulated clearly in the literature yet, we believe that flicker noise sources also have contribution to jitter. In the scope of this chapter, we will focus on short-term jitter where white noise effect is considerably larger than flicker noise effect. However, long-term jitter calculations should include the contribution of flicker noise, as well. In the following sections, IbRO and DRO cases are studied separately, in strong and weak inversion regions.

This chapter focuses on randomness analysis to gain design insights and predict randomness before starting to design a jittered oscillator based random number generator. Design parameters and their contribution to randomness are studied and randomness equations are derived in Sections 7.1 and 7.2 for both regions. In Section 7.3, a design example including estimation and simulation results is discussed. Furthermore, randomness behavior with power supply variation is investigated. The parameters which effect randomness are also examined in terms of the influences on

frequency and power.

7.1. Randomness in Strong Inversion Region

7.1.1. Inverter-Based Ring Oscillator

Average current can be rewritten as a function of frequency by the help of Equation 5.9 as

$$\mathbf{I = f_0 C M V_{DD}}. \tag{7.2}$$

Embedding the above current expression in Equation 5.11, the new jitter expression becomes

$$\sigma_\tau^{\mathbf{2}} = \frac{\mathbf{2kT}}{\mathbf{f_0^2 C M V_{DD}}}\left(\frac{\mathbf{2}}{\mathbf{V_{DD} - V_t}}(\gamma_\mathbf{N} + \gamma_\mathbf{P}) + \frac{\mathbf{2}}{\mathbf{V_{DD}}}\right). \tag{7.3}$$

Randomness equation $(\mathcal{R} = \sqrt{\sigma_\tau^2 f_0^2})$ can finally be expressed as

$$\mathcal{R} = \sqrt{\frac{\mathbf{2kT}}{\mathbf{C M V_{DD}}}\left(\frac{\mathbf{2}}{\mathbf{V_{DD} - V_t}}(\gamma_\mathbf{N} + \gamma_\mathbf{P}) + \frac{\mathbf{2}}{\mathbf{V_{DD}}}\right)}. \tag{7.4}$$

From this expression, it can be observed that randomness depends on the design parameters such as load capacitance (C), which is a function of inverter size and C_{ox}, stage number (M), and supply voltage (V_{DD}) and process parameters such as V_t, γ_N, and γ_P.

In order to derive randomness related to a given oscillation frequency, another approach is developed as follows. If f_0 is fixed and the current expression is rewritten taking velocity saturation into account, randomness can be rewritten as

$$\mathcal{R} = \sqrt{\frac{\mathbf{2kTf_0}}{\mathbf{k'\frac{W}{L}[V_{DSAT}(\frac{V_{DD}}{2} - Vt) - \frac{V_{DSAT}^2}{2}]}}(\frac{\mathbf{2(\gamma_N + \gamma_P)}}{\mathbf{V_{DD} - V_t}} + \frac{\mathbf{2}}{\mathbf{V_{DD}}})}, \tag{7.5}$$

where V_{DSAT} is the drain-source voltage of a transistor at the onset of velocity saturated region, and k' is the product of mobility (μ) and C_{ox}.

7.1.2. Differential Ring Oscillator

A similar procedure is followed for a DRO. Average current, which charges and discharges the load capacitance, can be restated in terms of oscillation frequency as

$$\mathbf{I = 2ln2f_0CMV_{op}}. \tag{7.6}$$

By using the above current equation and Equation 5.21, standard deviation of period jitter is rewritten as

$$\sigma_\tau^2 = \frac{\mathbf{kT}}{\mathbf{(ln2)^2 f_0^2 CMV_{op}}}[\gamma(\frac{\mathbf{3}}{\mathbf{4V_{effd}}} + \frac{\mathbf{1}}{\mathbf{V_{efft}}}) + \frac{\mathbf{1}}{\mathbf{V_{op}}}]. \tag{7.7}$$

Eventually, randomness equation for a DRO is extracted as

$$\mathcal{R} = \sqrt{\frac{\mathbf{kT}}{\mathbf{(ln2)^2 CMV_{op}}}[\gamma(\frac{\mathbf{3}}{\mathbf{4V_{effd}}} + \frac{\mathbf{1}}{\mathbf{V_{efft}}}) + \frac{\mathbf{1}}{\mathbf{V_{op}}}]}. \tag{7.8}$$

According to this expression, V_{op}, V_{efft} , and V_{effd} are fractions of supply voltage, M and C are the design parameters which randomness depends on. If the randomness

expression is rearranged for a given f_0, the following is obtained:

$$\mathcal{R} = \sqrt{\frac{\mathbf{kTf_0}}{\mathbf{ln2I}}[\gamma(\frac{\mathbf{3}}{\mathbf{4V_{effd}}} + \frac{\mathbf{1}}{\mathbf{V_{efft}}}) + \frac{\mathbf{1}}{\mathbf{V_{op}}}]}, \tag{7.9}$$

where I is the average current on the load capacitance expressed in terms of transistor parameters as

$$\mathbf{2I} = \mathbf{k'}(\frac{\mathbf{W}}{\mathbf{L}})_\mathbf{t}[\mathbf{V_{DSAT}}(\frac{\mathbf{V_{efft}}}{\mathbf{2}} - \mathbf{Vt}) - \frac{\mathbf{V_{DSAT}^2}}{\mathbf{2}}]. \tag{7.10}$$

During mirroring process of I, power supply dependence of V_{efft} is limited by saturation voltage limit of transistors because I is applied externally. Namely, V_{efft} can be ignored due to its small dependence on V_{DD}. On the other hand,

$$\mathbf{V_{op}} = \mathbf{V_{DD}} - \mathbf{IR_{eqload}}, \tag{7.11}$$

where R_{eqload} is given in [102] as

$$\mathbf{R_{eqload}} \approx \frac{\mathbf{3V_{DD}}}{\mathbf{4I}}(\mathbf{1} - \frac{\mathbf{5}\lambda\mathbf{V_{DD}}}{\mathbf{6}}), \tag{7.12}$$

where λ is a processes dependent coefficient. Finally, V_{effd} is

$$\mathbf{V_{effd}} = \mathbf{V_{DD}} - \mathbf{V_{op}} - \mathbf{V_{efft}}. \tag{7.13}$$

By taking into consideration the aforementioned equations, Equation 7.9 can be interpreted again. Due to the smaller value of V_{efft} among the voltages in Equation 7.9, it has the strongest effect in terms of magnitude on randomness value. Even though the effect is strongest, power supply dependence of V_{efft} is still limited as discussed above. Hence, randomness of DRO does not have a strong dependence on V_{DD}.

7.2. Randomness in Weak Inversion Region

By using the jitter expressions for weak inversion, randomness performance of CMOS ROs is examined. In the following subsections, IbRO and DRO cases are considered separately.

7.2.1. Inverter-Based Ring Oscillator

The average current given in Equation 7.2 can be used to derive the jitter equation as well. Using this current expression in Equation 6.3, the jitter can be written as

$$\sigma_{\tau}^{2} = \frac{\mathbf{q}}{\mathbf{f_0^2 C M V_{DD}}}(\mathbf{1} + \mathbf{e}^{-\mathbf{V_{DD}/2U_t}}). \tag{7.14}$$

The equation $(\mathcal{R} = \sqrt{\sigma_{\tau}^2 f_0^2})$ satisfies the randomness. In conjunction with this, randomness equation is attained for weak inversion region as

$$\mathcal{R} = \sqrt{\frac{\mathbf{q}}{\mathbf{C M V_{DD}}}(\mathbf{1} + \mathbf{e}^{-\mathbf{V_{DD}/2U_t}})}. \tag{7.15}$$

This expression asserts that randomness depends on the design parameters such as power supply voltage (V_{DD}), the load capacitance (C,) which is a function of inverter size and C_{ox}, and the number of stages (M).

7.2.2. Differential Ring Oscillator

White noise induced period jitter of DRO presented in Equation 6.13 is restated as

$$\sigma_\tau^2 = \frac{\mathbf{q}}{(\mathbf{ln2})^2\mathbf{f_0^2CMV_{op}}}[\frac{\mathbf{3}}{\mathbf{4}}(\mathbf{1}+\mathbf{e}^{-\mathbf{V_{dsd}/U_t}})+\frac{\mathbf{1}}{\mathbf{2}}(\mathbf{1}+\mathbf{e}^{-\mathbf{V_{dst}/U_t}})+\mathbf{2}] \tag{7.16}$$

for weak inversion operating region by including the average current given in Equation 7.6, where V_{op} is the differential peak output voltage swing, V_{dsd} is the drain-source voltage of differential pair transistors, and V_{dst} is the drain-source voltage of tail transistor. Afterwards, the randomness equation for a DRO is eventually obtained as

$$\mathcal{R} = \sqrt{\frac{\mathbf{q}}{(\mathbf{ln2})^2\mathbf{CMV_{op}}}[\frac{\mathbf{3}}{\mathbf{4}}(\mathbf{1}+\mathbf{e}^{-\mathbf{V_{dsd}/U_t}})+\frac{\mathbf{1}}{\mathbf{2}}(\mathbf{1}+\mathbf{e}^{-\mathbf{V_{dst}/U_t}})+\mathbf{2}]}. \tag{7.17}$$

According to this expression, randomness depends on the design parameters M, C, and voltages V_{op}, V_{dsd}, and V_{dst} which are functions of supply voltage (V_{DD}).

7.3. Performance of RO as RNG

After the verification of phase noise equations, randomness performance of oscillators is examined for security applications as mentioned previously. Therefore, we focus on how to increase the randomness of the IbRO and DRO by using the derived equations in Sections 7.1 and 7.2. As seen from Equations 7.4, 7.8, 7.15, and 7.17, randomness ($\mathcal{R}$) depends on M, C, V_{DD}. Similarly, oscillation frequency (f_{osc}) and power consumption (P) also depend on the same parameters expressed analytically,

$$\mathcal{R} = \mathbf{f_1}\{\mathbf{M},\mathbf{C},\mathbf{V_{DD}}\} \tag{7.18}$$

$$\mathbf{f_{osc}} = \mathbf{f_2}\{\mathbf{M},\mathbf{C},\mathbf{V_{DD}}\} \tag{7.19}$$

$$\mathbf{P} = \mathbf{f_3}\{\mathbf{M},\mathbf{C},\mathbf{V_{DD}}\}. \tag{7.20}$$

For this reason, one should be aware that a change in one of the M, C, and V_{DD} parameters not only affects the randomness but also the power consumption and the oscillation frequency. For a fair comparison, performance parameters of *power consumption* and *oscillation frequency* are combined into a single one as *energy* as shown in Equation 7.21. Hereafter, performance comparison of oscillators is done according to randomness versus energy measure.

$$\textbf{Energy} = \frac{\textbf{Power}}{\textbf{Frequency}} \tag{7.21}$$

In order to obtain maximum randomness from an RO, following cases should be provided according to randomness equations given in Equations 7.4, 7.8, 7.15, and 7.17,

(i) minimum transistor sizes which affect the capacitor (C) values
(ii) minimum number of stages (M)
(iii) minimum supply voltage (V_{DD})

By taking the above mentioned issues into account, randomness performance is examined by changing supply voltage and stage number, separately. Moreover, transistor sizes are adjusted to their minimum values as far as the technology and design permit. We compare the randomness behaviors of strong and weak inversion regions by sweeping V_{DD} starting from $0.4V$ for IbRO and $0.6V$ for DRO, where oscillators still have the oscillation behavior, up to $2.5V$. Furthermore, the effect of stage count is investigated by changing stage count of IbRO from three to 23 and DRO from three to seven. It is seen from Figure 7.1 that randomness increases with decreasing energy. In fact, if the speed specification meets the requirements of the system, this is the perfect condition for security applications. According to the figure, maximum randomness is obtained in the case of IbRO with three stages at $V_{DD} = 0.4V$ in weak inversion, while minimum energy is achieved as well. Big black circles in this

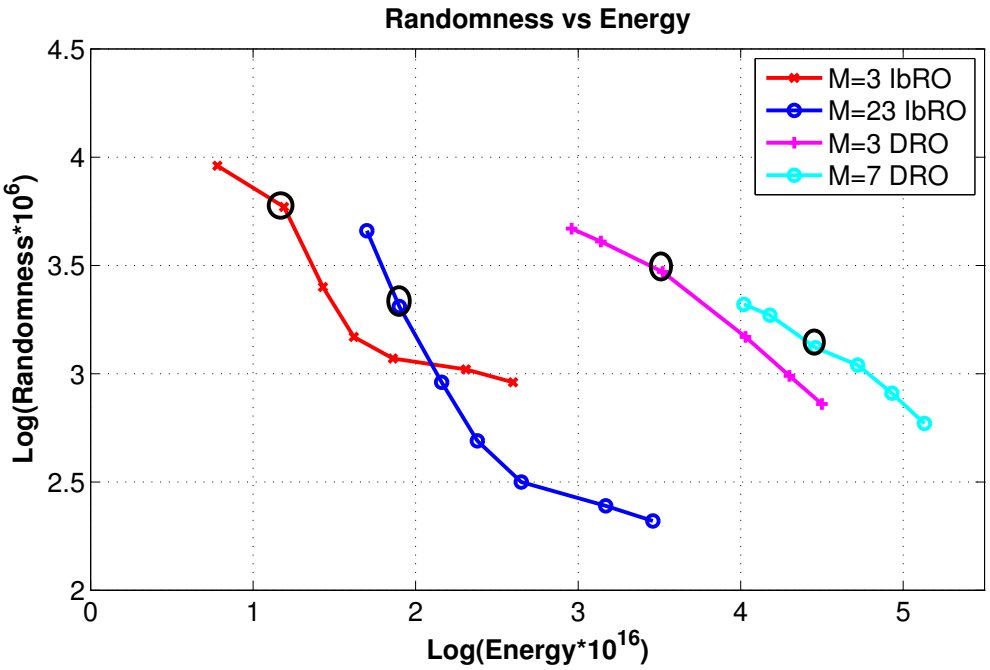

Figure 7.1. Randomness vs energy for IbRO.

figure and in the following figures indicates the weak and strong inversion borders.

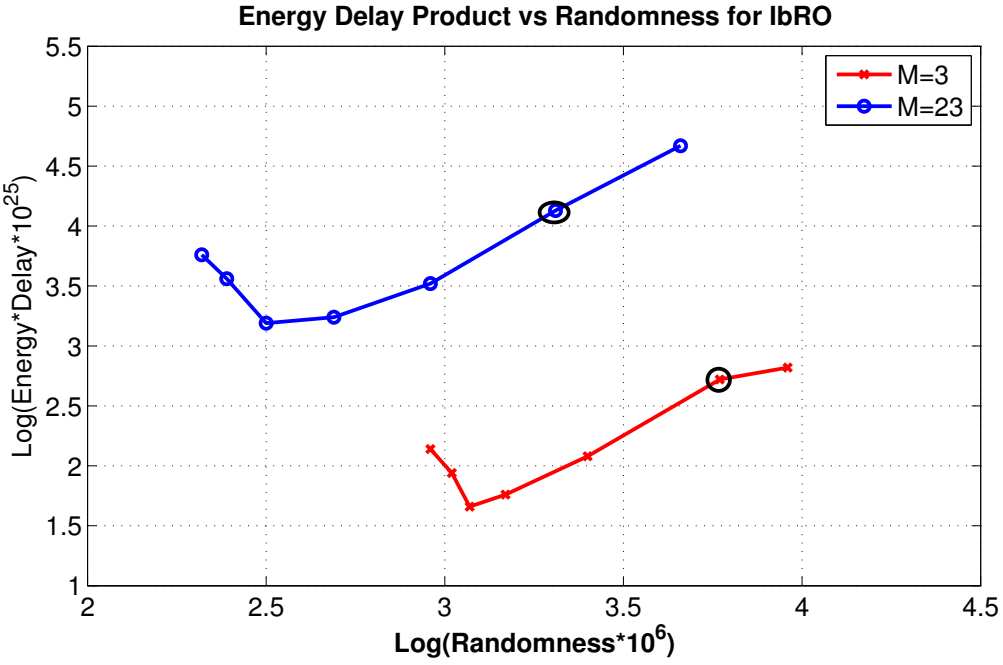

Figure 7.2. Randomness vs energy delay product for IbRO.

Randomness of a 23-stage IbRO increases 21.62 times by decreasing the supply voltage from $2.5V$ to 0.4V, while 56.58 times reduction in energy is achieved. On the other hand, decreasing the supply voltage from $2.5V$ to 0.6V causes 12.99 times increase in randomness and 33.23 times reduction in energy for a 7- stage DRO. With a 0.6V supply voltage, 7-stage DRO has 3.25 % jitter in a period while 23-stage IbRO with a supply voltage of 0.4V has only 0.45 %. However, for the same amount of randomness, DRO consumes more energy than inverter-based one.

In fact, performance comparison with energy is always in favor of low speed. For a fair comparison, we also investigate the energy delay product behavior versus randomness. The results are displayed in Figure 7.2 for IbRO. It is seen from the figure that, there is a minimum for energy delay product. If energy is the main concern, supply voltage can be adjusted to the low levels of sub-threshold. Hence maximum randomness can be performed with minimum energy. However, if speed is also an issue, the oscillator can be set to the optimum point of energy delay product.

f_0 can be adjusted to the desired frequency by changing V_{DD} and M inversely. As a consequence of changing V_{DD} and M, randomness may have different values for each case. Hence, randomness performance is investigated for a given frequency. Since weak inversion randomness parameters are also studied in this thesis with strong inversion counterparts, f_0 is selected $\sim 10MHz$ in order to observe weak inversion behavior of oscillators with strong inversion behavior comparatively. A DRO with 3-stage $@V_{DD} = 0.75V$ and an IbRO with 13-stage $@V_{DD} = 0.5V$ will oscillate around $10MHz$ in weak inversion, while a DRO with 259-stage $@V_{DD} = 2.5V$ and an IbRO with 1351-stage $@V_{DD} = 2.5V$ have an oscillation frequency of $\sim 10MHz$ in strong inversion. Figure 7.3 demonstrates randomness vs. V_{DD} behavior for a given frequency ($\sim 10MHz$) for both types of ring oscillators. Even for a given f_0, there is also a freedom for increasing randomness with decreasing V_{DD}. According to the figure, both ROs exhibit dependence on V_{DD}, while DRO has more randomness in both regions.

To compare performance parameters of oscillators, an IbRO and a DRO have

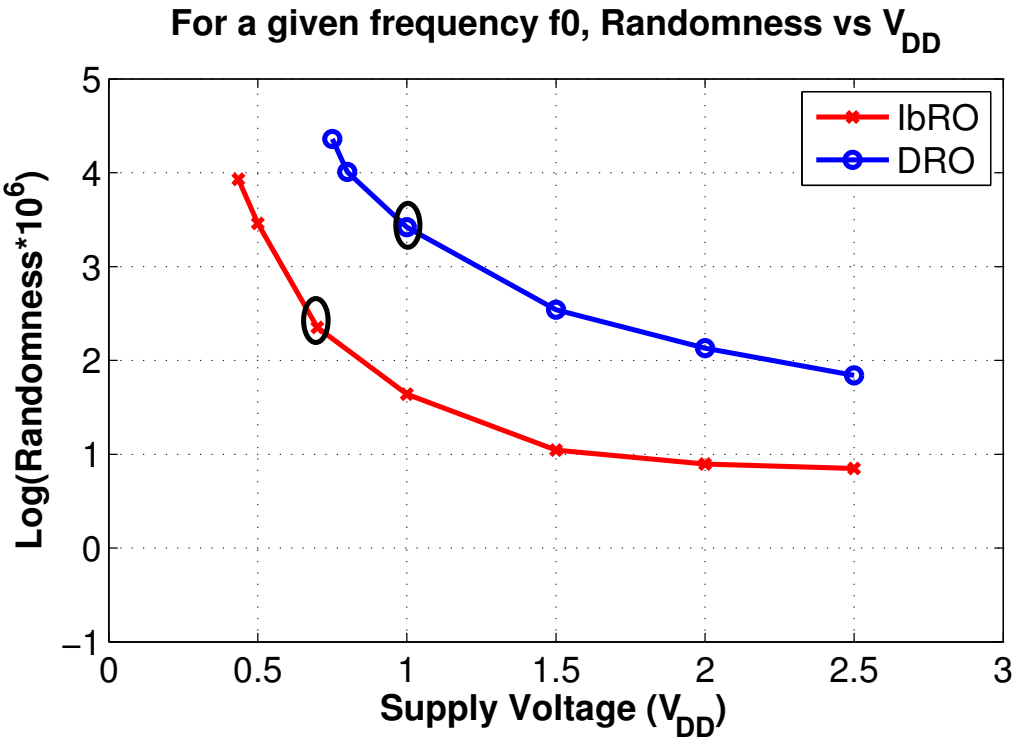

Figure 7.3. Supply voltage vs randomness behavior for a given frequency.

Table 7.1. Transistor aspect ratios of DRO.

Transistor	**Aspect Ratio W/L**
M_1	2/0.5
M_2	20/0.5
M_3	40/0.5
M_4, M_5	1.5/0.24
M_6, M_7, M_8	32/0.5

been designed. For IbRO, $(\frac{W}{L})_n = \frac{0.48}{0.24}$ and $(\frac{W}{L})_p = \frac{0.7}{0.24}$ is chosen in order to fit rule of minimum size transistors for maximum randomness. Aspect ratios of transistors used in DRO is given in Table 7.1, as well. In order to have an oscillation frequency $\sim 10MHz$, stage counts are selected as given in the previous paragraph. Their performances are summarized and compared in Table 7.2. According to this table, for the

given oscillation frequency of $\sim 10MHz$, ring oscillators have maximum randomness in weak inversion. For example, decreasing supply voltage from $2.5V$ to $0.75V$ results in 332 times more randomness for DRO, while decreasing supply voltage from $2.5V$ to $0.435V$ results 1203 times more randomness for IbRO. Operating in weak inversion rather than strong inversion benefits greater ratio of increase in randomness for IbRO. Moreover, if we compare the randomness performances of ROs in the same region, DRO presents 2.72 times more randomness than IbRO in weak inversion region; on the other hand, DRO has 9.85 times more randomness in strong inversion. For both cases DRO consumes more energy and needs more area.

7.4. Conclusion

Randomness equations are obtained for IbRO and DRO for both strong and weak inversion regions. Analysis shows that both ROs exhibit more randomness with less energy in the weak inversion region. Moreover, by using 3-stage ROs in the extremity of weak inversion, IbRO has almost twice the DRO's jitter where a supply voltage of $0.4V$ for IbRO and $0.6V$ for DRO is applied. Oscillation frequency and power budget are the two important parameters that should be considered. Maximizing randomness in a RO may cause a trade-off in oscillation frequency. Analyses showed that for a given f_0, there is a freedom for increasing randomness by decreasing V_{DD}. This result is valid for both ROs. Finally, for a given f_0, a case study was performed. It was seen that in strong inversion ($@V_{DD} = 2.5V$), DRO has 80.8 times higher randomness than IbRO, while in weak inversion it has 2.72 times more randomness. However, the sensitivity of randomness to supply voltage is greater in the IbRO case.

Table 7.2. Comparison of performance parameters for both RO ($f_0 \approx 10\ MHz$).

Parameters	**Inverter-based RO**		**Differential RO**	
	Weak Inversion $@V_{DD} = 0.435V$	**Strong Inversion** $@V_{DD} = 2.5V$	**Weak Inversion** $@V_{DD} = 0.75V$	**Strong Inversion** $@V_{DD} = 2.5V$
Power (μW)	0.037	144411	1.04	13778
Energy (fW/Hz)	1.19	11100	153	20.5×10^6
$\log 1e16 \times Energy$	0.077	4.044	2.18	7.31
Energy Delay Product $(10^{-22}W/Hz^2)$	1.16	11100	151	20.3×10^6
$\log 1e25 \times Energy \times Delay$	3.62	7.044	5.18	10.3
Randomness (μ)	8500	7.06	23119	69.55
$\log 1e6 \times Randomness$	3.93	0.85	4.36	1.84
Area $((\mu m)^2)$	88.56	38405	2136	184444

8. FLICKER NOISE EFFECT ON RANDOMNESS OF CMOS RO

Despite the abundance of research on RO-based RNGs, there are still some open problems, such as the effect of flicker noise on randomness. Flicker and white noise sources are the two main components of jitter in ROs. Both flicker and white noise are in the nature of semiconductors and both are inevitable. White noise has a uniform PSD, which enables obtaining uncorrelated random numbers. Therefore, it is preferred in random number generation. On the other hand, many researchers have questioned the use of flicker noise for randomness applications, because its colored PSD may cause correlation in neighboring bits. Although this is a valid concern, it requires further analysis. Since there is limited research on this topic, most researchers simply ignore flicker noise in their analyses.

Correlation between neighboring bits is inversely proportional to the sampling period of the RNG. In conjunction with this, the amount of the extracted entropy is directly relevant to sampling, since the folding effect of noise causes variations in the noise level in the sampled band. Consequently, variations in the apparent noise level lead to deviations in entropy levels as well.

In Section 8.1, folding effect of flicker noise on the sampling frequency is investigated. Flicker noise effect on randomness is examined by entropy analysis in Section 8.2. Models developed in MATLAB are utilized to produce synthetic bit streams with white noise, flicker noise, and combined noise. The results are discussed and demonstrated in the same section.

8.1. Folding Effect of Flicker Noise on the Sampling Frequency

Noise sampling has been studied in the scope of Switched Capacitor (SC) and Sample and Hold (S/H) circuits [103, 104]. A detailed noise analysis was done for comparator-based Analog to Digital Converter (ADC) circuits in [104], which breaks

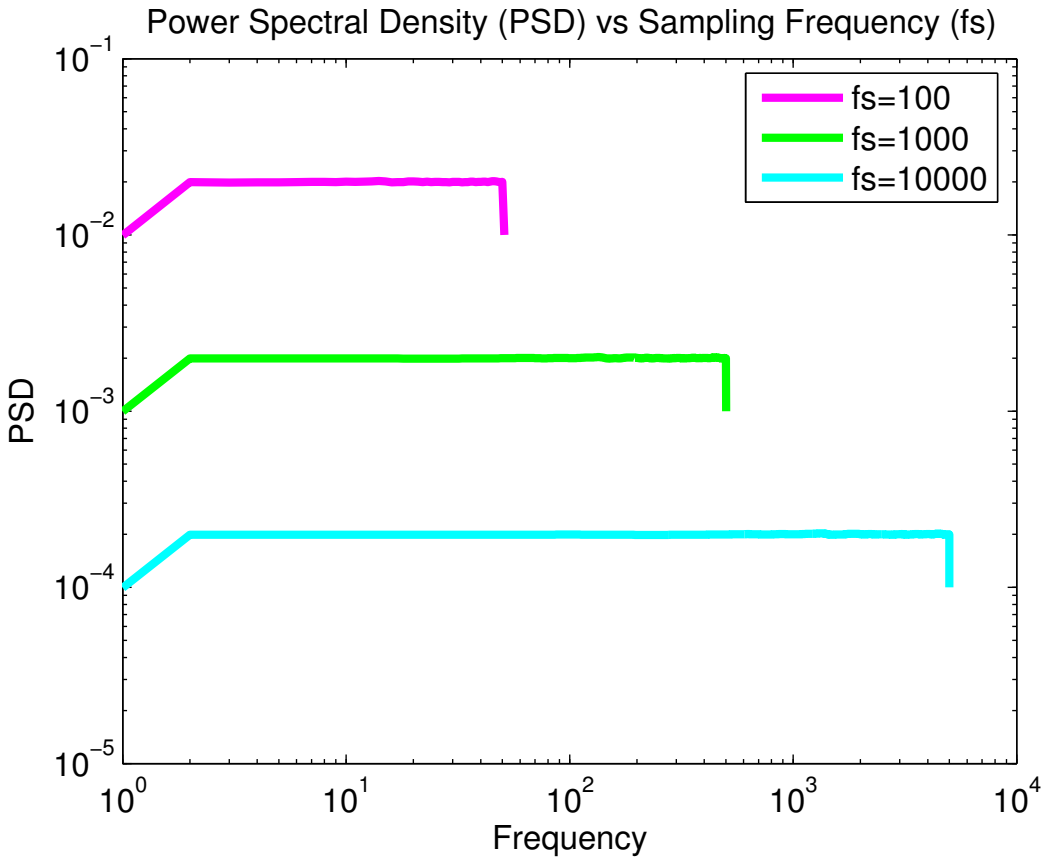

Figure 8.1. PSD of white noise with varying fs.

down the input referred noise as baseband flicker, apparent white, thermal, and folded flicker. Furthermore, it is stated in the same study that the dominant noise source in apparent white noise is the folded flicker noise of the preamplifier. This study emphasizes the importance of the folding effect of flicker noise due to sampling.

White noise power increases by decreasing sampling frequency as illustrated in Figure 8.1. According to that figure, when the sampling frequency rate is decreased to half, the PSD of white noise is doubled, since the total noise power remains the same. As can be seen in Figure 8.2 flicker noise also exhibits a similar behavior as white noise. Transformation of flicker noise by sampling and interpretation of its effects in connection with random number generation is well understood with the study in [105], where the aliasing phenomenon in $\frac{1}{f^{\alpha}}$ noise spectra was examined. It was shown that the spectral power of noise increases when sampling frequency goes below the Nyquist

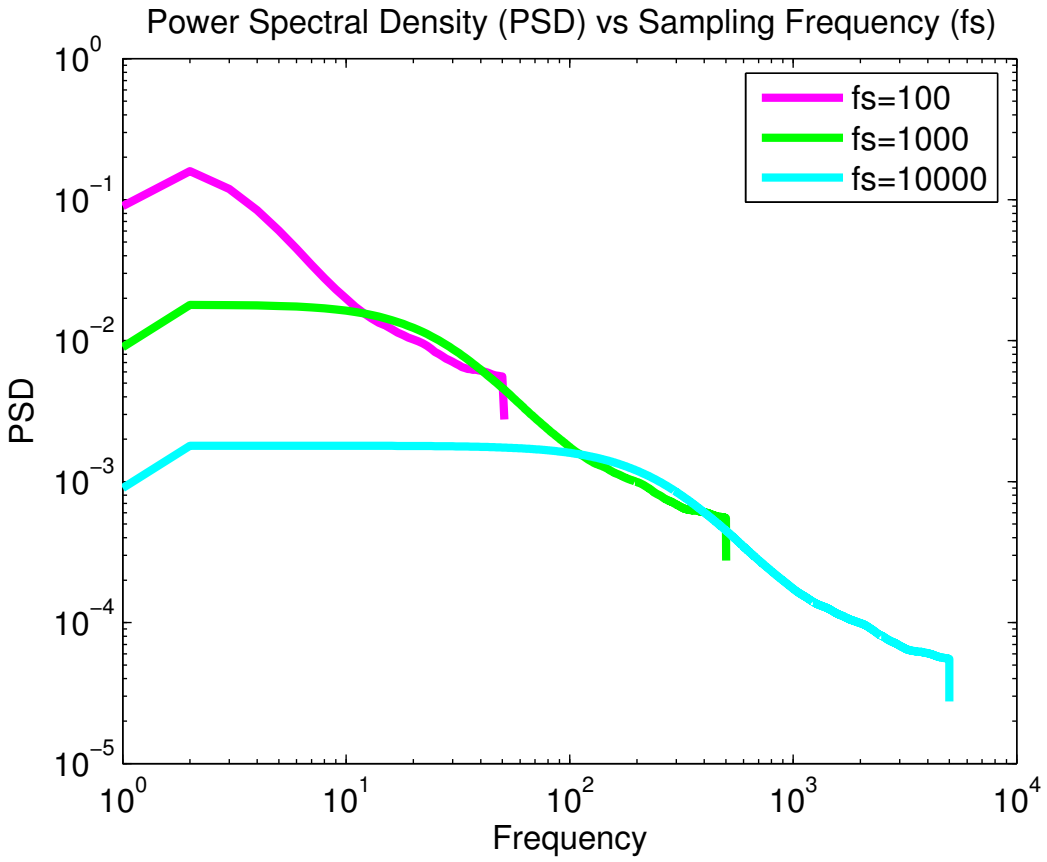

Figure 8.2. PSD of flicker noise with varying fs.

frequency, $f_N = 0.5f_s$. This is because of aliasing, i.e., the interference between the replicas of the flicker noise PSD. Moreover, in the same study, it was also claimed that flicker noise approximates white noise when the sampling frequency is low enough (under-sampling).

Aliasing phenomenon of flicker noise is demonstrated in Figure 8.3. In this simulation, the flicker noise model, which is used to generate synthetic bit streams with flicker noise and combined white and flicker noise, is utilized. The upper part of Figure 8.3 represents 1,000 times down-sampled version of flicker noise PSD. The lower part of the figure demonstrates 10,000 times down-sampled version of flicker noise PSD, which actually shows that if the flicker noise is down-sampled enough, it approximates white noise with higher power.

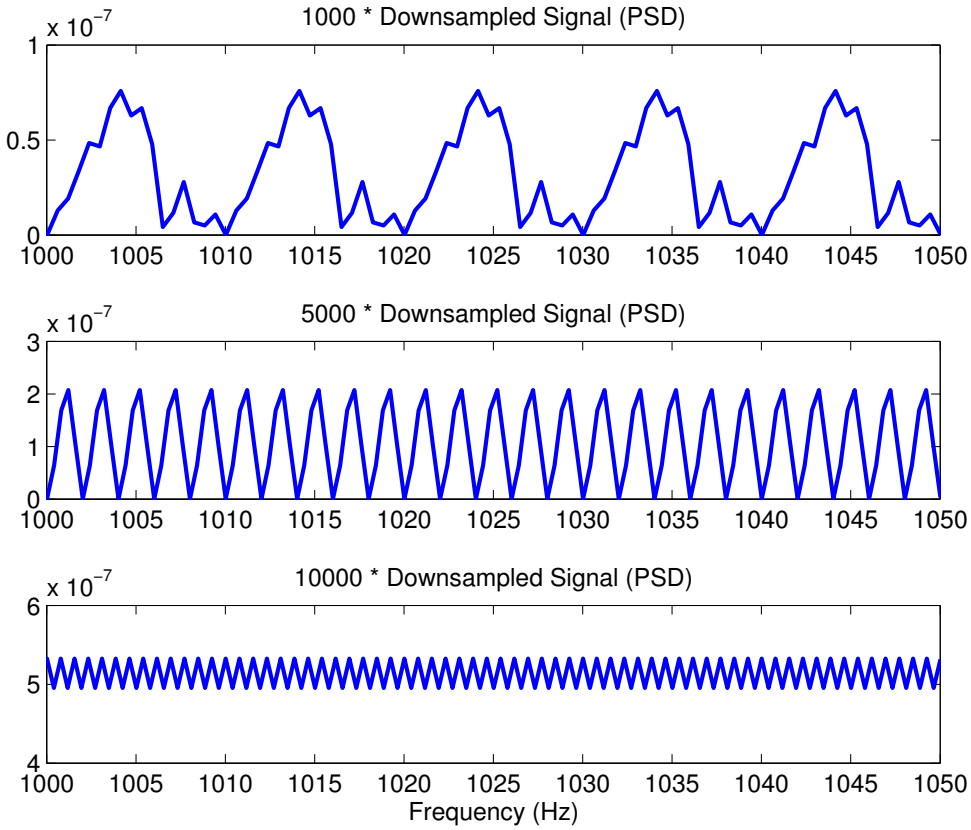

Figure 8.3. Aliasing in flicker noise model used in analysis.

8.2. Entropy Analysis of Noise Sources of an RO

The lack of flicker noise analysis in previous RNG design processes was a shortcoming. A complete noise analysis is mandatory to improve the randomness obtained from electronic oscillators. Such an understanding may allow the designer to constructively use flicker noise for entropy generation.

Flicker and white noise exist in the nature of devices and create jitter together. So, it is impossible to separate and turn them off individually in the natural world. Our strategy is therefore to produce synthetic bit streams with separate noise sources, to compare their entropy, and to assess their individual contributions to randomness.

Three types of synthetic bit streams are generated by MATLAB for ROs oper-

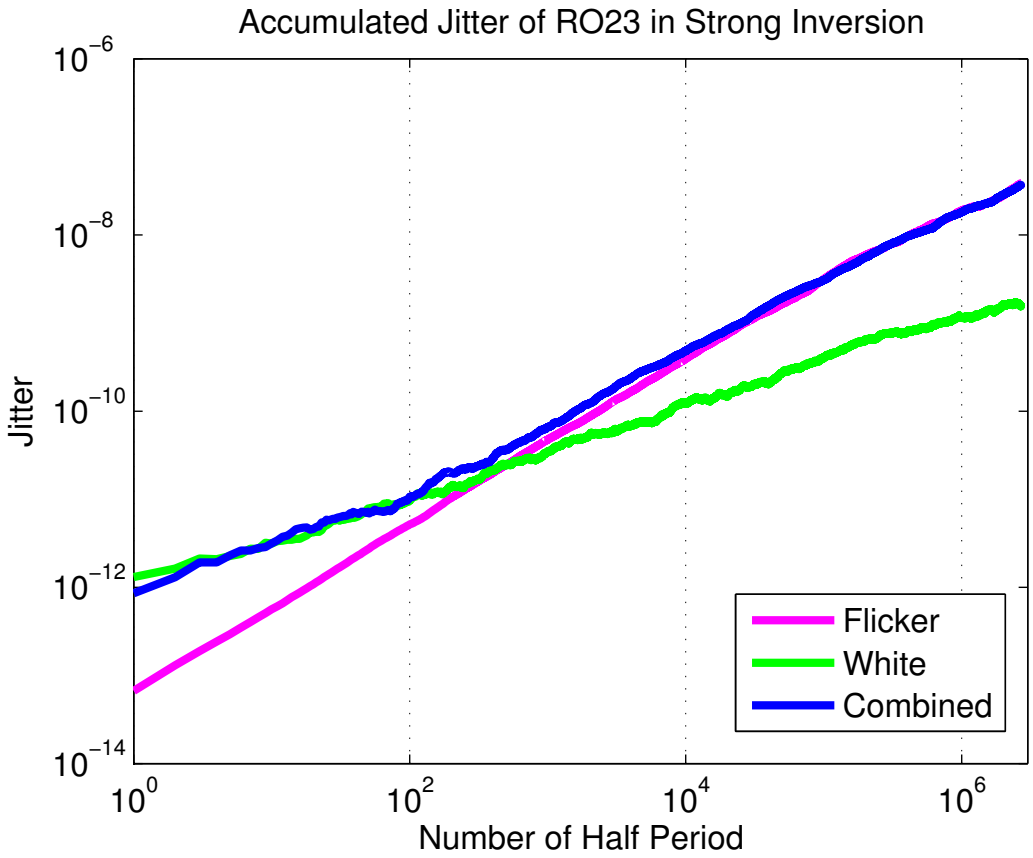

Figure 8.4. Accumulated jitter vs time for an RO with 23 inverters in strong inversion.

ating in strong inversion. One of the bit streams is generated with only flicker noise, another with only white noise, and the last one with both, called combined noise in the rest of the thesis. Utilizing these noise sources, a simulation model of the RO output, including jitter, can be generated. In order to investigate jitter, the long-term behavior of an RO operating in strong inversion is considered. Accumulated jitter behavior of this RO output is also generated in MATLAB, as shown in Figure 8.4. The accumulating behaviors of flicker and white noise induced jitter are in agreement with previous measurements given in [73]. The initial jitter levels are estimated from the phase noise measurements of an inverter-based ring oscillator (IbRO), consisting of 23 inverters [19]. Therefore, the jitter models in MATLAB are based on measurements.

At least 20 different outputs are generated for all three cases. T-entropies of

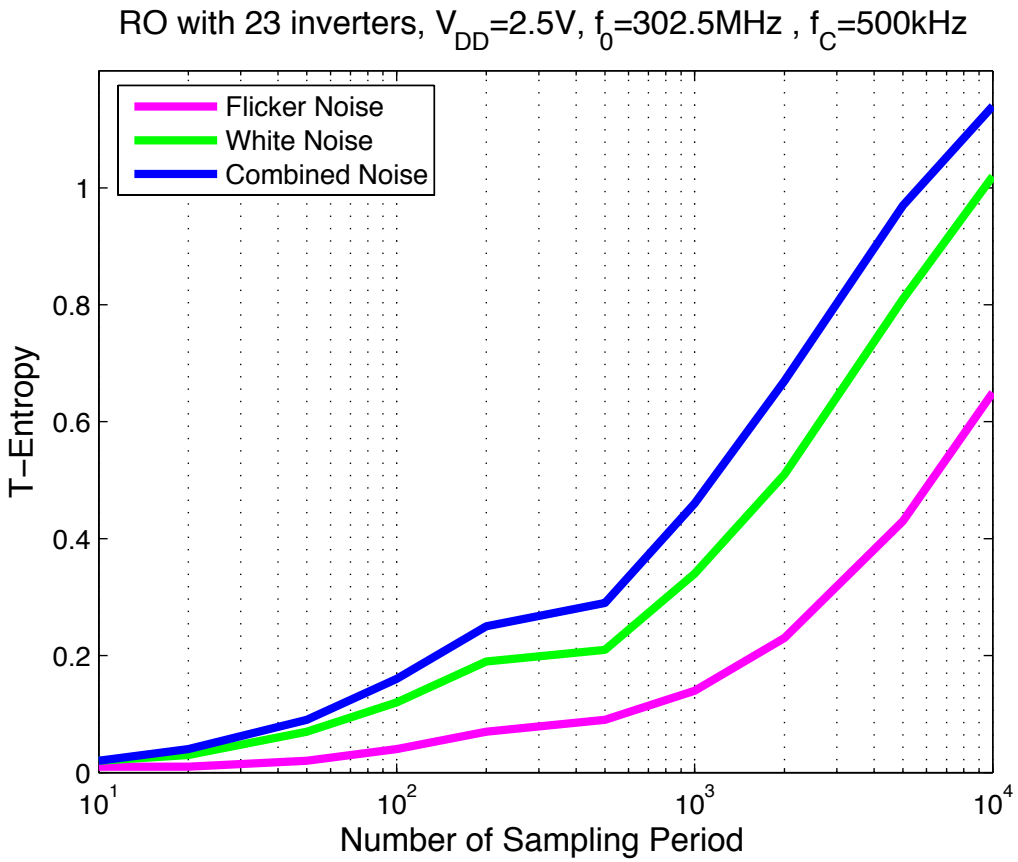

Figure 8.5. T-Entropy values of an RO in strong inversion with different noise sources induced bit streams.

the outputs are computed and an average entropy level is obtained from 20 entropy values. The results are presented in Figure 8.5, where combined noise source induced bit stream is observed to have more randomness than the white noise induced bit stream. Thus, we can conclude that flicker noise potentially has a positive effect on entropy. This conclusion promotes new approaches for flicker noise in random number generation.

T-entropy measure is used to calculate the randomness of ROs in the analysis. As mentioned in the introduction, T-entropy is an approximation of the Shannon entropy for finite bit streams. In T-entropy estimation, the theoretical maximum randomness level is eight for binary streams; however, an entropy level of at least 5.3 was observed to be required for the generated RO outputs to pass the NIST 140-2 test suite [29].

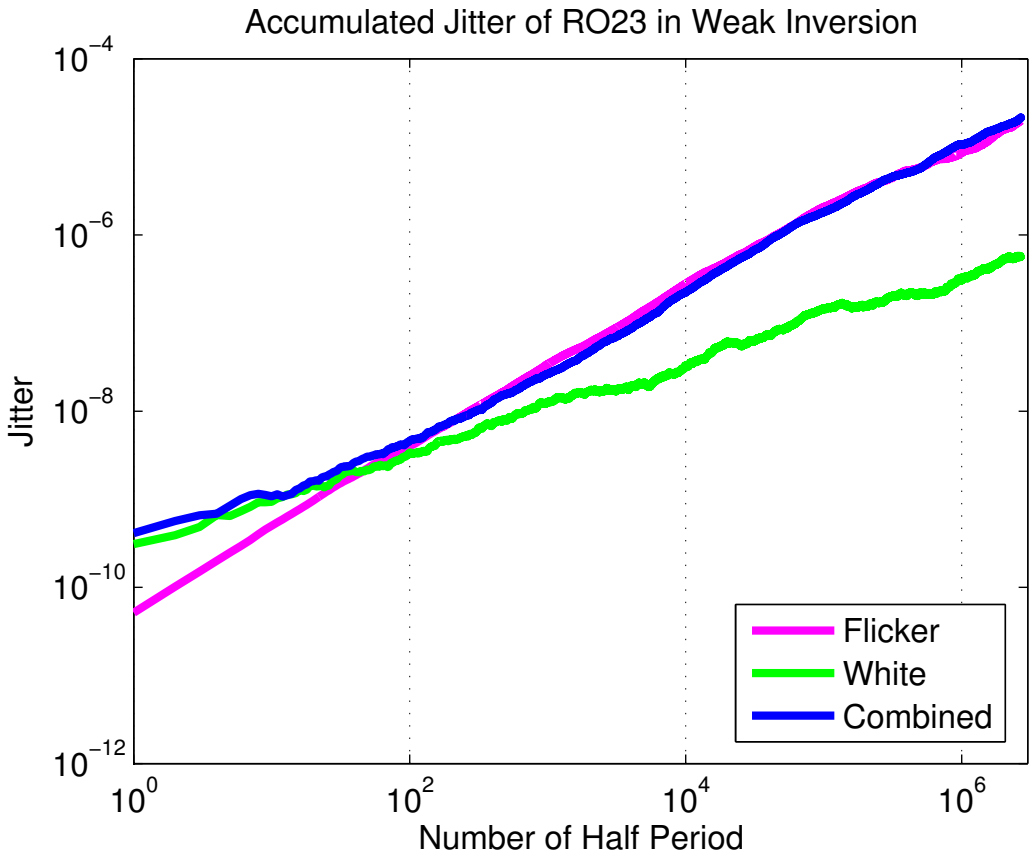

Figure 8.6. Accumulated jitter vs time for an RO with 23 inverters in weak inversion.

Similar to the investigation on weak inversion noise behavior of ROs in [19], Figure 8.6 shows the accumulated jitter behavior of an RO in weak inversion. This data is used for generating bit streams and Figure 8.7 presents the entropy values in weak inversion. Studying weak inversion besides strong inversion completes the study and shows the big picture.

Figure 8.7 also confirms the results obtained in strong inversion, where the combined noise still produces the highest entropy. An interesting result seen in Figure 8.7 is that the amount of entropy contribution of the flicker noise case is larger than its own entropy value. This observation leads us to believe that the interaction between white and flicker noise components results in an even bigger entropy level.

It is studied in [7] that increasing the sampling period leads to a decrease in

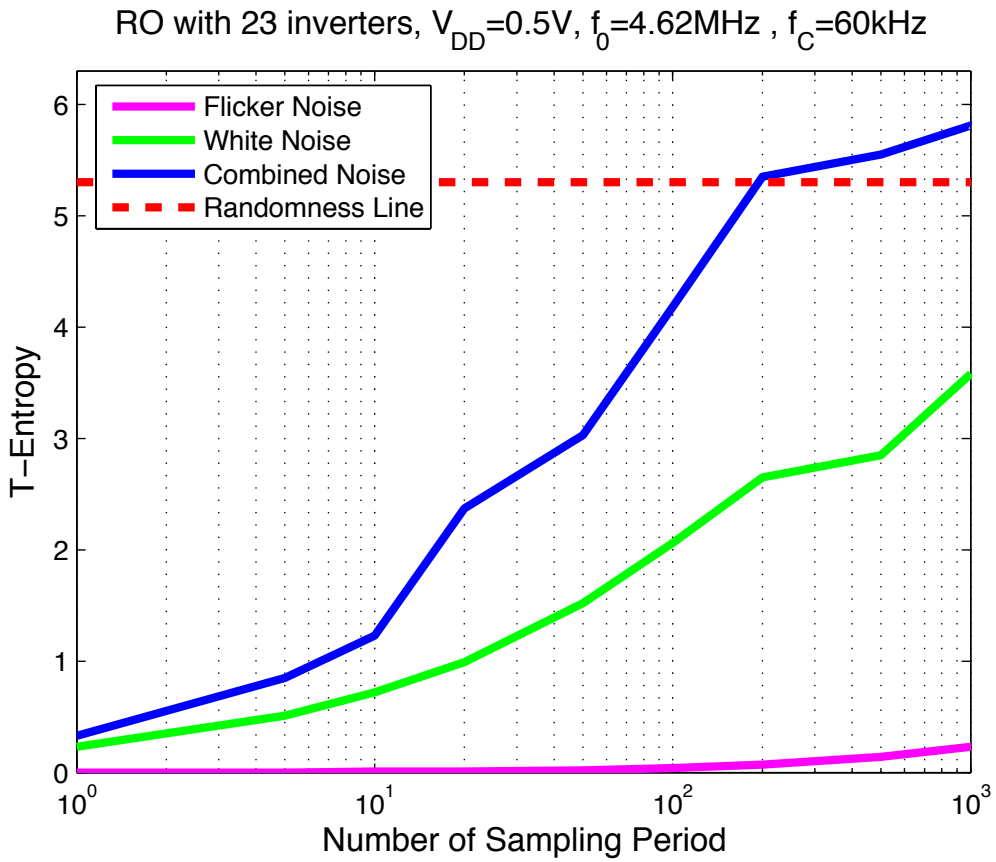

Figure 8.7. T-Entropy values of an RO in weak inversion with different noise sources induced bit streams.

autocorrelation. Obviously, jitter accumulates between sampling points over time, which is confirmed by our results. The effect of sampling on entropy could also be observed from Figures 8.5 and 8.7. According to these figures, entropy levels rise with sampling period.

From another point of view, if the raw data before the sampling process is considered, flicker noise causes an increase in randomness even if there is correlation between adjacent bits. This statement is confirmed by looking at the T-entropy level for unit sampling period in Figure 8.7.

8.3. Conclusion

The effect of flicker noise on randomness in ring oscillators is investigated. Three different types of bit streams are produced in MATLAB with the given jitter spectrum. These bit streams have the flicker, white, and combined noise effects. The simulations show that flicker noise has a positive effect on entropy and successively on randomness. This conclusion is opposite to the previous interpretations. Additionally, the sampling effect on randomness, which is an essential topic for RNG, is examined. Increasing the sampling period of RNG yields an increase in the entropy of generated bits, since the noise power is folded with sampling.

9. DESIGN OF EFFICIENT CMOS RO-Based RNG

The RNG in [6] uses hundreds of ROs which consume power and occupy a large area on the silicon. Moreover, the variety of RNG applications becomes limited due to the very high power consumption. These drawbacks motivate better RNG designs.

In order to design an efficient RO-based RNG, entropy analysis is performed for several cases and the minimum necessary number of ROs is analyzed in Section 9.1. The randomness is evaluated by T-Entropy. As a performance measure, entropy per energy is proposed. A discussion of the analysis results is presented in Section 9.2. The measurement results are given in Section 9.3.

9.1. RO-based RNG Design

Since we have the entropy information of RO with 23 inverters in weak and strong inversion, comparing their entropy levels is a good starting point. RO outputs generated with combined noise source modeling the real world noise are compared in Figures 9.1. As mentioned in the previous section, an entropy level of approximately 5.3 is enough for bit streams to be qualified as random. As observed in Figure 9.1, an RO operating in weak inversion can obtain the required entropy level when sampled once every 200 periods. This is a favorable result, since the average current consumption of the RO operating in weak inversion is around $0.83\mu A$. Thanks to the idea of operating ROs in weak inversion, RO-based RNGs may become very attractive for wireless, mobile, and low power applications. The decrease in the speed of the RNG is a drawback, but the power consumption decreases even faster.

On the other hand, entropy level of an RO operating in strong inversion is far from the required entropy level. Therefore, in strong inversion, it is necessary to use multiple ROs to form an RNG. We XORed the outputs of ROs operating in strong inversion. Figure 9.2 presents the entropy level of XORed ROs. Unfortunately, required entropy level of randomness cannot be obtained even if 40 ROs are XORed

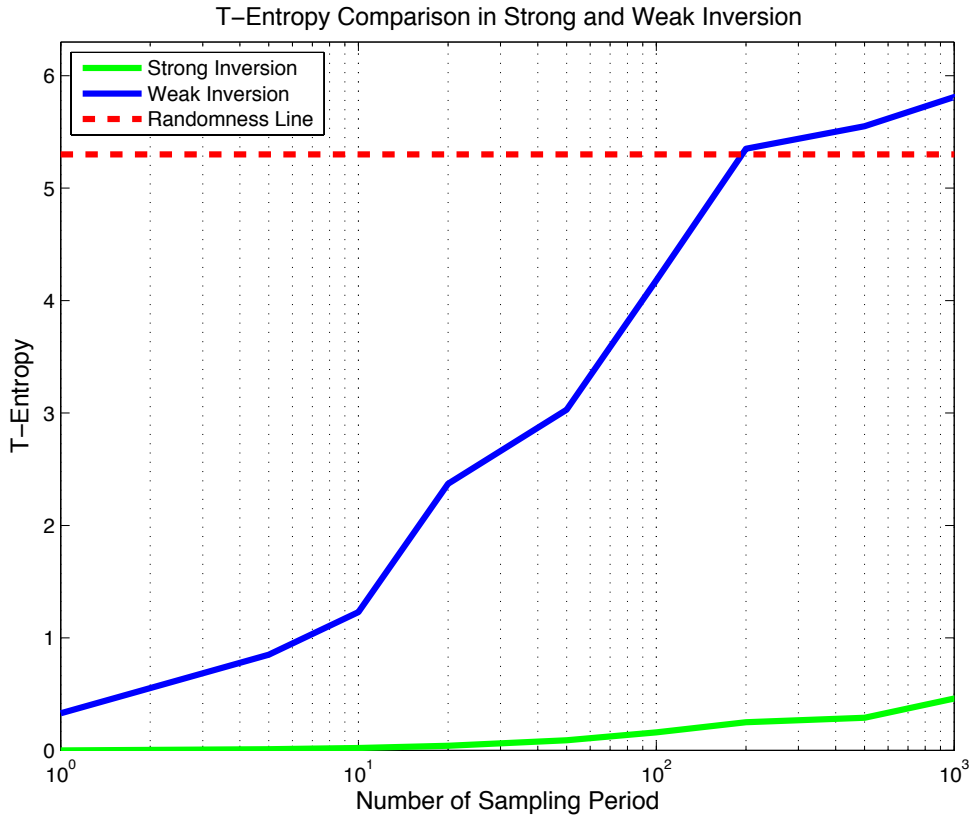

Figure 9.1. Entropy comparison of 23-inverter RO in strong and weak inversion.

according to this model.

Jitter is not the only source of randomness in RNGs there are other factors, such as drift in starting point of ROs and frequency differences among ROs. In Figure 9.2, jitter is used alone as a randomness source; thus, the required entropy level cannot be attained. Therefore, at least one of the other factors should also be included. The simulation is repeated by adding frequency scattering. To determine the amount of scattering, the measurements performed in [51] are used as a reference, where frequency scattering of ROs are shown to be confined to within ±2%. However, we should keep in mind that frequency scattering amounts may be somewhat different from those of [51] due to the different technologies involved. Therefore, the actual results may be slightly different from expected results. Figure 9.3 shows the number of XORed ROs and the corresponding entropy levels. The required entropy level can be attained

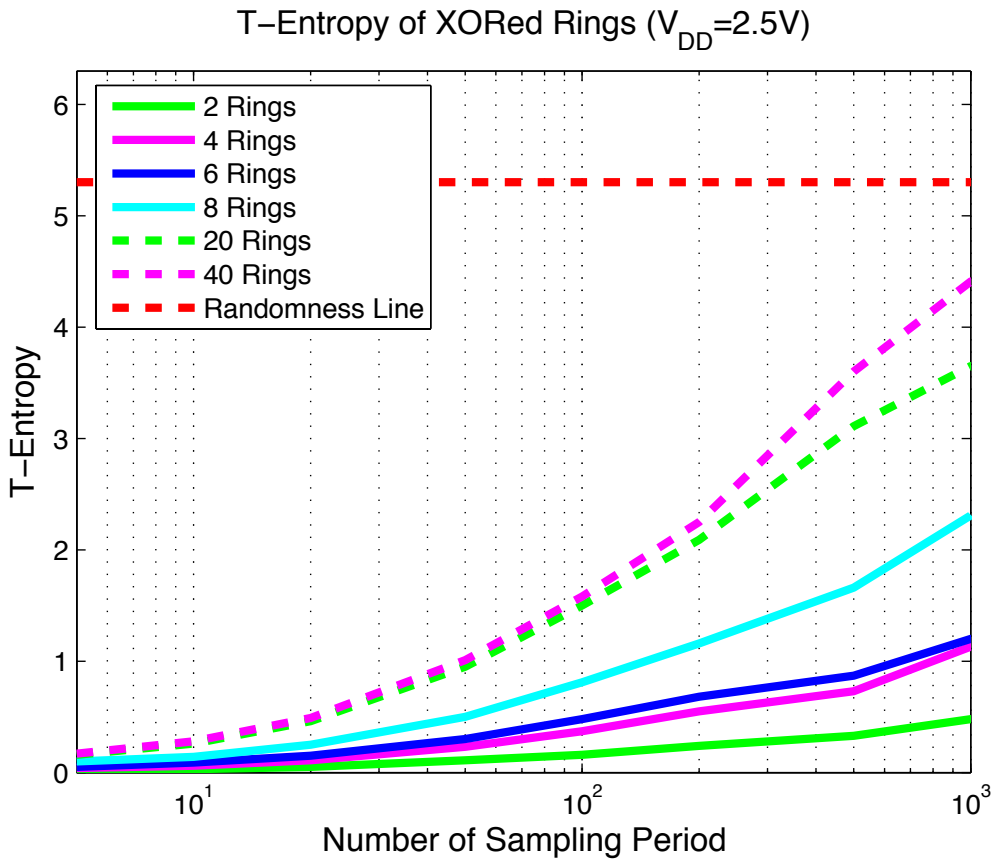

Figure 9.2. T-Entropy values of XORed ROs in strong inversion.

with at least nine XORed ROs. Moreover, while the XORed ROs are considered as operating independently from each other, there may be some phase locking among the ROs in practice, depending on the layout. We assume that by a careful layout design, locking effects are minimized. The dot on Figure 9.3 indicates the RNG configuration of 12 XORed ROs which is sampled once every 100 periods. This RNG configuration is chosen for evaluation with NIST 140-2 test suite.

It is observed from Figure 9.1 that even one RO operating in weak inversion can achieve the necessary randomness level. Nevertheless, there may be some statistical imperfections, such as bias at the output of ROs. Therefore, RNGs design with more than one RO should be preferred. Figure 9.4 exhibits the entropy of XORed ROs operating in weak inversion. It can also be observed from this figure that XORing helps to obtain the required entropy level with lower sampling periods which results

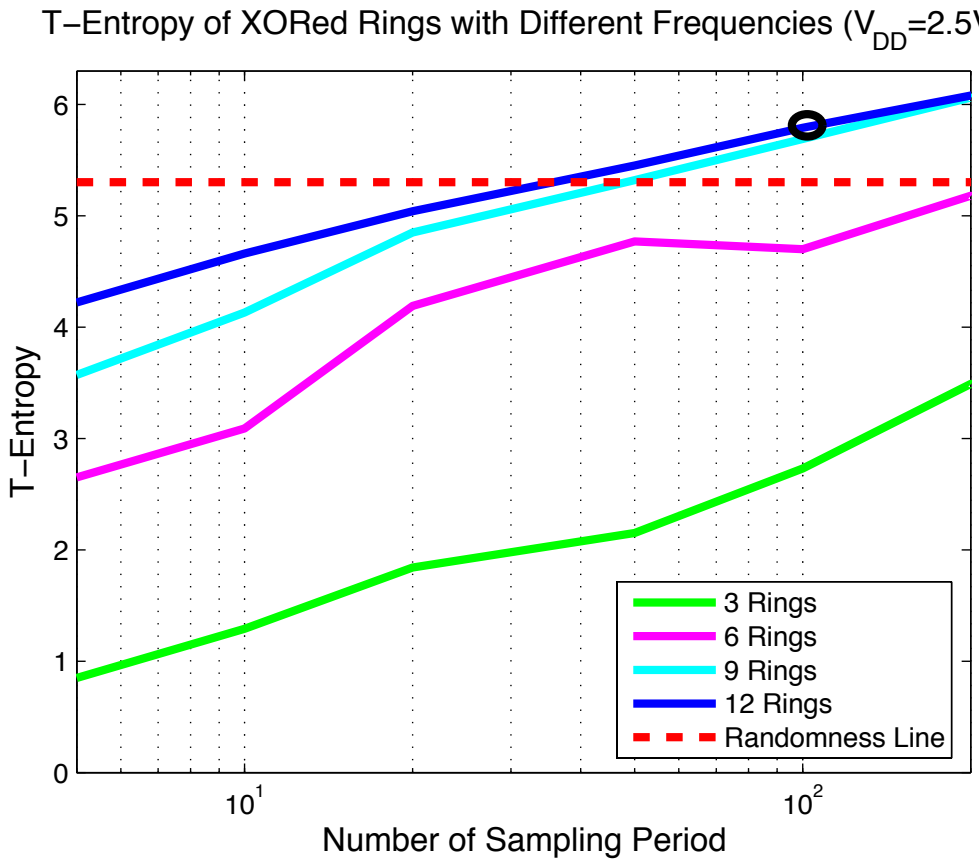

Figure 9.3. T-Entropy values of XORed ROs in strong inversion with scattered frequencies.

higher throughput.

Frequency scattering effect is also explored for weak inversion. Current variation is the most important factor that affects frequency scattering in an RO. Furthermore, V_{th} and dimension variations of transistors are the reasons of current variation. The effect of dimension variations on frequency scattering remains almost same in weak inversion. However, current has an exponential relation with V_{th} in weak inversion. The frequency variation of 1%, which is observed in strong inversion, may be caused from almost $100mV$ V_{th} variation. Since, the effect of V_{th} in weak inversion is exponential, for the same amount of $100mV$ V_{th} variation in weak inversion, it is estimated that 20% frequency scattering may be observed. In short, the V_{th} variation which causes 1% frequency scattering in strong inversion may cause around 20% frequency scatter-

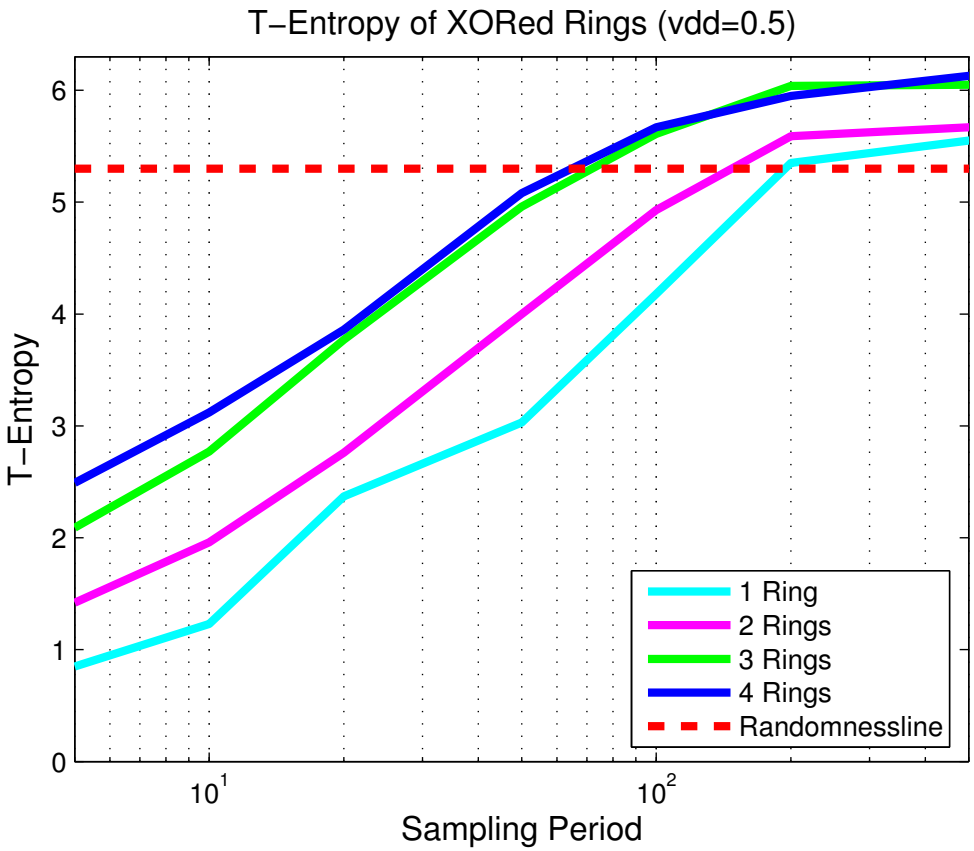

Figure 9.4. T-Entropy values of XORed ROs in weak inversion.

ing in weak inversion. Fig. 9.5 shows the T-entropy values of XORed ROs which are oscillating within 20% frequency difference. The dot on Figure 9.5 indicates the RNG configuration of three XORed ROs which is sampled once every 50 periods. This RNG configuration is chosen for evaluation with NIST 140-2 test suite.

9.2. Discussion

Randomness, energy, and throughput are the three main issues when designing an RNG, although area sometimes be a concern as well. A fair comparison parameter is necessary in order to determine and compare the performance of RNGs. Therefore, we have defined the entropy per energy metric. This metric indicates the amount of energy required for the acquired entropy. In Figure 9.6, entropy per energy versus throughput performance is demonstrated for various RNGs in log - log plot. In the figure, lines with

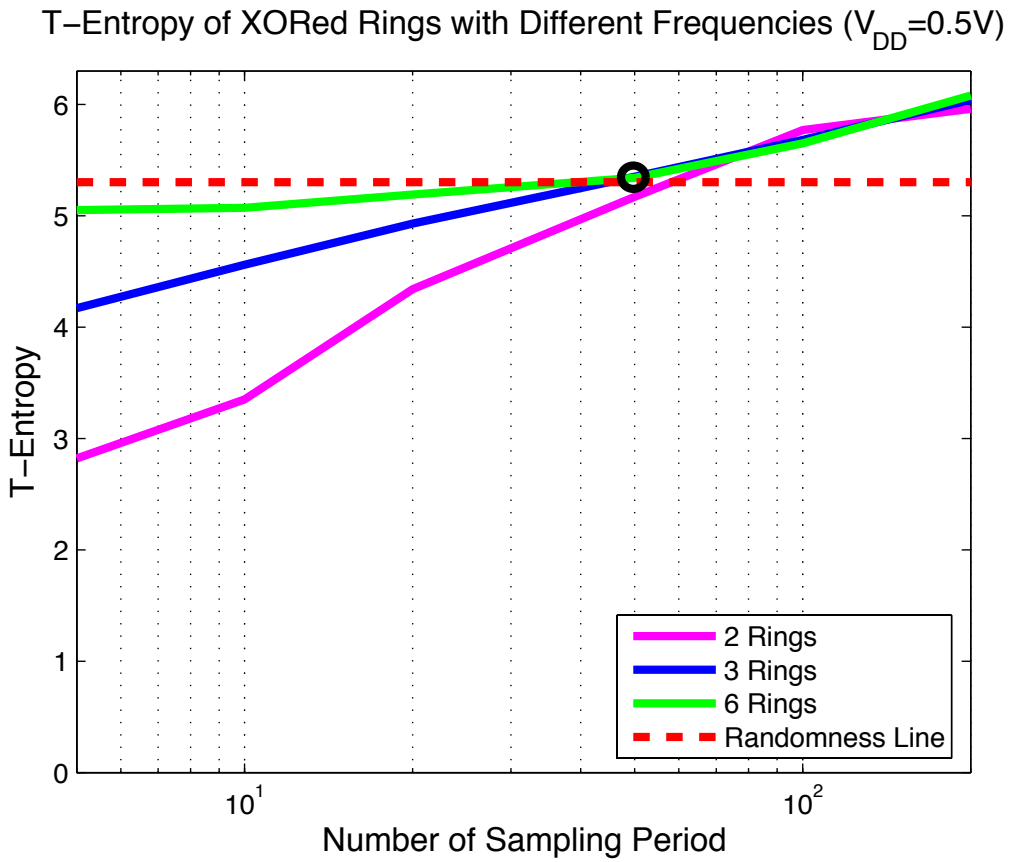

Figure 9.5. T-Entropy values of XORed ROs in weak inversion with scattered frequencies.

dots represent the RNGs operating in weak inversion and lines without dots represent the RNGs operating in strong inversion. All the RNGs are composed of ROs with 23 inverters. The amount of entropy per energy value is higher in RNGs operating in weak inversion than RNGs operating in strong inversion. The RNG which consists of fewer ROs exhibits a higher entropy per energy among the other RNGs operating in weak inversion. On the other hand, RNGs operating in weak inversion are slower than the others. For a given amount of throughput, one can obtain a higher entropy per energy with RNGs operating in weak inversion. When throughput is the main concern, obtained when RNGs operating in strong inversion should be preferred at the cost of high energy.

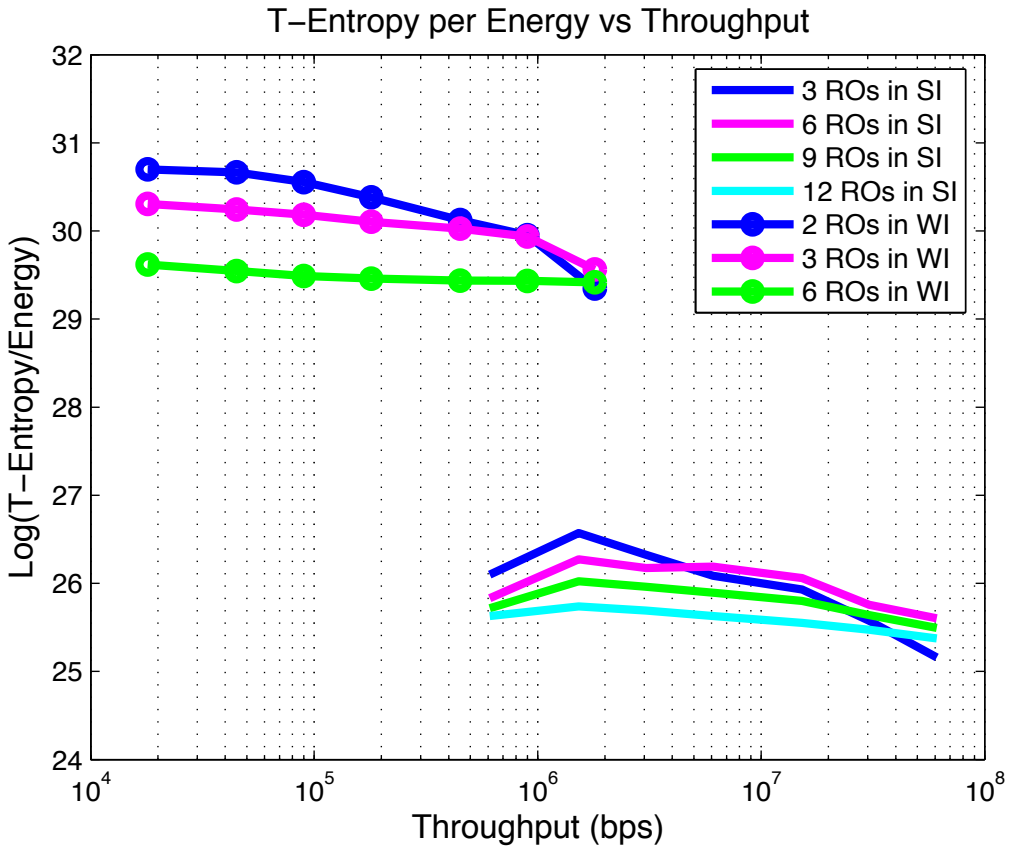

Figure 9.6. T-Entropy per energy versus throughput comparison of RNGs in both region.

9.3. Experiments

$0.25\mu m$ standard CMOS process has been used for design and fabrication of ROs. Photo of the fabricated chip can be seen in Figure 9.7. An FPGA based hardware has been used to upload the binary data to computer. It has a PCI interface and the maximum data storage rate is $2Gbps$. Data acquisition frequency can be set up to $250MHz$. The output amplitude of an RO operating in weak inversion is around 0.55 V. Data acquisition board requires an amplitude around 2.5 V. Therefore, the amplitude level should be amplified. The experiment set-up can be seen in Figure 9.8.

A ring oscillator with 23 inverters will exhibit a frequency around $305MHz$ @2.5 V in strong inversion and $9MHz$ @0.55 V in weak inversion. The output data of RO

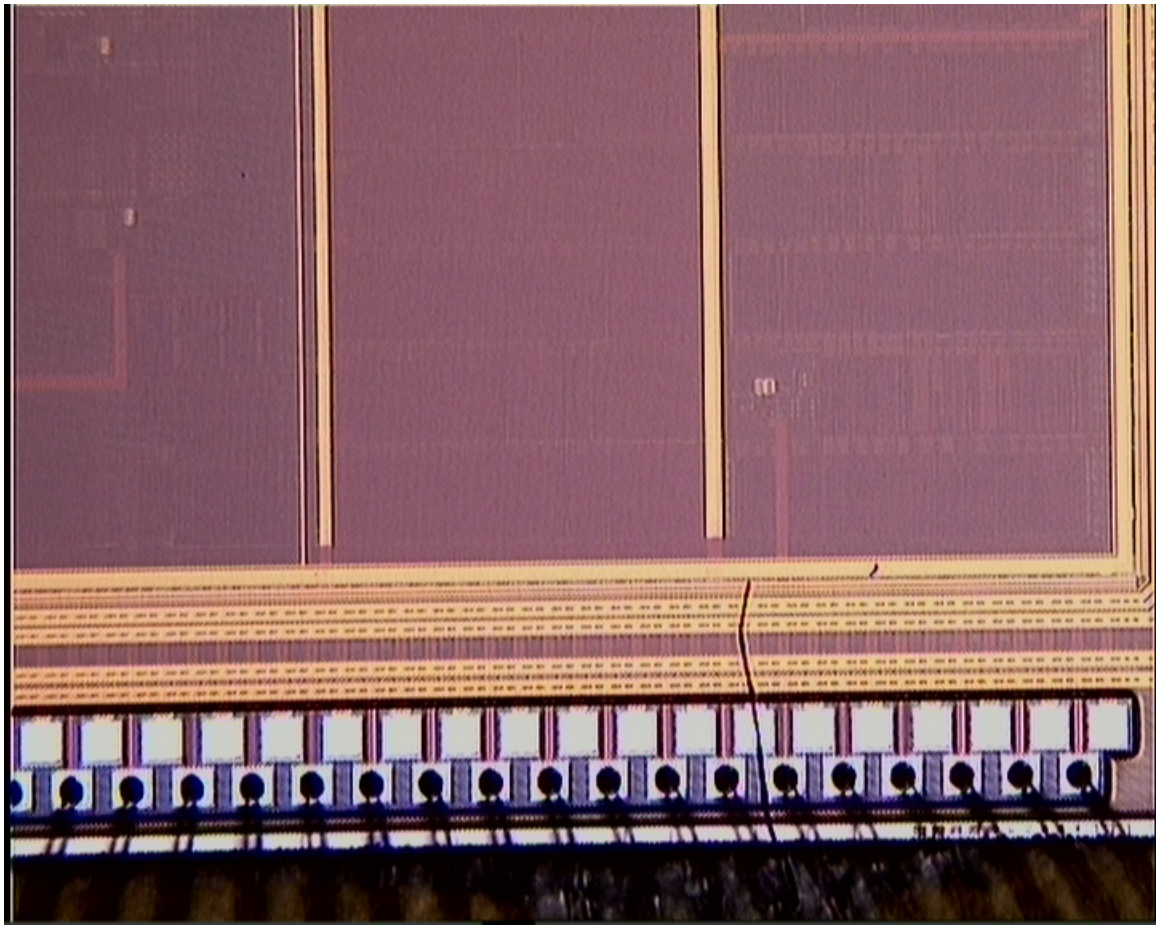

Figure 9.7. Chip photo.

is collected by the acquisition board. A bit stream of length 40 $MBytes$ was acquired through the PCI interface of the FPGA based hardware without any post processing for each acquisition. The obtained bits were processed by the full NIST 140-2 test suite [29].

First of all, the output of 12 XORed ROs operating in strong inversion (sampled once every 100 periods) is subjected to NIST 140-2 test suite. Table 9.1 presents the test results with pass rates and p-values. The proposed RNG with 12 XORed ROs, operating in strong inversion, passed the all test in NIST 140-2 test suite.

Next, the output of three XORed ROs operating in weak inversion (sampled once every 20 periods) is subjected to NIST 140-2 test suite. The test results with pass

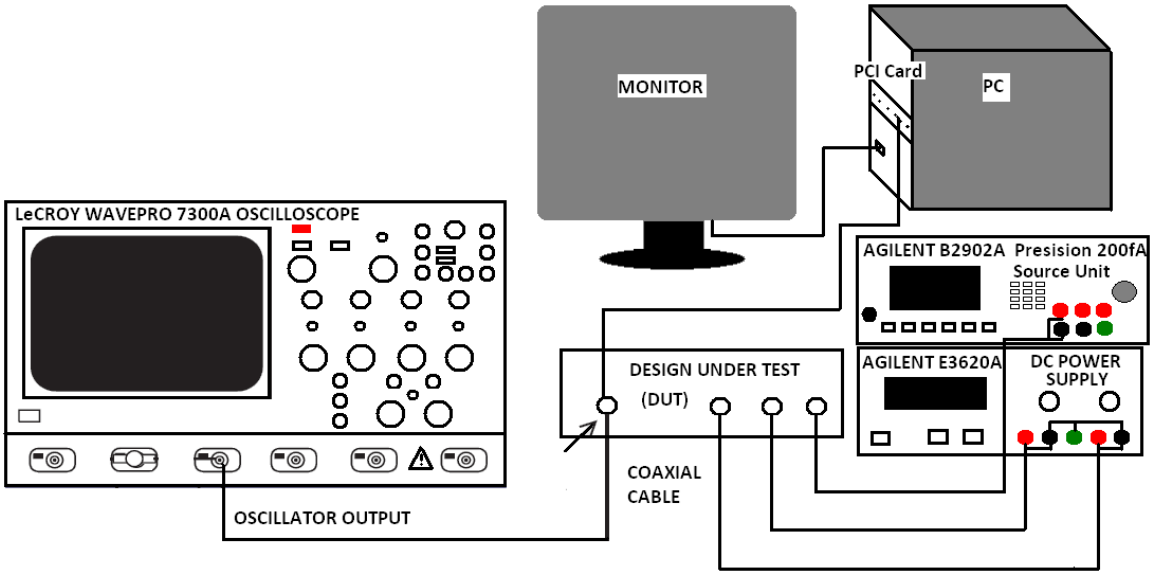

Figure 9.8. Measurement Setup.

rates and p-values are shown in Table 9.2. The proposed RNG with three XORed ROs, operating in weak inversion, passed the all test in NIST 140-2 test suite.

Table 9.1. Statistical test results of RNG with 12 XORed ring oscillators operating in strong inversion.

Statistical Test	**P Values**	**Pass Rates**
Frequency	0.437274	0.9967
Block Frequency	0.746572	0.9833
Cumulative Runs	0.664861	0.9867
Runs	0.630178	0.9900
Longest Run	0.134686	0.9900
Rank	0.822534	0.9933
FFT	0.224821	0.9967
NonPeriodic Template	0.915745	0.9967
Overlapping Templates	0.856907	0.9833
Universal	0.623240	0.9933
Approximate Entropy	0.053059	0.9900
Random Excursions	0.522989	0.9945
Random Excursion Variants	0.925420	0.9945
Serial	0.978072	0.9867
Lempel-Ziv	0.100508	0.9967
Linear Complexity	0.834308	0.9933

Table 9.2. Statistical test results of RNG with three XORed ring oscillators operating in weak inversion.

Statistical Test	**P Values**	**Pass Rates**
Frequency	0.804337	0.9833
Block Frequency	0.314042	0.9900
Cumulative Runs	0.739918	0.9900
Runs	0.077290	0.9833
Longest Run	0.616305	1.0000
Rank	0.534146	0.9933
FFT	0.240914	0.9967
NonPeriodic Template	0.973936	0.9933
Overlapping Templates	0.455937	0.9867
Universal	0.319084	0.9800
Approximate Entropy	0.220931	0.9867
Random Excursions	0.877806	0.9886
Random Excursion Variants	0.981664	0.9886
Serial	0.706149	0.9867
Lempel-Ziv	0.002898	0.9900
Linear Complexity	0.304126	0.9733

Table 9.3. Performance comparison.

Parameters	**RNG in Strong Inversion**	**RNG in Weak Inversion**
RO Oscillating Frequency	302.5 MHz	9 MHz
RO Number in RNG	12	3
Maximum Speed of Throughput	3.02 $Mbps$	0.18 $Mbps$
Supply Voltage	$2.5V$	0.55V
Power Dissipation	12.3 mW	28.09 uW
Energy Dissipation	40.3 pW/Hz	3.09 pW/Hz
Energy Delay Product	0.13 aW/Hz^2	0.345 aW/Hz^2
Typical 1/0 Ratio of Raw Data	1.0000194	0.999594
Area	20,600 $(um)^2$	5,420 $(um)^2$
Fabrication Technology	0.25 μm standard CMOS process	0.25 μm standard CMOS process

Furthermore, Table 9.3 presents a comparison between RNGs operating in weak and strong inversion in terms of power consumption, area, throughput, etc. Both of these RNGs fulfill randomness tests of NIST 140-2 [29]. RNG operating in strong inversion has 16.8 times more throughput; on the other hand, it consumes 437.8 times more power. In order to achieve the same throughput, multiple RNGs operating in weak inversion can be used in parallel. For the same throughput, it will still consume 26 times less power; however, the area overhead will be 1.5 times that of the RNG operating in strong inversion.

An ASIC implementation of [6] was published in [17], which was fabricated with a similar technology used in this thesis. That RNG passed the statistical tests of NIST with 18.5 $Mbps$ throughput after the von Neumann Corrector [93]. Although that RNG has 102 times higher throughput, its power consumption was given as 95 mW, which is 3382 times of that of the RNG operating in weak inversion.

9.4. Conclusion

A T-entropy based entropy analysis has been performed. Based on this analysis, an efficient method is developed to determine the minimum number of ROs to be XORed and the sampling period for RNG outputs to be classified as random. The entropy per energy metric is utilized to compare the performance of RNGs with different structures. The results are also confirmed with measurements. According to the analysis results some RNGs are selected and their outputs are subjected to NIST 140-2 Test Suite. Analysis and measurement test results are coherent.

10. CONCLUSION

Conventionally, it is thought that digital gate based IC RNGs have limited entropy source due to the limited jitter amount in a period. Hence, they were not widely used until recently. The increasing demand to digital platforms has attracted attention to RO-based RNGs. However, entropy of the RO-based RNGs should be increased due to the need for security applications in the digital platforms. This thesis mainly explores methods for increasing randomness in an RO-based RNGs.

In this thesis, three high speed, fully digital, different IC RNG samples are proposed. These designs have been implemented and fabricated with HHNEC's 0.25μm eFlash process with a supply voltage of 2.5V. Two of the implemented RNG circuits are based on ring oscillators which are modeled in [5], one of them is based on Fibonacci and Galois ring oscillators which is modeled in [36].

Existing phase noise and jitter models of ROs for strong inversion region have been reviewed. In order to find the flicker noise induced jitter equation, the missing link between flicker noise induced phase noise and jitter is obtained. After that, flicker noise induced jitter equation is derived. 0.25μm standard CMOS process has been used for design and analysis of ROs. A good match is attained between analyses and measurements. The analysis is extended to weak inversion. Phase noise and jitter equations of CMOS ROs are derived for weak inversion region. The analyses and measurements present a good match. It is observed that CMOS ROs in weak inversion have more phase noise than CMOS ROs operating in strong inversion. The same technology, 0.25μm standard CMOS process has also been used for weak inversion analysis.

In addition, randomness equations of IbRO and DRO are obtained for both strong and weak inversion regions after the derivation of phase noise and jitter models for weak inversion. Analysis shows that both ROs exhibit more randomness with less energy in the weak inversion region. Maximizing the randomness in an RO may cause

a trade-off in oscillation frequency.

Furthermore, the effect of flicker noise on randomness in ROs is investigated in this dissertation. Three different types of synthetic bit streams are produced in MATLAB with the given jitter spectrum which is derived from phase noise measurements. Natural like synthetic bit streams are produced in order to compare the effects of the noise sources individually. These bit streams have flicker, white, and combined noise effects. The T-Entropy based simulations show that flicker noise has a positive effect on entropy and successively on randomness.

Finally, an efficient design method is proposed based on the final interpretations of the analyses explored in this dissertation. Entropy per energy metric is proposed to compare the performance of RNGs with different number of XORed RO combinations. As in flicker noise analysis, T-entropy is used to evaluate randomness. The design parameters of an RO-based RNG, which are the minimum number of ROs to be XORed and the sampling period for RNG outputs to be classified as random, can be determined based on this analysis. According to the analysis results, some RNGs are selected and their outputs are subjected to NIST 140-2 Test Suite. Analysis and measurement test results are shown to be coherent.

10.1. Future Work

Considering this dissertation as a basis, some research extensions can be conducted. Possible research areas are listed as

- During derivation of randomness equations, short-term jitter behavior is considered; therefore, white noise induced jitter equations were used. Since it is demonstrated that flicker noise has an effect on randomness, randomness equations can be derived again by taking flicker noise induced jitter equations into account.

- The effect of temperature variations on randomness behavior of RO-based RNG operating in weak inversion region is an important topic, especially in terms of possible attack scenarios.

- The quality of generated random bit streams is directly related with the age of the RNG. Thus, in order to determine the life of an RNG, aging analysis should be performed. The same analysis should be also done for the RNG operating in weak inversion region.

- Determining the wake-up time is generally a design issue for an oscillator. It is caused by metastability when it starts to oscillate. In [106], an RO-based RNG has been developed by using the wake-up time of ROs. The wake-up time for the proposed RNGs can also be examined.

APPENDIX A: LIST OF PUBLICATIONS

Following journal and conference papers have been published during Ph.D. study.

A.1. JOURNAL PUBLICATIONS

(i) Guler, U., S. Ergun, "A High Speed, Fully Digital IC Random Number Generator", International Journal of Electronics and Communications (AEU), Vol. 44, No. 6, pp. 521-528, 2011.

(ii) Guler, U., G. Dundar, "Modeling CMOS Ring Oscillator Performance as a Randomness Source", IEEE Transactions Circuits and Systems I: Fundamental Theory Application, Vol. 61, No. 3, pp. 712-724, 2014.

(iii) Guler, U., A. E. Pusane and G. Dundar, "Design of Efficient CMOS Ring Oscillator Based Random Number Generator", IEEE Transactions Circuits and Systems I: Fundamental Theory Application, under revision

A.2. CONFERENCE PUBLICATIONS

(i) Guler, U., S. Ergun, "A High Speed, Fully Digital IC Random Number Generator", International Symposium on Circuits and Systems (ISCAS'10), pp. 425-428, 2010.

(ii) Guler, U., S. Ergun and G. Dundar, "A Digital IC Random Number Generator with Logic Gates Only", International Conference on Electronics, Circuits, and Systems (ICECS'10), pp. 239-242, 2010.

(iii) Guler, U., G. Dundar, "Maximizing Randomness in Ring Oscillators for Security Applications", European Conference on Circuit Theory and Design (ECCTD'2011), pp. 118-121, 2011.

(iv) Guler, U., G. Dundar, "Modeling Phase Noise and Jitter in Subthreshold Region and Assessing the Randomness Performance of CMOS Ring Oscillators", International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD'12), pp. 257-260, 2012.

(v) U. Guler, "A New Evaluation Method for RNG", International Common Criteria Conference (ICCC'13), pp. 1-2, 2013.

(vi) Guler, U., A. E. Pusane and G. Dundar, "Investigating Flicker Noise Effect on Randomness of CMOS Ring Oscillator based True Random Number Generator", International Conference on Information Science, Electronics and Electrical Engineering (ISEEE'14), pp. 845-849, 2014.

REFERENCES

1. Abidi, A. A., "Phase Noise and Jitter in CMOS Ring Oscillators", *IEEE Journal of Solid-State Circuits*, Vol. 41, No. 8, pp. 1803–1816, 2006.

2. Petrie, C. and J. Connelly, "A Noise-Based IC Random Number Generator for Applications in Cryptography", *IEEE Transactions Circuits and Systems I: Fundamental Theory Application*, Vol. 47, No. 5, pp. 615–621, 2000.

3. Bucci, M., L. Germani, R. Luzzi, P. Tommasino, A. Trifiletti and M. Varanonuovo, "A High-Speed IC Random-Number Source for SmartCard Microcontrollers", *IEEE Transactions Circuits and Systems I: Fundamental Theory Application*, Vol. 50, No. 11, pp. 1373–1380, 2003.

4. Bucci, M., L. Germani, R. Luzzi, A. Trifiletti and M. Varanonuovo, "A High Speed Oscillator-based Truly Random Number Source for Cryptographic Applications on a SmartCard IC", *IEEE Transactions Computers*, Vol. 52, No. 4, pp. 403–409, 2003.

5. Sunar, B., W. Martin and D. Stinson, "A Provably Secure True Random Number Generator with Built-In Tolerance to Active Attacks", *IEEE Transactions Computers*, Vol. 56, No. 1, pp. 109–119, 2007.

6. Sunar, B., W. Martin and D. Stinson, "A Provably Secure True Random Number Generator with Built-In Tolerance to Active Attacks", *Lecture Notes in Computer Science, Cryptographic Hardware and Embedded Systems (CHES '05)*, pp. 237–249, 2005.

7. Liu, C. and J. A. McNeill, "Jitter in Oscillators with 1/f Noise Sources", *IEEE International Symposium on Circuits and Systems (ISCAS'04)*, pp. 773–776, 2004.

8. Zhou, S., W. Zhang and N. Wu, "An Ultra-low Power CMOS Random Number

Generator", *Solid-State Electronics, Elsevier*, Vol. 52, No. 2, pp. 233–238, 2008.

9. Che, W., H. Deng, X. Tan and J. Wang, *Scheme of Truly Random Number Generator Application in RFID Tag*, White paper, Auto-ID Labs, Fudan, 2006.

10. Francillon, A. and C. Castelluccia, "TinyRNG: A Cryptographic Random Number Generator for Wireless Sensors Network Nodes", *5th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks and Workshops*, pp. 1–7, 2007.

11. Seetharam, D. and S. Rhee, "An Efficient Random Number Generator for Low-Power Sensor Networks", *29th Annual IEEE International Conference on Local Computer Networks*, pp. 560–562, 2004.

12. *MSP430 Ultra-Low-Power Micro-controller*, Datasheet, Texas Instruments, http://www.ti.com/lit/sg/slab034w/slab034w.pdf, 2008.

13. Fujdiak, R., J. Misurec, P. Mlynek and O. Raso, "Random Number Generator in MSP430 x5xx Families", *Elektro Revue*, Vol. 4, No. 4, 2013.

14. *Fast Time to Market with Proven Performance and Stability*, Datasheet, Inside Secure, 2013, http://www.insidesecure.com/content/download/1256/8316/version/1/file/03_FLYER-SecureMicrocontroller_0913_BD.pdf, [Accessed June 2014].

15. Guler, U., "A New Evaluation Method for RNG", *International Common Criteria Conference (ICCC'13)*, pp. 1–2, 2013.

16. Guler, U. and S. Ergun, "A High Speed, Fully Digital IC Random Number Generator", *International Symposium on Circuits and Systems (ISCAS'10)*, pp. 425–428, 2010.

17. Guler, U. and S. Ergun, "A High Speed, Fully Digital IC Random Number Gener-

ator", *International Journal of Electronics and Communications (AEU)*, Vol. 44, No. 6, pp. 521–528, 2011.

18. Guler, U., S. Ergun and G. Dundar, "A Digital IC Random Number Generator with Logic Gates Only", *International Conference on Electronics, Circuits, and Systems (ICECS'10)*, pp. 239–242, 2010.

19. Guler, U. and G. Dundar, "Modeling CMOS Ring Oscillator Performance as a Randomness Source", *IEEE Transactions Circuits and Systems I: Fundamental Theory Application*, Vol. 61, No. 3, pp. 712–724, 2014.

20. Guler, U. and G. Dundar, "Modeling Phase Noise and Jitter in Subthreshold Region and Assessing the Randomness Performance of CMOS Ring Oscillators", *International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD'12)*, pp. 257–260, 2012.

21. Guler, U. and G. Dundar, "Maximizing Randomness in Ring Oscillators for Security Applications", *European Conference on Circuit Theory and Design (ECCTD'2011)*, pp. 118–121, 2011.

22. Guler, U., A. E. Pusane and G. Dundar, "Investigating Flicker Noise Effect on Randomness of CMOS Ring Oscillator based True Random Number Generator", *International Conference on Information Science, Electronics and Electrical Engineering (ISEEE'14)*, pp. 845–849, 2014.

23. Guler, U., A. E. Pusane and G. Dundar, "Design of Efficient CMOS Ring Oscillator Based Random Number Generator", *IEEE Transactions Circuits and Systems I: Fundamental Theory Application*, under revision.

24. Knuth, D. E., *The Art of Computer Programming*, 3rd, Addison-Wesley Professional, Canada, 1998.

25. Zimmermann, P., *PGP Source Code and Internals*, MIT Press, Cambridge MA,

1995.

26. Schneier, B., *Applied Cryptography*, 2nd, John Wiley & Sons, 1996.

27. Davis, D., R. Ihaka and P. Fenstermacher, "Cryptographic Randomness from Air Turbulence in Disk Drives", *Lecture Notes in Computer Science, 14th Annual International Cryptology Conference (CRYPTO'94)*, Vol. 839, pp. 114–120, 1994.

28. Jun, B. and P. Kocher, *The Intel Random Number Generator*, White paper prepared for Intel Corp., Cryptography Research Inc. , 1999, `http://www.cryptography.com/resources/whitepapers/IntelRNG.pdf`, [Accessed June 2014].

29. *Security Requirements for Cryptographic Modules*, Fips pub 140-2, National Institute of Standard and Technology (NIST), Gaithersburg, 2001.

30. *A Statistical Test Suite for Random and Pseudo Random Number Generators for Cryptographic Applications*, 800-22, National Institute of Standard and Technology (NIST), May 2001.

31. Diehard, G. M., *The Marsaglia Random Number CDROM including the Diehard Battery of Tests of Randomness*, Tech. rep., Florida State University, 1995, http://stat.fsu.edu/pub/diehard/, [Accessed June 2014].

32. Bock, H., M. Bucci and R. Luzzi, "An Offset-Compensated Oscillator-Based Random Bit Source for Security Applications", *Cryptographic Hardware and and Embedded Systems (CHES'04)*, pp. 268–281, 2004.

33. Fischer, V., "A Closer Look at Security in Random Number Generators Design", *W. Schindler and S.A. Huss (Eds.): Constructive Side-Channel Analysis and Secure Design (COSADE '12), Lecture Notes in Computer Science (LNCS 7275), Springer-Verlag Berlin Heidelberg*, pp. 167–182, 2012.

34. Bagini, V. and M. Bucci, "A Design of Reliable True Random Number Generator for Cryptographic Applications", *Workshop Cryptographic Hardware and Embedded Systems (CHES'99)*, pp. 204–218, 1999.

35. Dichtl, M. and N. Janssen, "A High Quality Physical Random Number Generator", *Sophia Antipolis Forum Microelectronics (SAME 2000)*, pp. 48–53, 2000.

36. Dichtl, M. and J. Golic, "High Speed True Random Number Generation with Logic Gates Only", *Cryptographic Hardware and and Embedded Systems (CHES 2007)*, pp. 45–62, 2007.

37. Yalcin, M., J. Suykens and J. Vandewalle, "True Random Bit Generation from a Double Scroll Attractor", *IEEE Transactions on Circuits and Systems I*, Vol. 51, No. 7, pp. 1395–1404, 2004.

38. Ergun, S., "Modelling and Analysis of Chaos-Modulated Dual Oscillator-Based Random Number Generators", *EURASIP Europian Signal Processing Conference (EUSIPCO '08)*, 2008.

39. Ergun, S. and S. Ozoguz, "Truly Random Number Generators Based On Non-autonomous Continuous-time Chaos", *International Journal of Circuit Theory and Applications*, Vol. 38, No. 1, pp. 1–24, 2010.

40. Callegari, S., R. Rovatti and G. Setti, "Embeddable ADC-Based True Random Number Generator for Cryptographic Applications Exploiting Nonlinear Signal Processing and Chaos", *IEEE Transactions on Signal Processing*, Vol. 53, No. 2, pp. 793–805, 2005.

41. Stojanovski, T. and L. Kocarev, "Chaos-Based Random Number Generators-Part I: Analysis", *IEEE Transactions Circuits and Systems I: Fundamental Theory Application*, Vol. 48, No. 3, pp. 281–288, 2001.

42. Ergun, S. and S. Ozoguz, "Truly Random Number Generators Based on a Non-

Autonomous Chaotic Oscillator", *International Journal of Electronics and Communications (AEU)*, Vol. 61, No. 4, pp. 235–242, 2007.

43. Vasyltsov, I., E. Hambardzumyan, Y.-S. Kim and B. Karpinskyy, "Fast digital TRNG based on metastable ring oscillator", E. Oswald and E. P. Rohatgi (Editors), *Workshop Cryptographic Hardware and Embedded Systems (CHES '08)*, Vol. 5154, pp. 164–180, Springer, 2008.

44. *Security Requirements for Cryptographic Modules*, Fips pub 140-1, National Institute of Standard and Technology (NIST), Gaithersburg, 2001.

45. Mathew, S. K., S. Srinivasan, M. A. Anders, H. Kaul, S. K. Hsu, F. Sheikh, A. Agarwal, S. Satpathy and R. K. Krishnamurty, "2.4 Gbps, 7 mW All-Digital PVT-Variation Tolerant True Random Number Generator for 45 nm CMOS High-Performance Microprocessors", *IEEE Journal of Solid-State Circuit*, Vol. 47, No. 11, pp. 2807–2821, 2012.

46. *Evaluation Summary: VIA C3 Nehemiah Random Number Generator*, Evaluation summary prepared for via technologies, Cryptography Research, Inc., February 2003.

47. Roover, C. D. and M. Steyaert, "A 500 mV, 650 pW RNG in 130 nm CMOS for a UWB localization system", *European Solid State Circuit Conference (ESSCIRC 2010)*, pp. 278–281, 2010.

48. Golic, J. D., "New methods for digital generation and post processing of random data", *IEEE Transactions Computers*, Vol. 55, No. 10, pp. 1217–1229, 2006.

49. Schellekens, D., B. Preneel and I. Verbauwhede, "FPGA Vendor Agnostic True Random Number Generator", *16th International Conference on Field Programmable Logic and Applications (FPL 2006)*, pp. 1–6, 2006.

50. Sunar, B., *Response to Dichtl's Critism*, Tech. rep., Worchester Polytechnic Insti-

tute, 2008, http://www.crypto.wpi.edu/Research/truerandom.shtml, [Accessed June 1024].

51. Wold, K. and C. Tan, "Analysis and Enhancement of Random Number Generator in FPGA Based Oscillator Rings", *International Conference on ReConfigurable Computing and FPGAs (Reconfig 2008)*, pp. 385–390, 2008.

52. S.Yoo, D. Karakoyunlu, B. Birand and B. Sunar, "Improving the Robustness of Ring Oscillator TRNGs", *ACM Transactions on Reconfigurable Technology and Systems (TRETS)*, Vol. 3, No. 2, pp. 9:1– 9:30, 2010.

53. Markettos, A. T. and S. W. M. D. Stinson, "The Frequency Injection Attack on Ring-Oscillator-Based True Random Number Generators", *Springer, Lecture Notes in Computer Science, Cryptographic Hardware and Embedded Systems (CHES '09)*, pp. 317–331, 2009.

54. Bayon, P., L. Bossuet, A. Aubert, V. Fischer, F. Poucheret, B. Robisson and P. Maurine, "Contactless Electromagnetic Active Attack on Ring Oscillator Based True Random Number Generator", *W. Schindler and S.A. Huss (Eds.): Constructive Side-Channel Analysis and Secure Design (COSADE '12), Lecture Notes in Computer Science (LNCS 7275), Springer-Verlag Berlin Heidelberg*, pp. 151–166, 2012.

55. Bayon, P., L. Bossuet, A. Aubert and V. Fischer, "Electromagnetic Analysis on Ring Oscillator-Based True Random Number Generators", *IEEE International Symposium on Circuits and Systems (ISCAS'13)*, 2013.

56. Fechner, B. and A. Osterloh, "A True Random Number Generator with Built-in Attack Detection", *3rd International Conference on Dependability of Computer Systems (DepCoS'08)*, pp. 111–118, 2008.

57. Valtchanov, B., A. Aubert, F. Bernard and V. Fischer, "Modeling andobserving the jitter in ring oscillators implemented in FPGAs", *Design and Diagnostics of*

Electronic Circuits and Systems (DDECS'08), pp. 1–6, 2008.

58. Fischer, V., F. Bernard, N. Bochard and M. Varchola, "Enhancing Security of Ring Oscillator-based RNG implemented in FPGA", *Field-Programable Logic and Applications (FPL'08)*, pp. 245–250, 2008.

59. Bochard, N., F. Bernard, V. Fischer and B. Valtchanov, "True-randomness and Pseudo-randomness in Ring Oscillator-based True Random Number Generators", *International Journal of Reconfigurable Computing*, Vol. 2010, pp. 1–13, 2010.

60. Sokal, R. R. and F. J. Rohlf, *Biometry: the principles and practice of statistics in biological research*, W.H. Freeman, 1969.

61. Stefanau, N. and S. R. Sonkusale, "High Speed Array of Oscillator-based Truly Binary Random Number Generators", *IEEE International Symposium on Circuits and Systems (ISCAS'04)*, pp. 505–508, 2004.

62. Suh, G. E. and S. Devadas, "Physical Unclonable Functions for Device Authenticationand Secret Key Generation", *Design Automation Conference (DAC'07)*, pp. 9–14, 2007.

63. Maiti, A. and P. Schaumont, "Improving the Quality of a Physical Unclonable Function Using Configurable Ring Oscillators", *International Conference on Field Programmable Logic and Applications, (FPL'09)*, pp. 703–707, 2009.

64. Weigandt, T., *Low-Phase-Noise, Low-Timing-Jitter Design Techniques for Delay Cell Based VCOs and Frequency Synthesizers*, Ph.D. Thesis, UC Berkeley, 2008.

65. Maneatis, J., "Low-Jitter Process-Independent DLL and PLL Based on Self-Biased Techniques", *IEEE Journal of Solid-State Circuits*, Vol. 31, No. 11, pp. 1723–32, 1996.

66. Li, M. P., *Jitter, Noise and Signal Integrity at High Speeds*, 1st, Prentice Hall,

Westford MA, 2008.

67. *A Guide to Understanding and Characterizing Timing Jitter*, White paper, Tektronix, September 2002.

68. *Oscillator Jitter and How to Measure It*, Technical note, Epson, December 1998.

69. *An overview of Oscillator Jitter*, White paper, Statek Corporation, 2006-2007.

70. *Jitter Analysis: A Brief Guide to Jitter*, White paper, Tektronix, 2005.

71. *Clock Jitter and Measurement*, Application note, sit-an10007 rev. 1.0, SiTime, February 2009.

72. Hajimiri, A. and T. H. Lee, "A General Theory of Phase Noise in Electrical Oscillators", *IEEE Journal of Solid-State Circuits*, Vol. 33, No. 2, pp. 179–194, 1998.

73. Hajimiri, A., S. Limotyrakis and T. H. Lee, "Jitter and phase noise in ring oscillators", *IEEE Journal of Solid-State Circuits*, Vol. 34, pp. 790–804, 1999.

74. McNeill, J. A., "Jitter in Ring Oscillators", *IEEE Journal of Solid-State Circuits*, Vol. 32, No. 6, pp. 870–879, 1997.

75. Klumperink, E., S. Gierkink, H. Wallinga and B. Nauta, "Reduction of 1/f Noise in MOSFETs by Switched Bias Techniques", *9th IEEE/ProRISC Workshop on Circuits, Systems and Signal Processing*, Vol. 3, 1998.

76. *Getting Started Manual, WaveMaster 8Zi and 8ZiA Oscilloscopes*, Tech. rep., Teledyne LeCroy, December 2011, http://cdn.teledynelecroy.com/files/manuals/wm8zi_zia_gsm_revc.pdf, [Accessed June 2014].

77. *SDAIII-CompleteLinQ Software*, Operator's manual, Teledyne LeCroy, May

2013.

78. "Making More Accurate Jitter Measurements", *Sixth Annual Wireless Symposium*, 1998.

79. Johnson, J., "The Schottky-effect in low-frequency circuits", *Physical Review*, Vol. 26, pp. 71–85, 1925.

80. Schottky, W., "Über spontane Stromschwankungen in verschiedenen Elektrizitätsleitern", *Ann. der Phys.*, Vol. 57, pp. 541–567, 1918.

81. Schottky, W., "Small-shot effect and flicker effect", *Physical Review*, Vol. 28, pp. 74–103, 1926.

82. Bell, D. A., *Fundamentals and Physical Mechanism*, Van Nostrand, London, 1960.

83. Caloyannides, M. A., "Microcycle spectral estimates of 1/f noise in semiconductors", *Journal of Applied Physics*, Vol. 45, pp. 307–316, 1974.

84. Weigandt, T., K. Beomsup and P. Gray, "Analysis of Timing Jitter in CMOS Ring Oscillators", *IEEE International Symposium on Circuits and Systems (ISCAS 1994)*, Vol. 4, pp. 27–30, 1994.

85. Klimovitch, G., "Near-carrier oscillator spectrum due to flicker and white noise", *International Symposium on Circuits and Systems (ISCAS'00)*, pp. 703–706, 2000.

86. Liu, C. and J. A. McNeill, *Jitter in Oscillators with 1/f Noise Sources and Application to True Random Number Generator for Cryptography*, Ph.D. Thesis, Worchester Polytechnic Institute, 2006.

87. Sarpeshkar, R., T. Delbruck and P. C. A. Mead, "White Noise in MOS Transistors and Resistors", *IEEE Circuits and Devices*, Vol. 9, No. 6, pp. 23–29, 1993.

88. Sansen, W. M. C., *Analog Design Essentials*, 1st, Springer, 2006.

89. Shannon, C. E., "A Mathematical Theory of Communication", *Bell Systems Technical Journal*, Vol. 27, No. 3, pp. 379–423, 1948.

90. Titchener, M. R., U. Speidel and J. Yang, *A Comparison of Practical Information Measures*, CDMTCS Research Report Series, Auckland, May 2005.

91. Speidel, U., M. R. Titchener and J. Yang, "How well do practical information measures estimate the Shannon entropy?", *5th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP'06)*, pp. 861–865, 2006.

92. Cicek, I., A. E. Pusane and G. Dundar, "A Novel Design Method for Discrete Time Chaos Based True Random Number Generators", *Integration, the VLSI Journal*, Vol. 47, No. 1, pp. 38–47, 2014.

93. Neumann, J. V., "Various Techniques Used in Connection With Random Digits", *Applied Math Series.*, Vol. 12, pp. 36–38, 1951.

94. Garrett, P., *Making, Breaking Codes: An Introduction to Cryptology*, 1st, Prentice-Hall, NJ, 2001.

95. *NITROX Security Macro Processor*, Datasheet, Cavium Inc., 2012, http://www.caviumnetworks.com/pdfFiles/NITROX-IPsec-PB-1.3.pdf, [Accessed June 2014].

96. Leeson, D. B., "A simple model of feedback oscillator noises spectrum", *Proceeding of the IEEE*, Vol. 54, No. 2, pp. 329–330, 1966.

97. Rutman, J., "Characterization of phase and frequency instabilities in precision frequency sources: Fifteen years of progress", *Proceeding of the IEEE*, Vol. 66, No. 9, pp. 1048–1175, 1978.

98. Abidi, A. A. and R. G. Meyer, "Noise in relaxation oscillators", *IEEE Journal of Solid-State Circuits*, Vol. SC-18, pp. 794–802, 1983.

99. Razavi, B., "A Study of Phase Noise in CMOS Oscillators", *IEEE Journal of Solid-State Circuits*, Vol. 31, No. 3, pp. 331–343, 1996.

100. Chatfield, C., *The Analysis of Time Series: An Introduction*, 6th, Chapman and Hall, USA, 2003.

101. Demir, A., A. Mehrotra and J. Roychowdhury, "Phase Noise in Oscillators: A Unifying Theory and Numerical Methods for Characterization", *IEEE Transactions Circuits and Systems I: Fundamental Theory Application*, Vol. 47, No. 5, pp. 655–674, 2000.

102. Rabaey, J. M., A. Chandrakasan and B. Nikolic, *Digital Integrated Circuits - A Design Perspective*, 2nd, Prentice Hall, UK, 2003.

103. Fischer, J. H., "Noise Sources and calculation techniques for switched capacitor filters", *IEEE Journal of Solid-State Circuits*, Vol. SSC-17, No. 4, pp. 742–752, 1982.

104. Sepke, T., P. Holloway, C. G. Sodini and H. S. Lee, "Noise Analysis for Comparator-Based Circuits", *IEEE Transactions Circuits and Systems I: Fundamental Theory Application*, Vol. 56, No. 3, pp. 541–553, 2009.

105. Kirchner, J. W., "Aliasing in $\frac{1}{f^{\alpha}}$ noise spectra: Origins, consequences, and remedies", *The American Physical Society*, Vol. 71, pp. 1–19, 2005.

106. Nakura, T., M. Ikeda and K. Asada, "Ring Oscillator Based Random Number Generator Utilizing Wake-up Time Uncertainty", *IEEE Asian Solid-State Circuits Conference*, pp. 121–124, 2009.