

DOCTORAL DISSERTATION

Charge-Sharing Symmetric Adiabatic Logic: Comparative Analysis, Application and LSI Implementation for Cryptographic Hardware Design

March, 2015



Electronics and Information Systems Engineering Division
Graduate School of Engineering
Gifu University
Japan

Câncio Monteiro

**Charge-Sharing Symmetric Adiabatic Logic:
Comparative Analysis, Application and LSI
Implementation for Cryptographic
Hardware Design**

by

Câncio Monteiro

Submitted in partial satisfaction of the requirements for
the degree of Doctor of Philosophy
in Engineering



Electronics and Information Systems Engineering Division
Graduate School of Engineering
Gifu University
Japan

March, 2015

This dissertation has been examined and approved.

Dissertation Director, Dr. *Hiroshi KIMURA*
Professor of Electronics and Information Systems Engineering

Dissertation Supervisor, Dr. *Toshikazu SEKINE*
Associate Professor of Electronics and Information Systems
Engineering

Dissertation Co-Supervisor, Dr. *Takashi NAKAMURA*
Professor of Electronics and Information Systems Engineering

Dissertation Advisor, Dr. *Yasuhiro TAKAHASHI*
Associate Professor of Electronics and Information Systems
Engineering

16th February 2015

Date

“In the Name of Jesus Crist, Nothing is
Impossible”

Dedicate to

my mother,

Lina Monteiro

who brought me to this world, loves me, cares me during my life's
journey. She is the key of my success;

my late father,

Camilio Da Conceicao Monteiro

for his invisible spiritual supports;

all my families,

Monteiro

for their unlimited supports and tirelessness praying for my study;

my wife,

Lily Cipriana

who is always assist me for bringing the dream comes true

and my kids,

Alexander Keihiro & Ramos Kenji

you are my future.

Charge-Sharing Symmetric Adiabatic Logic: Comparative Analysis, Application and LSI Implementation for Cryptographic Hardware Design

Câncio Monteiro

Submitted for the degree of Doctor of Philosophy
in Engineering

March, 2015

Abstract

In this dissertation, the research study on low-power secure logic design using adiabatic logic techniques, and its implementation for cryptographic devices that require low-power, low frequency speed, and high security demands are summarized as follows:

1. The preliminary study was, researching and investigating all novel and well-known existing conventional secure logic styles, analyzing their CMOS cell structures from the point view of internal equivalent RC models, conducting some comparison study on the logic's ability for counteracting power analysis attacks by investigating the instantaneous peak supply current values and transitional energy fluctuation using SPICE simulator.
2. The results of preliminary study revealed that the existing secure logic styles are vulnerable for resistance against side-channel analysis attacks, and extremely power consuming, specifically, the implementation in the low-power and low frequency devices, such as IC-card, RFID tags and secure wireless sensors. Hence, the author has designed, simulated, and investigated the Charge-Sharing Symmetric Adiabatic Logic (CSSAL) circuits. The comparison results of individual logics have shown that the proposed CSSAL circuits exhibits low and uniform peak supply current traces for all possible dual-input transitions, which performs its logic immunity for side-channel attacks.

3. Two LSIs have been implemented and fabricated using 0.18 μm , 1.8-V CMOS process technology, with transistor size of wide (W)/length (L) = 0.6 μm / 0.18 μm for all PMOS and NMOS transistors.
 - (a) The first LSI was a bit-parallel cellular multiplier over $GF(2^4)$, where, in the same chip, the author has implemented two multiplier logic circuits, such as CSSAL multiplier and the conventional three-phase dual-rail pre-charged logic (TDPL) multiplier for the measurement comparison. The full custom layout was designed using cadence virtuoso IC6.1 with the chip size of $172 \times 155 \mu\text{m}^2$ and the global energy dissipation of 14 pJ at 12.5 MHz for the CSSAL multiplier has achieved, while the TDPL has $183 \times 173 \mu\text{m}^2$ of the chip size, and the global energy dissipation is 122.6 pJ, which is nine time higher than the proposed CSSAL multiplier at the post layout simulation level.
 - (b) Second LSI was an 8-bit AES S-box circuit using positive polarity Reed-Muller (PPRM) representation with a composite field technique. In the pre-layout simulation level, the author has carefully investigated the CSSAL and the other conventional dual-rail adiabatic logic styles from the view point of the transitional power fluctuation and the peak current traces in the 8-bit S-box in order to compare their resistance against side-channel attacks. A method to eliminate unwanted glitch current, the author applied triple-power clock signals to each respective inversion block; thus, the CSSAL S-box circuit performs uniform peak current traces and it has significant power reduction compare to the conventional logic style in the same S-box circuit. The full custom layout was designed using the same process technology as the multiplier one, and the chip are is $795 \times 614 \mu\text{m}^2$.
4. The fabricated LSIs were measured to check the input-output functionalities, to verify that whether the measurement results of the output signals are same as the simulation results or not. Moreover, the author has conducted further measurement of the supply current transition for power analysis attacks investigation. On the basis of the simulation and measurement results, the author assures that the proposed CSSAL logic has potential applicability for low power and secure low frequency devices, such as in IC-card, RFID tags and/or secure wireless sensors.

Keywords: Cryptography, Adiabatic logic, Differential Logic, Side-channel Analysis (SCA), Differential Power Analysis (DPA), Advanced Encryption Standard (AES), S-box, Multiplier, Smart card, SPICE simulation.

Declaration

The work in this dissertation is based on research carried out at the Toshikazu Sekine & Yasuhiro Takahashi Laboratory, Electronics and Information Systems Engineering Division, Graduate School of Engineering, Gifu University, Japan. No part of this dissertation has been submitted elsewhere for any other degree or qualification and they are my own work unless referenced to the contrary in the text.

Copyright © 2015 by Cândio Monteiro.

“The copyright of this dissertation rests with the author. No quotations from it should be published without the author’s prior written consent and information derived from it should be acknowledged”.

Acknowledgements

This dissertation could not have been fulfilled if it were not for the understanding and support of the following people:

The author is deeply grateful for a lot of things beyond words alone can express. Above all, praise to Almighty Holy God and then the author is most thankful to his family for their great support.

The author would like to thank his friendly supervisor Associate Prof. Dr. Yasuhiro Takahashi for the patient and intensive supervision, valuable technical opinion and deeply discussion during his meaningful-three-years study, at the Sekine-Takahashi Lab in Gifu University.

The author would like to thank his main supervisor, Associate Prof. Dr. Toshikazu Sekine for the participation and meaningful ideas during his weekly lab seminar, the tirelessness effort and professional discussion with all Lab students which is indirectly encourage the author and as his motivator during his three-years study.

The author thanks all the members of the analog/digital system LSI laboratory (also known as Sekine & Takahashi laboratory) in Gifu University for the daily discussion on the related matter.

Finally, the author would like to express his deep gratitude to those who have contributed in one way or another to the completion of this work. Last but not least, the author would like to thank you, for your interest in his dissertation.

Contents

Abstract	xi
Declaration	xv
Acknowledgements	xvii
1 Introduction	1
1.1 Background	1
1.1.1 Cryptography and Cryptographic Devices	1
1.1.2 Side-Channel Analysis Attacks on Cryptographic Devices	6
1.2 Motivation and Contribution	10
1.3 Organization of the Dissertation	13
2 CMOS Power Consumption, Power Analysis Model and Power Analysis Verification	15
2.1 Power Consumption of CMOS Circuits	16
2.1.1 Dynamic Power Consumption	16
2.1.2 Short-Circuit Power Consumption	19
2.1.3 Static Power Consumption	20
2.1.4 Another Factor of Power Consumption: Glitches	22
2.2 Adiabatic Logic Technique	23
2.3 Power Simulation Model in SCA Attacks	26
2.3.1 Power Model for Attacker	26
2.3.2 Power Model by Designer	28
2.4 Power Simulation Verification	29
2.4.1 Power Verification using Statistical Characteristic	30
2.4.2 Power Verification using Fast Fourier Transform	31
2.5 Summary	35

3	Survey on Side-Channel Information Leakage	37
3.1	Overview	37
3.2	Side-Channel Analysis	37
3.2.1	Timing Attacks	38
3.2.2	Power Analysis Attacks	39
3.2.3	Electromagnetic Analysis Attacks	44
3.3	SCA Countermeasures at Cell Level	45
3.3.1	Hiding	48
3.3.2	Masking	65
3.4	Summary	68
4	Dual-Rail Adiabatic Logic Approach for Secure Logic Implementa- tion	69
4.1	Overview	69
4.2	Conventional Secure Dual-Rail Adiabatic Logic Styles	69
4.2.1	Efficient Charge Recovery Logic	70
4.2.2	2N-2N2P Logic	74
4.2.3	Secure Adiabatic Logic	77
4.2.4	Symmetric Adiabatic Logic	80
4.3	Proposed CSSAL: Charge-Sharing Symmetric Adiabatic Logic	83
4.3.1	CSSAL Inverter	83
4.3.2	Equivalent RC Model Analysis of the CSSAL Inverter Logic	84
4.3.3	CSSAL NAND/AND	89
4.3.4	CSSAL XOR/XNOR	94
4.3.5	CSSAL OR/NOR	96
4.3.6	Further Analysis of CSSAL AND/NAND Logic	97
4.4	Comparison and Discussion	101
4.5	Summary	106
5	Logic Implementation	107
5.1	Introduction	107
5.2	Bit Parallel Cellular Multiplier over $GF(2^m)$	107
5.2.1	Review of AB^2 Multiplier over $GF(2^m)$	108
5.2.2	SPICE Simulation and Results	112
5.2.3	Frequency Spectrum Analysis Result: Multiplier Circuits	130
5.2.4	Analysis and Comparison: Multiplier Circuits	136
5.3	An 8-Bit AES S-Box Circuit using PPRM Representation	144

5.3.1	Overview of Advanced Encryption Standard	144
5.3.2	AES PPRM S-Box Circuit	147
5.3.3	Simulation Result	152
5.3.4	Frequency Spectrum Analysis Result: AES S-box Circuits . .	156
5.3.5	Analysis and Comparison: AES S-box Circuits	157
5.4	Summary	162
6	LSI Implementation	163
6.1	Introduction	163
6.2	Full-Custom LSI Layout Design	163
6.2.1	Individual Logic Layout	165
6.2.2	Multiplier Circuit Layout	177
6.2.3	CSSAL S-box Circuit Layout	182
6.3	Fabricated LSI Measurement	183
6.3.1	Measurement Equipment	183
6.3.2	Multiplier Measurement Results	188
6.3.3	CSSAL S-box Measurement Results	194
6.4	Summary	202
7	Conclusions and Future Works	203
7.1	Conclusions	203
7.2	Future Research Direction	204
	Bibliography	205
	List of publications	217
A	Simulation Result using 90 nm CMOS Process	221
A.1	Multiplier over $GF(2^4)$	221
A.2	An 8-bit AES S-box Circuit	225
B	PPRM Representation of AES S-Box Internal Logic Circuit	227
B.1	Stage 1 of PPRM S-Box	227
B.1.1	Combinational Logic Version 1 (ver.1)	227
B.1.2	Logic Sharing Technique (Combinational Logic ver.2)	228
B.2	Stage 2 of PPRM S-Box	229
B.3	Stage 3 of PPRM S-Box	229

C Trapezoidal Power Clock Generator	239
---------------------------------------	-----

List of Figures

1.1	Traditional cryptographic devices [3]– [5].	2
1.2	Modern cryptographic devices [6]– [9].	3
1.3	General assumption of cryptographic system and cryptanalysis. . . .	6
1.4	Overview of cryptanalysis.	7
1.5	Existing CMOS logic problems and our proposed solution for SCA countermeasures. This figure illustrates transitional current traces of (a) scCMOS, (b) DR-CMOS, (c) Target supply current trace of the proposed CSSAL.	12
2.1	Total power in a CMOS inverter: The dynamic power, short-circuit power and leakage power.	16
2.2	(a) Conventional CMOS inverter, (b) A CMOS pull up network (PUN) RC equivalent model for charging phase, (c) A CMOS pull down network (PDN) RC equivalent model for discharging phase.	17
2.3	Simulation result of the short-circuit and dynamic power consumptions of a static CMOS inverter.	20
2.4	Components of leakage power in CMOS (adopted from [38]).	21
2.5	(a) An adiabatic logic PUN equivalent RC model for charging phase, (b) An adiabatic PDN equivalent RC model for discharging phase. . .	24
2.6	Comparison of the supply current transitions for the equivalent RC models of the (a) CMOS logic step voltage and (b) Adiabatic logic ramped step voltage. (c) The peak supply current of the adiabatic logic is significantly lower than that of the conventional CMOS logic under the same parameters and conditions.	25
2.7	FFT schematic model.	33
2.8	LTspice transient analysis.	33

3.1	DPA traces: (a). One correct and two incorrect with power reference; (b). Quantitative DPA measurements (adopted from original article [47]).	42
3.2	Basic digital logics and their respective truth table.	47
3.3	Existing CMOS logic problems; (a) scCMOS inverter and its equivalent RC model, (b) DCVSL inverter and its equivalent RC model, (c) Current signature based in the input transitions	51
3.4	A 2-input SR cell and a corresponding 2-input DR cell.	51
3.5	SR scCMOS for AND and XOR transistor schematics and their respective LTspice simulation results.	52
3.6	DR DCVSL AND and XOR transistor schematics and their respective LTspice simulation results.	53
3.7	Generic dynamic logic (left) and its phases (right).	54
3.8	Dynamic logic: (a) NAND transistors schematic, (b) NAND input-output signals, (c) XOR transistors schematic, (d) XOR input-output signals.	55
3.9	Transistor schematic of generic SABL cells and its ideal input-output signals.	58
3.10	(a) Universal DR PDN (<i>i.e.</i> , used in DCVSL AND/NAND gate) and (b) Special DPDN of the SABL AND/NAND gate.	59
3.11	SABL: (a) NAND/AND transistors schematic, (b) NAND/AND input-output signals, (c) XNOR/XOR transistors schematic, (d) XNOR/XOR input-output signals.	60
3.12	Generic TDPL logic cells and its input-output timing graphs.	62
3.13	TDPL: (a) NAND/AND transistors schematic, (b) NAND/AND input-output signals, (c) XNOR/XOR transistors schematic, (d) XNOR/XOR input-output signals.	63
3.14	Supply current traces of 16-possible dual-input transitions (see Table 4.1) of individual gates investigates: scCMOS, DCVSL, Dynamic logic, SABL, and the TDPL, respectively. These results were achieved by setting 10-fF of the nominal capacitance at the output cells (between output nodes and the GND), and the horizontal time function has been enlarged for better readability of the peak current differences.	64
3.15	A dual-input unmasked cell and a corresponding dual-input masked cell.	66

3.16	MDPL: (a) Single-rail majority (SR MAJ) gate transistor schematic, (b) Cell schematic of MDPL NAND/AND gate, and (c) Input-output signals of the MDPL NAND/AND gate.	67
4.1	ECRL inverter: (a) Transistor schematic, (b) Timing chart.	71
4.2	ECRL: (a) Transistor schematic of AND/NAND circuit, (b) Input-output signals.	72
4.3	ECRL: (a) Transistor schematic of XOR/XNOR circuit, (b) Input-output signals.	73
4.4	2N-2N2P Inverter; (a) Generic logic structure, (b) Timing diagram. .	74
4.5	2N2N2P: (a) Transistor schematic of AND/NAND circuit, (b) Input-output signals.	75
4.6	2N2N2P: (a) Transistor schematic of XOR/XNOR circuit, (b) Input-output signals.	76
4.7	SAL; (a) Generic logic structure, (b) Timing diagram.	77
4.8	SAL: (a) Transistor schematic of AND/NAND circuit, (b) Input-output signals.	78
4.9	SAL: (a) Transistor schematic of XOR/XNOR circuit, (b) Input-output signals.	79
4.10	SyAL: (a) Inverter logic structure, (b) Timing diagram.	80
4.11	SyAL: (a) Transistor schematic of AND/NAND circuit, (b) Input-output signals.	81
4.12	SyAL: (a) Transistor schematic of XOR/XNOR circuit, (b) Input-output signals.	82
4.13	Proposed CSSAL logic; (a) Inverter logic structure, (b) Input and output signals of the proposed CSSAL inverter logic.	84
4.14	2N-2N2P: (a) Inverter logic structure and its input signals, (b) Internal RC model at each respective phases, and (c) Occurrences of the RC model at four-possible input transitions.	86
4.15	SyAL: (a) Inverter logic structure and its input signals, (b) Internal RC model at each respective phases, and (c) Occurrences of the RC model at four-possible input transitions.	87
4.16	Proposed CSSAL: (a) Inverter logic structure and its input signals, (b) Internal RC model at each respective phases, and (c) Occurrences of the RC model at four-possible input transitions.	87

4.17 Comparison of the supply peak current transition of inverter logic styles.	88
4.18 Proposed CSSAL vs. SyAL NAND/AND logic operation using RC model.	91
4.19 Symmetric vs. Asymmetric NAND/AND PDN topology for charging and discharging operation.	92
4.20 CSSAL: (a) Transistor schematic of AND/NAND with C_x pass-transistors, (b) Input-output signals with C_x pass-transistors, (c) Input-output signals without C_x pass-transistors.	93
4.21 Symmetric vs. Asymmetric XOR/XNOR PDN topology for charging and discharging operation.	94
4.22 CSSAL: (a) Transistor schematic of XOR/XNOR with C_x pass-transistors, (b) Input-output signals with C_x pass-transistors, (c) Input-output signals without C_x pass-transistors.	95
4.23 CSSAL: (a) Transistor schematic of OR/NOR with C_x pass-transistors, (b) Input-output signals with C_x pass-transistors, (c) Input-output signals without C_x pass-transistors.	96
4.24 CSSAL AND/NAND: (a) CSSAL ver.1: Transistor schematic with C_x pass-transistors, (b) CSSAL ver.2: Transistor schematic without C_x pass-transistors, (c) CSSAL ver.3: Transistor schematic without MP1 and MN18 (Eval cell) transistors, and (d) CSSAL ver.4: Transistor schematic without MP1, MN18 and the C_x pass-transistors.	98
4.25 Supply current traces of all CSSAL versions for AND/NAND and XOR/XNOR logic circuits @ 12.5 MHz.	99
4.26 CSSAL critical path verification: (a) 10-NAND/AND chains logic diagram, (b) Output signals using CSSAL ver.1, (c) CSSAL ver.1–ver.4 dynamic hazard signals.	100
4.27 All possible dual-input 16 transitions of investigated individual logics at 12.5 MHz for AND/NAND gate (left) and XOR/XNOR gate (right).102	
4.28 Supply current transitions of individual logics at different frequencies. (a) 1.25 MHz AND gate, (b) 1.25 MHz XOR gate, (c) 125 MHz AND gate. (d) 125MHz XOR gate.	103
5.1 Circuit diagram of the bit-parallel multiplier over $GF(2^4)$: (a) Inner cell using A-cell circuit and (b) Inner cell using B, C-cells circuit. . .	111

5.2	Inner cell circuits of Multiplier over $GF(2^4)$: (a) A-cell circuit, (b) B-cell circuit and (c) C-cell circuit.	112
5.3	4-bit input signals of the CSSAL multiplier over $GF(2^4)$	115
5.4	4-bit input signals of the SyAL multiplier over $GF(2^4)$	115
5.5	4-bit input signals of the SAL multiplier over $GF(2^4)$	116
5.6	4-bit input signals of the TDPL multiplier over $GF(2^4)$	116
5.7	Investigation of A-cell and B-cell supply current transitions. CSSAL individual logic without C_x pass-transistors.	117
5.8	Input signals connection of inner A-cell in bit-parallel multiplier over $GF(2^4)$: (a) Logic diagram of the first column, and (b) Block diagram.	118
5.9	Input signals connection of inner B, C-cells in bit-parallel multiplier over $GF(2^4)$: (a) Logic diagram of the first column, and (b) Block diagram.	119
5.10	Output signals of CSSAL bit-parallel multiplier over $GF(2^4)$ using A-Cell with C_x pass-transistors.	120
5.11	Output signals of CSSAL bit-parallel multiplier over $GF(2^4)$ using A-Cell without C_x pass-transistors.	121
5.12	Output signals of CSSAL bit-parallel multiplier over $GF(2^4)$ using B, C-Cell with C_x pass-transistors.	122
5.13	Output signals of CSSAL bit-parallel multiplier over $GF(2^4)$ using B, C-Cell without C_x pass-transistors.	123
5.14	Output signals of SyAL bit-parallel multiplier over $GF(2^4)$	124
5.15	Output signals of 2N-2N2P bit-parallel multiplier over $GF(2^4)$	125
5.16	Output signals of SAL bit-parallel multiplier over $GF(2^4)$	126
5.17	Output signals of SABL bit-parallel multiplier over $GF(2^4)$	127
5.18	Output signals of TDPL bit-parallel multiplier over $GF(2^4)$	128
5.19	Signal spectrum of the CSSAL multiplier @ 1.25 MHz: (a) Vertical y -axis decibel and horizontal x -axis logarithmic, (b) Both horizontal and vertical axis are set to linear.	131
5.20	Signal spectrum of the SyAL multiplier @ 1.25 MHz: (a) Vertical y -axis decibel and horizontal x -axis logarithmic, (b) Both horizontal and vertical axis are set to linear.	132
5.21	Signal spectrum of the SAL multiplier @ 1.25 MHz: (a) Vertical y -axis decibel and horizontal x -axis logarithmic, (b) Both horizontal and vertical axis are set to linear.	133

5.22	Signal spectrum of the 2N-2N2P multiplier @ 1.25 MHz: (a) Vertical y -axis decibel and horizontal x -axis logarithmic, (b) Both horizontal and vertical axis are set to linear.	134
5.23	Signal spectrum of the TDPL multiplier @ 1.25 MHz: (a) Vertical y -axis decibel and horizontal x -axis logarithmic, (b) Both horizontal and vertical axis are set to linear.	135
5.24	Instantaneous supply current transition of 8 input patterns representation of each adiabatic multiplier circuits; (a) CSSAL Multiplier; (b) SyAL Multiplier; (c) SAL Multiplier; (d) 2N-2N2P Multiplier.	137
5.25	NED occurrences of the CSSAL and SyAL multipliers using Monte Carlo simulation for process variation verification.	140
5.26	NSD occurrences of the CSSAL and SyAL multipliers using Monte Carlo simulation for process variation verification.	140
5.27	NED occurrences of the 2N-2N2P and SAL multipliers using Monte Carlo simulation for process variation verification.	141
5.28	NSD occurrences of the 2N-2N2P and SAL multipliers using Monte Carlo simulation for process variation verification.	142
5.29	Energy dissipation comparison of the bit-parallel multiplier over $GF(2^4)$ using different logic styles.	143
5.30	Encryption process of AES algorithm (adopted from original article [101]).	145
5.31	(a) Conventional composite field AES S-box architecture; (b) multi-stage PPRM representation with the implementation of the proposed triple V_{pcs} signals in the CSSAL 8-bit S-box circuit.	149
5.32	Triple power clock signals for CSSAL S-box circuit shown in Fig. 5.31(b).	149
5.33	Dynamic hazard voltage removal using triple power clock supplies. . .	151
5.34	Simulation result of single power clock signals versus triple power clock signal.	151
5.35	8-bit input signals of the CSSAL S-box. Input X0–X7 also supplied to SyAL and 2N-2N2P S-boxes.	152
5.36	Output voltage of the proposed CSSAL S-box at 12.5 MHz power clock frequency.	153
5.37	Output voltage of the 2N-2N2P S-box at 12.5 MHz power clock frequency.	154
5.38	Output voltage of the SyAL S-box at 12.5 MHz power clock frequency.	155
5.39	Signal spectrum of harmonic frequency of the AES S-box circuits. . .	156

5.40	Comparison of peak supply current traces of the multi-stage PPRM 8-bit S-box circuit at an operating frequency of 12.5 MHz (at the worst case of the current-to-data dependency) using different logic styles; (a) S-box circuit using CSSAL, (b) S-box circuit using SyAL, (c) S-box circuit using 2N-2N2P and (d) S-box circuit using Morioka's circuit.	158
5.41	Observed amount of energy consumed per input transition: proposed CSSAL, SyAL, 2N-2N2P Morioka's circuit [102] implementation in the 8-bit S-box for 256 energy data sample.	160
5.42	Simulation and calculation results of NSD using 0.18- μ m CMOS process variation for five samples of each circuit.	160
5.43	Simulated energy dissipation comparison of all the investigated adiabatic logics: CSSAL, SyAL, 2N-2N2P, and Morioka's circuit [102] in the multi-stage PPRM 8-bit S-box circuit at each operating frequency ranges.	161
6.1	Full-custom LSI design flow.	166
6.2	Layout of the CSSAL inverter/buffer of the circuit schematic diagram in Fig. 4.16(a).	167
6.3	Layout of the CSSAL ver.2 circuit; (a) NAND/AND, (b) XNOR/XOR.	167
6.4	Layout of the CSSAL ver.4 circuit; (a) NAND/AND, (b) XNOR/XOR.	168
6.5	Layout of the TDPL circuit; (a) NAND/AND, (b) XNOR/XOR.	168
6.6	Layout of the 2N-2N2P circuit; (a) NAND/AND, (b) XNOR/XOR.	169
6.7	Layout of the SyAL circuit; (a) NAND/AND, (b) XNOR/XOR.	169
6.8	Layout digram of 10-stages of NAND/AND chain using fundamental logic circuits as follows: (a) CSSAL ver.2, (b) CSSAL ver.4 and (c) SyAL.	170
6.9	Supply current traces of the CSSAL ver.2 : Post-layout simulation result of the 10-stages NAND/AND chain using 10-V _{pc} Power supplies with normal logic operation at 12.5 MHz.	171
6.10	Supply current traces of the CSSAL ver.4 : Post-layout simulation result of the 10-stages NAND/AND chain using 10-V _{pc} Power supplies with normal logic operation at 12.5 MHz.	172
6.11	Supply current traces of the SyAL : Post-layout simulation result of the 10-stages NAND/AND chain using 10-V _{pc} Power supplies with normal logic operation at 12.5 MHz.	173

6.12	CSSAL ver.2: 16-possible supply current transition from post-layout simulation of the 10-stages NAND/AND chain using 10-Vpc Power supplies. The results indicate that the possible stage of the CSSAL ver.1 is 4 or 5 stages of AND chain.	174
6.13	CSSAL ver.4: 16-possible supply current transition from post-layout simulation of the 10-stages NAND/AND chain using 10-Vpc Power supplies. The results indicate that the possible stage of the CSSAL ver.1 is 6 or 7 stages of AND chain.	175
6.14	SyAL: 16-possible supply current transition from post-layout simulation of the 10-stages NAND/AND chain using 10-Vpc Power supplies. The results indicate that the possible stage of the SyAL ver.1 is 6 or 7 stages of AND chain.	176
6.15	Layout of the multiplier circuits; (a) 2N-2N2P multiplier, (b) SyAL multiplier (c) CSSAL multiplier and (d) TDPL multiplier.	179
6.16	The comparison of peak supply current of the CSSAL, SyAL, 2N-2N2P and the TDPL multiplier when complementary dual output states are shifted for charging and discharging process. The proposed CSSAL has low and same peak current during charging and discharging processes, while the SyAL, 2N-2N2P have low and different peak current, and the TDPL has very high peak current traces.	181
6.17	Post-layout energy dissipation comparison of the bit-parallel cellular multiplier over $GF(2^4)$ in respect to the different input clock frequency.	181
6.18	Layout of the CSSAL 8-bit AES S-box circuit.	182
6.19	Multiplier over $GF(2^4)$: (a) LSI frame from the layout view, (b) Fabricated LSIs photomicrograph.	184
6.20	8-bit AES S-box: (a) LSI frame from the layout view, (b) Fabricated LSIs photomicrograph.	185
6.21	Capturing the LSI photomicrograph using Leica Light microscopes in the division of instrumental analysis, Gifu University.	186
6.22	(a) LSI image and (b) Unit under test.	187
6.23	Measurement photo including the used equipment.	187
6.24	Measurement diagram.	189
6.25	Input-output measurement signals of the TDPL bit-parallel cellular multiplier over $GF(2^4)$ at 1.25 MHz power clock frequency. Vertical scale: 2 V/div. Horizontal scale: 1 μ s/div.	190

6.26	Input-output measurement signals of the CSSAL bit-parallel cellular multiplier over $GF(2^4)$ at 1.25 MHz power clock frequency. Vertical scale: 2 V/div. Horizontal scale: 1 μ s/div.	191
6.27	Supply current traces of (a) CSSAL multiplier, (b) TDPL multiplier. Measurement is conducted as: (1) Transition of $In1 = 0 \rightarrow 1, 1 \rightarrow 0$ when $In2 = 0 \rightarrow 0$; (2) Transition of $In1 = 0 \rightarrow 1, 1 \rightarrow 0$ when $In2 = 1 \rightarrow 1$; (3) Transition of $In2 = 0 \rightarrow 1, 1 \rightarrow 0$ when $In1 = 0 \rightarrow 0$; (4) Transition of $In2 = 0 \rightarrow 1, 1 \rightarrow 0$ when $In1 = 1 \rightarrow 1$. Vertical scale: 100 mV/div. Horizontal scale: 1 μ s/div.	192
6.28	Dual-inputs (InA, InB) of the CSSAL bit-parallel cellular multiplier over $GF(2^4)$ at 1.25 MHz power clock frequency. Output signals show the correct AND/NAND logic function.	193
6.29	Input measurement signals of the CSSAL S-box LSI at 125 kHz power clock frequency. Vertical scale: 2 V/div. Horizontal scale: 5 μ s/div.	195
6.30	Output signals of internal wires a0–a3 of the CSSAL S-box LSI at 125 kHz power clock frequency. Top: Simulation signals. Bottom: Measurement signals with vertical scale: 2 V/div and horizontal scale: 10 μ s/div.	196
6.31	Output signals of internal wires b0–b3 of the CSSAL S-box LSI at 125 kHz power clock frequency. Top: Simulation signals. Bottom: Measurement signals with vertical scale: 2 V/div and horizontal scale: 10 μ s/div.	197
6.32	Output signals of internal wires c0–c3 of the CSSAL S-box LSI at 125 kHz power clock frequency. Top: Simulation signals. Bottom: Measurement signals with vertical scale: 2 V/div and horizontal scale: 10 μ s/div.	198
6.33	Output signals of internal wires d0–d3 of the CSSAL S-box LSI at 125 kHz power clock frequency. Top: Simulation signals. Bottom: Measurement signals with vertical scale: 2 V/div and horizontal scale: 10 μ s/div.	199
6.34	Output signals of Y0–Y7 of the CSSAL S-box LSI at 125 kHz power clock frequency. Top: Simulation signals. Bottom: Measurement signals with vertical scale: 2 V/div and horizontal scale: 10 μ s/div.	200
6.35	Supply current traces of the $I_{Vpc0}, I_{Vpc1}, I_{Vpc2}$ at 125 kHz power clock frequency: (a) Simulation and (b) Measurement results with input signal in Fig. 6.29.	201

A.1	Instantaneous supply current transition of 8 input patterns representation of each adiabatic multiplier circuits; (a) CSSAL Multiplier; (b) SyAL Multiplier; (c) 2N-2N2P Multiplier.	223
A.2	NSD occurrences of the CSSAL and SyAL multipliers using Monte Carlo simulation for process variation verification: (a) CSSAL vs SyAL and (b) 2N-2N2P.	224
A.3	Simulation and calculation results of NSD using 90 nm CMOS Process.	225
A.4	Simulated energy dissipation comparison of all the investigated adiabatic logics: CSSAL, SyAL, 2N-2N2P, and Morioka's circuit [102] implementation in the multi-stage PPRM 8-bit S-box circuit at each operating frequency ranges.	226
B.1	Combinational logic of internal wires of a_0 – a_3 and b_0 – b_3 in appendix of [102].	231
B.2	Optimized combinational logic of internal wires of a_0 – a_3 and b_0 – b_3 using logic sharing technique in this work.	232
B.3	Combinational logic of internal wires of c_0 – c_3 in appendix of [102]. . .	233
B.4	Optimized combinational logic of internal wires of c_0 – c_3 using logic sharing technique in this work.	234
B.5	(a) Internal wires of d_0 – d_3 in [102] and (b) Optimized combinational logic of internal wires of d_0 – d_3 using logic sharing technique.	235
B.6	Combinational logic of output y_0 – y_3	236
B.7	Combinational logic of output y_4 – y_7	237
C.1	Schematic diagram of the three-phase trapezoidal power clock generator circuit.	240
C.2	Input-output signals of the three-phase trapezoidal power clock generator.	241

List of Tables

2.1	Output transition of a CMOS inverter cell and the corresponding power consumption.	22
3.1	Input transition of NAND gate and its complementary that can be performed during one complete cycle.	50
3.2	Truth table of an MDPL NAND/AND cell for complementary input values.	66
4.1	Charging activity in asymmetric and symmetric DR PDN internal node capacitances (see Fig. 4.19). Symmetric PDN has balance charges than that of the asymmetric one.	92
4.2	Simulation and calculation results of AND/AND and XOR/XNOR individual logics.	104
4.3	Number of gates and the cyclical energy comparison of AND and XOR logic styles	105
5.1	Simulation and calculation results for A-cell and B, C-cells into bit-parallel cellular multiplier over $GF(2^4)$	129
5.2	FFT parameters and calculation results of the CSSAL, 2N-2N2P, SyAL, and TDPL multipliers circuits @ 1.25 MHz input clock frequency.	130
5.3	Simulation and calculation results of the bit-parallel cellular multiplier over $GF(2^4)$ at different input clock frequencies.	139
5.4	AES standard key-block-round combination	147
5.5	Gate size, transistor counts, layout area and delay of an 8-bit S-box circuit (0.18- μ m 1.8-V CMOS standard cell @ 12.5 MHz)	148

5.6	Simulation and calculation results of 8-bit S-box circuit in PPRM representation using proposed CSSAL, SyAL, 2N-2N2P, and Morioka's circuit [102] respectively at 1.25 MHz, 12.5 MHz, and 50 MHz input power clock frequencies	159
6.1	Comparison of gate numbers and chip area of all investigating logic styles.	177
6.2	Simulation and calculation results of the bit-parallel cellular multiplier over $GF(2^4)$ at 1.25MHz, 12.5MHz, 50MHz power clock frequency for the CSSAL, SyAL, 2N-2N2P, and the TDPL multiplier, respectively.	180
6.3	Proposed CSSAL multiplier over $GF(2^4)$ and the 8-bit AES S-box chip features summary	202
A.1	Simulation parameters	222
C.1	Parameters for elements in trapezoidal power clock generator circuit.	239

Chapter 1

Introduction

In this chapter, the background and the motivation of this dissertation are described. The major focus of this research work is on secure logic design for cryptographic hardware design and implementation. Therefore, our preliminary understanding on the basic knowledge of the cryptographic process, and the presence of cryptanalysis to break the confidentiality, the integrity and the authenticity of secure information on cryptosystem are required. Thereafter, the author envisages the motivations of this thesis work based on the identified problems in this research field. Moreover, in the end of this chapter, the flowing of this dissertation writing methodology will also be narrated for readers' better understanding of the whole history of this 3-years Ph.D. research results.

1.1 Background

1.1.1 Cryptography and Cryptographic Devices

Cryptographic devices are becoming ubiquitous computing devices to make the life easier, faster and safest in the modern era. The innovative cryptographic devices are moving beyond the traditional environments, such as e-government and e-banking systems, we see cryptographic techniques realized in web browsers, e-mail programs, cell phones, manufacturing systems, embedded software, smart buildings, cars, and even medical implant.

The etymology of cryptography is from the *Greek* words *kryptó* (κρύπτω), meaning “*hidden, secret*”, and *gráfo* (γράφω), meaning “*to write*” [1]. Therefore, the general meaning of the cryptology is the science of securely data communication with the presence of third party or known as cryptanalysis.

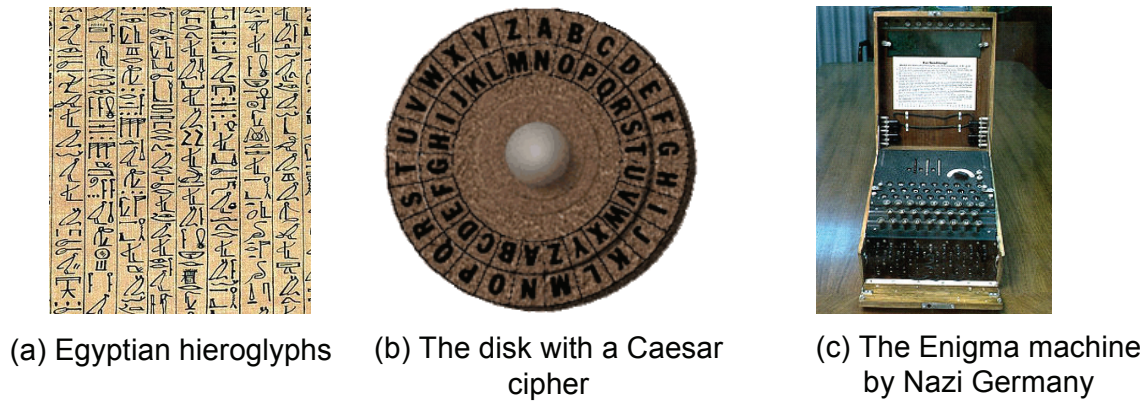


Figure 1.1: Traditional cryptographic devices [3]– [5].

The science of cryptology was been practicing in the old business and during the conflicts between nations, dating back in about 2000 BC [1], [2]. Some historical devices have been collected from internet resources, such as depicted in Fig. 1.1. Figure 1.1(a) depicts the Egyptian practices of hieroglyphics¹ in a non-standard fashion, presumably to hide the meaning from those who did not know the meaning. Another historical cipher, the shift cipher or known as Caesar cipher in ancient Roman’s cryptographic device, as shown in Fig. 1.1(b). This cipher has special case of the substitution cipher and has a very elegant mathematical description. Operation of the cipher in Fig. 1.1(b) is that, if we shift by 3 positions, then the letter A would be substitute by d, and B by e, and etc. Allegedly, Julius Caesar used this cipher with a three-position shift, which arises possible attack, because the adversary may guess the key to find the plaintext. Moreover, during the World War II, the Enigma machine shown in Fig. 1.1(c), which related to electro-mechanical rotor cipher machine which was used by Nazi Germany for enciphering and deciphering secret messages. Germany believed that its secret codes for radio messages by Enigma encryption machine were indecipherable to the Allies. However, the meticulous work of code breakers based at Britain’s Bletchley Park cracked the secrets of German wartime communication, and played a crucial role in the final defeat of Germany.

Cryptography primitives were solely concerned with converting messages into unreadable groups of figures to protect the message’s content during the time the message was being carried from one place to another. As early stated in this chapter

¹hieroglyphics (God’s word) is writing system used by the ancient Egyptians that combined logographic and alphabetic elements [3].

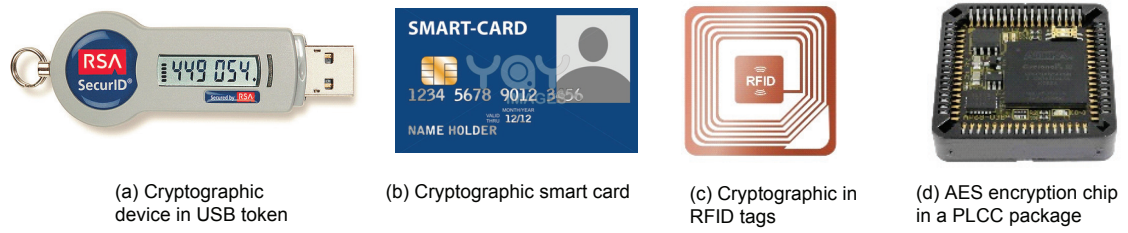


Figure 1.2: Modern cryptographic devices [6]– [9].

that the cryptographic environments have changed due to the aspect of the acquisition of knowledge, information, and the evolution of humanity and technology. For some examples out of many types of recent technology of cryptographic devices are listed in Fig. 1.2.

For security reasons, our digital certificate should be stored on a secure hardware device, such as a cryptographic universal serial bus (USB) token. The USB token is shown in Fig. 1.2(a). This device stores our digital certificate securely and ensures that nobody else has access to private key except the owner. One of the most important properties of the cryptographic USB token is that the private keys stored on the UBS token, can never be copied out of the token. As a result, during a digital signing operation, the digital signature is created directly on the token. Security for our private key is further enhanced through the mandatory use of a password to access the contents of the cryptographic USB token.

Another portable device that in very common use at this time is cryptographic smart card, as shown in Fig. 1.2(b). Based on the factual data [10], smart cards can be traced back to 1968 when using plastic cards as the carrier of microchips, that was first developed by the German inventors Jürgen Dethloff and Helmut Grötrupp. Two years later, in 1970, a patented Japanese, Dr. Kunitaka Arimura developed a similar application. The first formal reality of a smart card came with Roland Moreno’s smart card patents in France in 1974. With his patents, the semiconductor industries were able to manufacture and supply the required integrated circuits at a reasonable price. From those inventions and patents, the smart card has been commercially used for various applications.

The most common smart card applications are credit cards, electronic cash, computer security systems, banking systems, satellite TV payments, government identification, health care, entertainments, and transportations and so on. International Organization for Standardization (ISO) uses the term of integrated circuit

card (ICC) [11], is a plastic card that contains an embedded computer chip, either a memory or microprocessor type that stores and transacts data. These data are usually associated with either value, information, or both and are stored and processed within the card's chip. The card data is powered and transacted via a card reader that is part of a computing system.

Moreover, cryptographic system has been broadly expanded in secure wireless communication. In this regards, the light-weight radio frequency identification (RFID) tag is one of the devices to be introduced in this dissertation background, as depicted in Fig. 1.2(c) . An RFID tag is comprised of an integrated circuit (IC) attached to an antenna that has been printed, etched, stamped or vapor-deposited onto a mount which is often a paper substrate or Polyethylene Terephthalate (PET) [12]. The chip and antenna combo, called an inlay, is then converted between a printed label and its adhesive backing or inserted into a more durable structure. The tag's chip or IC delivers performance, memory and extended features to the tag. The chip is pre-programmed with a tag identifier (TID), a unique serial number assigned by the chip manufacturer, and includes a memory bank to store the items' unique tracking identifier called an electronic product code (EPC). An RFID reader is a network connected device (fixed or mobile) with an antenna that sends power as well as data and commands to the tags. The RFID reader acts like an access point for RFID tagged items so that the tags' data can be made available to business applications.

The component of cryptographic devices can be implemented either on separate chip or on a single chip. If they are implemented on separate chips, the chip need to be mounted on a printed circuit board (PCB). Suitable packages for chip that are mounted on a PCB are for example the dual in-line package (DIP) or the plastic landed chip carrier (PLCC), as shown in Fig. 1.2(d). Inside the PLCC package, mostly cryptographic devices are built based on multiple chips, such as cryptographic acceleration cards or hardware security modules (HSMs). Besides the modern cryptographic devices that the author has just introduced above, generally, they compose of several important components that shape the whole cryptosystems to perform specific task. The components of a cryptographic device can essentially divided into two groups. The components in the first group perform cryptographic operation, e.g., a digital circuit that performs encryptions. The components in the second groups handle data of cryptographic operations, e.g. non-volatile memory that provided the encryption key. Below, the author lists the component of typical cryptographic devices:

- **Dedicated Cryptographic Hardware :** This component includes all hardware that solely dedicate to performing cryptographic operations, e.g., a dedicated cryptographic circuit that implements Advanced Encryption Standard (AES).
- **General Purpose Hardware :** This components includes all general-purpose hardware that is used to preform cryptographic operations, e.g., a microcontroller that is programmed to performs AES encryption.
- **Cryptographic software :** This component consists of any type of software that implements cryptographic operations, e.g., software that implements AES.
- **Memory :** This component stores data of cryptographic operations, e.g. AES encryption keys.
- **Interface :** This component handles the data transfer to and from cryptographic device. For security reason, the cryptographic application impose special demands on the interface. It is for example crucial that the interface prevent sensitive data, like a cryptographic key, from unauthorized from the outside.

In practice, there are essentially two way to implement a digital circuit on a chip to execute cryptographic encryption. The first way is to implement the digital circuit as an application specific integrated circuit (ASIC). In this case, it is necessary to create a layout of a chip, which is later on will be discussed in our design work in Chapter 6. Based on the layout, the chip can be manufactured by a semiconductor foundry. The manufacturing is done based on so-called process technology. Cryptographic devices, like smart card or USB tokens, are often built on the basis of ASICs.

The second way to implement a digital circuit on a chip is to use a field-programmable gate array (FPGA). FPGAs essentially consist of programmable logic cell and programmable wires between this cells. By loading a configuration file into the FPGA, the engineer defines the behavior of the cells and the connection between the cells.

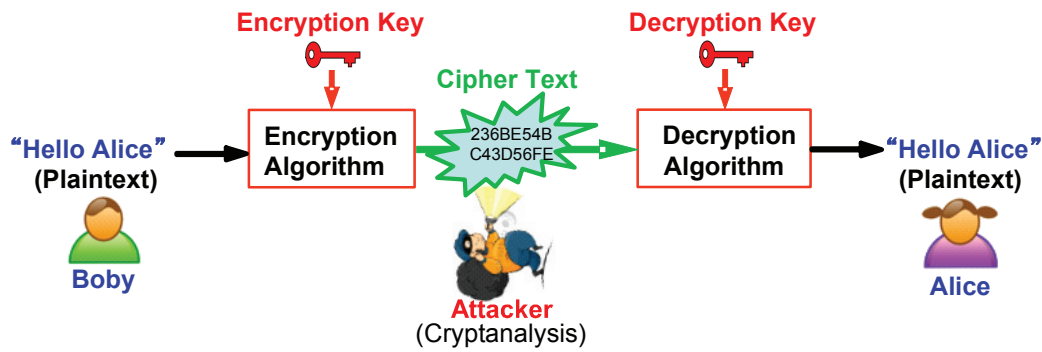


Figure 1.3: General assumption of cryptographic system and cryptanalysis.

1.1.2 Side-Channel Analysis Attacks on Cryptographic Devices

Brief introductory of cryptography and the cryptographic devices have been explored in the previous sub-section. Although, the innovative technology has transformed the physical devices from traditional way to modern styles and from analog transformation to digital coding, the generic cryptographic process itself still remains same, as depicted in Fig. 1.3.

This figure illustrates private communication between Bob and Alice. Bob sends a message "Hello Alice", which known as a plaintext, and then encoded into unreadable ciphertext using encryption algorithm. The cipher text is then decrypted by using decryption algorithm in order to be read by Alice. To encode and decode the original information, both algorithms have a secret key, and the type of keys are depend on the type of algorithm they use, such as symmetric algorithm or asymmetric algorithm.

Symmetric algorithms are what a common sense of cryptographic assumption is about: two parties have an encryption and decryption method for which they share same secret key. All cryptographic from ancient times up until 1976 was exclusively based on symmetric methods. Symmetric cipher are still in widespread use, especially for data encryption, such as Rijndael algorithm that was standardized as AES by the American National Institute of Standard and Technology (NIST) in 2001 [13], Twofish algorithm [14], Serpent algorithm [15], Blowfish algorithm [16], CAST5 [17], RC4 [18], 3DES [19], Skipjack [20], Safer+/++ (Bluetooth) [21] and IDEA [22].

Asymmetric (Public-Key) algorithm: In 1976, an entire of cipher was intro-

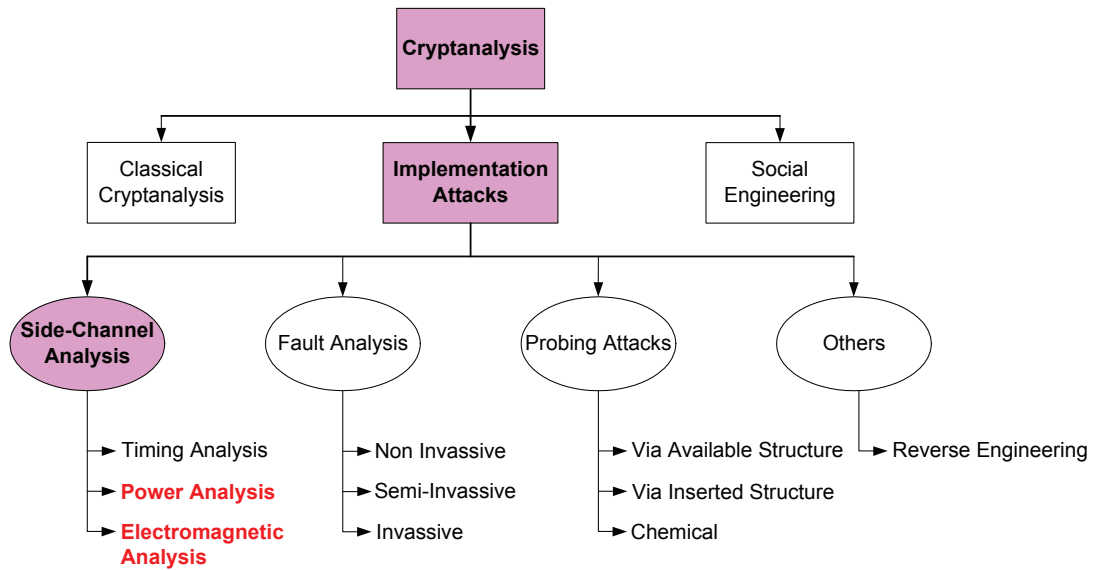


Figure 1.4: Overview of cryptanalysis.

duced by Whitfield Diffie, Martin Hellman and Ralph Merkle, which defined as Diffie–Hellman key exchange (D–H) [23], RSA [24] and DSA [25]. In public-key cryptographic, a user possesses a secret key as in symmetric cryptography but also a public key. Asymmetric algorithm can be used for applications such as digital signatures and key establishment.

This sub-section deals with cryptanalysis, means the art of “breaking the codes,” which is illustrated in Fig. 1.3 as an Attacker. The aim of the attacker is to reveal the secret key by eavesdropping secure information via communication channel between sender and receiver. We may think that the code breaking is for the intelligent community or perhaps organized crime, and should not be included in a serious classification of scientific discipline. However, the most cryptanalysis is done by respectable researchers in academia nowadays. Cryptanalysis is of central importance for modern cryptosystem. Without people who try to break our crypto methods, we will never know whether they are really secure or not.

To deeply understand what kind of cryptanalysis in the real world of cryptography, then, let us look at different ways of breaking cryptosystems as illustrated in Fig. 1.4.

• Classical Cryptanalysis

Classical cryptanalysis is understood as the science of recovering the plaintext x from the cipher text y , or, alternatively recovering the key k from the cipher text y . This cryptanalysis is something more analytical technique that require

some prior knowledge of how the code operates. The method also can be done with brute force attacks, where cryptanalysis simply iterate through all the possible permutations to test all possible keys. Briefly speaking, these all about mathematical algorithms attacks.

- **Social Engineering Attacks**

Bribing, blackmailing, tricking or classical espionage can be used to obtain a secret key by involving humans. For instance, forcing someone to reveal the secret key, e.g., by threatening someone with a knife or gun can be quite successful. Or, by getting the secret key through phone call, or, short message service (SMS), or via email by asking your password and bank account information to transfer money to you, or many others illegal black social activities to obtain your private secure information.

- **Implementation Attacks**

Several types of implementation attacks are amongst the strongest currently known attacks against cryptographic devices when looking at practicability, equipment, and costs. Attacking techniques on physical cryptographic devices are manifold as described in the thread of implementation attacks in Fig. 1.4.

- **Side-Channel Analysis (SCA) Attacks** is currently the most powerful attacks against cryptographic devices, and is categorized as passive attacks, where the cryptographic device is largely or even entirely within its specification. SCA attacks have become a special threat for cryptographic hardware engineers working to secure private information stored in cryptographic devices such as smart cards, RFID tags, USB tokens, and wireless sensors. Examples of side-channel attacks are attacks based upon sound, infrared radiation, time delays, simple or differential power analysis (SPA/DPA), and simple or differential electromagnetic analysis (SEMA/DEMA). The timing attacks documented by Kocher in 1996 [41] demonstrated how measuring computation time can reveal vital information about secret keys. The method of a power analysis attack involves probing device for physical measurements of its current consumption with respect to execution time. The author considers a power analysis an effective attack for revealing the secret key of a smart card by statistically analyzing power fluctuations that occur while the device encrypts and decrypts large blocks of data [47], [50]. The electromagnetic radiation

attacks have been extensively studied in [59]– [66]. DEMA attacks can reveal secret information because the electrical current flow during the switching of the CMOS gates causes a variation of the surrounding electromagnetic field that can be monitored by positioning an inductive probe around the microcontroller chip.

Various kind of attack techniques are being described in this chapter, however, this dissertation's work is a secure logic design for countermeasure against SCA attacks, therefore, SCA attacks will be extensively surveyed in Chapter 3.

- **Fault attacks (FA)** are categorized as active implementation attacks. In contrast to side-channel analysis, the behavior of the device is manipulated (tempered with) in order to gain erroneous computation results, or to make the device behave abnormally. Based on these results, it is possible to reveal secret information. Manipulating the device can be done in several ways. They differ in terms of effort and efficiency.

Invasive attacks: An invasive attacks typically starts with the depackaging of the device. Subsequently, different components of the device are accessed directly using a probing station. Invasive attacks are extremely powerful. However, they typically require quite expensive equipment. Consequently, only few publications on this topic are existed, such as reported in [26], [27].

Semi-invasive attacks: In semi-invasive attacks, the cryptographic devices is also depackaged. However, no direct electric contact to a chip surface is made—the passive layer stay intact. Semi-invasive attacks typically do not require as expensive equipment as invasive ones. However, the total effort that is needed to conduct semi-invasive attack still relatively high. In particular, the process of floating the positions for an attack on the surface of a modern chip require quite some time and expertise. The most comprehensive publication on semi-invasive attacks is in the Ph.D. thesis of Skorobogatov [27].

Non-invasive attacks : An active attack that can usually be performed at low cost. Thereby, the package of a device is not modified. Faults are injected by changing the working conditions of the device, e.g., a clock-glitch or power-glitch for short-term interrupt of the power supply, or by changing the temperature of the surrounding environment. These methods affect the whole device at once. Therefore, they belong to the global

attacks. Local attacks, on the other hand, target only at a limited area, like one or a few memory cells. A common local attack is optical fault induction. A survey on various methods that can be used to induce faults in semiconductors and exploit such errors maliciously has been reported in [28].

- **Probing Attacks:** While fault attacks try to manipulate the device, probing attacks aim to spy on inner values of the chip, like a data bus. This can be achieved by placing a probe on the chip. However, placing a probe is quite expensive and therefore the number of probes placed on a chip should be minimized. Thus, probing attack investigates how to reveal secret information with only one or a few probes, as experimentally described in [27].
- **Others:** As security by obscurity is still commonly used, reverse engineering aims at revealing the functionality and the inner life of a device [29]. Therefore, the chip is disassembled layer by layer. From each layer photos are taken to reconstruct the layout of the device afterwards.

A brief summary of this sub-section, the author wants to clarify that a solid cryptosystem should adhere to *Kerckhoffs's Principle*, postulated by a late cryptographer Auguste Kerckhoffs in 1883: *A cryptosystem should secure even if the attacker knows all details about the system, with the exception of the secret key* [2]. This postulate was reformulated later by a cryptographer and a Father of digital age, Claude Shannon in 1946 “*as the enemy know the systems being used*” [30], which means that the secret key must be kept secret, and is contradict with the concept of “Security by Obscurity.” Moreover, the author paid high consideration to the recommendation in [50] that, “Cryptosystem security should not rely on the secrecy of its implementation”. This recommendation strongly encourages the cipher designers, software developers, and hardware engineers to work closely together when producing secure cryptographic devices.

1.2 Motivation and Contribution

General overview of cryptanalysis has been introduced early in this chapter. In this thesis, the author restricts himself to focus exclusively on SCA attacks, and more specifically, power analysis attacks and some extensions to electromagnetic emission attacks. In power analysis attacks, secure information on cryptographic devices (for

instance, smart card) are possible to obtain by a probe measurement for several hypothetical power traces while a device executes encryption/decryption computation. Then, these measurement traces are calculated using statistical methods, to analyzed the different peaks of the power traces, and then reveal the secret keys of a device. In the same method, the electromagnet emission analysis also implemented, which the information signals are taken by using antenna sensor around the cryptographic chip area. We consider both power analysis and electromagnetic analysis are the most threatening attacks that can reveal the secret information at almost any symmetric or asymmetric algorithm.

It is commonly known that, mostly, the modern cryptographic devices are implemented using semiconductor logic gates, which are constructed out of CMOS transistors. Electron flows across the silicon substrate when charge is applied to (or remove from) a transistor's gate, consuming power and producing electromagnetic radiation. Information leakage on secure CMOS ICs possible because there are different power consumption occurs when the CMOS logic gates are switched ON or OFF, as shown in Fig. 1.5. For example, peak current transition of a static complementary CMOS (scCMOS) used for security ICs consumes different peak current for charging and discharging process as shown in Fig. 1.5(a). Furthermore, a technique to balance the charging and discharging load for uniform peak current trace, the dual-rail CMOS (DR-CMOS) logic in Fig. 1.5(b) becomes a solution for secure logic designing. However, it is same as scCMOS inverter logic that both $1 \rightarrow 1$ and $0 \rightarrow 0$ input transitions produce no supply current flow. Observing the current transition in Fig. 1.5(a) and (b), generally, transitional power consumption values hold that $(P_{0 \rightarrow 0} \approx P_{1 \rightarrow 1}) \ll (P_{0 \rightarrow 1}, P_{1 \rightarrow 0})$, which is attackable by using Hamming Distance (HD) model in power analysis attacks. The idea behind HD model is to count the number of $P_{0 \rightarrow 1}$ and $P_{1 \rightarrow 0}$, $P_{0 \rightarrow 0}$ and $P_{1 \rightarrow 1}$ transitions that occur in the digital circuit during a certain time interval with the assumption of $(P_{0 \rightarrow 1} \approx P_{1 \rightarrow 0}) \neq (P_{0 \rightarrow 0} \approx P_{1 \rightarrow 1})$ [50]. From the view point of DPA and DEMA attack techniques, the scCMOS and DR-CMOS are vulnerable, because they perform different peak current transition and different large magnitude which cause a sudden variation of the electromagnetic field surrounding the chip area as reported in [64]. As a result, the DPA and DEMA attacks are a bit difficult to avoid.

As a solution approach, the author has implemented a logic circuit that exhibits uniform peak current for all possible input transitions to avoid HD model by an expression $(P_{0 \rightarrow 1} \approx P_{1 \rightarrow 0}) = (P_{0 \rightarrow 0} \approx P_{1 \rightarrow 1})$ as graphically shown on the right side of Fig. 1.5(c). The design contents, the merit of the comparison results and the appli-

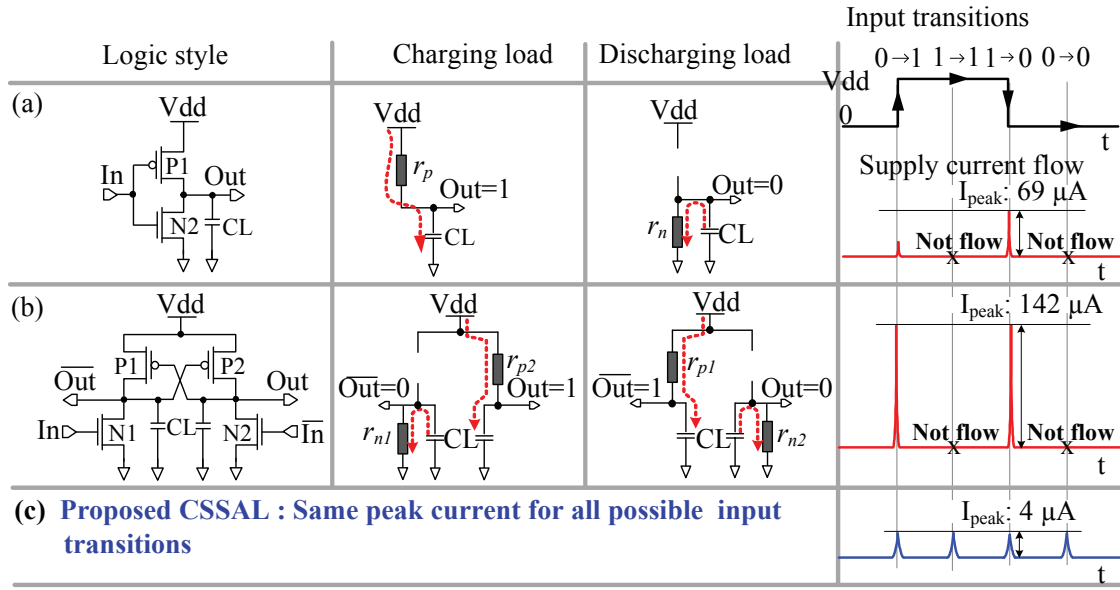


Figure 1.5: Existing CMOS logic problems and our proposed solution for SCA countermeasures. This figure illustrates transitional current traces of (a) scCMOS, (b) DR-CMOS, (c) Target supply current trace of the proposed CSSAL.

cations of the proposed logic have summarized and presented in ISCAS-2013 [31], and extensive results are in the published paper of Microelectronic Journal [32]. For contribution to side-channel attacks countermeasures, the author explores and investigates all the relevant conventional secure logic styles that available in the secure cell library, to identify the merit of logic's security, power efficiency, and possible applicability for cryptographic hardware design, such as in AES substitution box (S-box) implementation. Moreover, the author's proposed logic was implemented using adiabatic logic technique. The reason behind this implementation is to lower the peak supply current as low as possible, to balance supply current traces for all possible input logic transitions for counteracting power and electromagnetic analysis attacks. The limitation of designing digital logic circuit in adiabatic mode is that the logic speed is slow in comparing with the conventional CMOS logic styles. The maximum speed of adiabatic logic circuit is depend on power clock's phase delay (design technique) and propagation time delay. Accumulating all advantages and drawbacks of the proposed logic, the author deduces that the contribution work is for cryptographic devices that operate in low frequency band, require high security performance and low-power devices, such as smart cards (IC-cards), RFID tags, secure wireless sensors and other the battery-powered secure embedded devices.

1.3 Organization of the Dissertation

The construction of this dissertation is organized as following:

Chapter 1 (this chapter) has introduced the science of cryptology, short history of the cryptographic devices, and the presence of cryptanalysis. This chapter covers relevant cryptographic hardware implementation attacks that motivated the author to conduct this research on contributing side-channel attack countermeasures.

In Chapter 2, the author analyzes the CMOS logic's power consumption that contribute to the possibility of SCA attacks. Moreover, some basic techniques of power analysis model are derive in this chapter. The aim of the author in this chapter is to explore how the power consumption occurs in the complementary CMOS cells that cause supply current-to-data-dependencies which is vulnerable for secure logic implementation. Furthermore, power reduction technique using adiabatic switch principle to lower the peak supply current consumption of CMOS logic circuit for countermeasures against power analysis and electromagnetic emission will also be introduced.

Survey on side-channel information leakage and their countermeasures using existing conventional secure logic styles are covered in Chapter 3.

The author will introduce the proposed work in Chapter 4. Deep analysis of the proposed logic in terms of structure of the logic constructions, input timing design, individual logic operation, transitional power consumption using SPICE simulator are exclusively discussed here. To verify the merit of the proposed circuit, the comparative analysis results of the proposed work with the other conventional secure logic styles at the level of individual logic cells are conducted in this chapter.

In Chapter 5, the author will present the proposed logic implementation into two different circuit architectures, such as in the bit-parallel cellular multiplier over $FG(2^4)$ and in the 8-bit AES S-box circuit. At the same Chapter, the author will discuss some comparison study in pre-layout simulation results.

Measurement results of two LSIs that have been fabricated using $0.18\ \mu\text{m}$ CMOS process technology will be discussed in Chapter 6.

Finally, the author will draw the conclusion of this dissertation work and the perspective future research direction works in Chapter 7.

Chapter 2

CMOS Power Consumption, Power Analysis Model and Power Analysis Verification

Modern cryptographic devices mostly have implemented using CMOS digital circuit. A circuit that executes encryption/decryption algorithm digitally in a secure coding system to avoid information leakage from unintended parties. Digital circuits always consume power whenever they perform any computation. They draw current from power supply and then dissipate the received energy as heat. The power consumption determine whether a chip needs to be cooled or not, in the case of cryptographic devices, it determines whether a device can be attacked or not. Obviously, the fluctuation of power consumption is the most important property to be highly consider in this dissertation.

In this chapter, the author will discuss power consumption in a digital CMOS circuit, specifically at single-rail (SR) inverter level. A logic technique to reduce power consumption, and in the case of SCA, to lower the peak supply current transitions will be compared with the conventional CMOS logic style. The probability of switching power occurrences that lead to a power consumption model in power analysis attacks will be discussed. Moreover, the verification of the calculation parameters that will be used to measure the resistivity and the logic immunity for power analysis attack will also be introduced in this chapter.

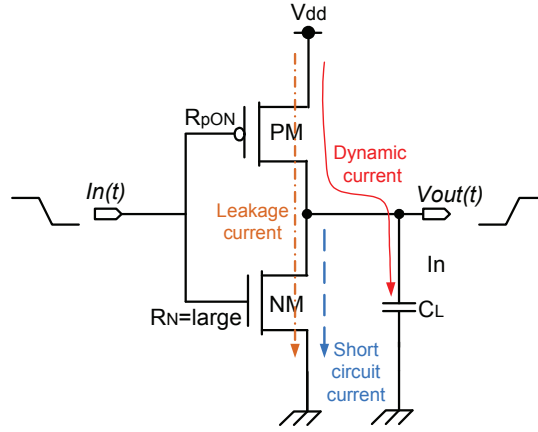


Figure 2.1: Total power in a CMOS inverter: The dynamic power, short-circuit power and leakage power.

2.1 Power Consumption of CMOS Circuits

Power consumption in a CMOS circuit composes of three main factors, such as dynamic power, short-circuit power and static (or leakage) power. The author illustrates those power consumption components in Fig. 2.1. This figure describes that dynamic power consumption occurs during the output node's capacitor C_L is switched, means charging process of the node capacitor. This power is also sometimes called as charging power. The short-circuit power happens when both transistors PMOS (PM) and NMOS (NM) operate simultaneously during short period of time of different input signal transitions (such as In signal changes from $0 \rightarrow 1$ and $1 \rightarrow 0$). The other contributing power is the static power, which is consumed at MOS transistors PM and NM are operating in the cutoff region, or during standby mode of real electronic devices. Mathematically, the author formulizes the overall CMOS power that illustrated in Fig. 2.1 as follows:

$$P_{total} = P_{dynamic} + P_{sc} + P_{stat} \quad (2.1)$$

To make it more clear for readers of this dissertation, the author will briefly explain each factor of Equation (2.1) in the following sub-sections.

2.1.1 Dynamic Power Consumption

Dynamic power consumption of a CMOS circuit is typically the dominant factor in the total power dissipation in micro-meter or earlier CMOS technology. The value of dynamic power depends on switching frequency f , amplitude of power supply V_{dd}

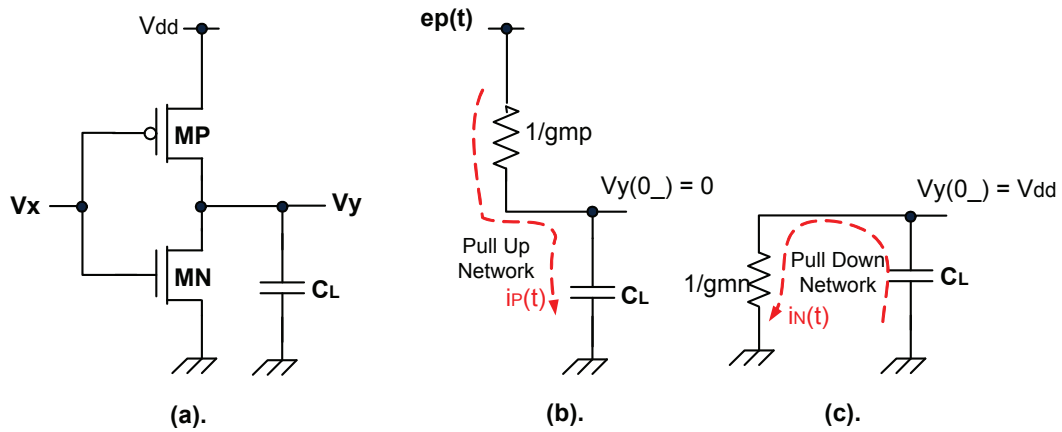


Figure 2.2: (a) Conventional CMOS inverter, (b) A CMOS pull up network (PUN) RC equivalent model for charging phase, (c) A CMOS pull down network (PDN) RC equivalent model for discharging phase.

and the load capacitance C_L of the output node. The size of C_L depends heavily on physical property of the used process technology, the length of the wires to the subsequent cell and the numbers of subsequent cells of an integrated circuit. To reduce this C_L in the circuit system is one of challenges of logic designers, particularly, in the placement and routing of the circuit's layout. This normally cause the mismatching values of the design parameters, such as power consumption and/or peak current traces. The author often experienced this phenomenon during the individual logic's layout design and implementation.

In this part, we analyze mathematically, how big the energy is stored in the output node's capacitor. Figure 2.2(a) shows a static CMOS logic inverter. The operation of this inverter logic is that when the state of input signal V_x changes from $1 \rightarrow 0$, the transistor MP is switched ON, the current supply from V_{dd} is flowing down to charge the output node of C_L from initial condition of $V_y(0_-) = 0 \rightarrow V_y = V_{dd}$. The internal equivalent RC model during this operation is called as pull-up network (PUN), as shown in Fig. 2.2(b). On the other hand, when the state of input signal V_x changes from $0 \rightarrow 1$, the transistor MN is switched ON, the output node of V_y is discharged from initial condition of $V_y(0_-) = V_{dd} \rightarrow V_y = 0$ level (grounded). The internal equivalent RC model during this operation is called pull-down network (PDN), as shown in Fig. 2.2(c). From the Fig. 2.2, the total power dissipation can be calculated using each network system. By considering the MOS resistance value of $1/g_{m_n} = 1/g_{m_p} = R$, $C_L = C$, we can calculate the current source that flows into the circuit, and then, further calculation of power dissipation of each network

system.

From the Kirchhoff Voltage Law (KVL), the equation for pull-up (charging) network shown in Fig. 2.2 has:

$$Ri(\tau) + \frac{1}{C} \int_0^t i(\tau) d\tau + v(0_-) = V_{dd} \quad (2.2)$$

By using Laplace transform for Eq. (2.2), we define the instantaneous current of $i(t)$:

$$RI(s) + \frac{1}{Cs} I(s) = \frac{V_{dd}}{s} \quad (2.3)$$

$$I(s) = \frac{CsV_{dd}}{(RCs + 1)s} = \frac{V_{dd}}{R(s + \frac{1}{RC})} \quad (2.4)$$

Applying the inverse Laplace transform for Eq. (2.4), we define the following equation:

$$i(t) = \frac{V_{dd}}{R} e^{-\frac{1}{RC}t} \quad (2.5)$$

Electrical analysis, the instantaneous power $p(t)$ is determined as following:

$$p(t) = i(t)V_R(t) = i(t)^2 R = \frac{V_{dd}^2}{R} e^{-2\frac{1}{RC}t} \quad (2.6)$$

Therefore, the energy dissipated over the period $t = 0$ to $t = \tau$ in Eq. (2.6) is expressed as:

$$E_{charge} = \int_0^\tau p(t) dt + E(0) = C \frac{V_{dd}^2}{2} (e^{-2\frac{1}{RC}\tau} + 1) \quad (2.7)$$

If, $\tau \rightarrow \infty$, then the energy is:

$$E_{charge} = \frac{1}{2} C V_{dd}^2 \quad (2.8)$$

Equation (2.8) shows that only half of the energy drawn from the power supply is stored in the load capacitance; the other half is dissipated as heat by PUN resistor. All the signal energy is dissipated during discharging in the PDN when the logic level in the output node is “0”, because no energy can enter the ground rail ($Q \cdot V_{gnd}$

$= Q \cdot 0 = 0$) [33]. Therefore, the total amount of energy dissipated as heat during charging and discharging is

$$\begin{aligned} E_{total} &= E_{charge} + E_{discharge} \\ &= (1/2)C_L V_{dd}^2 + (1/2)C_L V_{dd}^2 \\ &= C_L V_{dd}^2. \end{aligned} \tag{2.9}$$

Finally, the average dynamic power ($P_{dynamic}$) consumed by the cells during a certain period of time T can be formulated as shown in (2.10). In this equation, the f denotes the clock frequency, and α is the switching activity factor of the cell. The activity factor corresponds to the average number of 0→1 transitions that occur at the output cell in each clock cycle.

$$P_{dynamic} = \alpha f C_L V_{dd}^2. \tag{2.10}$$

From the aforementioned equation, it is apparent that the energy consumption in a conventional CMOS circuit can be reduced by scaling down the supply voltage of V_{dd} , decreasing the frequency of switching activity in the circuit, and/or lower the output node capacitance.

2.1.2 Short-Circuit Power Consumption

Short-circuit power (P_{SC}) occurs because there is no finite zero period of rise time and fall time of input signal transitions. During a certain interval time (herein after known as short-circuit time or t_{sc}) of the rise and fall edges of input signal, both PMOS and NMOS of an static inverter logic will be ON-state simultaneously. At this time, the short-circuit current path is established between the supply voltage V_{dd} and the common ground. The detail discussion of short-circuit power was reported in [36], with an expression shown in Equation (2.11); where β is a gain factor of a MOS transistor, τ represents the rise and fall time, f denotes a clock frequency, and the V_T is the MOS transistor threshold voltage.

$$P_{sc} = \frac{1}{12} \beta \tau f (V_{dd} - 2V_T)^3. \tag{2.11}$$

From the Equations (2.10) and (2.11), we observe that short-circuit power is smaller than the dynamic power. However, short-circuit power can also largely contribute to the huge dynamic power when it comes to several cascaded buffer circuits or in inverter chain circuit. For the clarity of dynamic and short circuit power, we have

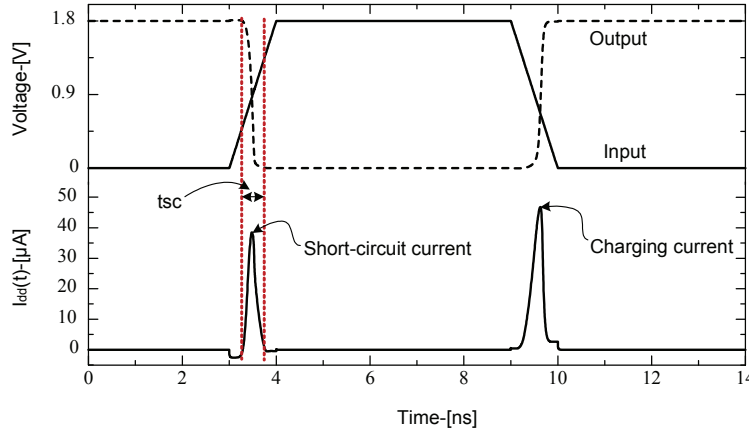


Figure 2.3: Simulation result of the short-circuit and dynamic power consumptions of a static CMOS inverter.

drawn both as depicted in Fig. 2.3¹. This figure is the modified one that appeared in previous chapter (see Fig. 1.5(a)) and was included in the author's proceeding paper presented in ISCAS-2014 [40].

2.1.3 Static Power Consumption

In a static CMOS inverter cell, static power consumption is typically very low. It essentially consists of the power used when the transistor is not in the process of switching. It occurs when small leakage current (I_{leak}) is flowing through the MOS transistor that is turned off. Static power is increasing significantly proportional to the shrinking of CMOS process technology.

There are several components that trigger the occurrence of leakage power [37]–[39]; such as: (1) Reverse bias diode leakage current (I_{rbdl}), which occurs due to the reverse bias current of p-n junction between diffusion region of the transistor and substrate. The amplitude of this current is define as: $I_{rbdl} = AJ_s(e^{qV_{bias}/kT} - 1)$, where A is a junction area, J_s is reverse saturations current density, V_{bias} is the reverse bias voltage across the junction and $V_{th} = kT/q$ is a thermal voltage; (2) Gate oxide tunneling current (I_{ox}) is the leak current that flows from oxide insulation to substrate. This value getting bigger and bigger proportional to the scaling down of CMOS process technology, where the gate oxide is also becoming thinner accordingly. The amplitude of this leak current is defined as $I_{ox} = AE_{ox}^2 e^{-B/E_{ox}}$.

¹Figure 2.3 obtained from SPICE simulation with the following parameters: $V_{dd} = 1.8$ V, $C_L = 1$ fF, MOS sizes $W/L = 0.6 \mu\text{m}/0.18 \mu\text{m}$ for both PMOS and NMOS transistors.

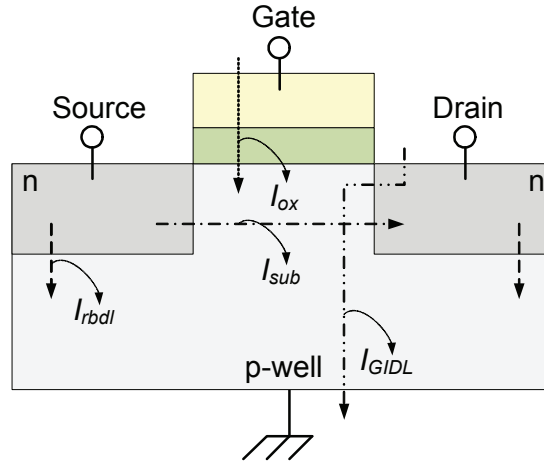


Figure 2.4: Components of leakage power in CMOS (adopted from [38]).

From this expression, the E_{ox} denotes the electric field across the oxide, B is the physically-based exponential parameter; (3) Gate induced drain leakage (GIDL) is another leakage current that increases exponentially due to the reduced gate oxide thickness as estimated: $I_{GIDL} = AE_s e^{-B/E_s}$, where E_s is the transverse electric field at the surface; (4) Another component of leakage power is the subthreshold leakage current (I_{sub}). The subthreshold current flows from source to drain even if gate source voltage V_{GS} below the threshold voltage of the device. This leak current occurs due to two reasons, such as the weak inversion effect (V_{GS} is lower but close to V_{th}) and the drain-induced barrier lowering (DIBL). DIBL is the reduction of the threshold voltage of transistor at high drain voltages. The subthreshold voltage is estimated as $I_{sub} = I_0 e^{(V_G - V_S - V_{T0} - \gamma V_S + \eta V_{DS}) / (n V_{th})} [1 - e^{-V_{DS} / n V_{th}}]$, where n is subthreshold swing coefficient constant, γ is the linearized body effect coefficient, η is the DIBL coefficient, and I_0 is the technology subthreshold leakage, which can be represent as $I_0 = \mu_0 C_{ox} (W/L) V_{th}^2 e^{1.8}$, with μ_0 is zero bias electron mobility. All components of leakage current that the author has just explained above are indicated in Fig. 2.4. Thereby, the total summation of all leakage current I_{leak} components aforementioned can be derived as:

$$E_{stat} = I_{leak} V_{dd} \quad (2.12)$$

From the view point of power analysis attacks, a leakage current of cryptographic circuit under 100-nm technology could leak the secret information. A novel class of attack to nanometer cryptographic circuit, named leakage power analysis (LPA) has been recently introduced by Alioto *et al.* in [34], [35]. Based on some prac-

Table 2.1: Output transition of a CMOS inverter cell and the corresponding power consumption.

Output Transition	Power	Type of Power	Power Value
0→0	$P_{0→0}$	static	Low
0→1	$P_{0→1}$	static + dynamic + short-circuit	High
1→0	$P_{1→0}$	static + short-circuit	High
1→1	$P_{1→1}$	static	Low

tical measurements of LPA attacks, they concluded that, even if transistor level countermeasures developed for thwarting DPA on cryptographic circuit have been implemented for balancing the overall dynamic consumption, an attacker is able to exploit the fraction of overall consumption due to the static power, which keeps on exhibiting dependence on the data input.

As the closing of this subsection, the author summarized the transitional power consumption of all contributing components in Table 2.1. In this table, 0→0 and 1→1 output transitions consume only static power, which is classified as low power value or even ignored in DPA attacks technique.

2.1.4 Another Factor of Power Consumption: Glitches

Glitches current in CMOS Circuits are data dependent and have the strong impact on the dynamic power consumption. Normally, the digital circuit comprises with the combinational cells that rises the complexity, multistage logic network. In this case, the input signals are generated from the outputs of previous logic stage, which cause non-zero delay time. The different arrival times of the input signals at the combinational cell lead to temporary states of the outputs condition (undesired state). Such temporary states of the output of a combinational cell are called *glitches*, *critical races*, or *dynamic hazard*. Based on the author's experience, even if only several nanosecond interval time of dynamic hazard voltage that surpass V_{th} value has strong influence to the stability of the logic inversion in more complex digital circuit. The author has challenged this troublesome during the secure logic design and its implementation. In many cases, this electric hazard often occurs when there is a long critical path without considering the balanced path, and hence, the skew time variation getting larger. Therefore, dynamic hazard elimination mostly can

be done with balanced path for multistage logic network. By doing this, of course will increase the gate size, but if the performance metric is the power efficiency, then, the balanced path delay is the option, since the glitches power is reportedly account for 20%–70% of dynamic power [97]. In the author’s design, the proposed logic adopted adiabatic operation, hence the glitch current is a critical issue to be considered, which will be discussed in the logic implementation in Chapter 5.

2.2 Adiabatic Logic Technique

The previous section has explained the conventional CMOS power consumption, where the global power dissipation is dominated by dynamic power (charging power). From the given mathematical equations of each contributing components, we observe that the common solution for power minimization is to reduce supply voltage V_{dd} . In this section, the author introduces the adiabatic logic principle that is adopted in the author’s secure logic design to lower the peak supply current for resistance against power and electromagnetic analysis attacks.

Adiabatic switching is commonly used to minimize energy loss during charging/discharging. The word “adiabatic” (Greek *adiabatos*, which means impassable) indicates a state change that occurs without heat loss or gain. During adiabatic switching, all the nodes are charged or discharged at a constant current in order to minimize power dissipation. This is accomplished by using AC power supplies to initially charge the circuit during specific adiabatic phases and then discharge the circuit to recover the supplied charge. The principle of adiabatic switching can best be explained by contrasting it with the conventional dissipative switching technique. The internal equivalent RC model for charging and discharging output load capacitance is shown in Fig. 2.5. The difference between the adiabatic inverter and the conventional CMOS inverter is that, the conventional CMOS inverter uses the step voltage of V_{dd} , whereas the adiabatic inverter uses power clock signal as source voltage, as can be seen in the Figs. 2.6(a) and (b) for CMOS and adiabatic model, respectively.

In adiabatic power consumption analysis, the following calculation are done to obtain the dissipated energy, where the τ is ramping time in Figure. 2.6(b):

$$Ri(t) + \frac{1}{C} \int i(t)dt = \frac{V_{dd}}{\tau}t \quad (2.13)$$

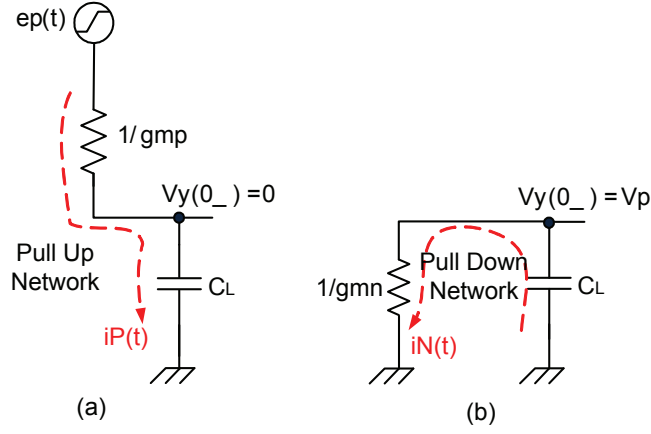


Figure 2.5: (a) An adiabatic logic PUN equivalent RC model for charging phase, (b) An adiabatic PDN equivalent RC model for discharging phase.

Applying the Laplace transform to Eq. (2.13), the following equations are derived

$$RI(s) + \frac{1}{Cs}I(s) = \frac{V_{dd}}{s^2\tau} \quad (2.14)$$

$$I(s) = \frac{CsV_{dd}}{(RCs + 1)s^2\tau} = \frac{V_{dd}C}{\tau} \left(\frac{1}{s} - \frac{1}{s + \frac{1}{RC}} \right) \quad (2.15)$$

Then, employing the inverse Laplace transform to Eq. (2.15), it becomes:

$$i(t) = \frac{V_{dd}C}{\tau} (1 - e^{-\frac{1}{RC}t}) \quad (2.16)$$

Same process as Eq. (2.6), then the $p(t)$ of adiabatic is:

$$p(t) = i(t)^2 R = R \frac{V_{dd}^2 C^2}{\tau^2} (1 - 2e^{-\frac{1}{RC}t} + e^{-\frac{2}{RC}t}) \quad (2.17)$$

Energy dissipated over the period $t = 0$ to $t = \tau$ is expressed as:

$$\begin{aligned} E_{charge} &= \int_0^\tau p(t)dt + E(0) \\ &= R \frac{V_{dd}^2 C^2}{\tau^2} \left[t - 2RCe^{-\frac{2}{RC}t} - \frac{RC}{2}e^{-\frac{1}{RC}t} \right]_0^\tau \\ &= R \frac{V_{dd}^2 C^2}{\tau^2} \left(2RCe^{-\frac{1}{RC}t} - \frac{RC}{2}e^{-\frac{2}{RC}t} - \frac{3RC}{2} + \tau \right) \end{aligned} \quad (2.18)$$

By using the approximation of $\tau \gg RC$, then, the energy dissipated is equal to:

$$E_{diss} = \frac{RC}{\tau} CV_{dd}^2 \quad (2.19)$$

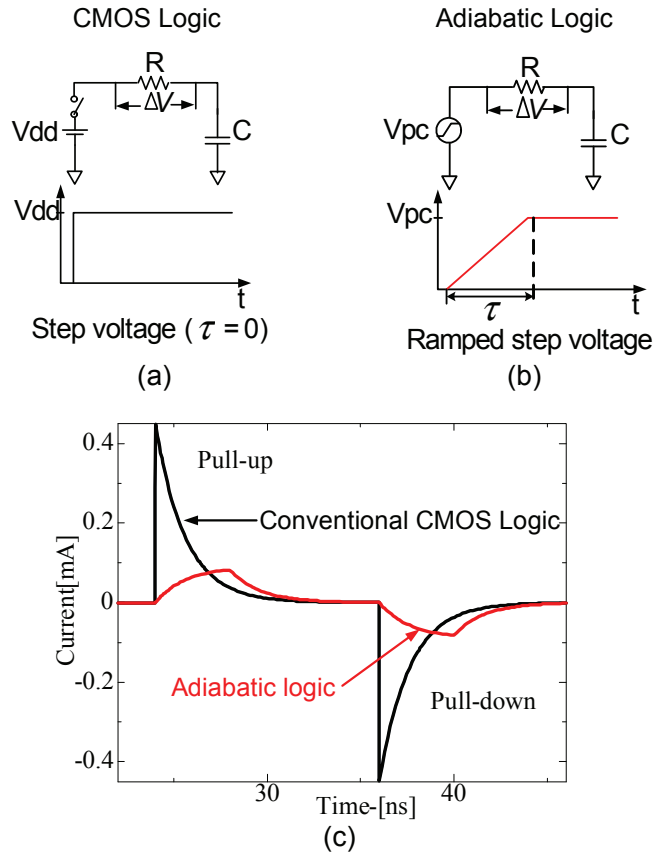


Figure 2.6: Comparison of the supply current transitions for the equivalent RC models of the (a) CMOS logic step voltage and (b) Adiabatic logic ramped step voltage. (c) The peak supply current of the adiabatic logic is significantly lower than that of the conventional CMOS logic under the same parameters and conditions.

Analytical understanding of the Eq. (2.8) and Eq. (2.19), the energy dissipated by the conventional CMOS inverter depends on the value of load C and the V_{dd} . On the other hand, the most logical understanding for adiabatic inverter is that the energy dissipated strongly depends on the transition switching time τ . Ideally, if the τ is longer, energy dissipation is nearly zero.

Figure 2.6(c) shown the comparison graph of the peak current traces of the conventional CMOS logic and adiabatic logic using each respective equivalent RC model². In this figure, when $t \leq 23$ ns, the circuit operates in pull-up network, and the pull-down network occurs at $t > 36$ ns. For CMOS, a large amount and sudden flow of current can be observed as indicated in Fig. 2.6(c). On the other hand, a

²Figure 2.6(c) obtained from SPICE simulation using Fig. 2.6 (a) and (b) with $R = 4$ k Ω and $C = 0.1$ pF.

gradual increase of supply current peak could be seen in the same figure with red color trace. Similar shapes are depicted during pull-down networks. By comparison, adiabatic circuit is showing low peak current about five times lower than that of the CMOS peak current. Power consumption is a function of instantaneous supply current and voltage, therefore, in adiabatic circuit, as the total amount of current flow through the circuit is less, the power dissipation definitely lower compare to the CMOS logic style.

2.3 Power Simulation Model in SCA Attacks

CMOS power consumption has been clearly described in the previous section. Table 2.1 summarizes the power consumption by an static CMOS inverter, with the clarification to where the power is lost at each possible data flipping. Observing the power transition in Table 2.1, we can map output the data values that represent the power transition relationship as following expression:

$$(P_{0 \rightarrow 0} \approx P_{1 \rightarrow 1}) \ll (P_{0 \rightarrow 1}, P_{1 \rightarrow 0}) \quad (2.20)$$

Based on this equation, the author intends to explain the most common sense power simulation model by an attacker and the logic designers. Some power models are mapped, so that one can design a proper secure logic accordingly, in order to diminish information leakage in SCA attacks.

In this section, the author first describes two generic power models that are commonly used for power analysis attacks, such as Hamming-distance (HD) and Hamming-weight (HW) models. Thereafter, the author will describe a simple condition of transitional power relationship, which become a target to be fulfill in the logic design and implementation. Because, the author acknowledges that the power simulations are crucially important for designers in order to build reliable system, power efficient and secure digital circuit.

2.3.1 Power Model for Attacker

Hamming-Distance Model

The basic idea of HD model is to count the number of $(0 \rightarrow 1)$ and $(1 \rightarrow 0)$ that occur in a digital circuit during a certain time interval. This power trace does not contain actual power values in watt but the number of transitions that occur in corresponding time interval to determine the power usage.

In HD model, the following assumptions are made; first, it is assumed that all $(0 \rightarrow 1)$ and $(1 \rightarrow 0)$ transitions are consuming equal power consumption, and second, all $(0 \rightarrow 0)$ and $(1 \rightarrow 1)$ transitions are also assumed to contribute same equal power consumption. Hamming-distance model is a basic of power simulation model for power analysis attacks, which mainly concentrated in dynamic power consumption. It does not consider the differences in parasitic capacitances of wires or cells. It also completely ignores the static power consumption of the cells. In this model we only assume that all cells contribute equally to the power consumption. The author dictates the definition of HD model as: the HD of two string of registers R_0 and R_1 correspond to the Hamming-weight, as follows: $HD(R_0, R_1) = HW(R_0 \oplus R_1)$ [50]. Observing the concept of HD model and associated with the power consumption in Equation (2.20), proportionately, the author defines following expression:

$$(P_{0 \rightarrow 0} = P_{1 \rightarrow 1}) \neq (P_{0 \rightarrow 1} = P_{1 \rightarrow 0}) \quad (2.21)$$

This equation is the reformulation of the Eq. (2.20) that relies on hamming distance model.

Practically, the attacker often has some clues about parts of the attacked device netlist; for example, if the attacked device is a microcontroller, then it should be considered as publicly known microcontroller. Hence, it has register, data bus, some memory, and arithmetic logic unit, and some other communication interfaces. In this case, the HD is suitable to map the data flipping on the data bus, because the data bus is quite long and connected to many components. Thus, the capacitive loads of such a bus is quite big, and hence, it contribute significantly to the overall power consumptions of the microcontroller. One of the other attackable components using HD is for example, the register. The register is triggered by a clock signal, consequently, they change their value only once in each clock cycle. Therefore, the attacker can simulate the power consumption of a register by calculating the HD of the value stored in the consecutive clock cycle.

From this short explanation, we understand that, generally, the HD model can be used to simulate the power consumption of a part of a netlist, whenever the attacker knows the consecutive data values that are processed by this part of the netlist.

Hamming-Weight Model

In Hamming Weight (HW) model, the attacker assumes that the power consumption is proportional to the number of bits that are set in a processed data value. It usually

applied if the attacker does not know the consecutive data values for some known part of the netlist, which means that the values that are processed before and after the current value are ignored; which is different with HD model that the attacker is required to have some knowledge of preceding and succeeding values of the data. Or in other words, we can say that HW model is not very well suited to describe the power consumption of a CMOS circuit. The author has mentioned in the previous section that the power consumption of a CMOS circuit is rather depends on the fact that whether there occurs a transition in the circuit or not, and not on the processed value.

As stated above that HW is unrelated to the power consumption value, however, in practice it can be calculated same as HD model with some trial and error analysis by setting preceding data in digits values, either set to all “0” or all “1” before processing the current data values. For better understanding of this method, the author shows some cases as following: it assumes to have some data bus strings, such as D_0 , D_1 and D_2 . The goal of attacker using HW model is to calculate the power consumption that is caused by the processing of D_1 , without knowing the value of D_0 or D_2 . D_1 is involved in two possible transitions, *i.e.*, $D_0 \rightarrow D_1$ and $D_1 \rightarrow D_2$. The first scenario is to set data bus string of D_0 to all “0” (*i.e.*, n -bit data bus are set to 0), and then, calculate the $D_0 \rightarrow D_1$ transition. In this case, HW model is equivalent to HD model, such as $HD(D_0, D_1) = HW(D_0 \oplus D_1) = HW(D_1)$. The other scenario is to set D_0 value to all “1,” and it holds that $HD(D_0, D_1) = n - HW(D_1)$. To calculate $D_1 \rightarrow D_2$ transition, the same process of $D_0 \rightarrow D_1$ are conducted.

For power analysis attacks, it does not matter whether the simulated power consumption is directly or inversely proportional to the actual power consumption. It is only important that it is proportional. Hence, the HW and HD models are equivalent in terms of power analysis attacks, if the bits of D_0 are set to “0” or “1” before the $D_0 \rightarrow D_1$ transition occurs.

2.3.2 Power Model by Designer

The most commonly used levels to verify the accuracy of power simulation model by the designers are so called analog level, digital level and the behavioral level.

Power modeling in analog simulations are based on transistor netlist that also contains parasitic elements (capacitance and resistance) of the circuit. The precision of the simulation essentially depends on the precision of the modeling of the circuit parasitic. The validity of the analog model are usually verified using analog circuit

simulators, such as *SPICE* from the university of California at Berkley, *Spectre* from Cadence Design System and *Nanosim* from Synopsys.

The power simulations at the logic level are based on the netlist of the cells that ideally also contain back-annotated³ information about signal delay, rise-time and fall-time. The precision of the logic simulation depends on the quality of back-annotated information and on the power model used for the cells. The power model at the logic level is to map the simulated transitions to a power traces. Less attention for this level will raise hazardous glitches that will affect the overall circuit performance. For this purpose, it is important to have power model that describe how the transitions at the output cell are related to the power consumption. The author has paid high consideration on this level during the tough time on circuit design and implementation. On this regards, the author has mapped the power model as follows:

$$(P_{0 \rightarrow 0} = P_{1 \rightarrow 1}) = (P_{0 \rightarrow 1} = P_{1 \rightarrow 0}) \quad (2.22)$$

This condition is set to withstand the classical idea of HD model that the author has shown in Equation (2.21). Equation (2.22) is the most important property that the author has put in account to be fulfilled during the logic design. To fully understand the relation is equation (2.22), the author will address some more concrete explanation that proven by some power simulation values at the CMOS inverter in Chapter 4.

Power simulation model at behavioral level are based on high-level description and high-level power model of digital circuit. This high-level description contains the major components of the digital circuit (such as microcontrollers, memories, dedicated hardware modules, etc.). This model is used during the simulation to map the activities of components, such as processing of data or the execution of instructions, to power consumption values.

2.4 Power Simulation Verification

After having at glance of the CMOS power contribution to side-channel information leakage, and followed by mapping some power simulation models for both attacker

³Back-annotation is the term that generally used in connection to netlist logical simulation. This is specifically related to the static timing analysis (timing delay), and often include more accurate physical information from the layout (post-layout simulation for static timing analysis).

and designer, here, the author will introduce some important parameters that will be used to check whether the logic circuit is strong enough to thwart power analysis or is not worth for secure logic implementation. This is more important information that behind the merit of this dissertation works. There are two techniques that the author has adopted to validate the proposed logic circuit; those are power consumption verification using statistical calculation and the other method is by using Fast Fourier Transform (FFT).

2.4.1 Power Verification using Statistical Characteristic

Power analysis attacks rely on the fact that power consumption of a cryptographic devices depend on the operation they perform and on the data they process. Therefore, it is important to verify the logic circuit that fulfils some minimum standards for secure logic implementation, such as the logic immunity for counteracting power analysis attacks. In this case, the author has verified the proposed logic from the view point of the instantaneous power transitions due to the probability of input-output transitions. The variations of energy dissipation by the individual logic styles (Inverter/Buffer, NAND/AND and XNOR/XOR) and their implementation in LSIs are calculated, to validate the merit of the proposed logic circuit in comparison with the conventional secure logic styles. The simple statistical calculation parameters that the author has utilized are normalized energy deviation (NED) and normalized standard deviation (NSD) [74], as formulated in the following equations:

$$NED = \frac{E_{max} - E_{min}}{E_{max}}, \quad (2.23)$$

where E_{min} and E_{max} are the minimum and maximum values of energy data samples, respectively. The value of NED should be as low as possible, ideally, expected to be zero value. However, it is pretty difficult to realize in practice, due to the physical leakage power of the CMOS that we have delved previously.

Then, an average power is calculated as:

$$\overline{E} = \frac{1}{n} \sum_{i=E_1}^{E_n} E_i. \quad (2.24)$$

This equation is the average of the energy dissipation over each respective transition, where n is the total number of energy transition (e.g., Inverter has 4 data transitions, AND or XOR has 16 data transitions), and the E_i is the i th number of energy transition being calculated. It is important to note that, the energy E values discuss

here are from SPICE simulation values that calculated by using this equation $E_{diss} = \int_0^T V_{pc}(t)I_{pc}(t)dt$.

Furthermore, the energy variance or standard deviation is estimated as follows:

$$\sigma_E = \sqrt{\sum_{i=E_1}^{E_n} (E_i - \bar{E})^2 / n}. \quad (2.25)$$

The standard deviation essentially reflects the width of power distribution. The wider the distributions of transitional energy dissipations, the wider the standard deviation. This phenomenon represents the vulnerability of the logic circuit. Therefore, this value should be as small as possible (nearly zero).

Then finally, the normalized standard deviation is define as

$$NSD = \sigma_E / \bar{E}, \quad (2.26)$$

which is to measure the resistance against power analysis attacks. Notable information from this NSD property is that, if we achieve extremely low value of NSD, or if we convert it into a percentage unit, it is expected to be below 5%. If so, then this circuit is classified as robust secure logic against DPA attacks.

The author usually measures NED and NSD which indicate the ability of the logic against power analysis attack. These parameters indicate how the consumed energy is more constant for different input transitions, only for small values which obtained by these parameters.

Moreover, the objective of this research is to design a robust secure logic that consumes constant power for every power clock cycle, and to reduce the average energy as low as possible. Therefore, the author of this dissertation has defined a figure of merit (FoM) as follows

$$FoM = \sigma_E \bar{E}. \quad (2.27)$$

2.4.2 Power Verification using Fast Fourier Transform

Another approach to check and validate of stability of uniformly power transitions of a CMOS logic circuit in analog level using SPICE simulator is the frequency spectrum of the fast Fourier transform (FFT). The merit of this power simulation model is that, by simulating a logic circuit in a certain speed of clock frequency during a certain period of time, there should be only a single spectrum appears exactly at original frequency. Let us say, if the original frequency of the clock speed

is f_0 , then, the frequency spectrum below this f_0 , such as $f_0/2$, $f_0/3$, $f_0/4$, etc., should not appear. If not, then the logic circuit is consuming various power values whenever the data are flipped.

A fast Fourier transform (FFT) is an algorithm to compute the discrete Fourier transform (DFT) and its inverse in a very efficient computation time. The FFT is a mathematical method for transforming time domain into frequency domain. It operates by decomposing an M point time domain signals into M time domain signals that each composed of a single point. The second step is to calculate the N frequency spectra corresponding to these M time domain signals. Lastly, the N spectra are synthesized into a single frequency spectrum.

Time Step using FFT

In this sub-section, the time-step of each data point out of M time domain signals are defined as follows:

$$\Delta t = \frac{t_2 - t_1}{2M}, \quad (2.28)$$

where t_1 is the initial simulation time, t_2 is the simulation end time, M is the total number of data plotted when t_2 is ended. Based on the Equation (2.28), if, the sampling time of $T_s = \Delta t$, then, the sampling frequency f_s becomes

$$\Delta t = \frac{1}{f_s}. \quad (2.29)$$

According to FFT schematic model in Fig. (2.7), the initial frequency is 0, therefore, the frequency increments of Δf for $M - 1$ output data point becomes the following equation:

$$f_{N-1} = \frac{f_s}{2} - \Delta f \quad (2.30)$$

$$\Delta f = \frac{f_{M-1}}{M - 1} \quad (2.31)$$

$$\Delta f = \frac{f_s}{2M} \quad (2.32)$$

Then, Equation (2.28) can be rearranged as following equation

$$2M = \frac{t_2 - t_1}{\Delta t}. \quad (2.33)$$

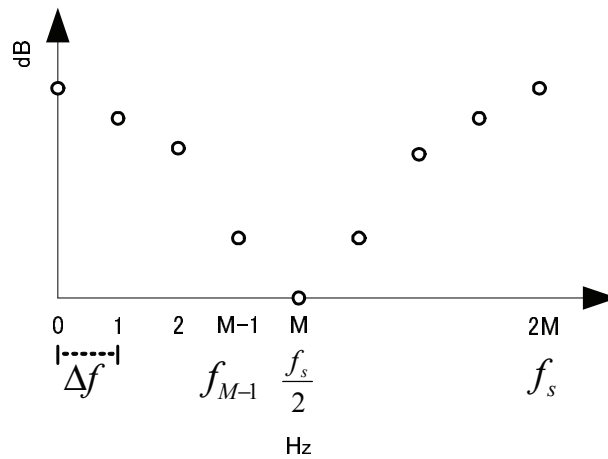


Figure 2.7: FFT schematic model.

Substituting the Equations (2.29) and (2.33) into Equation (2.32), we can achieve an FFT frequency increment of Δf which decided by initial time t_1 and end time t_2 , as expressed in following equation:

$$\Delta f = \frac{1}{t_2 - t_1}. \quad (2.34)$$

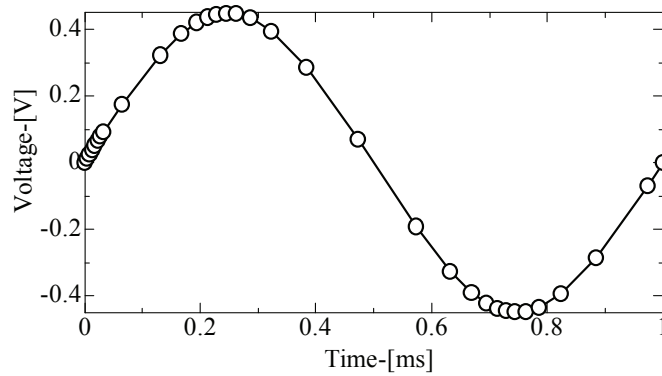


Figure 2.8: LTspice transient analysis.

The aforementioned formulas dictate the interval time required to plot each data of M string. This actually reveals the irregular data transition shown in Fig.2.8, since it is invisible in SPICE simulator to graphically observe the actual time of each data point.

Frequency Increment using DFT Transient Analysis

In this sub-section, the frequency increment of each FFT data in SPICE simulation using discrete Fourier transform (DFT) equations are as follows:

$$H(\omega) = \int_{-\infty}^{\infty} h(t)e^{-j\omega t} dt \quad (2.35)$$

$$H(\omega - \omega_{0s}) = \int_{-\infty}^{\infty} h(t)e^{-j(\omega - \omega_{0s})t} dt \quad (2.36)$$

Equation (2.36) is used if there exists initial frequency of ω_{0s} .

$$h_s(t) = h(t)e^{j\omega_{0s}t} \quad (2.37)$$

$$H_s(\omega) = \int_{-\infty}^{\infty} h_s(t)e^{-j\omega t} dt \quad (2.38)$$

Then, the Equations (2.37) and (2.38) are used to define real part and imaginary part of the $h(t)$ function as the following Equations (2.39) and (2.40).

$$\text{Re}H_s(\omega) = \int_{-\infty}^{\infty} h(t) \cos(\omega - \omega_{0s})t dt \quad (2.39)$$

$$\text{Im}H_s(\omega) = \int_{-\infty}^{\infty} h(t) \sin(\omega - \omega_{0s})t dt \quad (2.40)$$

Observing the irregular data points of the sine wave in Fig. 2.8, then the parameters in equation (2.41) are applied to calculate the real and imaginary part as shown in the following equations (2.42) and (2.43).

$$h_k = h(t_k), H_{sn} = H_s(\omega_n) \quad (2.41)$$

$$\text{Re}H_n = \sum_{k=1}^{K-1} h_k \cos[(\omega_n - \omega_{0s})t_k](t_k - t_{k-1}) \quad (2.42)$$

$$\text{Im}H_n = \sum_{k=1}^{K-1} h_k \sin[(\omega_n - \omega_{0s})t_k](t_k - t_{k-1}) \quad (2.43)$$

where, $n = 0, 1, 2, \dots, N-1$.

From equations above, we consider the ω_n as an regular interval, then, frequency increments of Δf becomes the following Equation (2.44).

$$\Delta f = \frac{f_{N-1} - f_0}{N} = \frac{f_W}{N} \quad (2.44)$$

From equation (2.44) the following equations are defined

$$f_W = f_{N-1} - f_0, f = \frac{\omega}{2\pi}, \quad (2.45)$$

$$f_n = n\Delta f. \quad (2.46)$$

Furthermore, time steps from Equations (2.42) and (2.43) become Equation (2.47); where, ω_{0s} consider as 0, then the sampling frequency becomes equation (2.48).

$$\Delta t = \frac{t_{K-1} - t_0}{K} = \frac{1}{f_s}. \quad (2.47)$$

$$f_W \leq f_s. \quad (2.48)$$

Finally, by substituting Equations (2.44) and (2.47) into (2.48), we now define Equation (2.49).

$$\Delta f = \frac{f_W}{N} \leq \frac{K}{N(t_{K-1} - t_0)} \quad (2.49)$$

Equation (2.49) shows how often the data transitions happen regularly during a certain frequency range.

2.5 Summary

The author underlines several important notes of this chapter as follows:

- To ensure the long-battery-life for battery-powered embedded cryptographic devices, the CMOS power consumption need to be highly considered.
- Adiabatic logic technique is a promising technique that can guarantee the efficiency of power usage. The minimum energy point in relation with the logic speed of any logic circuit need to be identified for effective design and implementation in appropriate devices.
- Peak supply current of an adiabatic logic is significantly low if compare to the CMOS logic style. Hence, a proper secure logic design using adiabatic switching technique can diminish the information leakage in SCA attacks.
- It is important for logic designers to formulate power simulation model in order to justify the resistivity, reliability, and applicability of the secure logic implementation.
- Logic verification methods are necessarily important to validate the effectiveness of the logic circuit for secure logic implementation.

Chapter 3

Survey on Side-Channel Information Leakage

3.1 Overview

A general overview of cryptanalysis has been prescribed in Fig. 1.4. In this survey, the author limits himself to only deal with the side-channel analysis (SCA) attacks implementation. After having an insightful view of the SCA attacks, the author will continue to elaborate the SCA countermeasures at cell level. Careful investigations on the existing logic styles, such as single-rail and dual-rail vulnerability will be emphasized. Moreover, a compilation of well-known secure logic styles that literately available in secure logic cell library, which specifically designed for resistance against SCA attacks will be discussed. The importance of this survey is to investigate, analyze and identify the existing secure logic problems and their merit as well, for comparative study with the proposed logic in this dissertation.

3.2 Side-Channel Analysis

The nature of any computation system, information processing or any circuit operation can be defined as $\text{Input} \rightarrow \text{Process} \rightarrow \text{Output}$. The output composes of two component: (1) The main-channel which is usually the expected/intended results, and (2) The side-channel, where the unintended/undesired outputs that occur during and after processing. The side-channel may include energy or information leakage in many different forms. In the context of information security, if the computation/processing is being performed, it is expected to be processed in a secure channel. Whilst the processing peripheral devices may not definitely keeping the information

under processed in secure way due to a lot of factors, such speed, power, surrounding temperature (heat), radiation, etc. Acquiring some essential information from side-channel leakage by irrelevant party is known as side-channel analysis (SCA). In the cryptanalysis field in the current decade, SCA attacks are the most threatening type of implementation attacks that frightening the modern crypto devices. These attacks are based on physical implementation attacks that gain information via physical properties of the device, such as computation time analysis, power consumption analysis and electromagnetic emission analysis. This section will illustrate in detail of each of the aforementioned SCA elements.

3.2.1 Timing Attacks

Timing attack is the first type of SCA attacks that was defined by Paul Kocher in his seminal work in 1996 [41]. This attack was firstly implemented to gain the secret key of the Diffie-Hellman [23], RSA [24] and DSA [25] algorithms. The computation of cryptographic algorithms in the dedicated hardware implementations often requires some times to finalize the operations from a given input queries to the intended output results. The computational time differences from input to output data depends on the complexity of the algorithm itself, such as square-and-multiply algorithm used in modular exponentiation. For example, the simple modular exponentiation in Diffie-Hellman and RSA algorithms for private key operations consist of computing $R = y^x \bmod n$, where n is publicly known and y can be found by an eavesdropper. The aim of the attacker is to find private key of x by computing $y^x \bmod n$ for several values of y , while the attacker has knowledge about the value of y , n and the required computation time; whereby, the corresponding timing measurement can be observed, as: $T = e + \sum_{i=0}^{w-1} t_i$, in which w is bits length of the key x , t_i is the time required for the multiplication and squaring steps for bit i and e includes measurement error.

The discussion and results of Kocher were quite theoretical, and more ideal posture of how the attack will be in the implementation. The first practical implementation of the timing attack was conducted by Dhem *et al.* as reported in [42]. They developed the basic ideas of Kocher, and they realized in practical implementation that was able to break a 512-bit key in a few minutes, as reported. The attack was implemented in CASCADE smart card, in which, the experimental results shown that by collecting much higher sample of timing measurements can lead to very high success rate of secret key recovery.

More successful extensive work of the timing attacks has been extended to steal the secret-key in the web servers, such as in open-source implementation of the secure socket layer (OpenSSL) [43], [44]. The timing data were procured by measuring the execution time of part of the transport layer security (TLS) handshakes between attacker client and OpenSSL's own TLS server, where the server provide an elliptic curve digital signature algorithm (ECDSA) on a number of exchange messages. This timing information are used to mount a lattice attacks¹ that exploits the vulnerability and recover the ECDSA private key from given small number of signature along with the timing data.

Brief information on timing attacks above reveals that the cryptographic algorithm, protocols, systems and devices need some modification to prevent timing attack. Based on author's observation from literature, timing analysis technique reveals that although inexpensive to be implemented, it requires some knowledge about detailed design of the target system in order to perform this attacks successfully.

3.2.2 Power Analysis Attacks

The basic idea of power analysis (PA) attacks is to reveal the secret key of cryptographic devices via probe measurement technique. Depend on the type of PA attacks implemented, basically, the attacker measures one or multiple power traces from the same device, and then, statistically analyzing the key dependent differences to reveal the secret key on the attacked device. The author dictates the definition of PA postulated by Kocher *et al.* [47] in 1999 that, the power analysis attacks exploit the fact that the instantaneous power consumption of cryptographic devices depends on the data it processes and the operation it performs. Paul Kocher introduced the information leakage on cryptographic hardware, which is previously may not be realized by cryptographers to maintain the secrecy of secret-key from cryptanalysis. The attack was implemented on symmetric algorithm of the data encryption standard (DES) which cannot be done by timing analysis attack. At the same year, the PA was formalized by Tomas Messerges *et al.* [48], through practical implementation of PA attacks on smart cards. They confirmed the work of Kocher

¹Lattice attack is an implementation attack based on the lattice reduction algorithm that proposed in LLL algorithm [45]. Some notably works using lattice attacks on digital signature (DSA and ECDSA) can be seen in [46].

that power analysis attacks are very effective and powerful that need to be addressed in order to provide maximum amount of security on cryptographic devices.

The author considers this section as the strongest motivation of this dissertation, and hence, it is necessary to delve more detail understanding on how the PA attacks work in the implementation. In recent decade, PA has gained increasingly important attentions from the cryptographic research communities to conduct research in this issue. More extensive works, improvements and even some new ideas on PA analysis have been realized. Therefore, the author wants to explain more detail in following subsections, about the simple and differential power analysis, correlation power analysis and leakage power analysis, respectively.

Simple Power Analysis

Simple power analysis (SPA) is a technique that involves directly interpreting power consumption measurement collected during cryptographic operation. In other words, the attacker tries to derive the key more or less directly from the single trace of power consumption along the function of time. SPA attacks often require detail knowledge about the implementation of the cryptographic algorithm that is executed by the device under attack. Furthermore, if only one power trace is available, usually complex statistical methods have to be used in order to extract the signal.

SPA attacks are useful in practice if only one or few traces are available for a given set of inputs. For example scenario where a consumer has to refill the gas tank of the car on a regular basis and always buys a similar amount of gas. A malicious smart card reader could record the power consumption of the card. In this way, the attacker could gather a couple of traces for similar plaintext. In the other cases where the IC-card is used for multipurpose, that require more measurement sample for accurate statistical analysis, then SPA will not work, instead the differential analysis style must be used

Further work on SPA attack was conducted by Mangard in 2002 [49], a year after the announcement of AES (a successor of DES). He utilized the SPA attack to the AES key expansion on 8-bit and 32-bit processors that it can succeeded to reveal the 128-bit secret key on a device that implemented in an unprotected AES implementations.

Differential Power Analysis

Differential power analysis (DPA) is the most powerful attack which relies on statistical test to isolate a signal of interest from noise and complex power signal on a device. The goal of DPA attack is to reveal the secret key of cryptographic devices based on the large number of power traces that have been recorded while the devices encrypt and decrypt different data blocks. The power of DPA lies on its ability to discover useful information whenever there is a correlation between power traces and processed data. Although the attacker does not need to know details of how an algorithm is programmed, they still must know which algorithm to attack since the differential attack requires a known model of cipher behavior. Using a DPA attack, the attacker can discover a group of key bits at once and dramatically reduce the key search space.

In DPA attack, the attacker first records a large number of power traces for thousands of encryptions, for instance, he observes m encryption operation (*i.e.*, $m = 10^3$ DES operations), and store the ciphertexts in an array C_i where i is the index of the operation, for instance $i = 0, 1, 2, \dots, 999$. The power traces are recorded in an array $S_{i,t}$ where t is the index of the time sample recorded by the high speed oscilloscope, for instance, they measure from 0–9,999 for 10^4 data samples. Then, for each ciphertext value C and a single bit b of the intermediate value (*i.e.*, $b = 0, 1, 2, \dots, 31$) of the computation, the attacker chooses a partition function $D(C, b, K_i)$, taking input of a few key bits K_i . The aim is to select the value of a single-bit result (perhaps the most significant bit) of the sub-computation performed on K_i and C . If K_i is incorrect, the probability of $D(C, b, K_i)$ being correct will be about 50%. If K_i is correct, the probability will be 1. $D(C, b, K_i)$ is a definition model and can be accurately calculated by a simulation. The values of $D(C, b, K_i)$ are used to divide both average of the real recorded traces and the power consumption model traces into two groups: $D = 0$ and $D = 1$. The attacker then takes the mean of each group and subtract the two means to get the power difference for correct key searching as follows:

$$T(t, C, b, K_i) = \frac{1}{|D_0|} \sum_{S_x \in D_0} S_i(t) - \frac{1}{|D_1|} \sum_{S_x \in D_1} S_i(t), \quad (3.1)$$

where $|D_n|$ is the number of members in the set D_n . The $T(t)$ is the differential trace, a function of the difference of the two means over time (all other C , b and K values being constant for each run). When plotted, the trace will be flat noise, except if the key hypothesis K_i is correct, which should show as a peak at the time instant b

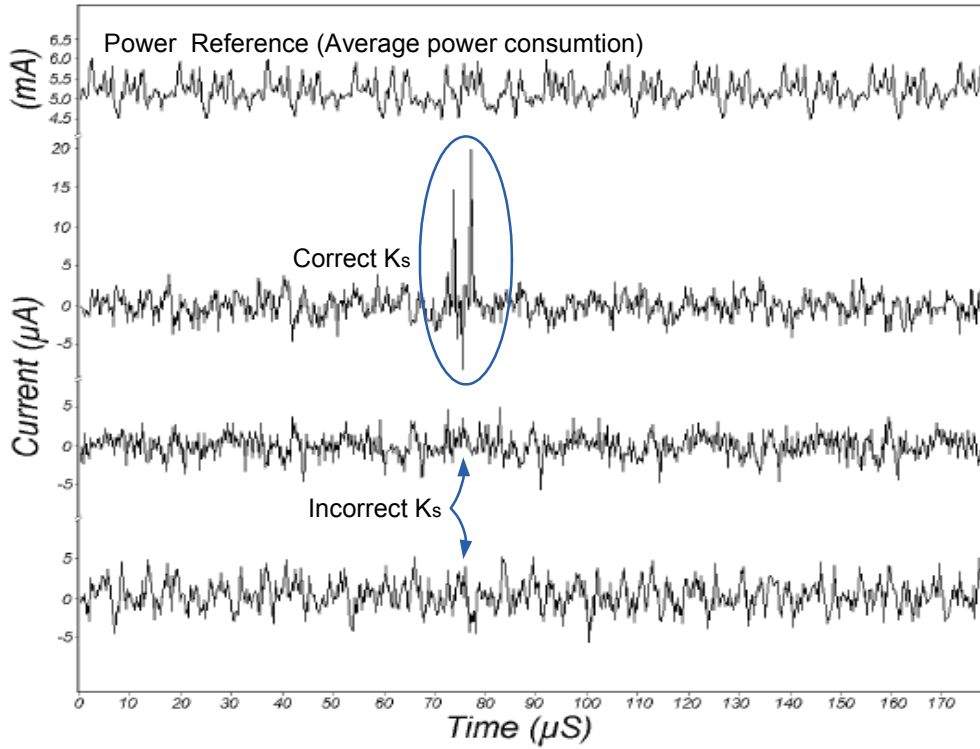


Figure 3.1: DPA traces: (a). One correct and two incorrect with power reference; (b). Quantitative DPA measurements (adopted from original article [47]).

is used, which obviously depicted in Fig. 3.1. This figure shows four traces prepared using known plaintexts entering a DES encryption function on smart card. The top trace is the reference power trace showing the average power consumption during DES operations. Below are three differential traces, where the first was produced using a correct guess for K_i . The lower two traces were produced using incorrect values for K_i . These traces were prepared using 1000 samples ($m = 10^3$).

Based in the idea of DPA technique, attractive research works on DPA attack implementation have been expanded. Just in case, if the readers want to know more about DPA attacks, the author referred to few papers out of uncountable published works in this research area [51]– [55].

Correlation Power Analysis Attacks

Correlation power analysis (CPA) attack is the enhancement of DPA attacks, which deduced the correct key by using correlation coefficient of statistics between the power traces and the values of intermediate result of the key guess. CPA was first introduced by Eric Brier *et al.* [56], in which the implementation was also targeted

to reveal the secret key of DES and AES algorithms. CPA attack considers that the power dissipation of an operation at a specific time is proportional to the hamming weight of the processing data. Suppose W is the random variable of the measured power and H is the random variable of the HW of the data D and R is unknown, but not necessarily zero. The basic HW model for the data dependency can be described as $HW = aH(D \oplus R) + b$, where a is a scalar and b is the random variable for all the other power consumption of a chip. More accurate power model such as HW model can be used if some reference states are predictable.

CPA was implemented according to most familiar statistic measure of Pearson correlation, as following estimation [56]:

$$\rho_{WH}(R) = \frac{N \sum W_i H_{i,R} - \sum W_i \sum H_{i,R}}{\sqrt{N \sum W_i^2 - (\sum W_i)^2} \sqrt{N \sum H_{i,R}^2 - (\sum H_{i,R})^2}}, \quad (3.2)$$

where N is the number of power traces W_i , and N associated to random data words M_i , R is reference states. Thus, the predicted Hamming distance $H_{i,R} = H(M_i \oplus R)$. The summations of equation above are taken over N samples ($i = 1, 2, \dots, N$) at each step within the power traces $W_i(t)$.

Equation (3.2) shown that it is only necessary to only have certain number of measured power traces and predicted HD for statistical calculation resources, while in DPA technique, the attacker is required to know several resources in order to implement the attack successful. For further analysis and implementation of CPA attacks, the author refers the readers to [50], [57], [58].

Leakage Power Analysis Attacks

Earlier PA techniques, such SPA, DPA and CPA, all these attacks are exploiting the dependence of dynamic power consumption on the input of cryptographic algorithm. In contrast, the leakage power analysis (LPA) [34], [35] was recently implemented by measuring the static (leakage) power, while the device is in standby mode (transistor off state or cut off region). In the current CMOS nano scale technology, the dynamic power is no longer the dominant factor of total CMOS power consumption, due to the much faster increase of static power when the gate size is getting thinner proportional to the shrinking of technology. The procedure of LPA implementation is as simple as CPA in previous section.

The confirmation work in [35] has proven the effectiveness of the LPA attacks. They have tested secure logic styles for countermeasures against DPA/CPA attacks that none of those logic styles can counteracting the LPA attacks using 65 nm

and 90 nm CMOS process. The DPA inventor, Paul Kocher suggested that one of the methods to prevent DPA attacks is to introduce noise into power consumption measurement. However, LPA proves that on-chip noise and process variations are all contribute to the effectiveness and applicability of LPA attack in practice.

The author of this dissertation fully aware that, his results perhaps irrelevant to CPA attack technique, because the CMOS parameter 180 nm process used in this work is still dominated by dynamic power, where leakage power is fully eliminated by the proposed logic. However, in one hand, it is important information for future research direction that probably will utilize the latest nano meter technology. In the other hand, to direct the readers, who are interested in secure logic design to put in account in order to design more robust secure logic style that can immune to any kind of attack techniques in the current technology.

3.2.3 Electromagnetic Analysis Attacks

In general sense, the terms of electromagnetic analysis (EMA) covers all forms of electromagnetic radiations, including radio-wave, microwave, infra-red (thermal), optical and high frequency radiations, etc. In electrical engineering, the electromagnetic emission is commonly used, that is related to electromagnetic interferences (EMI) and electromagnetic compatibility (EMC), by which, this subject is the electrical/electronic circuits that emit electromagnetic radiation when electrical power is applied to the circuit.

In regards to the cryptographic security, the EMA is one of the SCA attacks that practiced to reveal the secret key by analyzing the information leakage in the secure devices causes by electromagnetic emanation. As for concrete meaning of EMA in this research filed is that, the current that flows during the switching of the CMOS gates, causes a variation of the electromagnetic field surrounding the chip that can be monitored by inductive probes which are particularly sensitive to the related impulse.

The method of data sampling is by using antenna sensor, such as loop antenna around the secure chip area while the chip is under operating. By then, the electromotive force across the sensor using Lentz's law²; thus, it will relate to the variation

²Lentz's law is named after the German scientist Heinrich Friedrich Emil Lenz (1804–1865). Lenz's law was invented in in 1834, based on Faraday's law of electromagnetic induction “ An induced electromotive force (emf) always gives rise to a current whose magnetic field opposes (-)

of magnetic flux as follows: $V = -\frac{d\phi}{dt}$ and $\phi = \oint \vec{B} \cdot d\vec{A}$, where V is the probe's output voltage, ϕ is the magnetic flux sensed by probe, t is the time, \vec{B} is the magnetic field and \vec{A} is the area that it penetrates [59].

The author does not know very well who was the first realized the EMA implementation, but the idea of using EMA in SCA attacks was hinted by Paul Kocher who introduced the DPA attack. There are two types of electromagnetic emission attacks: the simple electromagnetic analysis (SEMA) and the differential electromagnetic analysis (DEMA) attacks. The SEMA and DEMA attacks are using similar principle as done by SPA and DPA, such as, the SEMA attack uses the side-channel information from one measurement directly to determine the secret key, while the DEMA requires many measurement traces in order to filter out noises. This research field has attracted more attention to the cryptographic researchers to evaluate and analyze the security metric of the devices using EMA attacks, and hence, the author refers some related literature list for further reading [60]– [66].

3.3 SCA Countermeasures at Cell Level

Hardware countermeasure against SCA attacks means to designing an appropriate hardware architecture that could be resistive against the SCA attacks described in the previous section. The design flow of crypto chips generally starts with high level design capture, a design of digital circuit that will be used in the hardware system. This is the backbone step, a barometer to measure the ability of the entire circuit system suitability and resistivity for cryptographic hardware implementation.

When Paul Kocher *et al.* [47] introduced the DPA attacks, at the same time, they suggested also some techniques on how to prevent DPA and related attacks, such as reducing the signal size, balancing HW, balancing state transition (referred to HD), aggressive shielding in practice, introducing the noise into power consumptions and randomizing the execution time. Since then, it has been more than a decade, the cryptographers and circuit engineers have been doing a lot of efforts to strengthen the security of crypto devices. From the literature resources, the author observes that, mostly, the logic designers dedicate more effort in the cell level security that falls into two techniques: hiding and masking.

The author emphasizes that the fundamental issue of power analysis attacks on cryptographic devices is closely related to the electrical power consumption of an

the original change in magnetic flux”.

endpoint hardware. Therefore, a logic designed to hide or mask the data being processed should be highly considered.

Numerous studies on hiding intermediate values at cell level have been published, but most of these employed a conventional CMOS logic style, which means the devices are still susceptible to DPA and DEMA attacks because of their large fluctuation of energy dissipation that leads to increasing the signal size.

Over the last few years, there have been several reports pertaining to cell-level uses of dual-rail (DR) pre-charged logic, such as sense-amplifier-based logic (SABL) circuits and their implementations [69]– [71], where the input logic structure is designed to balance all internal node capacitances for constant power consumption under all input conditions at every clock cycle. Furthermore, the simple or wave dynamic differential logic (SDDL/WDDL) [72] was designed, which has achieved an important reduction in the power variation for both ASIC and FPGA, but their drawback are the increased area, computation time, and power consumption.

It is interesting to know that, the presence of DPA attacks has encouraged the relevant research community with a very lively competitive works, aiming to create a better protected circuit system to thwart any kind of adversaries. For instance, evaluation study of the gate's resistance to DPA, the time-varying power traces of the circuit element was conducted by Lee *et al.* in [73]. They proposed a symmetric discharged logic (SDL) and comparison was done with the SABL. The results in [73] shown that SDL reduced power differences to immeasurable small values compare to the SABL and the other conventional logic styles. Furthermore, an enhanced SABL was developed into the well-known three-phase dual-rail pre-charge logic (TDPL) [74] to unbalance load conditions, thus allowing a semi-custom design flow without any constraint on the routing of the complementary wire. Moreover, an asynchronous dual-rail gate design has been proposed [83] for balancing power, requires no capacitance matching of data outputs, and tolerates process variability in the routed interconnect between gates. Additional work on the masking approach was proposed for randomizing intermediate values that are processed by cryptographic devices [79].

Some of innumerable well designed secure CMOS logic styles have been published as proceedings, periodicals and technical reports [80]– [86], in which the author could not able to investigate each of them in this dissertation.

The fact is that, all the existing conventional CMOS logic techniques aforementioned have been successfully implemented and practical security evaluation were done. However, most of the real crypto-devices were designed in the conventional

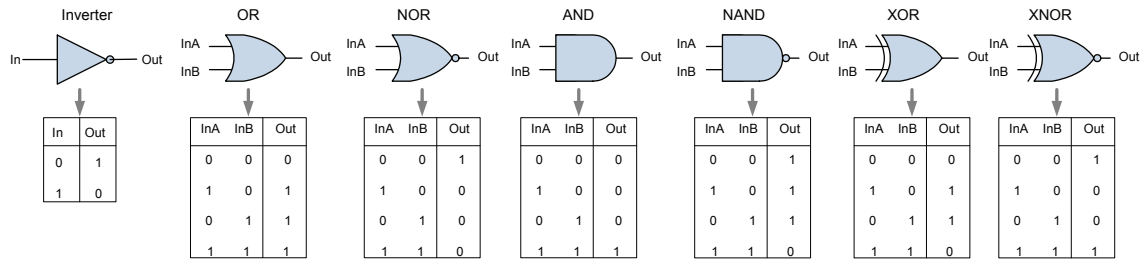


Figure 3.2: Basic digital logics and their respective truth table.

CMOS logic style, which means they are highly power consuming, and hence exist some detectable high-peaks power trace that make the system presumably vulnerable to a practical measurement of power and electromagnetic analysis.

In this section, the author will explore and investigate some of the relevant CMOS logic styles that implementing two techniques aforementioned (hiding & masking). The term of “conventional logic” will be used more often. It intends to distinguish the previously published and referred logic styles with the proposed logic one. Moreover, the CMOS logic techniques (with supply voltage of constant V_{dd}) will cover in this section, whilst the conventional secure adiabatic logic styles will be placed in chapter 4, including the proposed logic to arrange better comparison data of individual logic circuits.

Starting from this section, the author will explain some basic digital logic construction using CMOS cells, such as Inverter Logic, NAND/AND logic and XNOR/XOR logic. These basic digital logics’ functions will be tested by LTSpice simulation using an $0.18\mu m$, 1.8 V standard CMOS process technology. The author will analyze the ability of the logic function against power analysis attacks from the view point of the instantaneous peak supply current differences for every possible input (or output) transitions.

In addition, the author also provides the basic digital logic symbols and their respective truth table for your convenience, as described in the Fig. 3.2. Some of these symbols will be used in the logic implementation, and hence, you will not spend extra time to search for the fundamental logic operation.

3.3.1 Hiding

The goal of hiding countermeasures is to make the power consumption of cryptographic devices independent of the intermediate values and independent of the operations that are performed. There are two essential approaches to achieve this independence: (1) To build devices in such a way that the power is randomly consumed (not depend on the input signals), (2) To build devices that consume an equal amount of power for all operation and for all data values. This means that the equal amounts of power are consumed in each clock cycle. Power constant in each logic cycle refers to the instantaneous power consumptions of a cell are similar (or same) in every clock cycle. To the best of author's knowledge, the logic styles with a constant power consumption are typically implemented in dual-rail pre-charge (DRP) logic style. The author proves it by re-simulating and analyzing all the logic styles in both single-rail (SR) and dual-rail (DR) logic techniques, and therefore the comprehensive discussion is presented in this section.

Imperfection of the Existing Logic Style

In this part, the author presents two types of the most commonly known and simplest CMOS logic circuits of both SR and DR logic styles; thus are SR static complementary CMOS (SR-scCMOS) and the DR differential cascode voltage switch logic (DR-DCVSL) [87]. In this section, the author aimed to clarify the Fig. 1.5 that early appeared in the objective of this study in Chapter 1.

- **SR-scCMOS:** As shown in Fig. 3.3(a), a static complementary CMOS (scCMOS), which is the most simplest and default inverter logic in the standard cell library used for security ICs only consumes energy from the power supply when its input has a 1→0 transition. At this transition, a high instantaneous supply current occurs, as can be observed in Fig. 3.3(c). This figure shown that during the 0→1 input transition, the energy previously store in the scCMOS output capacitance will be dissipated at PDN load; and it supposes to be no supply current flow from V_{dd} , yet there are still some small current flowing because of the short circuit during the PMOS and NMOS change the logic condition (simultaneously ON-state for short period of time).
- **DR-DCVSL:** In generic DR differential logic families, the circuits are operated in two outputs conditions: (1) When the input vector $\mathbf{IN} = (in_1, \dots, in_n)$ is the true vector of the switching function $Q(\mathbf{IN})$, node Q is disconnected

from the ground by the unique path of NMOS Differential Pull Down Network (DPDN) tree. (2) When $\mathbf{IN} = (in_1, \dots, in_n)$ is false vector of $Q(\mathbf{IN})$, the reverse function holds. The DCVSL inverter/buffer logic circuit is shown Fig. 3.3(b), including the internal equivalent RC model that performing the charging and discharging load connection.

In contrast to SR-scCMOS inverter circuit that performs unbalanced peak current transitions, in this DR-DCVSL technique, both $0 \rightarrow 1$ and $1 \rightarrow 0$ transitions produce balanced supply current trace. In the two cases of $0 \rightarrow 0$ and $1 \rightarrow 1$, the cell consumes only static power, which considerably very low power or event neglected in DPA attacks. Thus, the general power consumption values of both circuits hold that $(P_{0 \rightarrow 0} \approx P_{1 \rightarrow 1}) \ll (P_{0 \rightarrow 1}, P_{1 \rightarrow 0})$, which susceptible to Hamming Distance (HD) model in power analysis attacks (see Equation 2.21).

The basic differences between SR logic and DR logic (differential logic) for countermeasures against DPA attacks are illustrated in Fig. 3.4. Typically, a logic function, such as NAND gate of SR cell has two inputs A and B and one output q as shown in left of Fig. 3.4. One digit input of the NAND logic can perform one output in four possible transitions, such as $0 \rightarrow 0$, $0 \rightarrow 1$, $1 \rightarrow 0$ and $1 \rightarrow 1$ during one clock cycle. Hence, in total there exist 16 possible combinations of input transition that can occur during one cycle, as shown in Table. 4.1. Base on this table, the author will evaluate the merit of each individual logic styles to check and validate the logic's resistance against SCA attacks.

In cryptanalysis, the power analysis are split into two groups [79]. The first group contains all measurements, where $q = 0$ (or $0 \rightarrow 0$, $1 \rightarrow 0$) at the end of one cycle and the second group contains all measurements, where $q = 1$ (or $0 \rightarrow 1$, $1 \rightarrow 1$). By analyzing Table. 4.1 without complementary signals, the energy comparison of $q = 0$ and $q = 1$ for SR gates can be calculated as following :

$$\frac{3E_{1 \rightarrow 0} + E_{0 \rightarrow 0}}{4} \neq \frac{9E_{1 \rightarrow 1} + 3E_{0 \rightarrow 1}}{12} \quad (3.3)$$

Analyzing Table. 4.1 for differential logic gates, the probability of q , $\bar{q} = 1$ and q , $\bar{q} = 0$ can be calculated as following:

$$\frac{6E_{1 \rightarrow 0} + 10E_{0 \rightarrow 0}}{16} = \frac{10E_{1 \rightarrow 1} + 6E_{0 \rightarrow 1}}{16} \quad (3.4)$$

The attackers are normally calculates the means of the energies of both groups and subtract to each other, hence there is a side-channel information leakage for SR cells. It happens because the energy of $q = 1$ and $q = 0$ is not symmetric as shown in

Eq. (3.3), while for differential logic cells is counterbalanced for $q = 1$ and $q = 0$, as shown in Eq. (3.4). Therefore, a logic properly designed in DR logic style has ability to hide the correlation between data and power consumption.

The SPICE simulation results of the SR scCMOS (AND and XOR) and DR DCVSL (AND/NAND and XOR/XNOR) are depicted in Figs. 3.5 and 3.6, respectively. From this figures, we can observe that the circuits operation for logic inversion are correct, same as the logic truth table in Fig. 3.2. By comparing Figs. 3.5, 3.6 and Figs. 3.3, we may now realize that there are completely different story. For instance, there are no current flow at input $0 \rightarrow 0$ and $1 \rightarrow 1$ transitions for inverter level; however, the gate counts will increase when it comes to the dual-input logic inversions, and hence the different peak supply current spikes always occur for all input-output transitions. These circumstances are caused by the short circuit current and unbalanced load condition when PDN cells charge their states. Therefore, to investigate the logic circuit immunity for secure logic design implementation, it is better to investigate and perform the comparison at dual-input logic styles (for instance, AND or XOR logic circuit).

Table 3.1: Input transition of NAND gate and its complementary that can be performed during one complete cycle.

No.	SR & DR				DR Complementary Signals			
	$a_i \rightarrow a_{i+1}$	$b_i \rightarrow b_{i+1}$	$q_i \rightarrow q_{i+1}$	$E_i \rightarrow E_{i+1}$	$\bar{a}_i \rightarrow \bar{a}_{i+1}$	$\bar{b}_i \rightarrow \bar{b}_{i+1}$	$\bar{q}_i \rightarrow \bar{q}_{i+1}$	$E_i \rightarrow E_{i+1}$
1	0→0	0→0	1→1	1→1	1→1	1→1	0→0	0→0
2	0→0	0→1	1→1	1→1	1→1	1→0	0→0	0→0
3	0→0	1→1	1→1	1→1	1→1	0→0	0→0	0→0
4	0→0	1→0	1→1	1→1	1→1	0→1	0→0	0→0
5	0→1	0→0	1→1	1→1	1→0	1→1	0→0	0→0
6	0→1	0→1	1→0	1→0	1→0	1→0	0→1	0→1
7	0→1	1→1	1→0	1→0	1→0	0→0	0→1	0→1
8	0→1	1→0	1→1	1→1	1→0	0→1	0→0	0→0
9	1→1	0→0	1→1	1→1	0→0	1→1	0→0	0→0
10	1→1	0→1	1→0	1→0	0→0	1→0	0→1	0→1
11	1→1	1→1	0→0	0→0	0→0	0→0	1→1	1→1
12	1→1	1→0	0→1	0→1	0→0	0→1	1→0	1→0
13	1→0	0→0	1→1	1→1	0→1	1→1	0→0	0→0
14	1→0	0→1	1→0	1→1	0→1	1→0	0→0	0→0
15	1→0	1→1	0→1	0→1	0→1	0→0	1→0	1→0
16	1→0	1→0	0→0	0→1	0→1	0→1	1→0	1→0

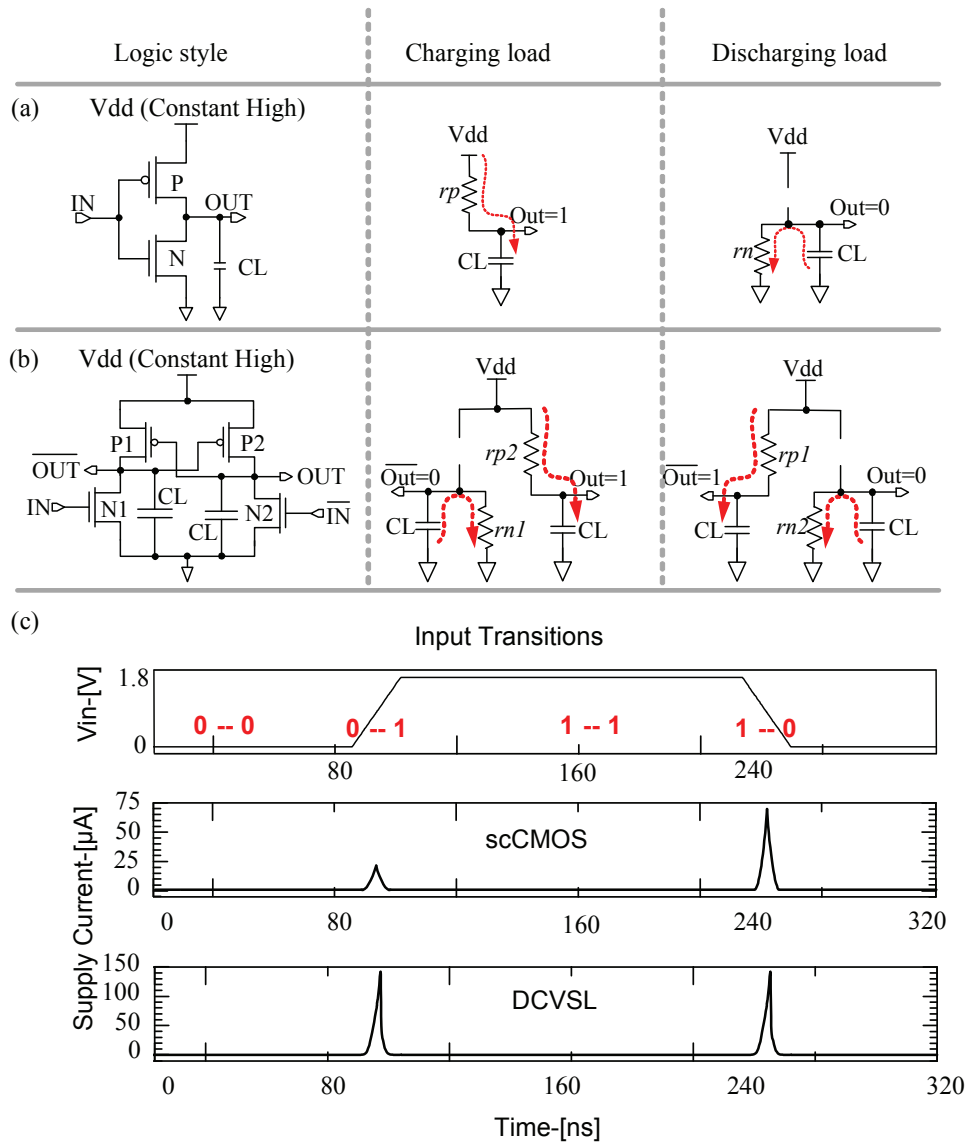


Figure 3.3: Existing CMOS logic problems; (a) scCMOS inverter and its equivalent RC model, (b) DCVSL inverter and its equivalent RC model, (c) Current signature based on the input transitions

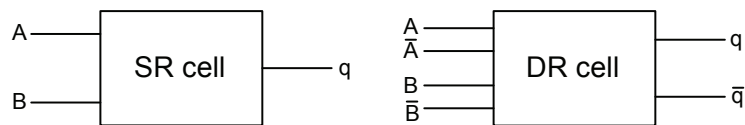


Figure 3.4: A 2-input SR cell and a corresponding 2-input DR cell.

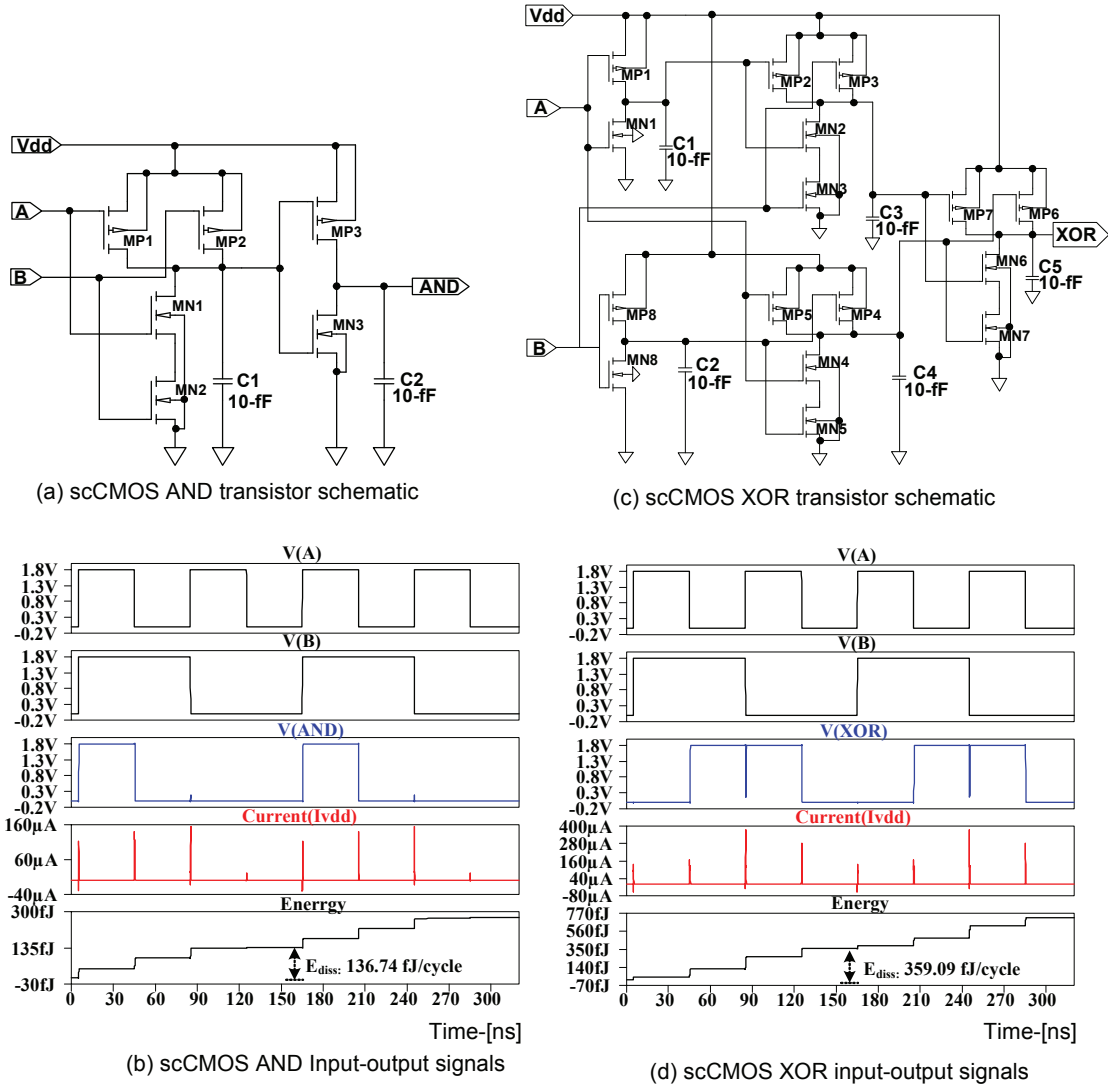


Figure 3.5: SR scCMOS for AND and XOR transistor schematics and their respective LTspice simulation results.

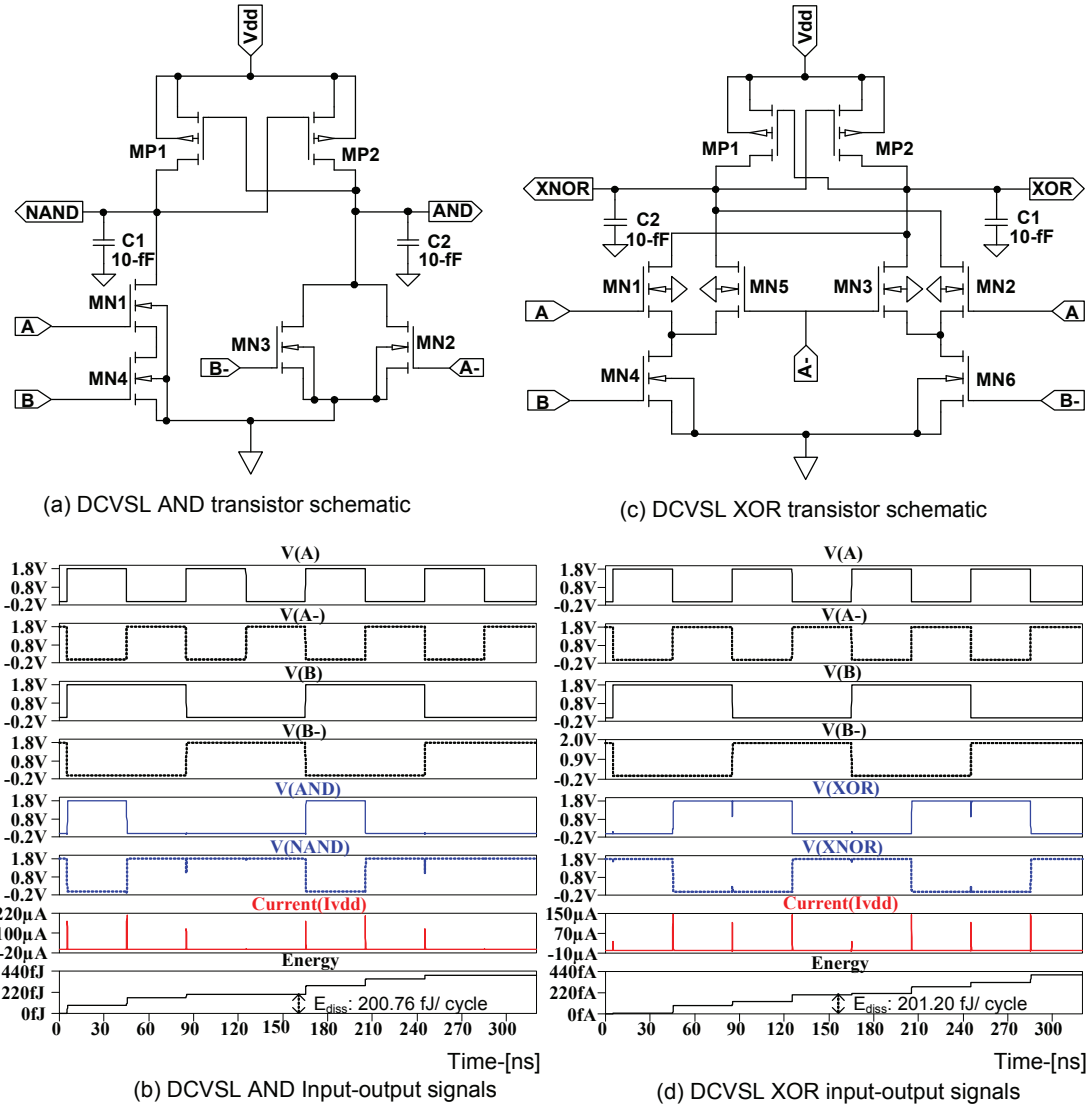


Figure 3.6: DR DCVSL AND and XOR transistor schematics and their respective LTspice simulation results.

Dynamic Logics

The basic construction of dynamic logic gate is shown in Figure. 3.7. The dynamic logics are always driven by a clock (clk), which operates in two phases: Pre-charge phase and Evaluation phase [67], [68]. When $\text{clk}=0$, M_p is *ON* and M_n is *OFF*, and under this condition, the pre-charge phase occurs, which drives $\text{Out} = 1$ and energy is stored in node capacitance C_L . On the other hand, if $\text{clk}=1$, M_p is *OFF* and M_n is *ON*, and this condition is called the evaluation phase. The output is conditionally discharged based on the input values and the pull-down network topology.

Logic function of dynamic NAND and XOR logic in SPICE simulation is shown in Fig. 3.8. This figure shown that, the logic operated in dynamic mode is suitable to reduce/avoid short-circuit current. However, the SR dynamic logic performs current-to-data dependencies. Every output signal goes from $0 \rightarrow 1$, it draws high current from power supply. From the view point of cryptanalysis, this logic has high susceptibility towards HD model.

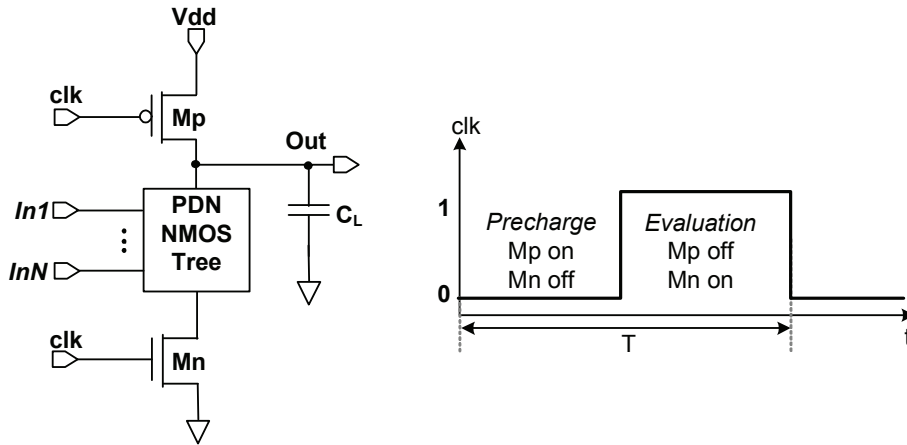


Figure 3.7: Generic dynamic logic (left) and its phases (right).

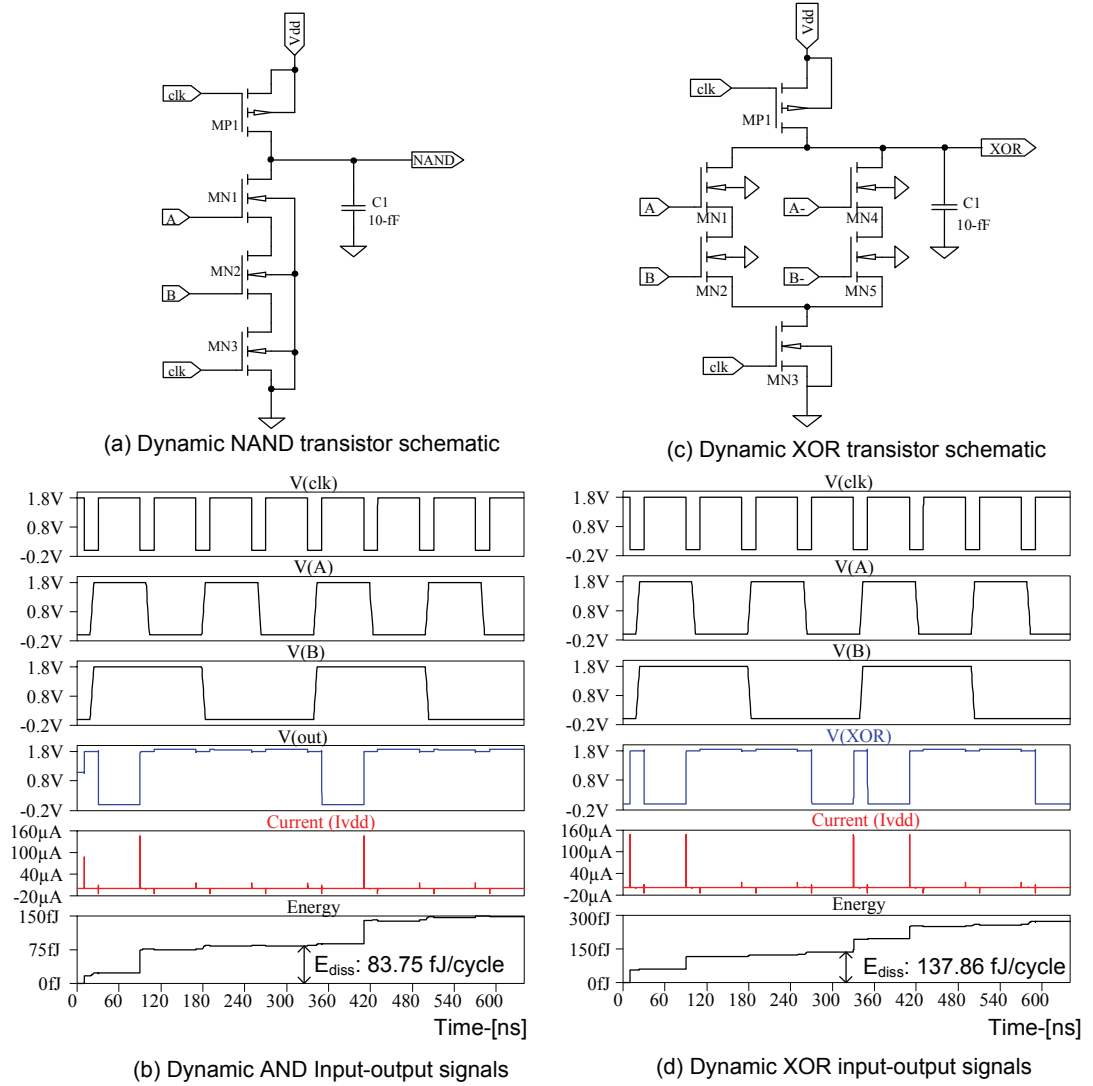


Figure 3.8: Dynamic logic: (a) NAND transistors schematic, (b) NAND input-output signals, (c) XOR transistors schematic, (d) XOR input-output signals.

Pre-Charged Dual-Rail Logic

Countless research works on secure logic design and application in cryptographic implementation such as smart card have been periodically published by scientific researchers. To the best of author's knowledge, the most of those secure logic cells are constructed in DR pre-charge logic, such as sense amplifier based logic (SABL) in [69]–[71], wave dynamic differential logic (WDDL) [72] and the three-phase dual-rail pre-charge logic (TDPL) [74].

In this section, the author of this dissertation describes the SPICE simulation results of the SABL and the TDPL, analyzes and compares the resistance of the logic ability to withstand power analysis attacks.

Sense Amplifier Based Logic (SABL)

The SABL was designed based on DR pre-charged logic style, where its cells are specifically designed to have constant internal power consumption independent of the processed logic values. A characteristic of SABL circuit is that all cells are connected to clock signal and all of them are pre-charged simultaneously. Figure 3.9 shows the transistor schematic of generic SABL cell. In order to achieve the same transition at the output nodes *out* and \overline{out} in each clock cycle, the SABL cell consist of the differential pull-down network (DPDN) and the cross-couple CMOS latch as indicated in Fig. 3.9. Same as the dynamic logic operation, the SABL also operates in two phases as follows (see Fig. 3.9):

Evaluation phase : At the end of pre-charge phase, all internal nodes of SABL cell have been set to 1. At the onset of the evaluation phase (*i.e.*, the clock signal switches from 0→1), the PMOS transistor Pr1 and Pr2 are turned off, and the NMOS transistor Eval is turned on. At this time, the evaluation phase is occurred based on the input logic state. The input rails of SABL are designed in specific way that during the evaluation, all internal node capacitances and one of the outputs are discharged to ground through Eval cell.

Precharge Phase: At the onset of pre-charged phase, the clock signal is switched from 1→0. The NMOS transition Eval is turned off, which disconnect the inputs node to ground. Simultaneously the PMOS Pr1 and Pr2 are turned on. Thus the both output nodes are pre-charged to high and all internal node capacitance are re-charged at the same time. As a result, the huge energy is consumed at this phase. Until the end of pre-charge phase, the clock signal changes the transition from 0→1 again, and then, the next cycle phases are occurred again.

Input cells construction of SABL: The DPDN of SABL cells are built in such way that all input nodes are connected to each other to maintain constant charging and discharging capacitive load, as shown in Fig. 3.10(b). Observing more closely to this figure, the most commonly used DR-PDN (universal DR-PDN) in Fig. 3.10(a) indicates that there exists unbalanced load (some floating capacitance) for each input condition. In none of the four different events, the same combination capacitances have to be charged. In contrast to universal DR-PDN, the connection of SABL cells exhibits free of floating capacitance. By making it possible, the NMOS transistor MN1 must be attached, and the MN2 is connected to the node between MN3 and MN4. In other word, both wires of every complementary input cells must be connected to the same number of gate terminal of transistors with identical parameters. This ensure that the capacitances of complementary inputs of SABL cells are pair wise balanced. As a result, the SABL cells consume very constant supply current (see current I_{vdd} in Fig. 3.11) at every clock signal falls down, which shown the current-to-data independent. The input-output logic operation of the complementary AND/NAND and XOR/XNOR and their respective current traces can be conformed in Fig. 3.11.

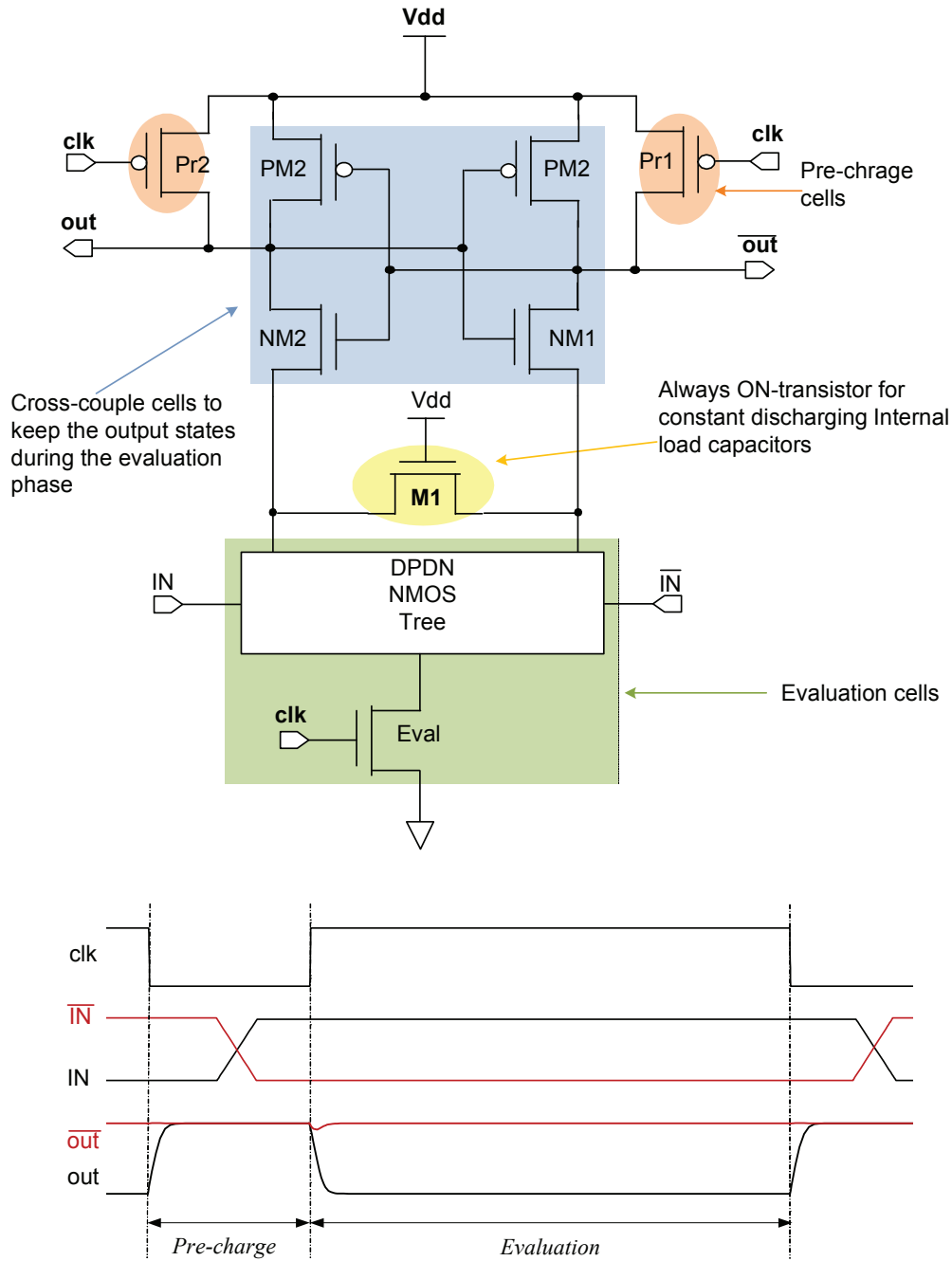


Figure 3.9: Transistor schematic of generic SABL cells and its ideal input-output signals.

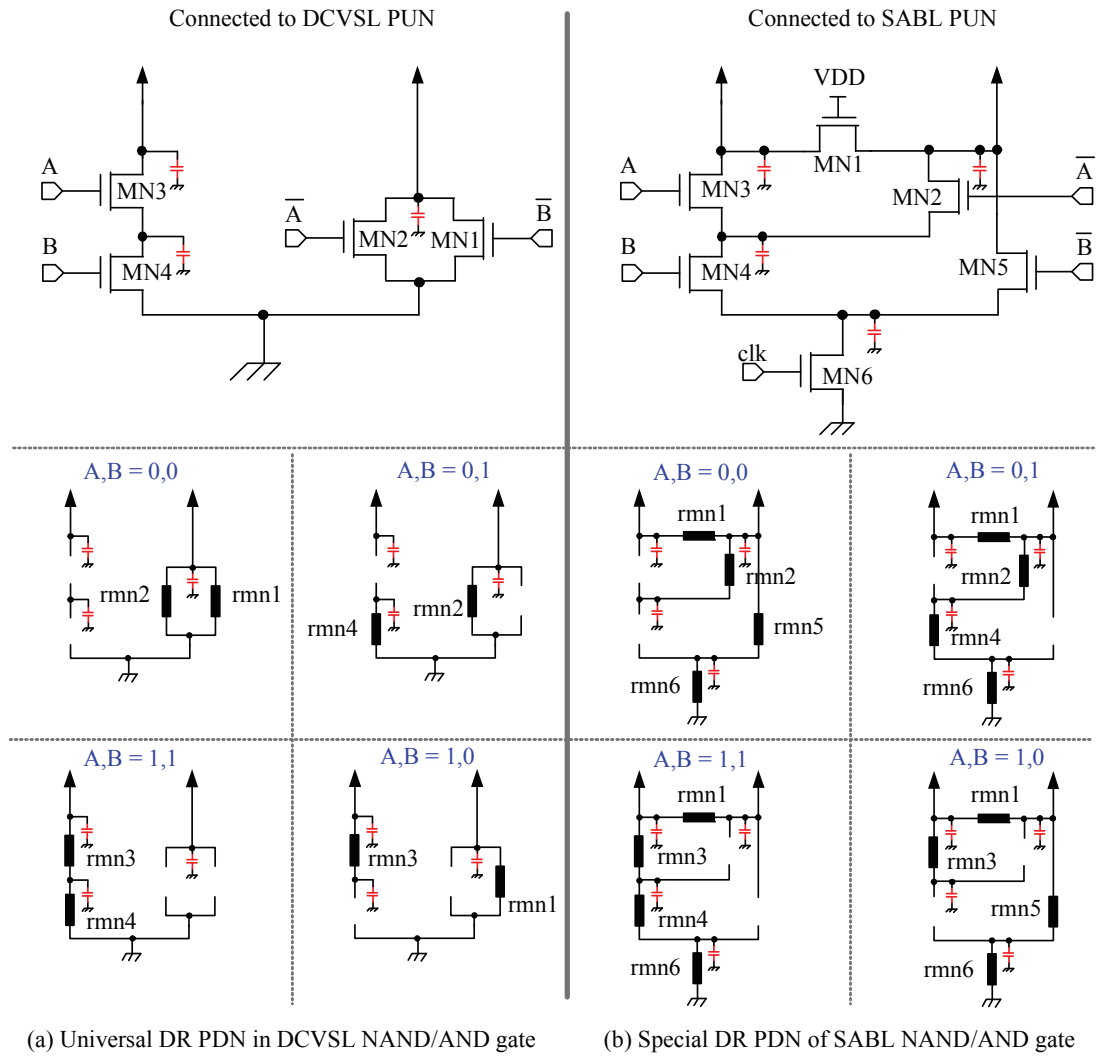
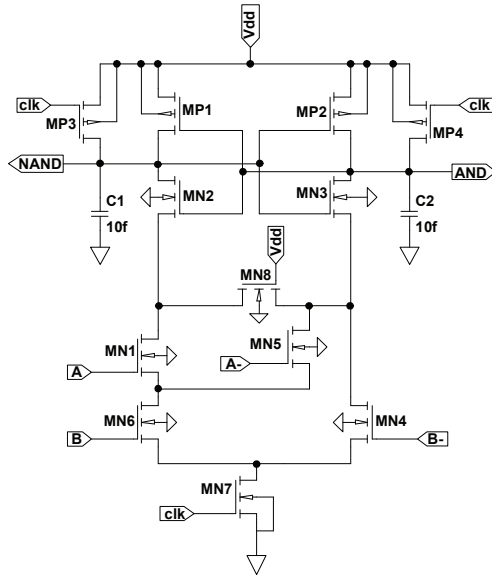
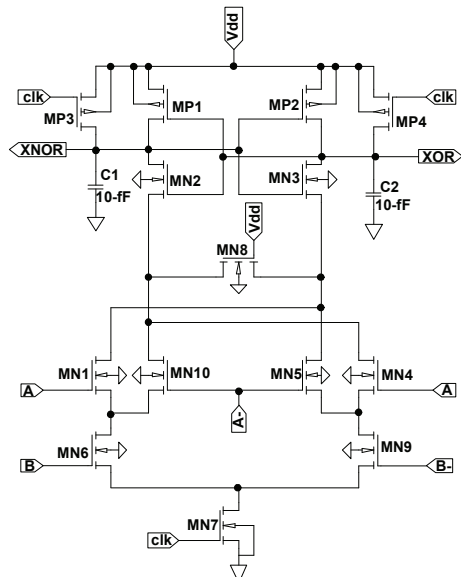


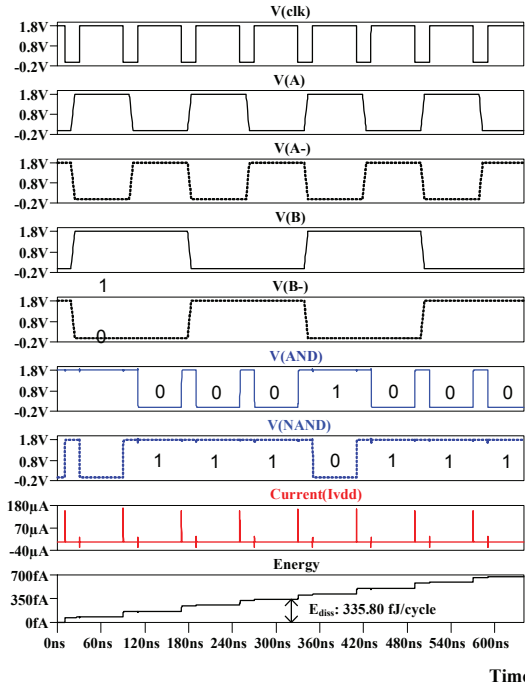
Figure 3.10: (a) Universal DR PDN (*i.e.*, used in DCVSL AND/NAND gate) and (b) Special DPDN of the SABL AND/NAND gate.



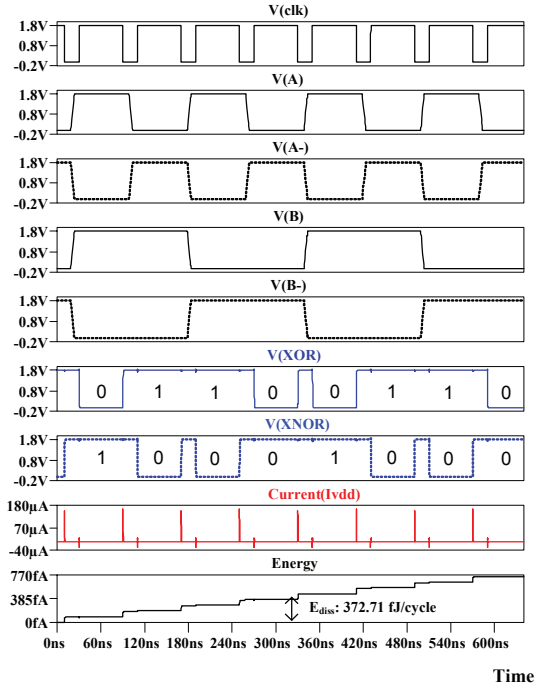
(a) SABL NAND/AND transistor schematic



(c) SABL XNOR/XOR transistor schematic



(b) SABL NAND/AND input-output signals



(d) SABL XNOR/XOR input-output signals

Figure 3.11: SABL: (a) NAND/AND transistors schematic, (b) NAND/AND input-output signals, (c) XNOR/XOR transistors schematic, (d) XNOR/XOR input-output signals.

Three-Phase Dual-Rail Pre-charged Logic (TDPL)

The TDPL was designed based on dual-rail pre-charge logic family whose power consumption is insensitive to unbalanced load conditions thus allowing adopting a semi-custom design flow (automatic place and route) without any constraint on the routing of the complementary wires [74]. In comparison with the SABL, the TDPL has three phases: pre-charge, evaluation and discharge during one cycle operation, while the SABL logic has two phases only without discharge phase. The basic operation of generic TDPL in Fig. 3.12 is described as following:

Charge: At the beginning of each cycle, signal discharge at low level, thus switching transistor MP1 is ON. At the same time, signal charge goes low too and both output lines are pre-charged to V_{dd} .

Evaluation: During this phase new input data are presented to the circuit. On the rise-edge of the signal eval, the transistor MNeval is turned on to discharge one of the output lines according to the active input path.

Discharge: At the end of each operation cycle, the signal discharge is activated in order to pull down (through the additional pull-down transistors MN1 and MN4) the output line which has not been discharged during the evaluation phase.

The schematic diagrams of the TDPL AND/NAND and XOR/XNOR, and the SPICE simulation results confirming the input-output functionality and the constant current traces for every clock cycle can be seen in Fig. 3.13.

The authors of the TDPL concluded in [74], the energy consumption per cycle is up to 100 times more balanced than the corresponding SABL gates without requiring any constraint on the geometry of the complementary wires. To verify this conclusion, the transitional supply current traces of all logic styles investigated on this section have been checked, as depicted in Fig. 3.14. From this figure, we can observe that, the scCMOS, DCVSL, and the dynamic logic styles consume various power fluctuation, and there are visible different peak current between zero levels to the maximum peak level. On the other hand, the well-designed SABL and TDPL perform undetectable peak variance. On top of the SABL cell, the TDPL has very constant current traces for 16-different data.

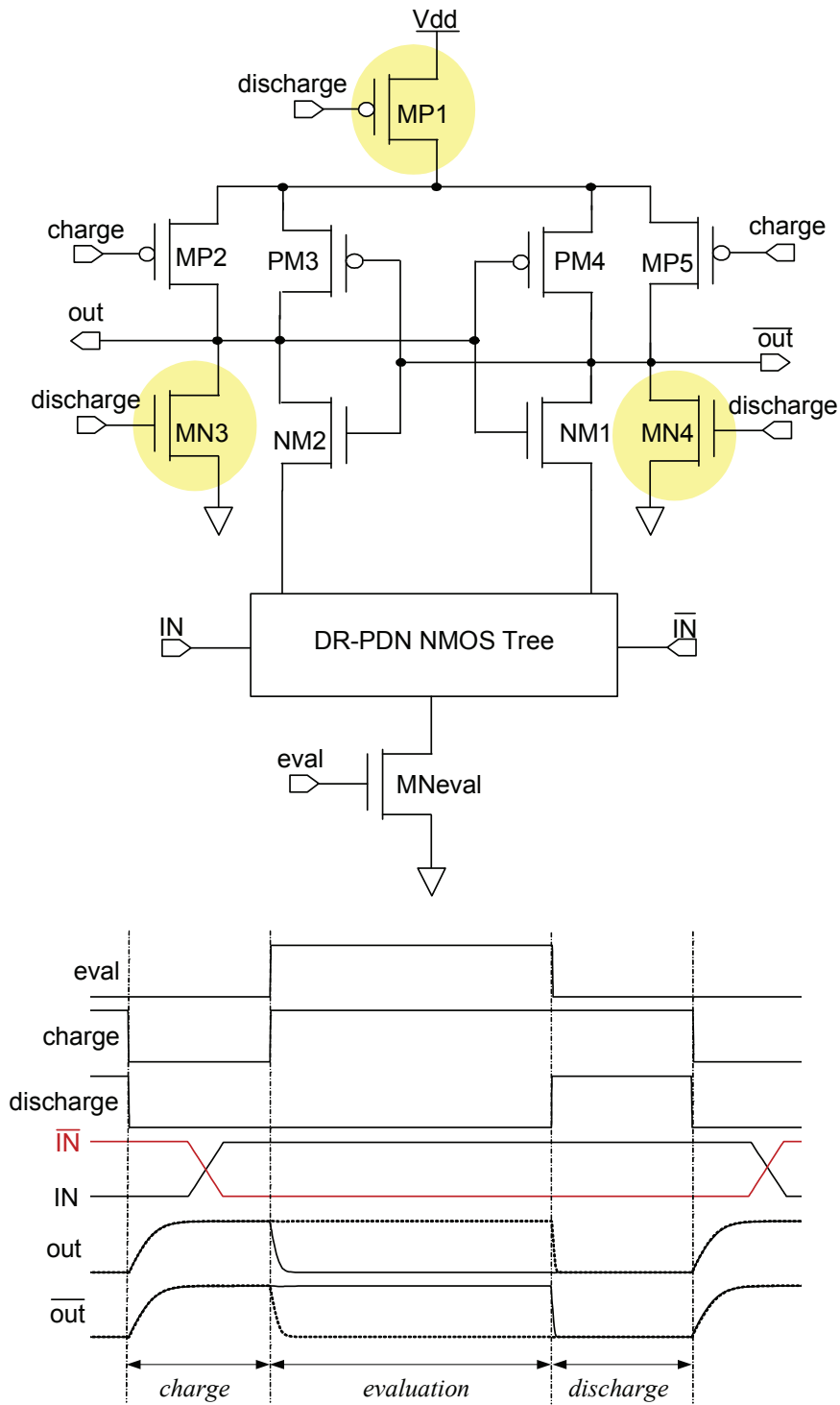


Figure 3.12: Generic TDPL logic cells and its input-output timing graphs.

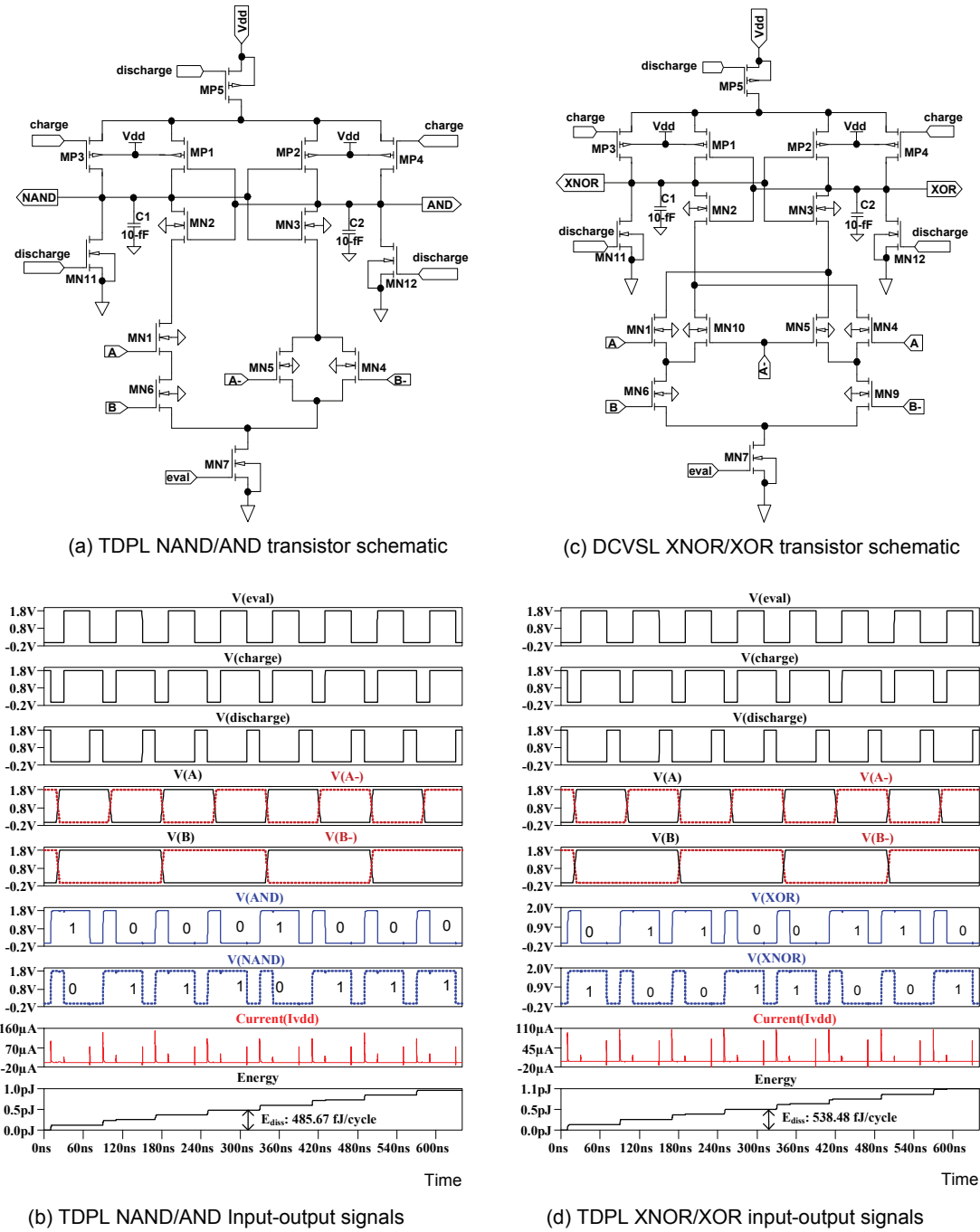


Figure 3.13: TDPL: (a) NAND/AND transistors schematic, (b) NAND/AND input-output signals, (c) XNOR/XOR transistors schematic, (d) XNOR/XOR input-output signals.

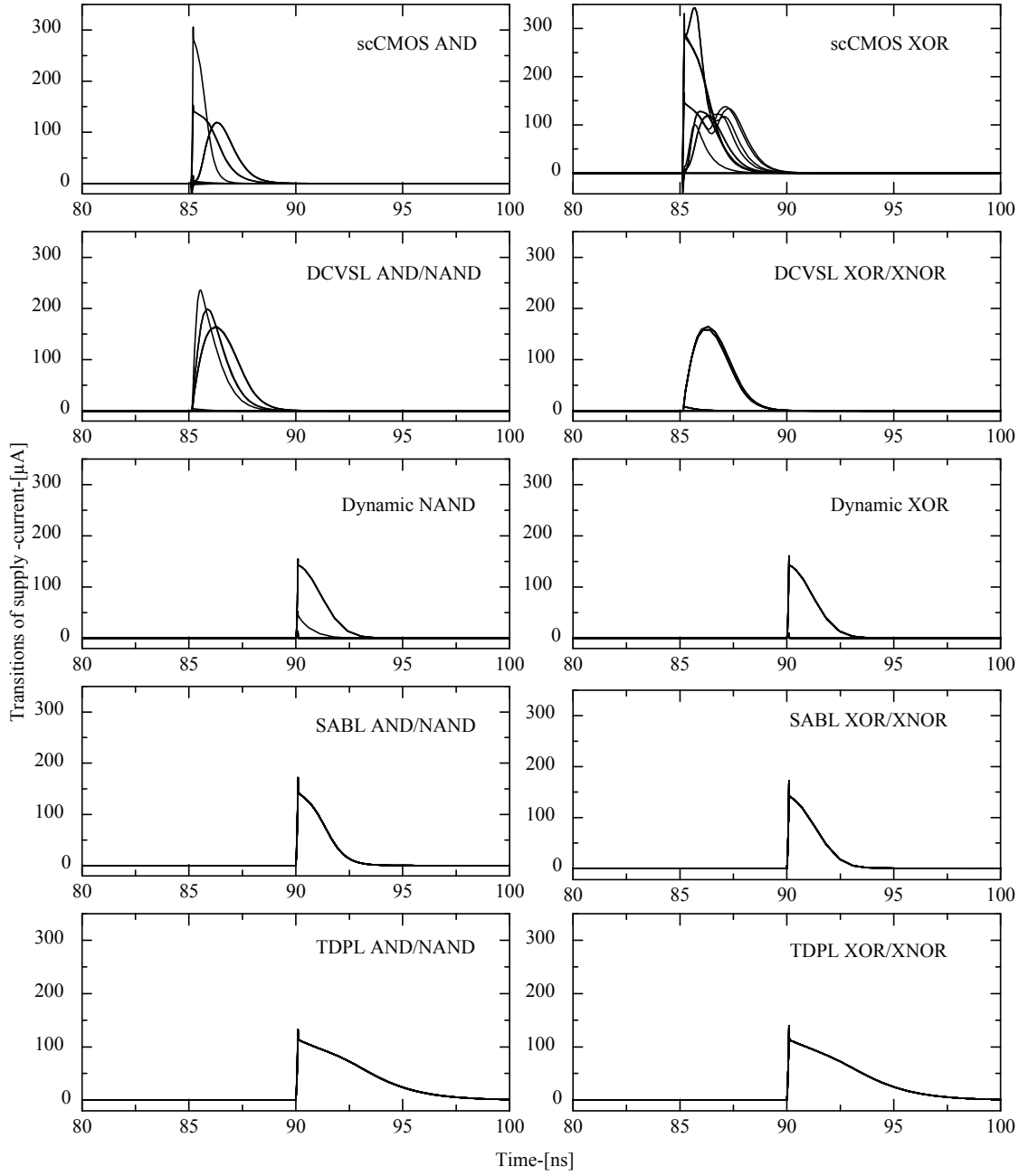


Figure 3.14: Supply current traces of 16-possible dual-input transitions (see Table 4.1) of individual gates investigated: scCMOS, DCVSL, Dynamic logic, SABL, and the TDPL, respectively. These results were achieved by setting 10-fF of the nominal capacitance at the output cells (between output nodes and the GND), and the horizontal time function has been enlarged for better readability of the peak current differences.

3.3.2 Masking

The masking countermeasure is a technique to remove the correlation between the input data and the side-channel leakage from intermediate nodes in a functional blocks, such multiplicative masking of AES transformation blocks conducted in [78]. The masking method is accomplished either at each gate level or in a block level (algorithm level). Analytically, the masking is expressed as: $v_m = v * m$, where the input value of v is concealed by random value of m (called as mask) [50]. The value of m must not be known by the attacker. The operation $*$ represents for the operations that are used in cryptographic algorithm, such as Boolean masking operation of exclusive-or function ($v_m = v \oplus m$), arithmetic masking operation of the modular addition ($v_m = v + m \pmod{n}$), or the modular multiplication ($v_m = v \times m \pmod{n}$) [76], [77].

Generally, in masked gate level implementation, the input signals are masked with some random values in order to make input signals invisible, and thus the masking technique also called as blinding. For example, in power analysis, although the attacker may draw some leaked signals, he will not know the actual input value that it represents for, because it is blinded by random value m . Generic normal cell and masked cell are shown Fig. 3.15. From this figure, the normal cell's input a and b are masked, and becomes dual-input DR logic style. To implement this masking method at cell level, of course, it will cost some additional gates compare to the normal cell style.

The first research work on masked logic implementation was conducted by Popp and Mangard [79], called as masked dual-rail pre-charge logic (MDPL). MPDL circuit was adopting DR pre-charge logic style by employing the single-rail majority (SR-MAJ) gate, as depicted in Fig. 3.16. Figure. 3.16(a) shows a SR-MAJ transistor schematic diagram. An MDPL NAND/AND requires two SR-MAJ cells with complementary input signals same as the normal differential logic styles, except the MDPL has its own property of mask signal as shown Fig. 3.16(b). The author investigates this MDPL AND/NAND gates in SPICE simulation, and the results confirm the truth table in Table 3.2. From this table, we observe that the output signals q_m and $\overline{q_m}$ perform NAND and AND functions, but obviously, it is difficult to predict actual input condition. For example, in line no. 4 of the truth table, q_m should be logic 1 same as the previous line no. 3, since they have same input a_m and b_m ; however with the unknown mask value inverts it into logic 0, surprisingly. Based on the simulation results in Fig. 3.16(c), different high supply current spikes always appear

when the output data flip, and hence, the attacker may use HD model to reveal the secret key. Moreover, the masking logic styles are area consuming, consequently, power dissipation also high. For instance, to construct an XOR/XNOR gate will require six SR-MAJ gates, which is very costly, about 4–5 times higher than hiding logic technique (*i.e.*, TDPL or SABL).

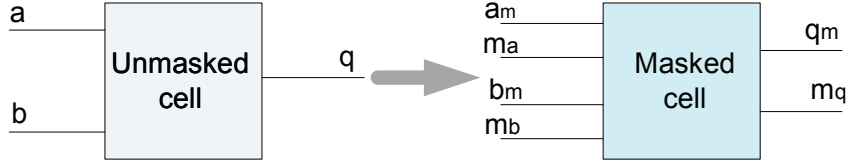
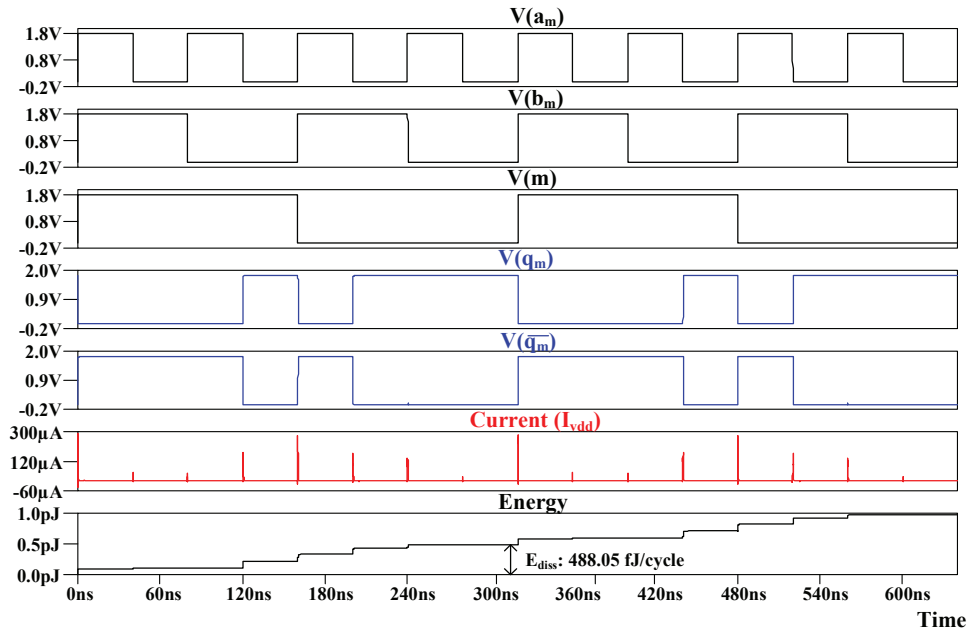
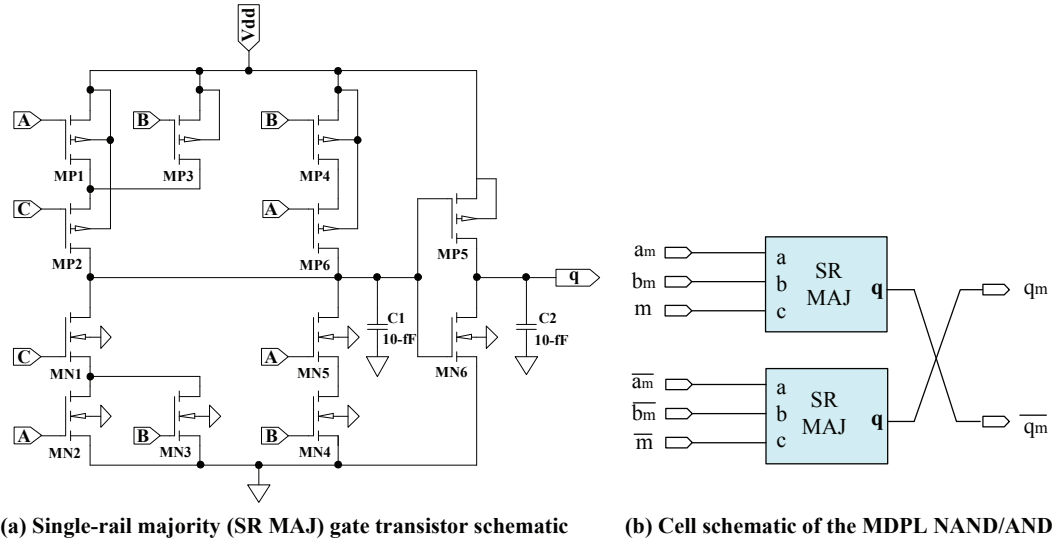


Figure 3.15: A dual-input unmasked cell and a corresponding dual-input masked cell.

Table 3.2: Truth table of an MDPL NAND/AND cell for complementary input values.

Line no.	m	b_m	a_m	\overline{m}	$\overline{b_m}$	$\overline{a_m}$	$q_m = MAJ(\overline{a_m}, \overline{b_m}, \overline{m})$	$\overline{q_m} = MAJ(a_m b_m, m)$
1	0	0	0	1	1	1	1	0
2	1	0	0	0	1	1	1	0
3	0	1	0	1	0	1	1	0
4	1	1	0	0	0	1	0	1
5	0	0	1	1	1	0	1	0
6	1	0	1	0	1	0	0	1
7	0	1	1	1	0	0	0	1
8	1	1	1	0	0	0	0	1



(c) Input-output signals of the MDPL NAND/AND gate

Figure 3.16: MDPL: (a) Single-rail majority (SR MAJ) gate transistor schematic, (b) Cell schematic of MDPL NAND/AND gate, and (C) Input-output signals of the MDPL NAND/AND gate.

3.4 Summary

The survey conducted in this chapter on the basis of the literature resources and some confirmation works in SPICE simulation level has revealed the following important points:

- SCA attacks, specifically the power analysis attacks are possible because there exist a correlation between the processed data and power consumption when a crypto device executes encryption/decryption calculation. To counteract these attacks, numerous work on digital cell level were reported; those efforts are to hide or mask the processed data by making constant power and randomizing power consumption to avoid current-to-data dependencies.
- Secret key of crypto chip can also be revealed using EMA attacks, because the high supply current that flows during the switching of the CMOS gates, causes a variation of the electromagnetic field surrounding the chip that can be monitored by inductive probes. To reduce the possibility of this attack, the electric current signals must be lowered as low as possible. To handle this in the logic level design, the adiabatically low-power design technique could be one of the solutions.
- There are two basic techniques have been widely developed to countermeasure against side-channel attacks at cell level; such as hiding and masking techniques. In hiding technique, the logic designer makes a circuit to consume constant power for every input-output transitions. In masking method, the logic circuit is design in such a way that power is randomly consumed; however, this technique requires extra signal to mask the main signals, and hence it is area consuming (for instance, the majority gate for XOR function). Based on this drawback, the proposed logic will be designed using hiding logic technique.
- SCA countermeasures at cell level using various logic styles have been investigated in this chapter. Based on the simulation results in Fig. 3.14, the author has confirmed that the single-rail static or dynamic logic styles are data dependent. Therefore, a proper logic design using complementary dual-rail technique should be employed.

Chapter 4

Dual-Rail Adiabatic Logic

Approach for Secure Logic Implementation

4.1 Overview

In this chapter, the author will discuss the proposed logic. To validate the security metric of the proposed logic circuits, then the author will first introduce some conventional adiabatic logic styles that previously published for secure logic implementation, or were under investigation for power analysis attacks countermeasures. The author will verify the security of the proposed logic by carefully analyzing the individual logic functions corresponding to 16-possible dual-input transitions. Then, compares the results with those of the previous secure logic styles using the same parameters and under the same conditions. The parameters to measure the resistance of the logic against DPA attacks, such as the means on energy, the variance of energy, the normalized standard deviation, and the figure of merit will be calculated and the results will be compared to each other.

4.2 Conventional Secure Dual-Rail Adiabatic Logic Styles

Adiabatic logic technique was originally proposed for energy recycling rather than dissipated as heat [33]. On the basis of adiabatic technique, further works on dynamic power minimization using adiabatic switching principle for cryptographic

hardware implementation were published [88], [89]. They employed single-rail adiabatic quasi-static CMOS (AQS-CMOS) that claimed for low-power, low-complexity smart card circuitry than the conventional smart card architecture. Moreover, the comparison work of adiabatic logic as a countermeasure against power analysis attacks [90] has been reported. The comparison done in [90] claimed that adiabatic dynamic logic (SR logic) seems to be secure against DPA attacks. However, all those works are ignoring the investigation of the ability of logic function for counteracting power analysis attacks. A visibility study on SR and DR CMOS logic styles was done by the author of this dissertation [91]. The study was conducted to investigate static logic, dynamic logic, precharge dual-rail logic styles in adiabatic operation. The investigations are carried out by calculating the normalized energy deviation (NED) and normalized standard deviation (NSD) of every dissipated energy transitions according to input logic function, in which the results shown that the single-rail static and dynamic logic styles are data dependent meaning, they are vulnerable for secure logic implementation.

In this section, the author will investigate the DR adiabatic logic styles, such as, an efficient charge recovery logic (ECRL) [92], the 2N-2N2P [93], secure adiabatic logic (SAL) circuit [94] and symmetric adiabatic logic (SyAL) circuit [95]. These logic styles will be re-simulated in LTspice simulator. The results will be compared with the proposed logic in the next section, and the detail discussion on the logic's immunity for SCA attacks will be drawn.

4.2.1 Efficient Charge Recovery Logic

Efficient charge recovery logic (ECRL) is the simplest DR adiabatic logic since ever been proposed. The generic logic structure of ECRL and its input and output waveform are shown in Fig. 4.1. The ECRL is basically operated in four phases as follows:

1. **Input phase:** In this phase, the input signal slowly goes high from $0 \rightarrow V_{dd}$, while the power clock signal is low. At the end of this phase, inputs have taken their own valid values. Suppose that $In=1$ (high) and $\overline{In}=0$ (low); meaning, the transistor MN2 is closed and MN1 is open.
2. **Evaluation phase:** The power clock signal slowly goes high; thus, the *Out* node is charged through MP2. In contrast, \overline{Out} node remains low since it is connected to ground through the MN2.

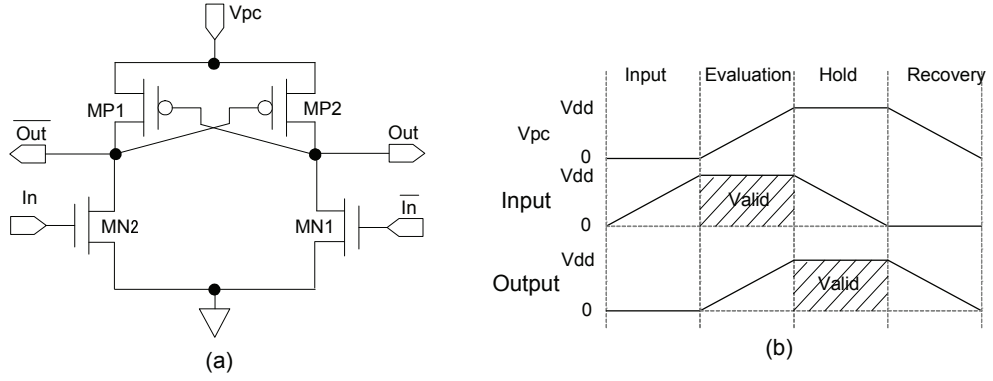


Figure 4.1: ECRL inverter: (a) Transistor schematic, (b) Timing chart.

3. **Hold phase:** In this phase, the present condition of valid output remains stable at high level.
4. **Recovery phase:** The power clock steadily decreases towards low level. By falling power clock signal, *Out* goes low via MP2, and hence the charge recovery takes place to save power instead of dissipated as heat.

The ECRL AND/NAND and XOR/XNOR logic circuits are shown in Figs. 4.2(a), 4.3(a), respectively. The SPICE simulation results of respective circuits are shown in Figs. 4.2(b), 4.3(b). Input-output signals are correctly operated in adiabatic mode. Observing these figures, the supply current traces appear with different peak values. For example, let us look at input pattern $(A, B) = (1, 0)$ at 180-ns of horizontal axes in Figs. 4.2(b). We can see here that when the power clock signal at recovery phase ($V_{pc} = V_{dd} \rightarrow 0$ V) the circuit is under charge recovery, means the output node NAND that previously charged (high) is transferred back to power supply V_{pc} line. Ironically, this charges are not fully discharged to zero level, it remains some floating charges when the output voltage transition of NAND node reaches the PMOS threshold voltage V_{tp} level; and then, the same node is re-charged again within the same load, thus cause the low peak current at this point compare to the previous current signal. This different peak current transitions is the main concern on this dissertation work. Because, it leads to the logic vulnerability to power analysis attack.

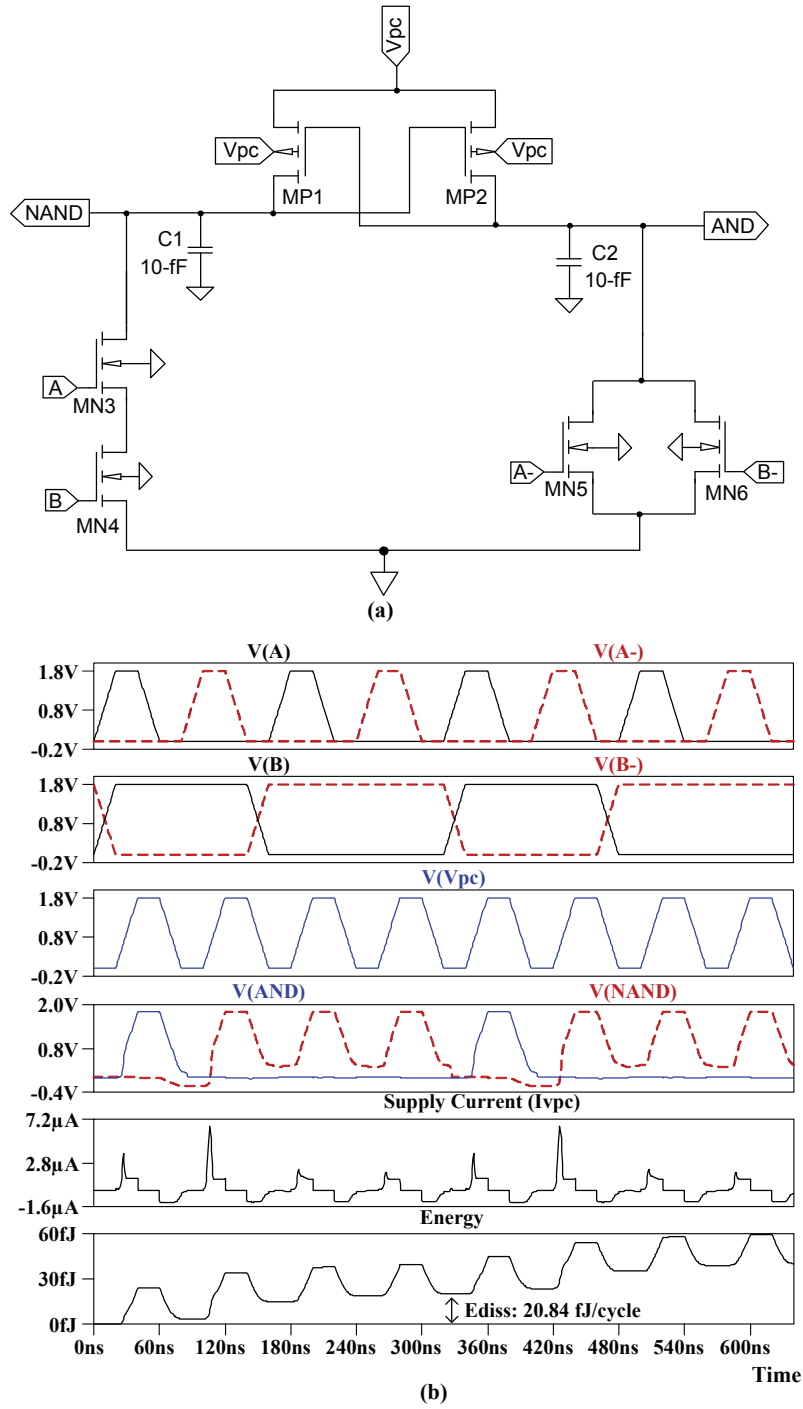


Figure 4.2: ECRL: (a) Transistor schematic of AND/NAND circuit, (b) Input-output signals.

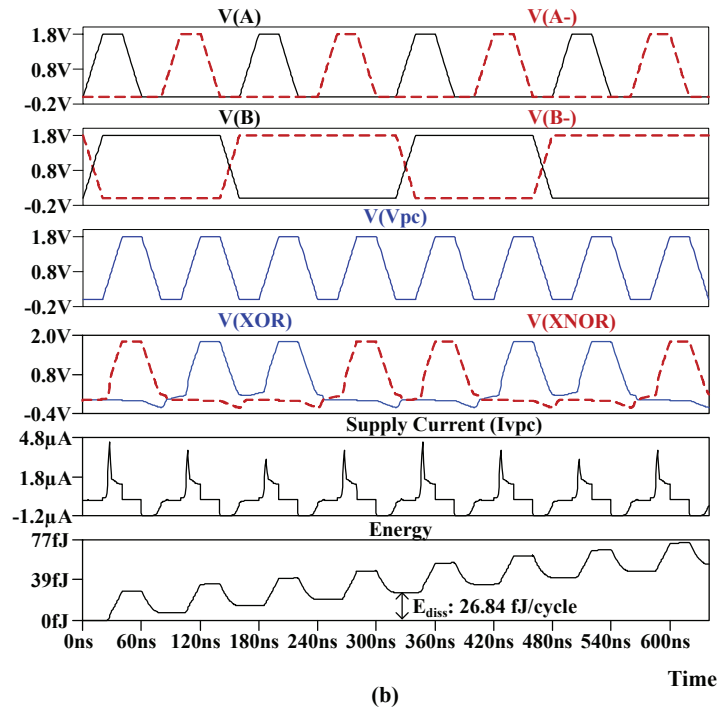
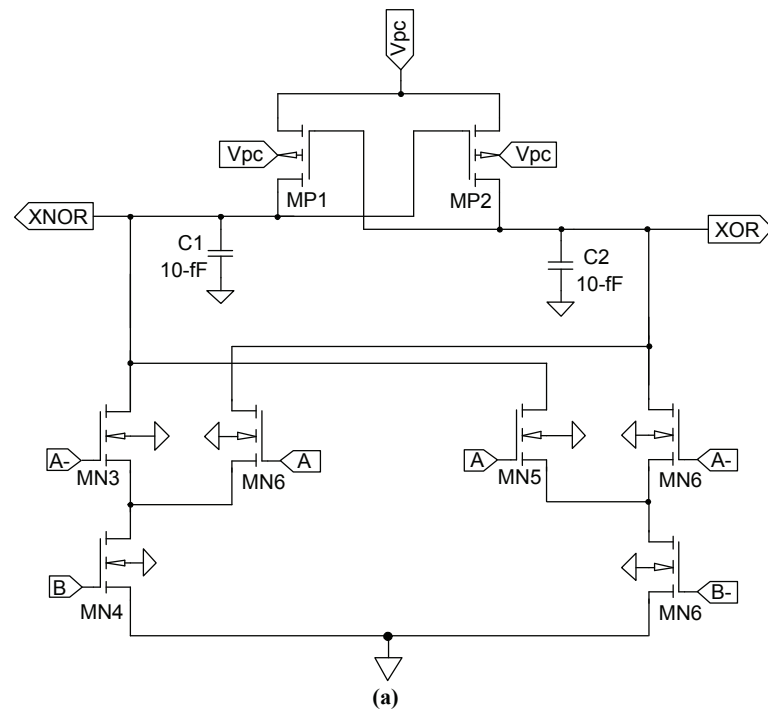


Figure 4.3: ECRL: (a) Transistor schematic of XOR/XNOR circuit, (b) Input-output signals.

4.2.2 2N-2N2P Logic

The operational function of 2N-2N2P gate is similar to ECRL logic operation, they have same phases. Generic logic of the 2N-2N2P and its waveform are shown in Fig. 4.4. Observing Figs. 4.4(a), the 2N-2N2P gate consists of two main parts: (i) Two functional input gates MN3 and MN4 whose duty is to construct the logic outputs Out and \overline{Out} , and (ii) A latch circuit which is made by two cross-couple NMOS transistor to avoid floating charges at output nodes. The primary advantage of 2N-2N2P is in fact that the addition of the cross-coupled MN1 and MN2 result in non-floating data valid over 100% of the *Hold* phase, as opposed to ECRL logic style, because the input are ramping down during the *Hold* phase, in which a logic output is only clamped for the first 50% of *Hold* phase [93].

The 2N-2N2P AND/NAND and XOR/XNOR logic circuits are shown in Figs. 4.5(a), 4.6(a), respectively. The SPICE simulation results of respective circuits are shown in Figs. 4.5(b), 4.6(b). Same as ECRL, the input-output signals are also correctly operated in adiabatic mode. Observing these figures, the supply current traces appear with different peak values, and the reason of this phenomenon has been explained in previous section of ECRL logic style. Both ECRL and 2N-2N2P were implemented using same PDN transistor topology. Thus, they have similar supply current behavior.

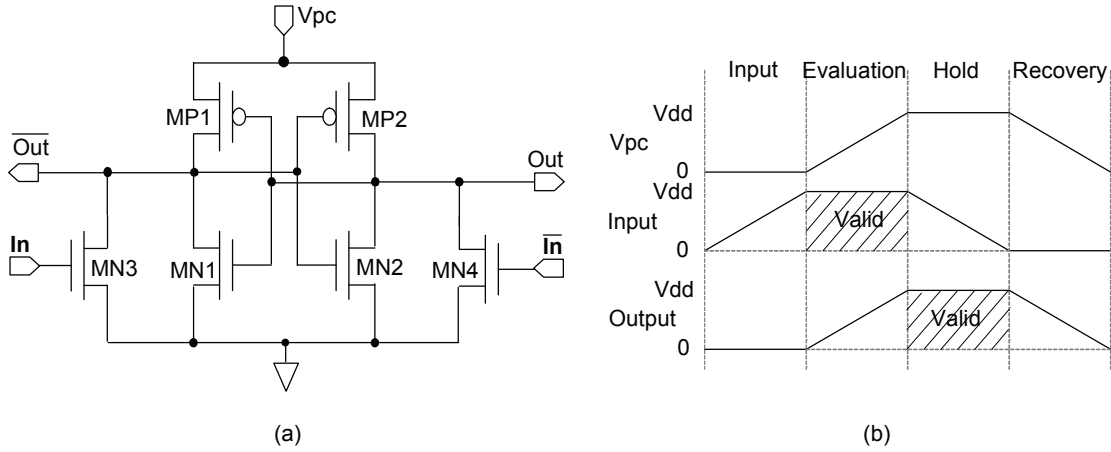


Figure 4.4: 2N-2N2P Inverter; (a) Generic logic structure, (b) Timing diagram.

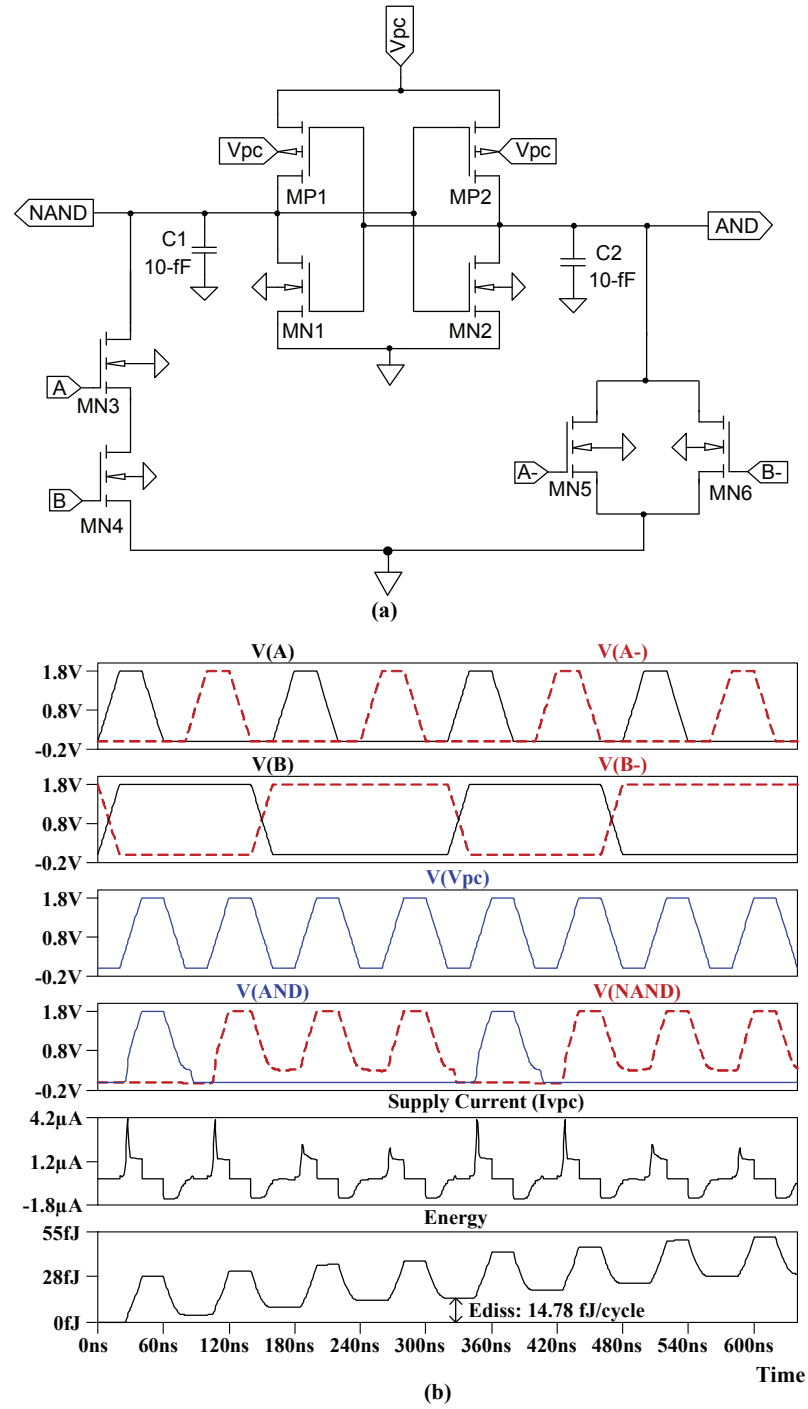


Figure 4.5: 2N2N2P: (a) Transistor schematic of AND/NAND circuit, (b) Input-output signals.

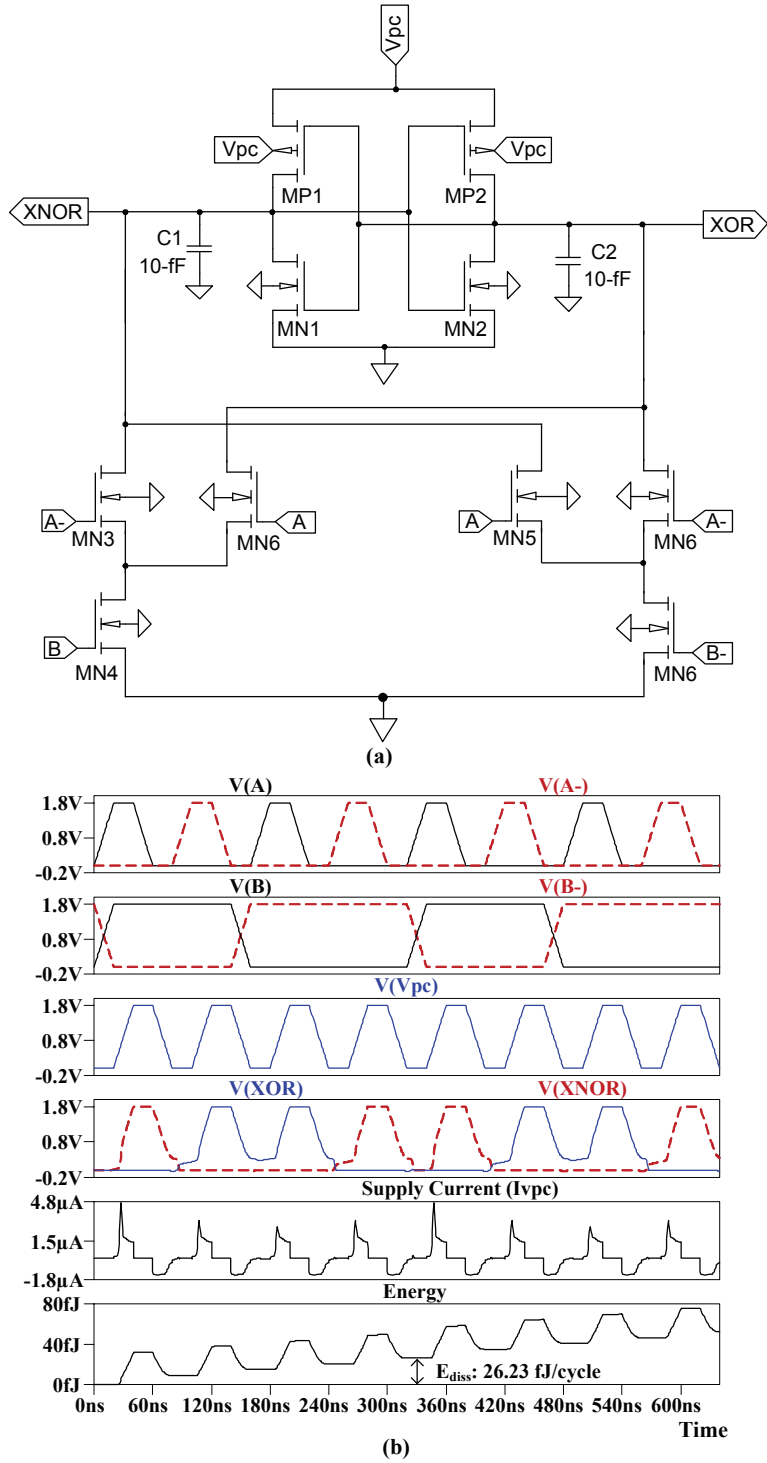


Figure 4.6: 2N2N2P: (a) Transistor schematic of XOR/XNOR circuit, (b) Input-output signals.

4.2.3 Secure Adiabatic Logic

The generic cell construction of secure adiabatic logic (SAL) and its timing chart are depicted in Fig. 4.7. The SAL consists of three main parts: (i) Two function blocks construct the outputs. These functions are implemented by NMOS transistors, (ii) A latch circuit which is made by two cross-coupled NMOS transistors, i.e., MN1 and MN2 to keep the output stable in respect to input conditions, and (iii) Extra pass transistors, i.e., MN3 to MN8, that are responsible to discharge internal capacitances of the function blocks adiabatically.

In SAL design, the function blocks and the two cross-coupled NMOS transistors are connected to a DC bias voltage equal to V_{tp} instead of connected to GND, in order to avoid the non-adiabatic energy dissipation due to incomplete discharge of C_{load} . The SPICE simulation in this work, the DC bias = 0.2-V was chosen based on the investigation, where this value is suitable for power reduction. There are eight power clock signals operated in eight phases as reported in original paper [94]; however, only four power clock signals (V_{pc0} , V_{pc1} , V_{pc5} , V_{pc6}) are used for individual logic investigation in this work, as shown in Fig. 4.7.

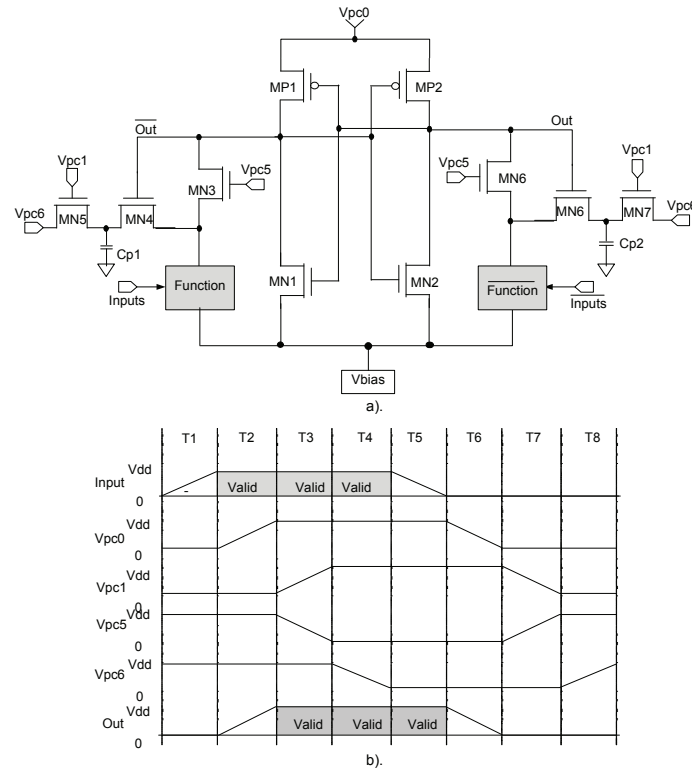


Figure 4.7: SAL; (a) Generic logic structure, (b) Timing diagram.

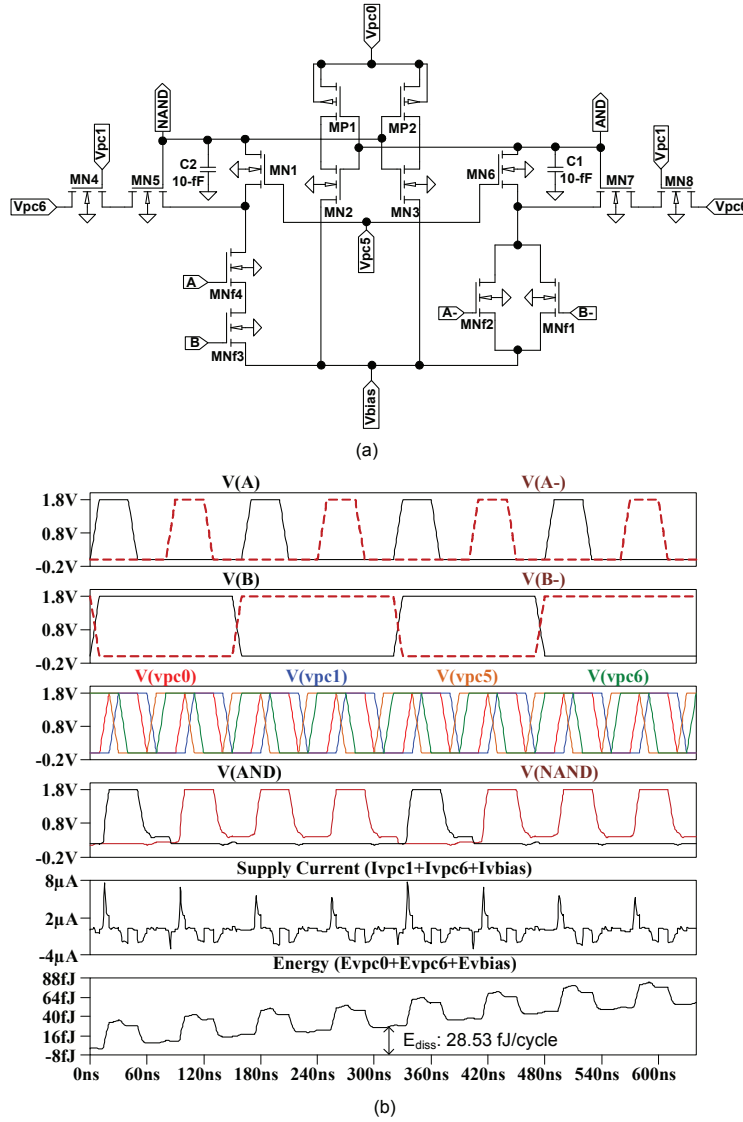


Figure 4.8: SAL: (a) Transistor schematic of AND/NAND circuit, (b) Input-output signals.

The logic operation in the SPICE simulation results of the SAL AND/NAND and XOR/XNOR can be confirmed in Figs. 4.8, 4.9. In these figures, the energy and supply current traces are plotted from the total summation of power supplies V_{pc0} , V_{pc6} and the V_{bias} , considering the current that flows from drain-source direction and/or reversely. Although, the SAL has discharge cells to discharge the internal wires that previously undischarged, the peak current differences remain visible for dual-input logic operations, because CSSAL also implemented using universal DR PDN same as ECRL and the 2N-2N2P.

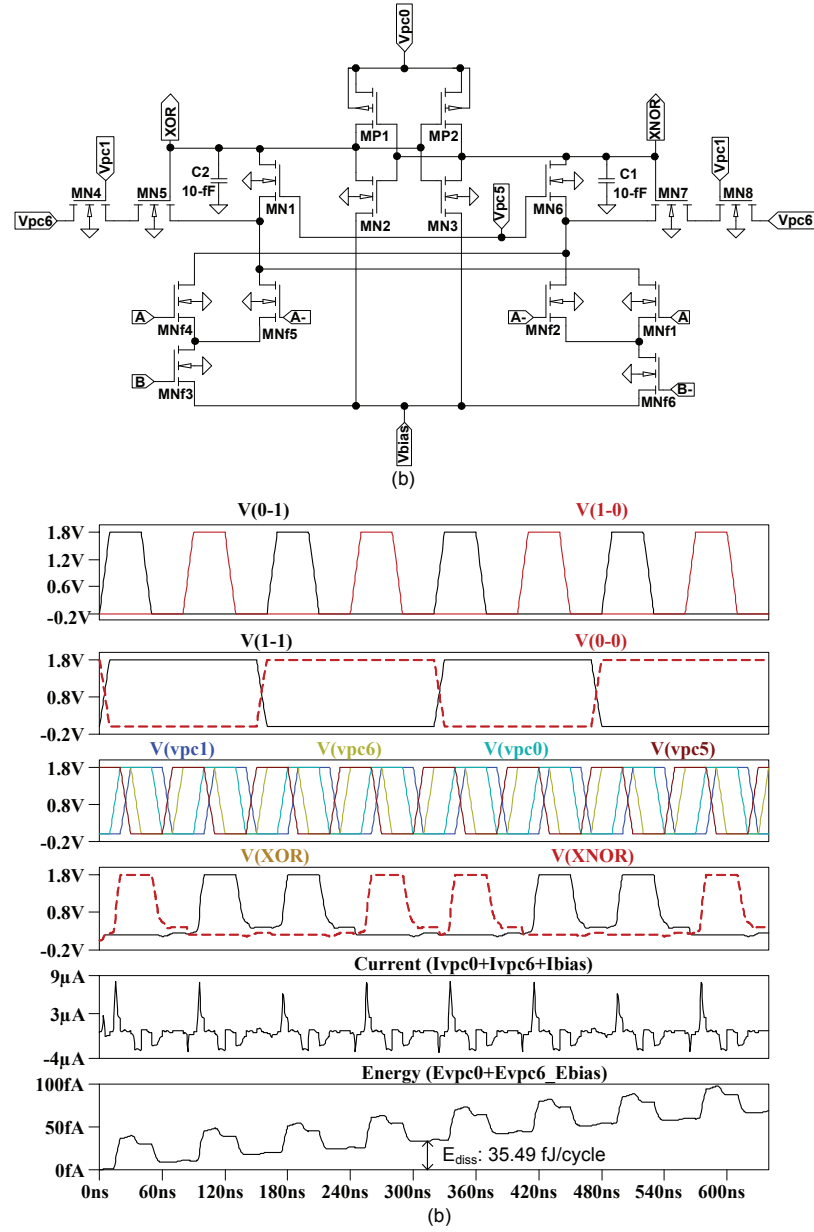


Figure 4.9: SAL: (a) Transistor schematic of XOR/XNOR circuit, (b) Input-output signals.

4.2.4 Symmetric Adiabatic Logic

Symmetric adiabatic logic (SyAL) [95] employed a symmetric pull-down network transistors that was proposed in symmetric discharge logic [73] to minimize differences in power traces for resistances to DPA attacks. The principal idea of the SyAL circuit is assigned to discharge the internal wires of the symmetric PDN transistors. The SyAL was constructed in such a way that on-and-off transistors are configured equally for all cases. It has been explained in the logic operation of the ECRL that the output nodes of adiabatic logic are not fully discharge to zero level through PMOS transistor; therefore, SyAL is designed to share all internal parasitic capacitors by inserting the BR transistors that operate when power clock and both complementary input signals are at low level, as shown in Fig. 4.10. Similar to other DR adiabatic logic styles, SyAL also operates in input phase (Wait), evaluation phase, hold phase and recovery phase, with additional bridge phase, thus making five phases for one period of power clock cycle. When the BR signal at high level, the voltage of both output nodes are equalized, thus the charge sharing occurs. As the result, it claimed that the supply current is not affected by the previous input data.

The SyAL AND/NAND and XOR/XNOR logic circuits and their SPICE simulation are shown in Figs. 4.11, 4.12, respectively. In these figures, the supply current transition for four dual-input patterns has similar amplitude, however small differences still can be observed.

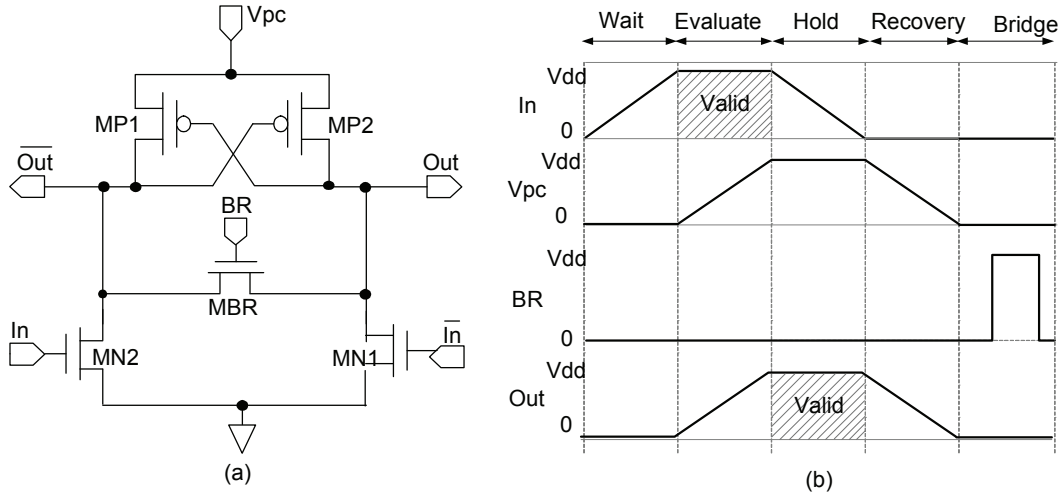
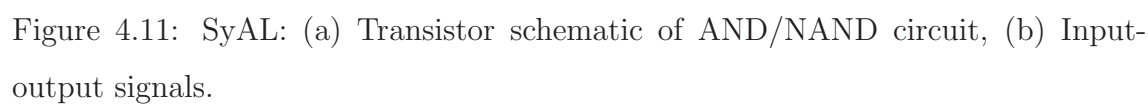


Figure 4.10: SyAL: (a) Inverter logic structure, (b) Timing diagram.



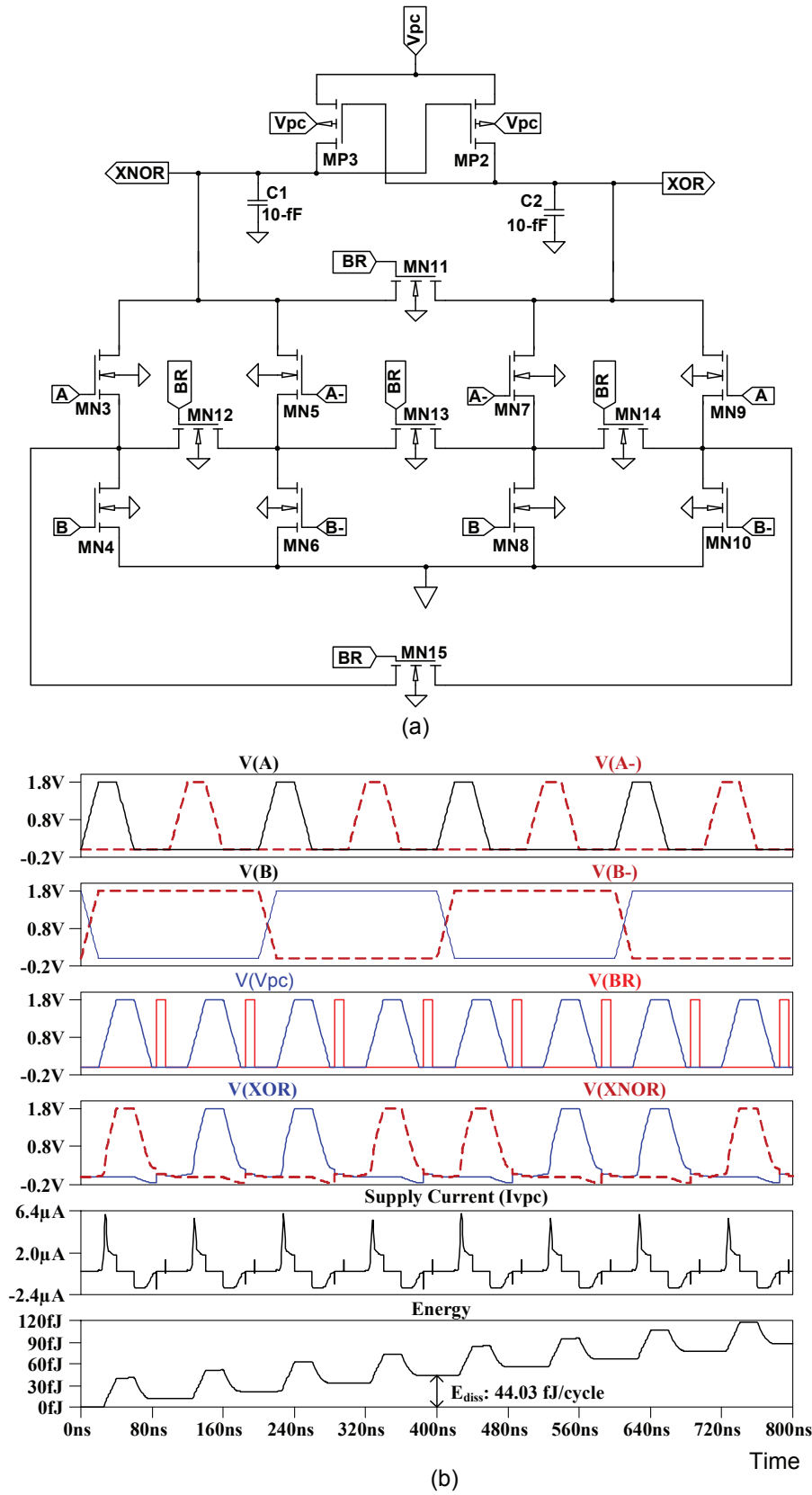


Figure 4.12: SyAL: (a) Transistor schematic of XOR/XNOR circuit, (b) Input-output signals.

4.3 Proposed CSSAL: Charge-Sharing Symmetric Adiabatic Logic

Learning from the previous works, conducting tireless investigative studies, scrutinize the conventional logic styles, identify the existing problems, and then accumulating the prestigious ideas from the other researchers have been passed through by the author of this dissertation. After all, the author has succeeded to present a proposed robust secure logic styles. In this section, the author introduces the proposed charge-sharing symmetric adiabatic logic (herein after known as CSSAL). The CSSAL circuitry were realized based on several ideas and techniques, such as the idea the TDPL, SyAL, 2N-2N2P and an optional idea of clocked adiabatic logic (CAL) [96]. These four logic techniques have contributed to the robust secure CSSAL realization.

4.3.1 CSSAL Inverter

The proposed CSSAL inverter/buffer circuit is depicted in Fig. 4.13(a), and its input and output signals are shown in Fig. 4.13(b). Obviously shown in this figure, the CSSAL operates in four phases as follows:

1. **Charge-sharing:** The discharge (*Dischg*) signal increases with a rate twice that of the input signal. In this phase, the power-clock voltage (V_{pc}) is stable at a low level, and the evaluation path signal which is established by *In* or \overline{In} (MN5 or MN6) and *Eval* (MN8) cells simultaneously also slowly increases. During this phase, all the internal node capacitances are discharged to ground before the logic function is evaluated, in order to prevent the circuit from depending on the previous input data.
2. **Evaluation:** In this *Eval* phase, the *Dischg* signal is already stable at a low level, which turns on the MP1 for supply current to flow into the logic circuit. The output wires are evaluated through one of the active input cells and *Cx* transistors that are already at a high level.
3. **Hold:** During the hold phase, the presently active input and *Eval* signals slowly decrease to become low, but the outputs remain stable because those are controlled by cross-coupled NMOSs MN1 and MN2.

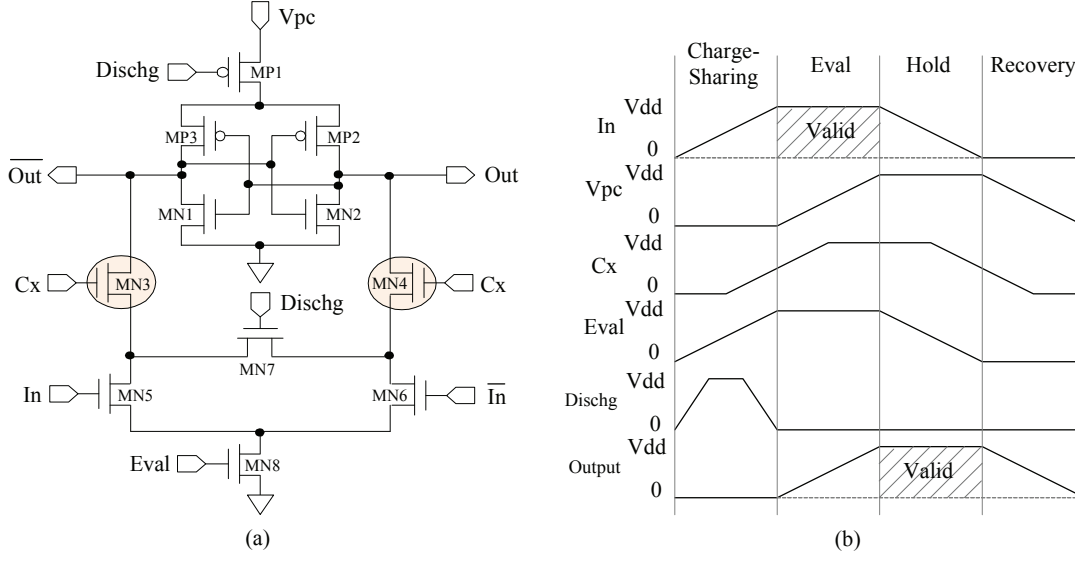


Figure 4.13: Proposed CSSAL logic; (a) Inverter logic structure, (b) Input and output signals of the proposed CSSAL inverter logic.

4. **Recovery:** The power clock voltage (V_{pc}) is steadily decreases to a low level, and the presently active output is discharged to low via the active MP2 or MP3 and MP1 since the Dischg signal is still low. Consequently, charge recovery concept occurs for every power-clock cycle to minimize the energy lost through charging or discharging.

4.3.2 Equivalent RC Model Analysis of the CSSAL Inverter Logic

The proposed CSSAL is an enhancement of the SyAL form of the symmetric input logic style. The SyAL is one of secure DR logic styles that employing adiabatic principles to equalize the voltage between the output nodes. It applied charge-sharing techniques to reduce the data dependences. In this part, the author will conduct a deep discussion of the proposed CSSAL inverter in comparison with the 2N-2N2P and the SyAL inverter logic styles.

Figure 4.14(a) recalls the 2N2N2P inverter circuit that was originally proposed in adiabatic operation with four phases as shown in the timing diagram of the same figure. In this investigation, a carefully analysis of the internal RC connection at each respective phase when input state transition occurs. For example, the input condition at 0→1 transition, there will be four internal RC connection models hap-

pen at every phase, which is denoted as number 1, 2, 3, and 4 indicating each phase as shown in Fig. 4.14(b). Generic inverter logic normally has 4-possible input transitions, and hence in the case of 2N-2N2P for input condition at $1 \rightarrow 1$ transition, only RC model number 2, 3, 4 are occurred. The number 1 RC model (as shown in Fig. 4.14(b)) does not occur for this input condition. Same condition happen for opposite input transition ($0 \rightarrow 0$). Therefore, the occurrence of RC model are not balance for 2N-2N2P logic style as noted in Fig. 4.14(c), which cause various energy dissipation for different input transitions.

In the same way with 2N-2N2P, the Fig. 4.15(a) also recalls the SyAL inverter circuit that was originally proposed in adiabatic operation with five phases that can observed in the same figure. The equivalent RC model of each phase shown in Fig. 4.15(b) indicates that, at bridge-phase for $0 \rightarrow 1$ and $1 \rightarrow 1$ transitions has different internal RC model connection. Moreover, during the transition $1 \rightarrow 1$ or $0 \rightarrow 0$, the internal RC model of the Wait phase is omitted. Consequently, there are unbalanced load for 4-possible input transitions as labeled in Fig. 4.15(c). In addition, the charge sharing is performed at number 5 and 5' in Fig. 4.15(b). This figure shows that the path of electric current of *Out* and \overline{Out} flow to low V_{pc} through PMOS transistors MP1 and MP2, which is not fully discharged to zero level as early stated in section 4.2.1.

The proposed CSSAL has four phases as early described in section 4.3.1, where initially, it discharges all internal node charges to ground level before the other phases occur as can be seen in Fig. 4.16(b) during charge-sharing phase. In contrast to SyAL, internal wire charges are grounded through active evaluation cells. The CSSAL avoids to operate the charge sharing via PMOS transistors, and hence the top level MP1 was inserted that will be in the OFF state during charge-sharing phase in order to cut-off the current's path to low level of V_{pc} line. As the result, all internal nodes charges are fully discharged to become zero level before the logic evaluation takes place.

In comparison to the 2N-2N2P and the SyAL internal load conditions in regards to input signal transitions, the numbers of RC model occurrences of the CSSAL are same for all possible input transitions, which makes the CSSAL circuit able to consume more uniform and constant energy for different input transitions. This balanced load can be observed in Fig. 4.16(c), and the unique differences can be reconfirmed in Fig. 4.15(c) and Fig. 4.14(c) that belong to SyAL and 2N-2N2P, respectively.

The explanation of the inverter logic styles aforementioned have been verified

by means of SPICE simulation as depicted in Fig. 4.17. The author has checked each peak current value of 4-transition as summarized in the table of the same figure. The results shown that, the CSSAL has more constant value compare to the other logic styles. It is important to check this transitional peak current values in order to validate the proposed logic security measures that has been mapped in Eq. (2.22). The CSSAL proves HD model in Eq. (2.22) with very small acceptable differences about $0.03 \mu\text{A}$ order between $1 \rightarrow 0$ and $0 \rightarrow 0$ transitions, which the author classifies as very strong logic in this comparison result. Moreover, the SyAL peak differences is less next to the proposed CSASAL with peak difference value about $0.3 \mu\text{A}$ order between $1 \rightarrow 0$ and $0 \rightarrow 0$ transitions. The 2N-2N2P has about 50% differences, scCMOS and DR-CMOS both has 100% peak differences between the same transitions. Note that the peak current values of the scCMOS and DR-CMOS logic styles in this figure are recalled from the current traces in Fig. 3.3(c).

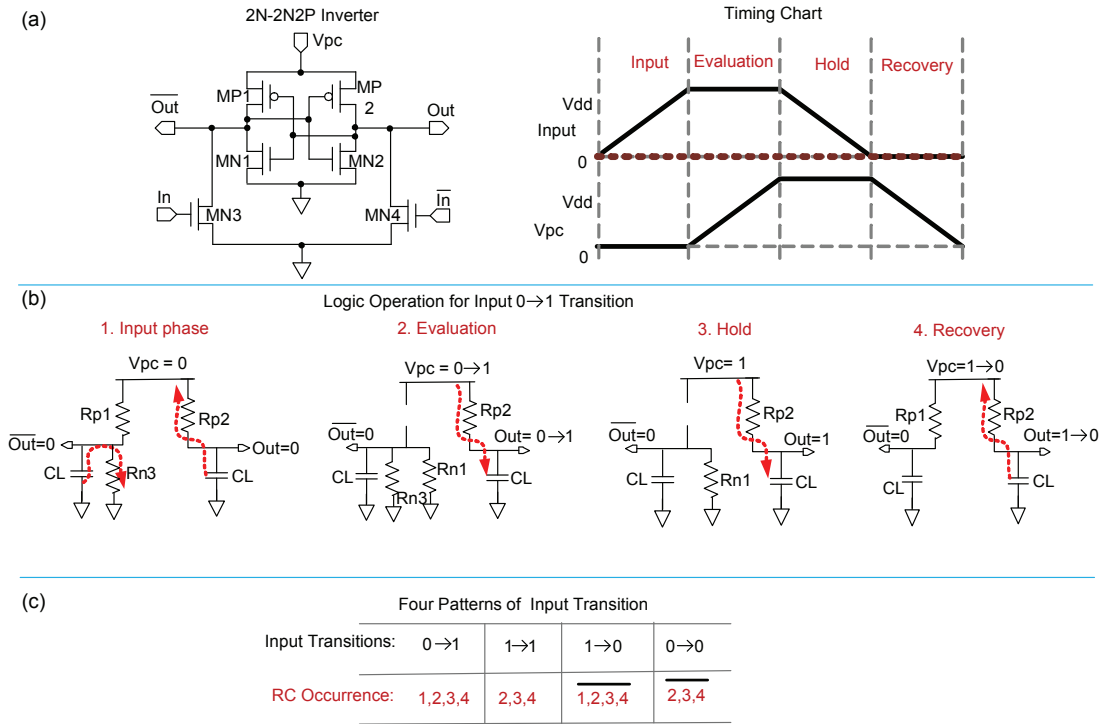


Figure 4.14: 2N-2N2P: (a) Inverter logic structure and its input signals, (b) Internal RC model at each respective phases, and (c) Occurrences of the RC model at four-possible input transitions.

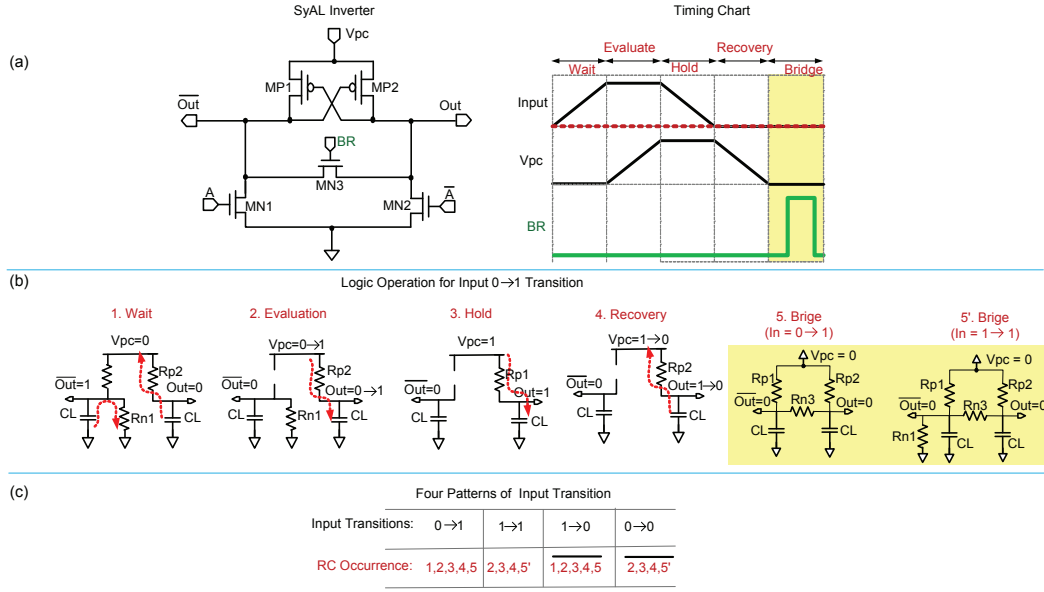


Figure 4.15: SyAL: (a) Inverter logic structure and its input signals, (b) Internal RC model at each respective phases, and (c) Occurrences of the RC model at four-possible input transitions.

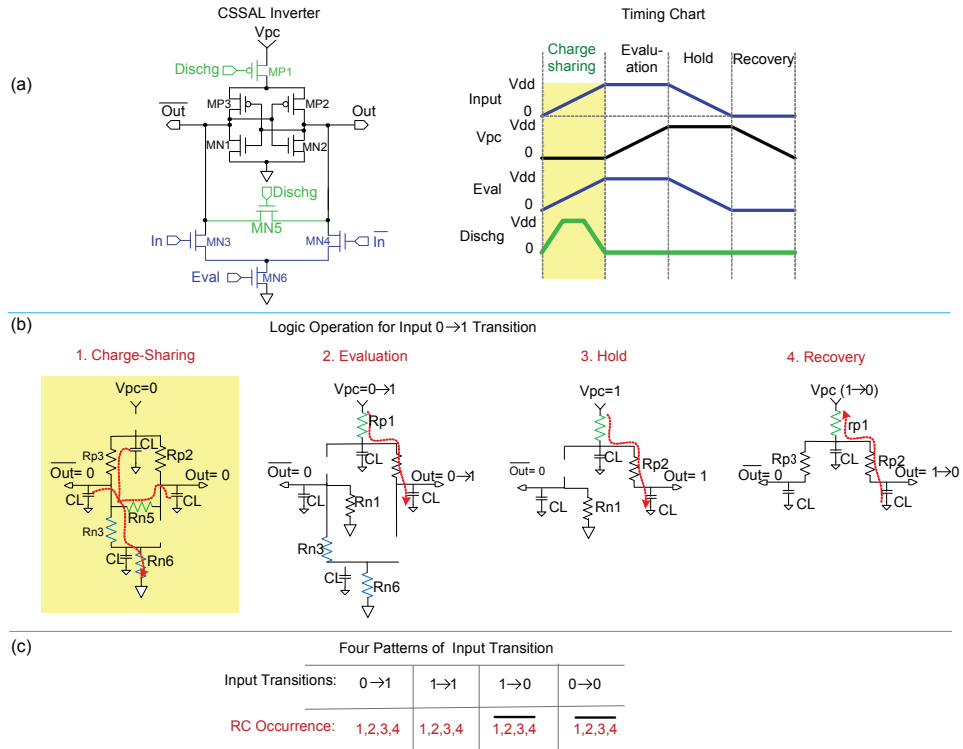
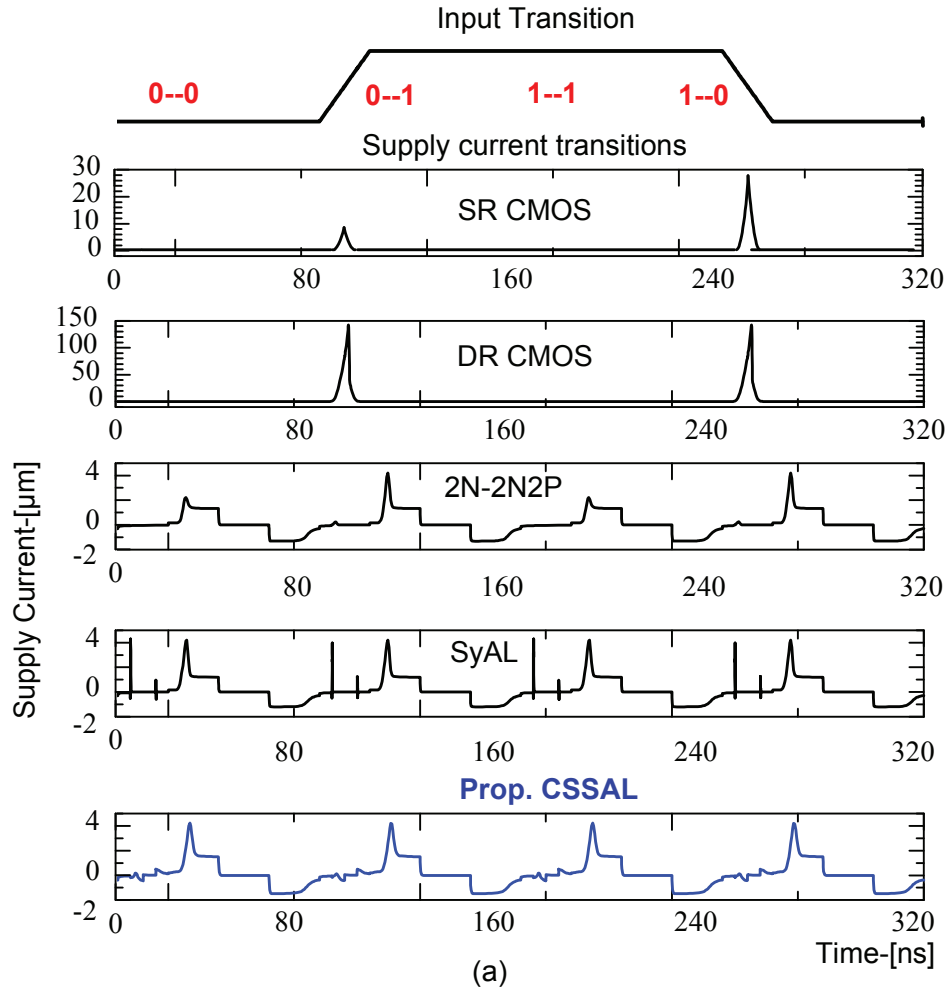


Figure 4.16: Proposed CSSAL: (a) Inverter logic structure and its input signals, (b) Internal RC model at each respective phases, and (c) Occurrences of the RC model at four-possible input transitions.



Transition	0-0	0-1	1-1	1-0	DPA Resistance
sCMOS	0	7	0	69	Vulnerable
DR-CMOS	0	142.404	0	142.403	Vulnerable
2N-2N2P	2.2116	4.20056	2.21223	4.26681	Weak
SyAL	4.28177	4.00384	4.28137	3.97839	Strong
Prop. CSSAL	4.24782	4.21374	4.24782	4.21352	Very Strong

Unit: μA order

(b)

Figure 4.17: Comparison of the supply peak current transition of inverter logic styles.

4.3.3 CSSAL NAND/AND

A transistor schematic of the NAND/AND logic of the CSSAL is depicted in Fig. 4.18 (a). Firstly, the author compares the circuit diagram with the SyAL, the improvement points and clarifies the logic operation at dual-input logic styles. As shown in the logic construction in Fig. 4.18(a) and Fig. 4.18(c) that the similarity is clear at PDN topology, however, the logic operation phases has been modified in CSSAL style. In SyAL, BR signal was set to pulse signal that caused some visible spikes at supply current traces, for example, these spikes can be observed in supply current trace of Fig. 4.11(b) at around 80-ns when BR signal edge rises up and falls down. The CSSAL completely removes these unwanted current spikes by setting the charge-sharing signal in trapezoidal waveform to slowly trigger the charge-sharing transistors. Moreover, the signal position was set to rise edge of input signals. The top level PMOS MP1 is OFF, hence no energy dissipation from power supply during charge-sharing phase, means no supply current flow at this particular phase. By doing this, the proposed CSSAL starts by setting all internal node capacitances to ground level when the input signal is in such that the active input signal $\geq V_{THN}$ before the power-clock signal arrives. This makes the proposed logic has balance low-peak supply current transitions, which is the unique different from SyAL, and is the idea behind the name so called charge-sharing symmetric adiabatic logic. In other word, the SyAL firstly evaluates output nodes before charge sharing phase, whilst the CSSAL is in opposite operation. The equivalent RC model of discharge (charge-sharing) and the evaluation phase can observed in Fig. 4.18(b) and Fig. 4.18(d).

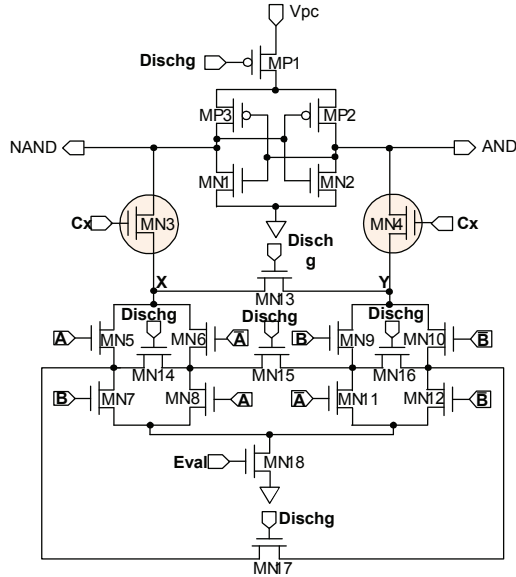
The advantage of using symmetric PDN topology instead of employing asymmetric universal DR PDN is described using RC model in Fig. 4.19. This analysis is done for evaluation/charging phase, the phase where dynamic power is consumed. In DR logic style, both complementary output wires are charge and discharged simultaneously. Let us consider the charges of internal parasitic node capacitance at charging line (logical 1 state) as Q1, Q2, and at discharge line (logical 0 state) as Qa, Qb, Qc, Qd. In symmetric PDN technique, same amount of energy are dissipated by internal resistance (R_L) and store as well in internal node capacitances (Q1+Q2). In contrary, the asymmetric PDN has unbalanced load, such as input pair (A,B = 1,0) has two nodes to be charged, the Q1+Q2, whilst the other pairs only Q1 is charged, as shown in Fig. 4.19(a). For better understanding, the author has summarized in Table 4.1. Moreover, asymmetric PDN has one floating internal

charge, labeled as Q_{float} at input pair (A,B = 0,0) that indicated in RC model of Fig. 4.19(a). In addition, there is no internal resistance for charge line at input pairs (A,B) = (0,0), (0,1), (1,1); however, the input pair (A,B = 1,0) has r_3 for additional energy loss.

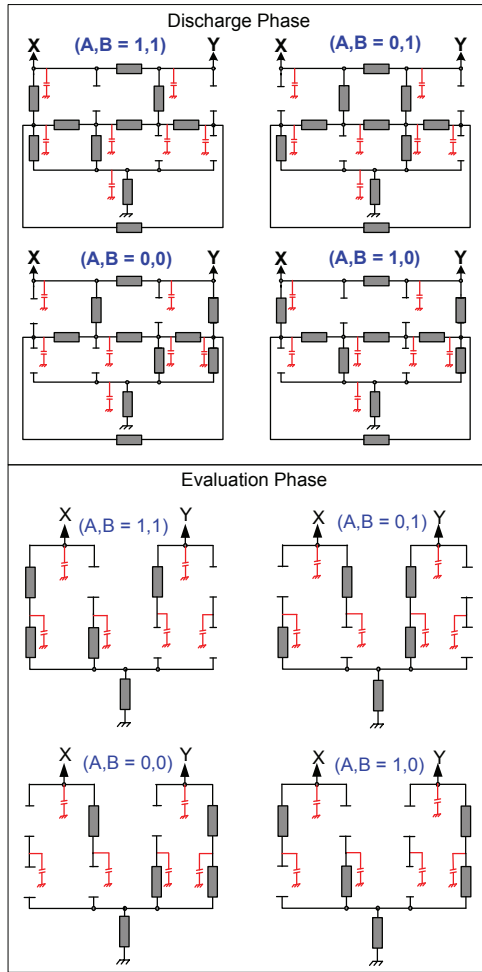
In the CSSAL circuit topology, the author inserts control signal (C_x) pass-transistors into the CSSAL structure as indicated in Fig. 4.13(a). This idea was obtained from CAL circuit [96]. The main role of the C_x pass transistors is to maintain the stability of the output during the charge-sharing phase. They also enable the proposed logic to consume the same amount of energy for all possible input transitions. However, the disadvantage of these pass transistors is high-energy consumption, which can be seen in the bottom of Fig. 4.20(b). Moreover, it is important to note that, applying extra pass transistors in the fundamental logic circuits and their implementation in the more complex digital circuit (e.g., multiplier) may affect the electric hazard (glitch) occurrence when the logic state is stable at a low or high level, which has been extensively analyzed in [98]– [100]. Careful analysis of simulations and physical measurements in [99] has shown that both unmasked and masked implementations leak side-channel information due to glitches at the output of logic gates. Furthermore, the author has found out this glitch current phenomenon in SPICE simulation results when C_x transistors were inserted into the CSSAL in a bit-parallel cellular multiplier over $GF(2^4)$. Therefore, the application of the CSSAL into the AES S-box circuit implementation, the control signal C_x pass transistors in the proposed CSSAL is considered as conditional transistors. For this reason, the inverter logic analysis in Fig. 4.16, the C_x pass-transistors were excluded.

The logic operation and supply current traces with and without C_x pass-transistors has similar results. Simulation results can be confirmed in Fig.4.20.

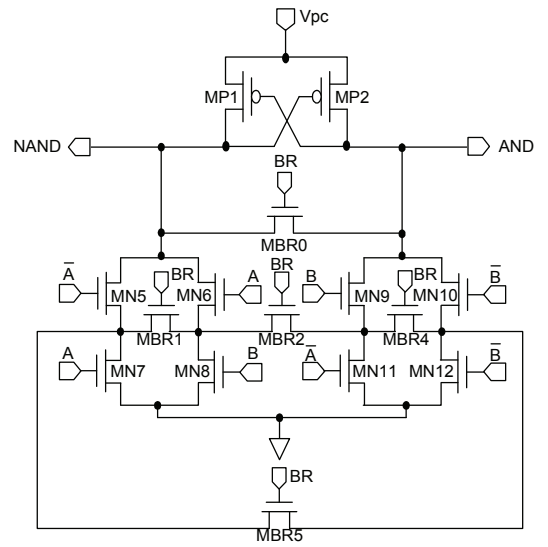
(a) Proposed CSSAL AND/NAND



(b) Proposed CSSAL AND/NAND internal RC model



(c) SyAL AND/NAND



(d) SyAL AND/NAND internal RC model

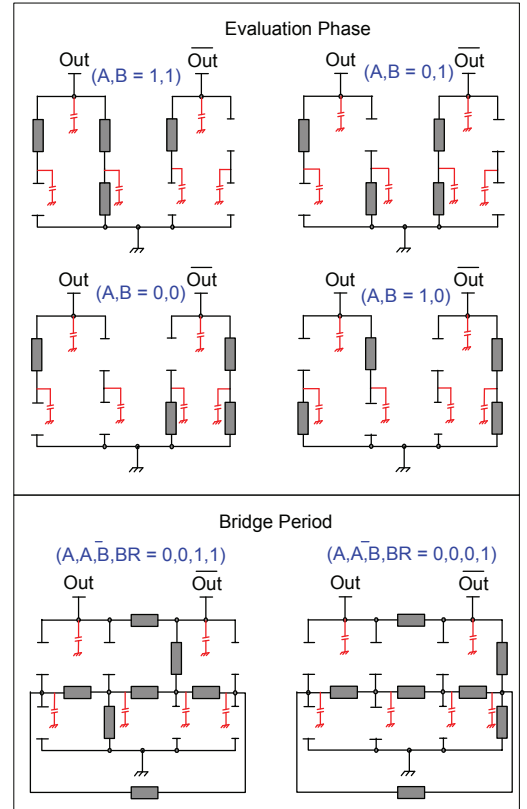


Figure 4.18: Proposed CSSAL vs. SyAL NAND/AND logic operation using RC model.

Table 4.1: Charging activity in asymmetric and symmetric DR PDN internal node capacitances (see Fig. 4.19). Symmetric PDN has balance charges than that of the asymmetric one.

Input activity	Asymmetric		Symmetric	
	Charge	Discharge	Charge	Discharge
A,B = 0,0	Q1	Qa	Q1, Q2	Qa, Qb, Qc, Qd
A,B = 0,1	Q1	Qa, Qb	Q1, Q2	Qa, Qb, Qc, Qd
A,B = 1,1	Q1	Qa, Qb	Q1, Q2	Qa, Qb, Qc, Qd
A,B = 1,0	Q1, Q2	Qa	Q1, Q2	Qa, Qb, Qc, Qd

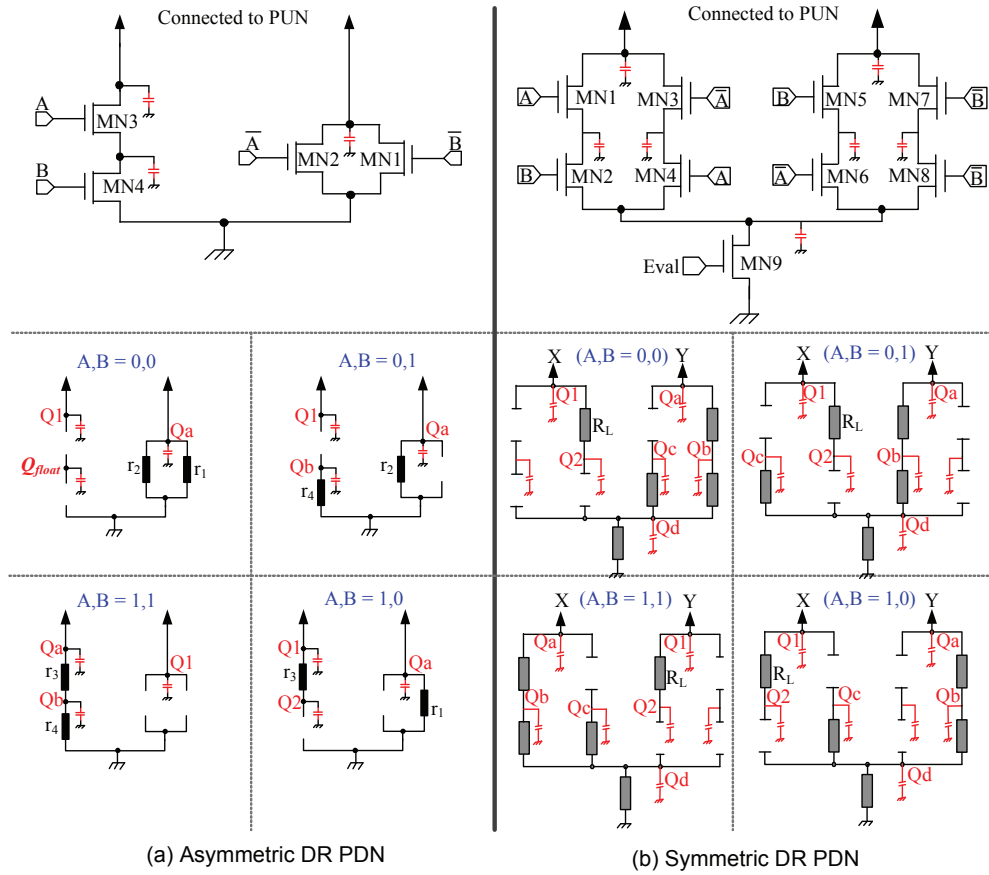


Figure 4.19: Symmetric vs. Asymmetric NAND/AND PDN topology for charging and discharging operation.

4.3.4 CSSAL XOR/XNOR

The logic structure of the CSSAL XNOR/XOR is same as the NAND/AND circuit topology, except that the positions of the input signals are different. Internal equivalent RC model, charging and discharging node parasitic capacitances are also same, as being previously explained. It can be observed in Fig. 4.21(b). The most commonly used DR XOR/XNOR internal equivalent RC model also depicted in Fig. 4.21(a). In this figure, it has same charges for all input conditions; for instance, (A,B) is from (1,1) to (0,1) transition has same (Q1,Q2) to (Q1,Q2), but the charge Q2 is still in the same node which has not changed during this transition. Consequently, different peak supply current still can be observed, for example can be seen in Fig. 4.6. The SPICE simulation results of CSSAL XNOR/XOR logic operation with and without C_x transistors can be seen in Fig. 4.22.

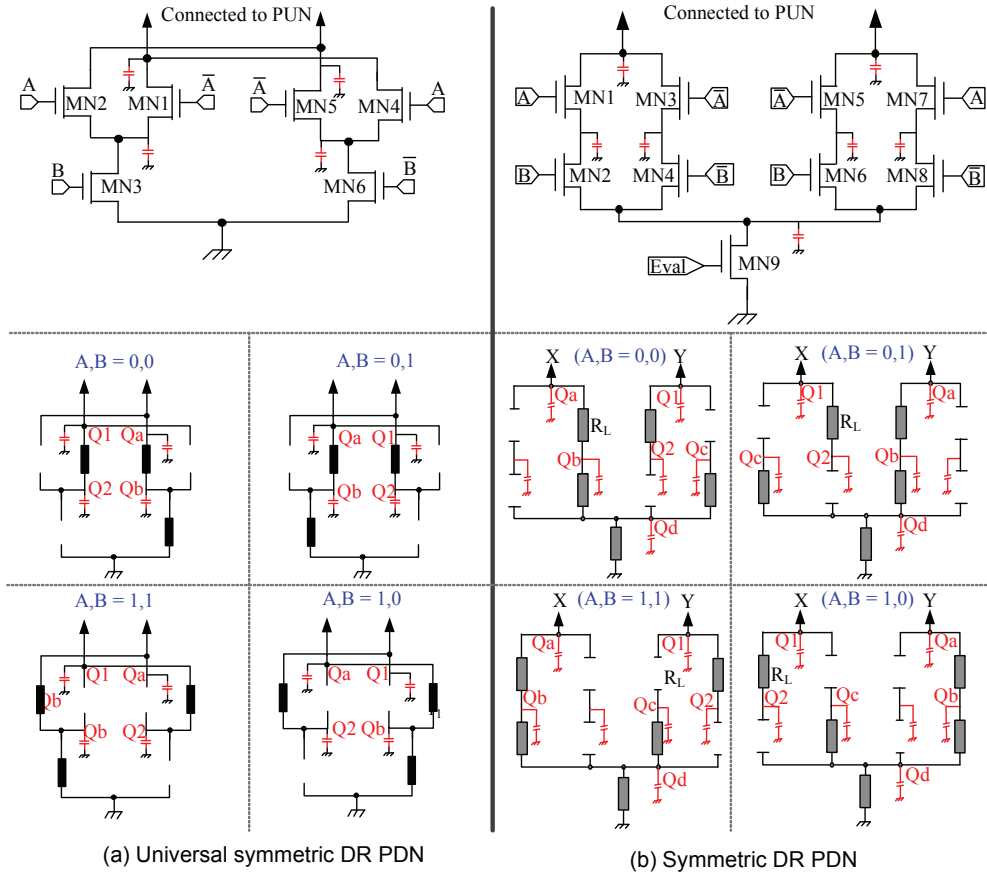


Figure 4.21: Symmetric vs. Asymmetric XOR/XNOR PDN topology for charging and discharging operation.

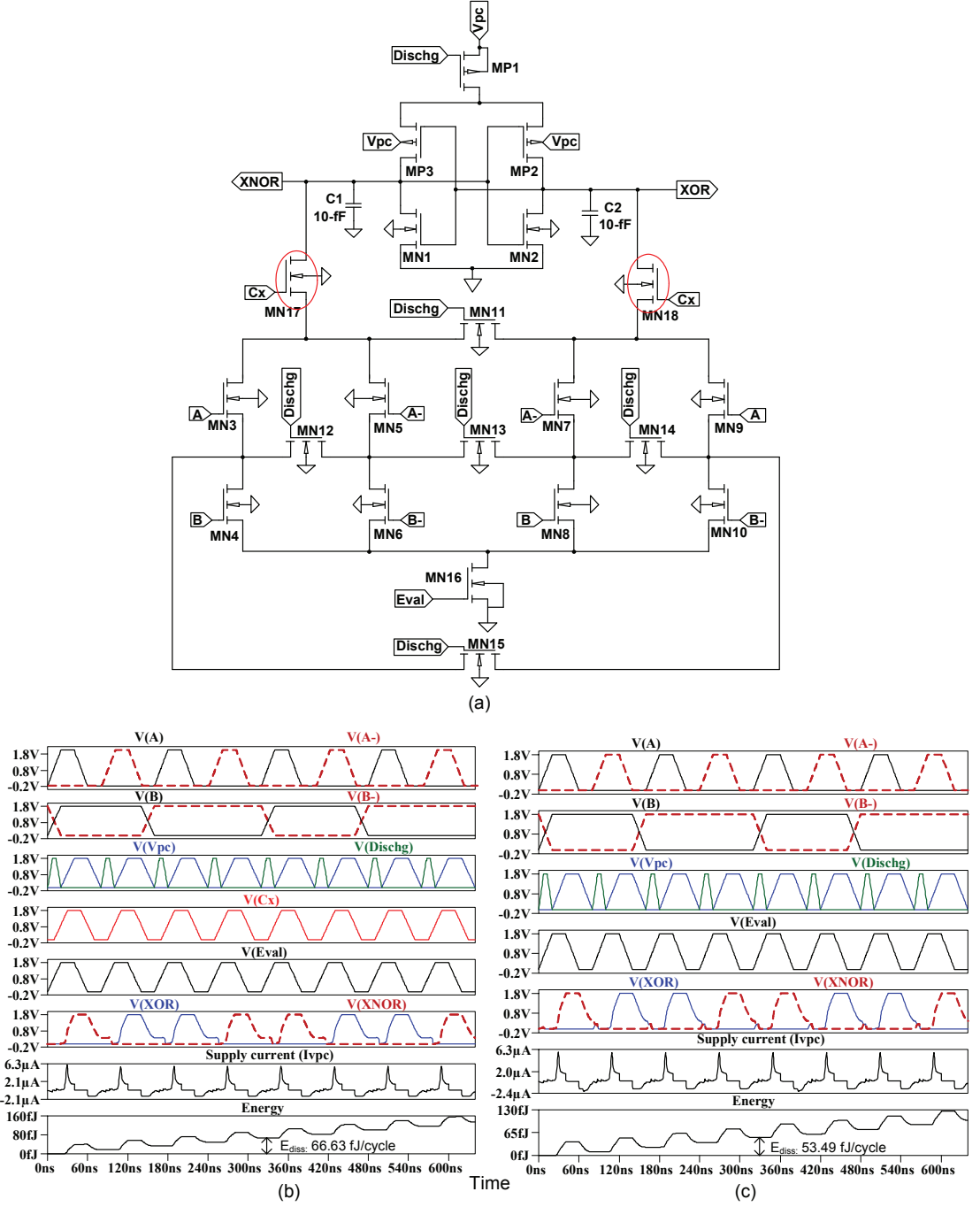


Figure 4.22: CSSAL: (a) Transistor schematic of XOR/XNOR with C_x pass-transistors, (b) Input-output signals with C_x pass-transistors, (c) Input-output signals without C_x pass-transistors.

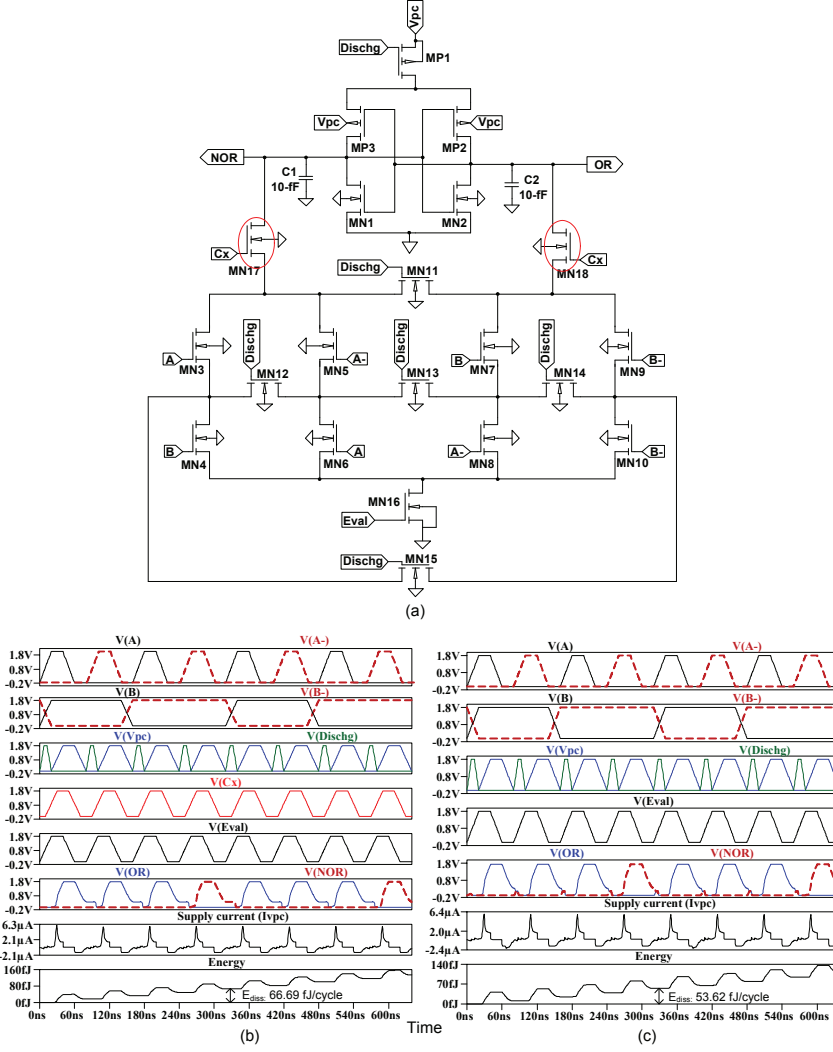


Figure 4.23: CSSAL: (a) Transistor schematic of OR/NOR with C_x pass-transistors, (b) Input-output signals with C_x pass-transistors, (c) Input-output signals without C_x pass-transistors.

4.3.5 CSSAL OR/NOR

The LSI implementation in this dissertation, the author only utilized XOR and AND gates. In this section the author just want to show the CSSAL OR/NOR logic operation function in SPICE simulation. Transistor schematic, the input-output waveforms with and without C_x transistor of CSSAL OR/NOR can be observed in Fig. 4.23.

4.3.6 Further Analysis of CSSAL AND/NAND Logic

In this section, the author discusses more detail about the CSSAL AND/NAND logic. The purpose is to investigate the possibility of gate reduction without violating the logic resistance to DPA attacks. The CSSAL AND/NAND logic namely ver.1, ver.2, ver.3 and ver.4 are depicted in Fig. 4.24. It represents also for XOR/XNOR and OR/NOR logic functions. The author has investigated each logic style and the peak current traces for 16-possible dual-input transition has been graphed, as depicted in Fig. 4.25. This result shows that ver.1–ver.4 has similar peak values, and hence they are applicable for secure LSI implementation with the preconditions of avoiding glitch current meaning, the dynamic hazards must be diminished or avoid by utilizing any suitable circuit among these four versions. The LSIs that will be presented in the next chapter has utilized the circuit ver.2 of XOR and AND logic functions.

In some specific complex digital circuitry, such bit serial multiplier or multiple stages logic circuitry where the input signals of the current stage are coming from the output of the previous stage may affect the circuit performance, in terms of delay and unwanted signals. This has become serious concern by the logic designers, specifically for those who are designing secure logic implementation, where the current-to-data dependence becomes the main challenge to be faced. Long path logic inversion (known as critical path) also produces dynamic hazard voltage (glitch current). To avoid this issues, the author has conducted further analysis using CSSAL NAND/AND chains for 10-serial blocks, as shown in Fig. 4.26(a). All output AND nodes from AND1–AND10 are plotted in order to check the possible block numbers that can be implemented, without the presence of dynamic hazards. The CSSAL ver.1–ver.4 in Fig. 4.24 are tested into Fig. 4.26(a). For instance, all output signals of CSSAL AND ver.1 is plotted in Fig. 4.26(b). This figure represents all CSSAL versions meaning, same simulation processes were done. As indicated in this figure, the dynamic hazard of all circuit versions has enlarged as shown in Fig. 4.26(c). From this figure we can see that dynamic hazard peak above transistor NMOS V_{thn} level as indicated in each graph should be avoided. Hence, CSSAL ver.1 and ver.3 are applicable for maximum 2-chains, CSSAL ver.2 can be used for 7-chains, and the CSSAL ver.4 can be used up until 9-chains.

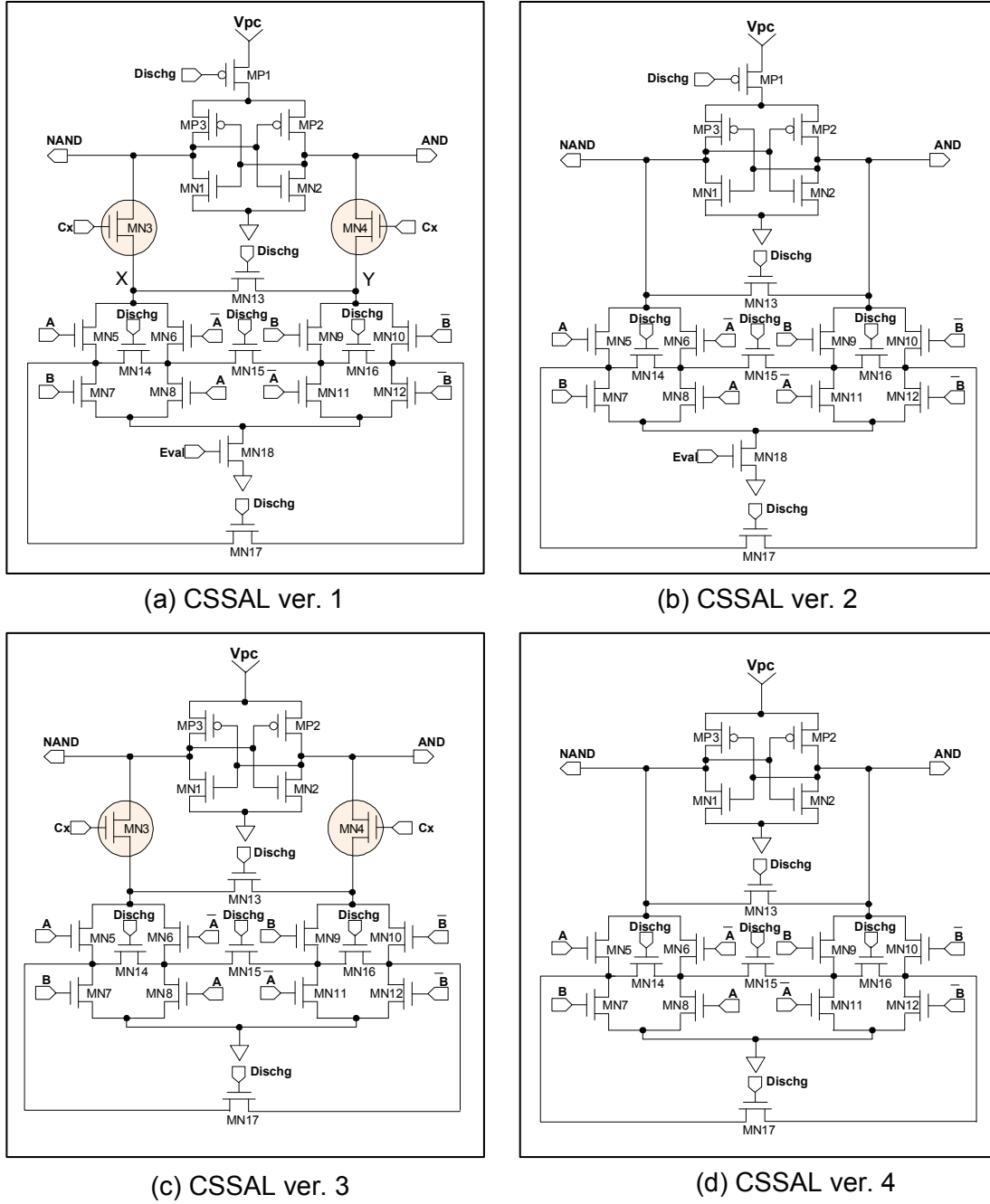


Figure 4.24: CSSAL AND/NAND: (a) CSSAL ver.1: Transistor schematic with C_x pass-transistors, (b) CSSAL ver.2: Transistor schematic without C_x pass-transistors, (c) CSSAL ver.3: Transistor schematic without MP1 and MN18 (Eval cell) transistors, and (d) CSSAL ver.4: Transistor schematic without MP1, MN18 and the C_x pass-transistors.

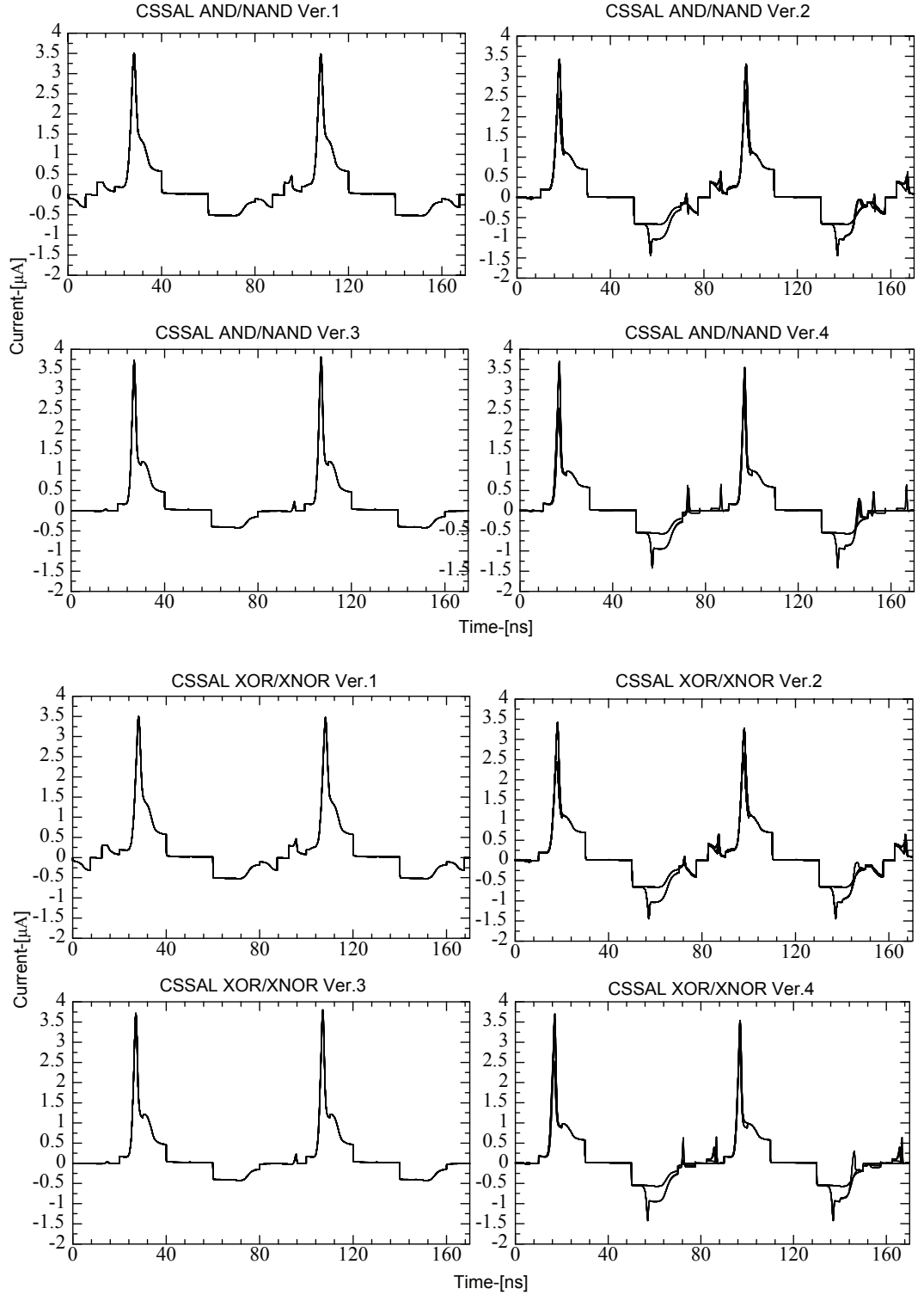


Figure 4.25: Supply current traces of all CSSAL versions for AND/NAND and XOR/XNOR logic circuits @ 12.5 MHz.

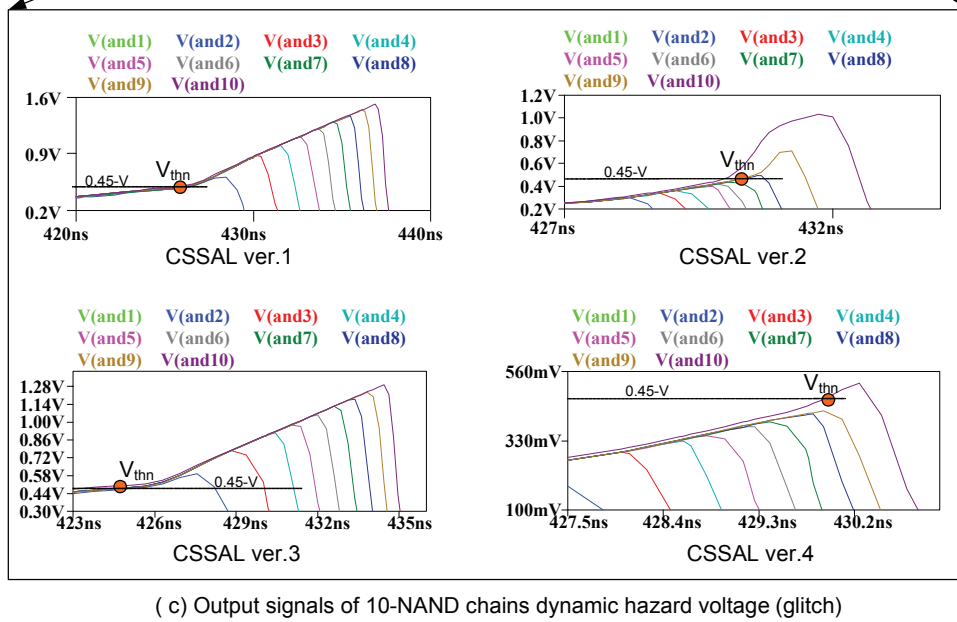
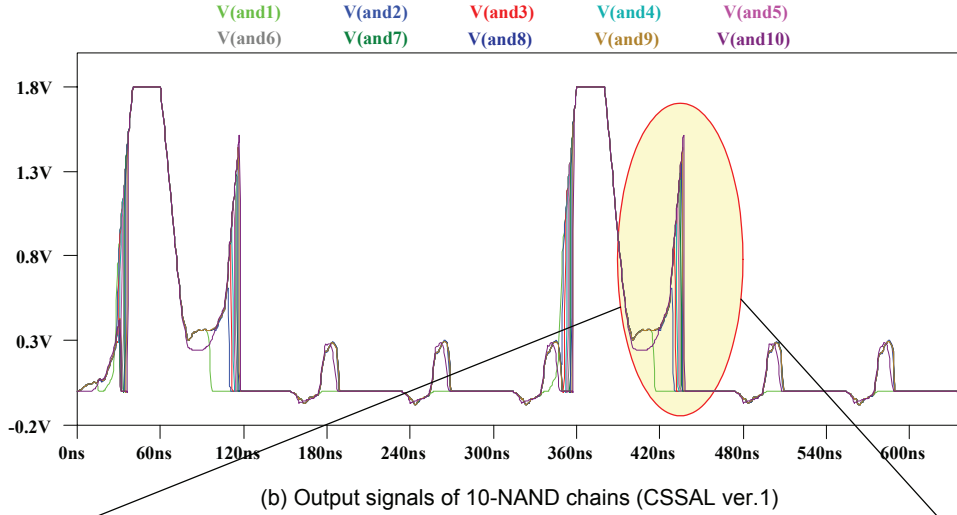
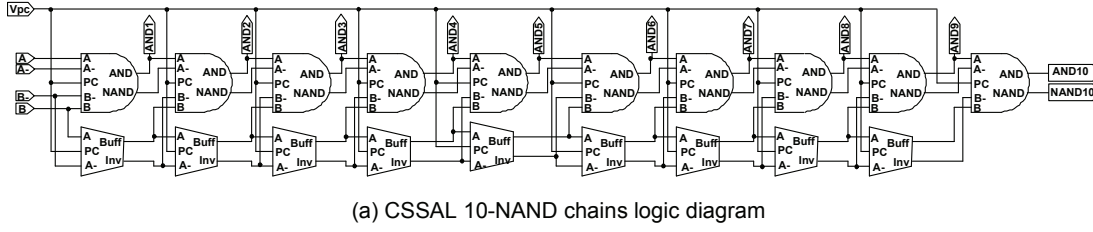


Figure 4.26: CSSAL critical path verification: (a) 10-NAND/AND chains logic diagram, (b) Output signals using CSSAL ver.1, (c) CSSAL ver.1–ver.4 dynamic hazard signals.

4.4 Comparison and Discussion

Figure of merit to measure the resistance of the logic against power analysis attack is discussed in this section. The comparison data compose of all logic circuits that implemented using hiding technique, *i.e.*, the conventional TDPL, SABL, 2N-2N2P, SAL, SyAL and the proposed CSSAL NAND/AND and XNOR/XOR logic styles. The simulation and investigation results of these logics are compared in three different operating frequencies, at 1.25 MHz, 12.5 MHz and 125 MHz. Figure 4.27 shown supply current traces of all 16 possible dual-input transition for NAND/AND investigating logics at power clock frequency of 12.5 MHz. Similar simulation results at 1.25 MHz and 125 MHz are shown in Fig. 4.28. From these results, we can see that the CSSAL shows only a single line for 16 different data among the conventional SyAL and 2N-2N2P logic styles, which means that CSSAL consume more uniform energy in adiabatic mode at low frequency band. At high frequency speed (125 MHz) the proposed logic circuit performance is degraded; in this matter, the appropriate frequency range should be chosen when implementing the proposed logic. For instance, the crypto devices that operate at low speed and low-power requirement, such as smart card or RFID tags. The SAL also exhibits uniformly peak current at individual logic level, however, it controls by several clock signals which may difficult in more complex system LSI. The conventional TDPL and SABL CMOS logic style also consume uniform energy, however their peak current are about 27 and 42 times higher than that of the CSSAL.

Calculation results in Table 4.2 show the important parameters of power simulation verification. It has been explained earlier in Section 2.4.1 that if we achieve extremely low value of those parameters, then the circuit is classified as robust secure logic against DPA attacks. Therefore, the results in this table indicate that the proposed CSSAL consumes more uniformly low-power at lower frequency bands. This table also confirms that the TDPL has more constant power for all frequency range in this comparison study.

In addition, the author has provided information of the gate size in corresponding to their respective cyclical energy dissipation of all logic styles investigated as summarized in Table 4.3. From this data, we can identify that the drawback of the proposed logic is area consuming. Moreover, its energy dissipation is higher among the adiabatic logic styles, but significantly low in comparison with conventional secure CMOS logic styles, such as TDPL or SABL.

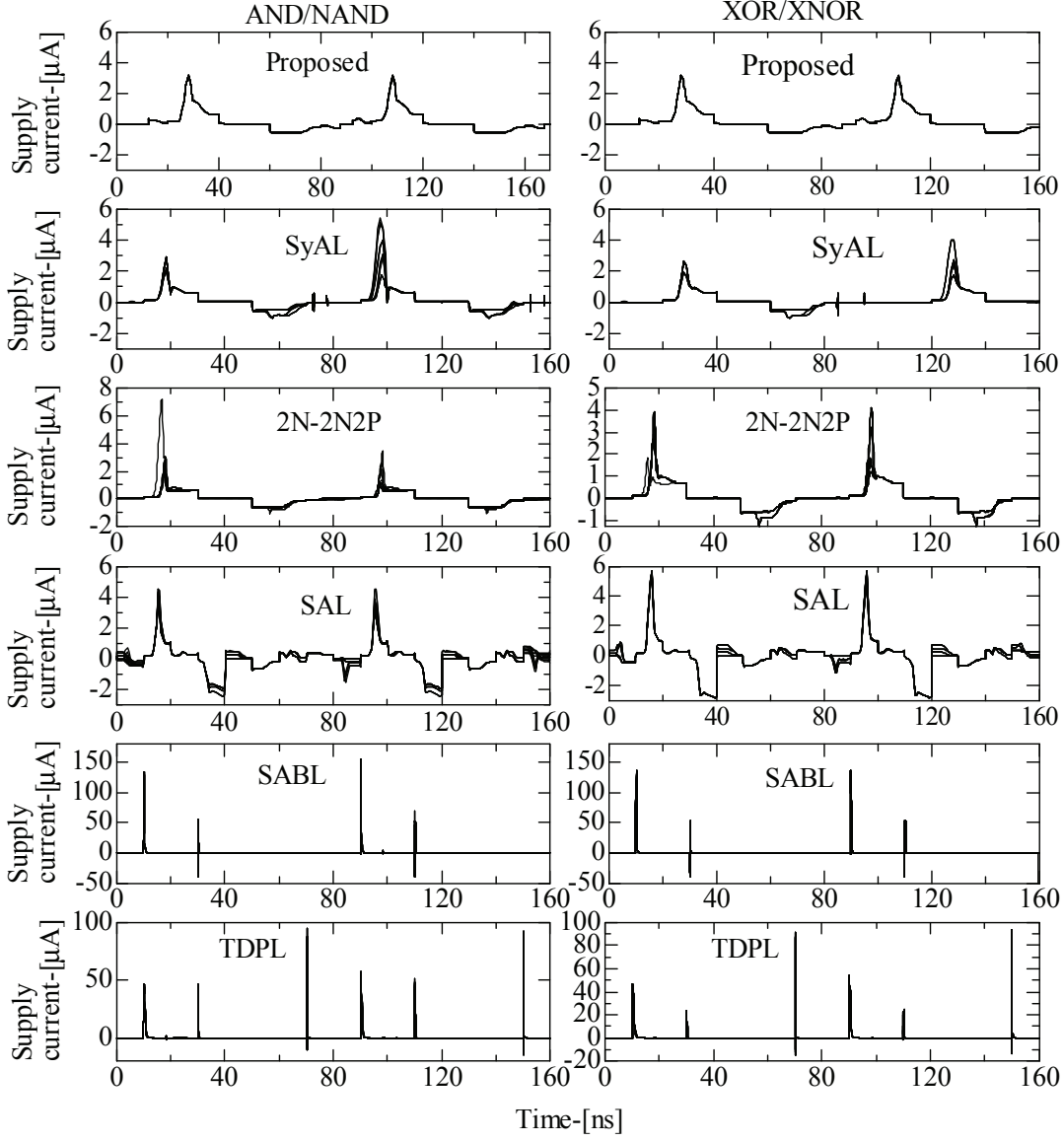


Figure 4.27: All possible dual-input 16 transitions of investigated individual logics at 12.5 MHz for AND/NAND gate (left) and XOR/XNOR gate (right).

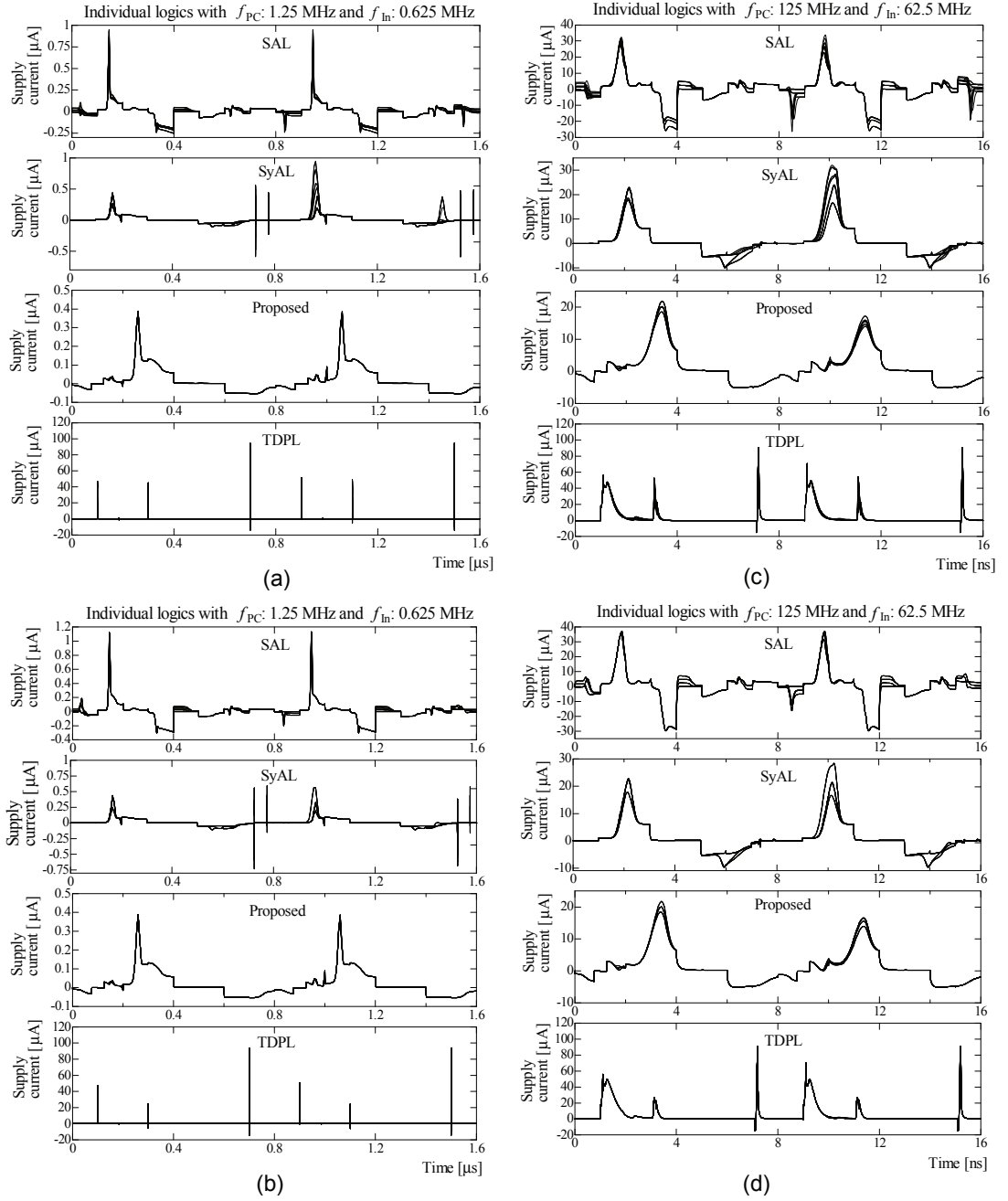


Figure 4.28: Supply current transitions of individual logics at different frequencies. (a) 1.25 MHz AND gate, (b) 1.25 MHz XOR gate, (c) 125 MHz AND gate. (d) 125MHz XOR gate.

Table 4.2: Simulation and calculation results of AND/AND and XOR/XNOR individual logics.

Individual NAND/AND logic												
Freq. [MHz]	SAL			SyAL			Proposed			TDPL		
	1.25	12.5	125	1.25	12.5	125	1.25	12.5	125	1.25	12.5	125
E_{min} [fJ]	5.23	7.43	15.21	9.03	0.73	13.53	19.79	21.45	16.65	128.08	121.77	119.77
E_{max} [fJ]	12.13	12.09	21.78	20.14	19.66	24.20	20.07	21.70	21.47	134.82	124.39	125.29
\overline{E} [fJ]	6.36	8.60	17.50	12.39	12.78	18.48	19.92	21.59	19.48	130.77	124.39	121.97
σ_E [fJ]	1.55	1.11	1.85	3.83	2.98	3.45	0.08	0.09	1.48	2.17	0.57	1.69
NED [%]	56.99	38.51	30.17	55.18	50.46	44.08	1.39	1.15	22.45	4.99	2.11	4.41
NSD [%]	24.23	12.85	10.56	30.94	23.32	18.68	0.44	0.42	7.59	1.66	0.46	1.39
$F_oM(\sigma_E \overline{E})$ [fJ ²]	9.86	9.55	32.38	47.45	38.08	63.76	1.59	1.95	28.79	283.77	70.90	206.13
Individual XNOR/XOR logic												
Freq. [MHz]	SAL			SyAL			Proposed			TDPL		
	1.25	12.5	125	1.25	12.5	125	1.25	12.5	125	1.25	12.5	125
E_{min} [fJ]	5.38	8.95	21.11	5.32	6.77	12.96	19.80	21.59	16.65	142.86	127.26	132.39
E_{max} [fJ]	8.12	12.22	29.49	12.79	13.27	20.11	20.09	21.79	19.84	143.79	128.14	132.77
\overline{E} [fJ]	6.86	10.74	25.74	9.58	10.20	9.58	19.92	21.68	18.87	143.33	127.74	132.63
σ_E [fJ]	1.07	1.35	3.20	2.75	2.59	2.49	0.10	0.07	1.29	0.33	0.36	0.15
NED [%]	33.78	26.78	28.43	58.39	48.98	35.33	1.38	0.92	16.09	0.65	0.69	0.29
NSD [%]	15.63	12.57	12.43	28.70	25.39	18.37	0.52	0.32	6.86	0.23	0.28	0.11
$F_oM(\sigma_E \overline{E})$ [fJ ²]	7.34	14.50	82.04	26.34	26.42	23.85	1.99	1.52	24.34	47.30	45.99	19.89

Table 4.3: Number of gates and the cyclical energy comparison of AND and XOR logic styles .

Number of Gate and Energy Comparison @ 12.5 MHz				
Circuit	AND		XOR	
	No. Gate	E/cycle [fJ]	No. Gate	E/cycle [fJ]
scCMOS	6	136.76	16	399.09
DCVSL	6	200.76	8	201.20
Dynamic logic	4	83.75	6	137.86
SABL	12	335.80	14	372.71
TDPL	14	485..67	16	504.67
ECRL	6	20.12	8	26.84
2N-2N2P	8	14.78	10	26.23
SAL	14	28.53	16	35.49
SyAL	15	45.60	15	44.03
CSSAL ver.1	21	66.63	21	66.69
CSSAL ver.2	19	50.37	19	53.62
CSSAL ver.3	19	57..10	19	57.16
CSSAL ver.4	17	38.56	17	42.19

4.5 Summary

The author has introduced the proposed CSSAL along with the insightful analysis and investigation of the fundamental logic styles. Comparative study which composed of the conventional dual-rail adiabatic logic styles has been conducted, aiming to understand deeply about the figure of merit and as well as the drawbacks of the proposed logic. Throughout the analysis of the circuit topology and some accurate SPICE simulation results ensure the author to underline the contents of this chapter in the following summary:

1. It is difficult to design a robust secure digital logic circuit that can responds to all costs; such as small gate (low area), ultra-low power and high speed. Therefore, the circuit design merit and the characteristic of the application target should be prioritized.
2. Thoroughly analysis and investigation at the fundamental logic level are crucial important. This step is a pre-condition for better design of secure LSI implementation. The author has suggested that the long critical path of a cell circuit should be paid extra attention; this may severely degrade the circuit performance against power analysis attack. Long critical path in the logic inversion normally produces glitch current and undesirable delay time, as a result, the information leakage difficult to prevent [99].
3. Investigation and evaluation results have shown that the proposed CSSAL consumes more uniform (nearly constant) energy at low frequency ranges. This can be verify in Table 4.2, that CSSAL has very small values of NSD and the energy variance of σ_E . Moreover the supply current transition results, for instance, in Fig. 4.27 has shown that the CSSAL exhibits only a single line for 16 different data transitions in comparison with adiabatic logic styles, and significantly low peak values in comparison to that of the TDPL and SABL.

Chapter 5

Logic Implementation

5.1 Introduction

This chapter deals with the proposed logic implementation in the bit-parallel cellular multiplier over $GF(2^4)$ and an 8-bit AES S-box circuit. The circuits' resistance will be verified using power simulation verification that the author has introduced in the earlier section 2.4. Using the same circuit diagram, the author also implements several benches of the conventional logic styles in order to validate the proposed logic's resistant to thwart power analysis attack. After all, the author will discuss the SPICE simulation results from the view point of security metric and the global power consumption in the end of this chapter.

5.2 Bit Parallel Cellular Multiplier over $GF(2^m)$

Galois field (GF) (named after the French Mathematician Évariste Galois) arithmetic has an important role in coding theory, computer algebra and cryptographic algorithms. For these applications, the fundamental arithmetic operations, *i.e.*, addition, subtraction, multiplication, inversion and exponentiation are efficiently used in the algorithm design and implementation. In the system VLSI design, this field has been widely employed based on the laws presented in [106], the first work that described parallel-in-parallel-out multiplier of a cellular-array architecture. The current useful AES encryption standard algorithm also operates over finite field $GF(2^8)$ for computational efficiency, high resistance to cryptanalysis, hardware and software compatibility, and flexibility. Since the new AES standard was announced, much effort has been expanded [101]–[105] to simplify the finite field over $GF(2^8)$ in S-Box transformation to $GF((2^4)^2)$ and $GF(((2^2)^2)^2)$ for low cost, low power consumption,

and low complexity.

In order to verify the capability of the proposed CSSAL for counteracting power analysis attacks, the author has implemented the individual logics with the existing bit-parallel cellular multiplier in [107]. This multiplier analytically explored the inner product multiplication algorithm for calculating AB^2 in a class field $GF(2^m)$ using a cellular architecture that involves low-complexity and less computation time. In this implementation, the polynomial of degree $m = 4$, with arithmetic calculations have been performed to define the cellular array multiplication of $AB^2 = \sum_{j=0}^m A^{(2j)}[B^2]^{(-j)}$ to calculate the function block of the bit-parallel multiplier over $GF(2^4)$.

5.2.1 Review of AB^2 Multiplier over $GF(2^m)$

In [107], finite field $GF(2^m)$ contains 2^m element $\{0, 1, \beta, \beta^2, \dots, \beta^{N-1}\}$, where $N = 2^m - 1$ and β is root of primitive polynomial. Let α be a root of $p(x)$, where $p(x)$ is all one polynomial (AOP) of degree m , i.e., $p(x) = 1 + x + x^2 + \dots + x^m$ over $GF(2)$. Then, any $a \in GF(2^m)$ can be represented by a canonical basis such as $a = a_0 + a_1\alpha + a_2\alpha^2 + \dots + a_{m-1}\alpha^{m-1}$, where $a_i \in GF(2)$ for $i = 1, 2, 3, \dots, m-1$, and $\{1, \alpha, \alpha^2, \dots, \alpha^{m-1}\}$ is canonical basis of $GF(2^m)$. Since $p(\alpha) = 0$, $\alpha^m = 1 + \alpha + \alpha^2 + \dots + \alpha^{m-1}$, and a common computation in both multiplication is multiplying by α . We can obtain as follows:

$$\alpha^{m+1} + 1 = 0 \quad (5.1)$$

There are several definitions and theorem based on the property of AOP that will facilitate the design of the bit-parallel cellular architectures for computing AB^2 multiplication. The following definitions will be used to configure the bit-parallel multiplier and the connection of the inner cells in respect to how the primary input signals will be connected. **Definition1** : $A^{(j)}$ will donates the element A as the periodic shift- j -bit-by-right operation, i.e., let

$$A = a_0 + a_1\alpha + a_2\alpha^2 + \dots + a_{m-1}\alpha^{m-1} + a_m\alpha^m, \quad \text{then}$$

$$A^{(j)} = a_{\langle j \rangle} + a_{\langle 1+j \rangle}\alpha + a_{\langle 2+j \rangle}\alpha^2 + \dots + a_{\langle m-1+j \rangle}\alpha^{m-1} + a_{\langle m+j \rangle}\alpha^m \quad (5.2)$$

where $\langle x \rangle$ denotes x modulo $m + 1$. Similarly, $A^{(-j)}$ is equivalent to the periodical shifting j bits to the left, such as

$$A^{(-j)} = a_{\langle -j \rangle} + a_{\langle 1-j \rangle}\alpha + a_{\langle 2-j \rangle}\alpha^2 + \dots + a_{\langle m-1-j \rangle}\alpha^{m-1} + a_{\langle m-j \rangle}\alpha^m. \quad (5.3)$$

Generalizing that the coefficient of A relate between $A^{(j)}$ and $A^{(-j)}$, then

$$A = A^{(-j)}\alpha^j = A^{(j)}\alpha^{(-j)} \quad (5.4)$$

Definition2 : $[A^2]^{(j)}$ will donate the square element, A^2 , as periodic shift-j-bit-by-right operation, i.e., let

$$\begin{aligned} A^2 &= (a_0 + a_1\alpha + a_2\alpha^2 + \cdots + a_{m-1}\alpha^{m-1} + a_m\alpha^m)^2 \\ &= a_0 + a_1\alpha^2 + a_2\alpha^4 + \cdots + a_{m-1}\alpha^{2(m-1)} + a_m\alpha^{2m}, \quad \text{then} \end{aligned}$$

$$[A^2]^{(j)} = a_{\langle j \rangle} + a_{\langle 1+j \rangle}\alpha^2 + a_{\langle 2+j \rangle}\alpha^4 + \cdots + a_{\langle m-1+j \rangle}\alpha^{2(m-1)} + a_{\langle m+j \rangle}\alpha^{2m} \quad (5.5)$$

Similarly, $[A^2]^{(-j)}$ is equivalent to the periodically shifting j bit to the left, such as

$$[A^2]^{(-j)} = a_{\langle -j \rangle} + a_{\langle 1-j \rangle}\alpha^2 + a_{\langle 2-j \rangle}\alpha^4 + \cdots + a_{\langle m-1-j \rangle}\alpha^{2(m-1)} + a_{\langle m-j \rangle}\alpha^{2m} \quad (5.6)$$

Generalizing that the coefficient of A^2 relate between $[A^2]^{(j)}$ and $[A^2]^{(-j)}$, the

$$A = [A^2]^{(-j)}\alpha^{2j} = [A^2]^{(j)}\alpha^{-2j} \quad (5.7)$$

Definition3 : Let A and B over $GF(2^m)$ be generated by an irreducible AOP $p(x)$, that is

$$A = a_0 + a_1\alpha + a_2\alpha^2 + \cdots + a_{m-1}\alpha^{m-1} + a_m\alpha^m, \quad a_m = 0$$

$$B^2 = b_0 + b_1\alpha^2 + b_2\alpha^4 + \cdots + b_{m-1}\alpha^{2(m-1)} + b_m\alpha^{2m}, \quad b_m = 0.$$

Then the inner product of $A.B^2$ is defined as follows

$$A.B^2 = a_0b_0 + a_1b_1\alpha^3 + a_2b_2\alpha^6 + \cdots + a_{m-1}b_{m-1}\alpha^{3(m-1)} + a_mb_m\alpha^{3m} \quad (5.8)$$

Definition4 : Based on the Definition 3, the i th inner product can be defined as follows:

$$\begin{aligned} A^{(j)}.[B^2]^{(-j)} &= a_{\langle j \rangle}b_{\langle -j \rangle} + a_{\langle 1+j \rangle}b_{\langle 1-j \rangle}\alpha^3 + a_{\langle 2+j \rangle}b_{\langle 2-j \rangle}\alpha^6 + \cdots \\ &\quad + a_{\langle m-1+j \rangle}b_{\langle m-1-j \rangle}\alpha^{3(m-1)} + a_{\langle m+j \rangle}b_{\langle m-j \rangle}\alpha^{3m}, \end{aligned} \quad (5.9)$$

where $A^{(0)}.[B^2]^{(0)} = A.B^2$.

Theorem1 : Let A and B over $GF(2^m)$ be produced by an irreducible AOP $p(x)$ of degree m , the product of AB^2 can be represented by the following recursive formula:

$$AB^2 = A^{(0)}[B^2]^{(0)} + A^{(2)}[B^2]^{(-1)} + A^{(4)}[B^2]^{(-2)} + \dots + A^{(2m)}[B^2]^{(-m)} \quad (5.10)$$

In order to proof the Theorem 1, let define the summation of Eq. 5.9 in Definition 4 as follows:

$$\begin{aligned} \sum_{j=0}^m A^{(2j)}[B^2]^{(-j)} &= \sum_{j=0}^m (a_{\langle j \rangle} b_{\langle -j \rangle} + a_{\langle 1+j \rangle} b_{\langle 1-j \rangle} \alpha^3 + a_{\langle 2+j \rangle} b_{\langle 2-j \rangle} \alpha^6 + \dots \\ &\quad + a_{\langle m+j \rangle} b_{\langle m-j \rangle} \alpha^{3m}) \\ &= \sum_{j=0}^m \sum_{i=0}^m a_{\langle i+j \rangle} b_{\langle i-j \rangle} \alpha^{3i} \end{aligned} \quad (5.11)$$

Let $AB^2 = d_0 + d_1 \alpha^3 + \dots + d_m \alpha^{2m}$, where d_i is data at i th as follows:

$$d_i = \sum_{j=0}^m (a_{\langle i+j \rangle} b_{\langle i-j \rangle} \pmod{2}). \quad (5.12)$$

By using the AOP property of $\alpha^{m+1} = 1$, then, AB^2 can be simplified as follows:

$$\begin{aligned} AB^2 &= a_0 b_0 + a_1 b_1 \alpha^3 + a_2 b_2 \alpha^6 + \dots + a_{m-1} b_{m-1} \alpha^{3(m-1)} + a_m b_m \alpha^{3m} \\ &= c_0 + c_1 \alpha + c_2 \alpha^2 + \dots + c_m \alpha^m, \end{aligned} \quad (5.13)$$

where the c_k is defined as

$$c_k = \sum_{k=i+j \pmod{m+1}}^m a_i b_j \pmod{2}. \quad (5.14)$$

To compare the coefficient of Eqs. (11) and (13), obtain

$$d_j = c_{\langle 3j \rangle}, \quad \text{for } 0 \leq j \leq m$$

Therefore, it turns out that the Theorem 1 becomes $AB^2 = \sum_{j=0}^m A^{(2j)}[B^2]^{(-j)}$. Based on this definitions and theorem, a low complexity and low computation time of multiplier architecture were successfully designed.

The configuration of the inner cell multiplier circuit in Fig. 5.1(a) composes of $(m+1)^2$ of the basic cells, denoted as A-cell which operates $ab + c$ computations. The complexity of A-cell includes one 2-input AND and one 2-input XOR gate, as depicted in Fig. 5.2(a). Moreover, complexity of the improved multiplier circuit in Fig. 5.1(b) has $(m+1)^2 + (m+1)$ of B-cell + C-cell, respectively, as shown in Fig. 5.2(b) and (c).

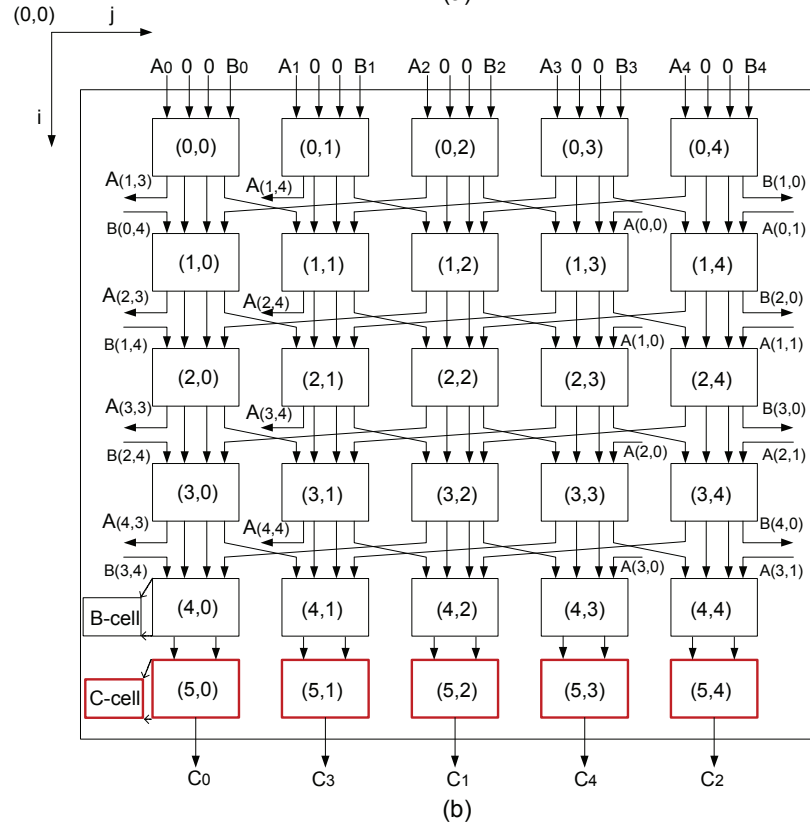
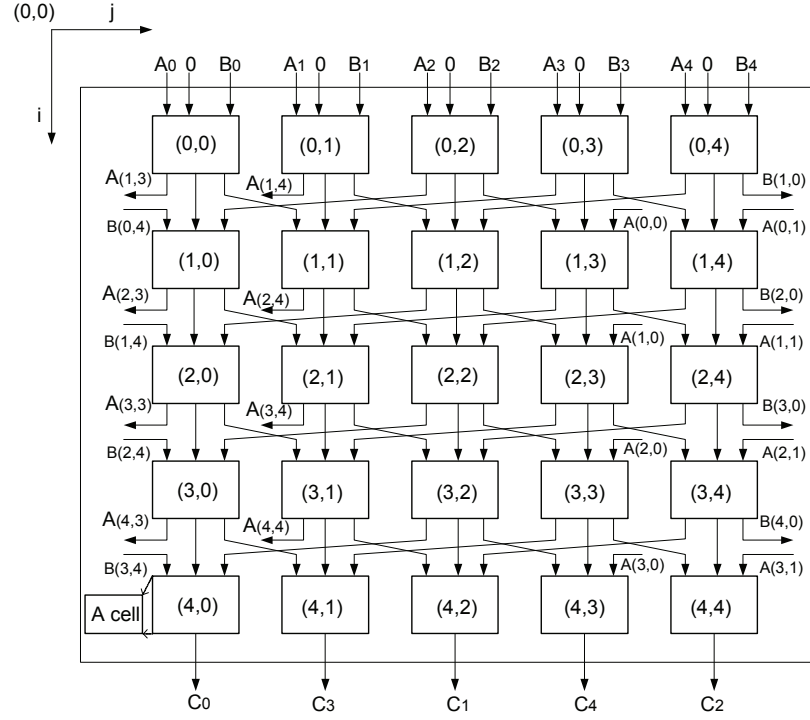


Figure 5.1: Circuit diagram of the bit-parallel multiplier over $GF(2^4)$: (a) Inner cell using A-cell circuit and (b) Inner cell using B, C-cells circuit.

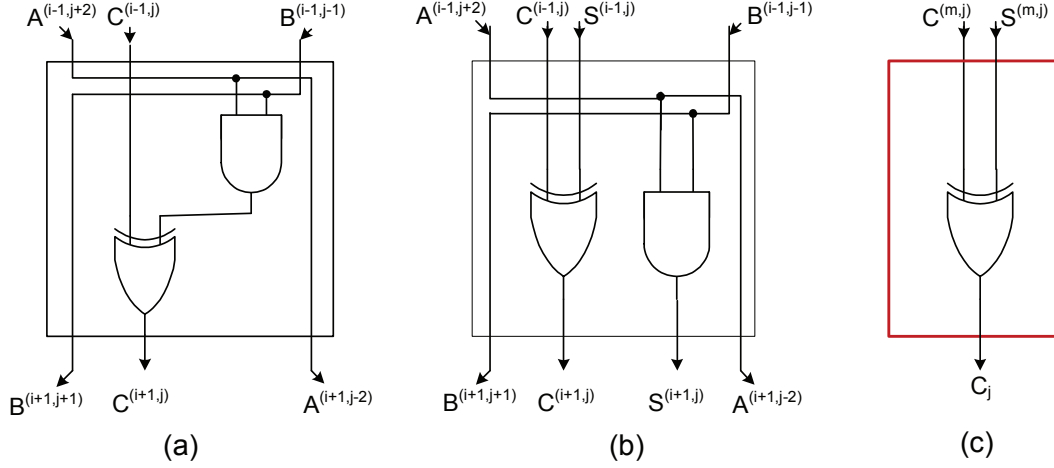


Figure 5.2: Inner cell circuits of Multiplier over $GF(2^4)$: (a) A-cell circuit, (b) B-cell circuit and (c) C-cell circuit.

5.2.2 SPICE Simulation and Results

The implementation of fundamental logic into multiplier are AND and XOR gates, as shown in Fig. 5.2. The author implements all the logic styles (proposed CSSAL, SAL, 2N-2N2P, SyAL, TDPL and SABL) into the multiplier circuit in Fig. 5.1(a). The objective of utilizing these six different logic styles is to investigate and compare more fair data to validate the merit of the proposed logic in the LSI implementation.

The input signals of CSSAL, SyAL, SAL, and TDPL are shown in Figs. 5.3– 5.6, respectively. Power clock signals and primary dual-inputs (A, B) signals are same as individual logics discussed in the previous chapter. In the multiplier simulation, 4-bit input signals are applied to all circuit styles. And therefore the primary input signals of multiplier in Fig. 5.1 are grouped as: $A_0=A_2=A_4$, $B_0=B_2=B_4$, $A_1=A_3$, and $B_1=B_3$ from the least significant bit to the most significant bit, respectively. The Vpc signal, input A_0 – A_4 and B_0 – B_4 signals in Fig. 5.3 are also apply for 2N-2N2P, except the phase delay of Vpc is set to 50% of individual logic phase delay in order to obtain good output result of the multiplier circuit. More over the input signals of the SABL multiplier are same with the TDPL signals in Fig. 5.6, where the Discharge and Eval signals are omitted and Charge signals of the TDPL is same as clock signal of the SABL.

Initially, the author has investigated the A-cell and B-cell as shown in Fig. 5.7 using the proposed CSSAL ver.1, SAL, SyAL and the TDPL. The supply current transitions corresponding to all possible input transitions has been checked as graphed in the same figure. Input A-cell consists of 3-bit with 64-possible input transitions, whilst the B-cell has 2-bit of input, and hence only 16-possible transitions happen which same as dual-input fundamental logic presented in the previous chapter. The SPICE simulation results in Figs. 5.7(c), (d) shown that the CSSAL performs uniformly peak current traces than that of the adiabatic circuit of SAL and SyAL. TDPL has very constant results for both A-cell and B-cell, however, it has very high signal compare to the CSSAL.

The author wants to state here before going further that, the TDPL was designed as an improvement of the SALB, and hence it has better performance in terms of security merit; therefore, in this chapter, the SABL has less discussion compare to the TDPL in this comparison study. And therefore, the readers will find that some graphical information including tables, the comparison data are randomly exist. Above all, the main purpose is to validate the propose logic resistance, and hence the CSSAL discussion is more likely emphasized.

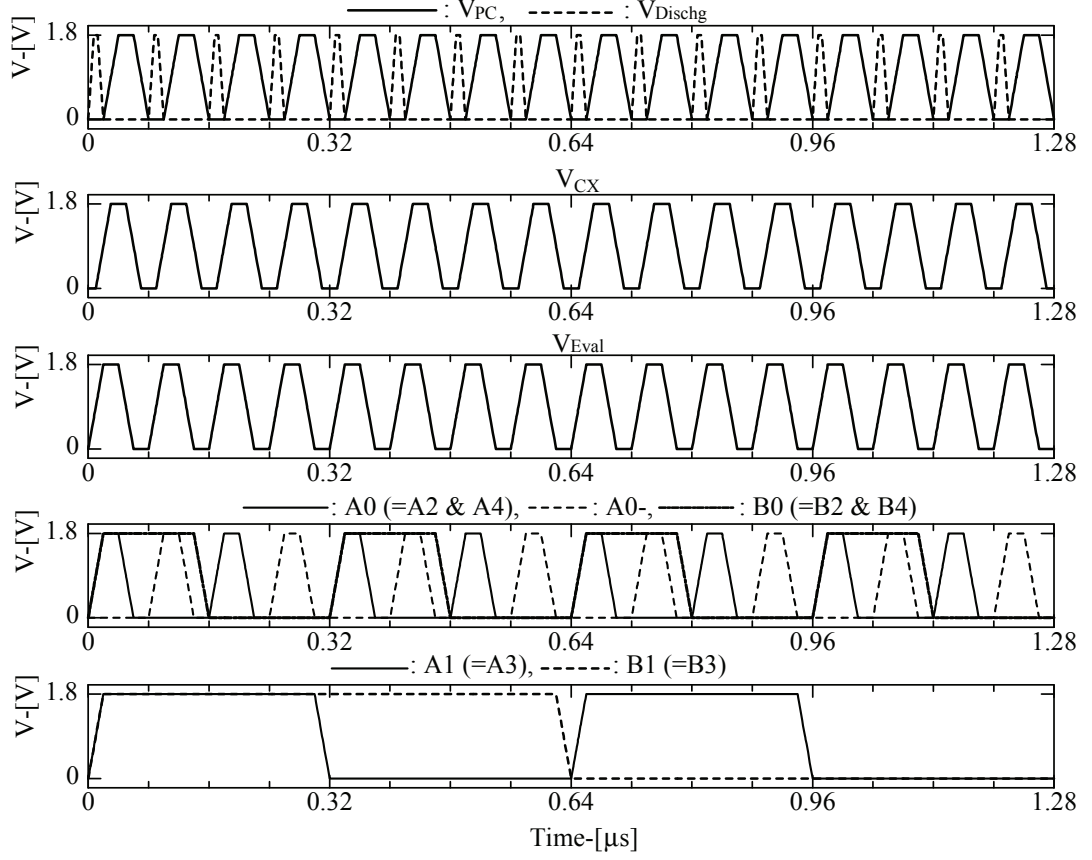
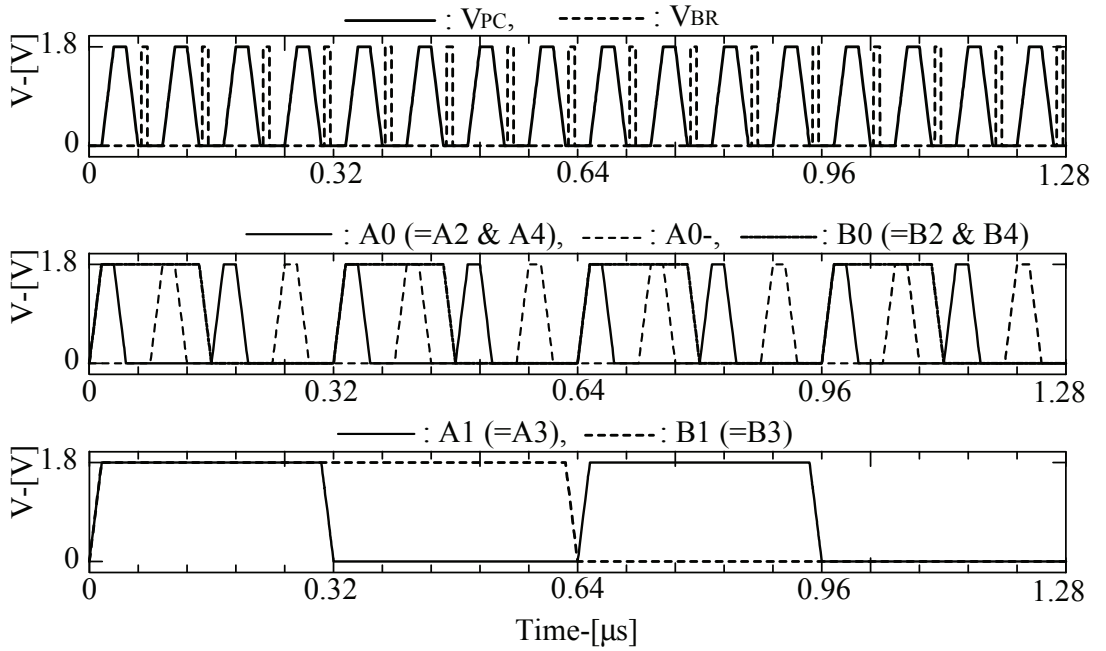
Inner Cell Wire Connection of the Multiplier Circuits

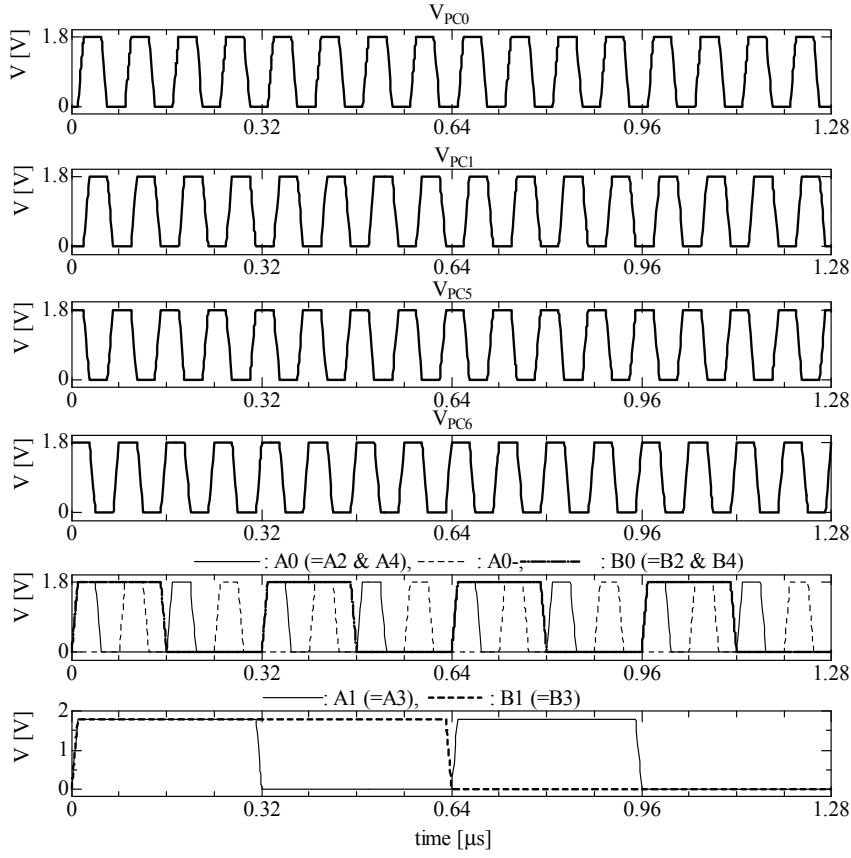
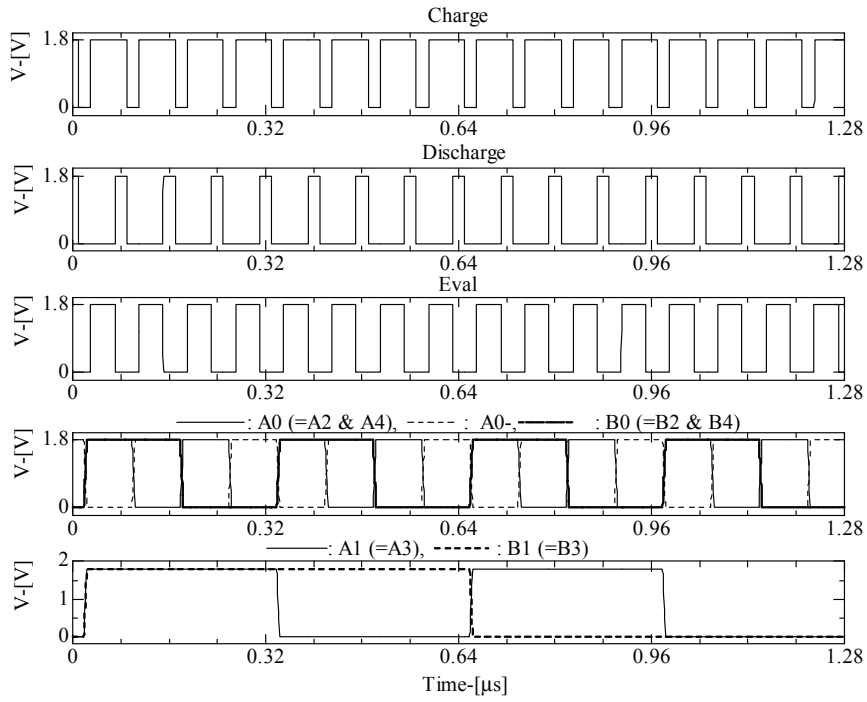
The author has analyzed the meaning of definition 1–4 for periodical to-right and to-left shifting operations, thus the connection based on the definition 1 mentioned that the $A^{(j)}$ will donates the element A as the periodic shift- j -bit-by-right operation. Similarly, $A^{(-j)}$ is identical to the period shifting j bits to the left. For instance, as indicated in Fig. 5.2(a), the input $A^{(i-1, j+2)}$ means that it comes from the previous i row at the position of j column-shifted-2times-to-the-right. Similarly, same as the one opposite direction, such as $B^{(i-1, j-1)}$. Hereafter, the AND logic unit performs $A^{(i-1, j+2)}$ and $B^{(i-1, j-1)}$, and XOR gate adds the previous result of inner cell computation and the output of the AND gate in the same cell. The output of XOR gate propagates the cell $(i+1, j)$, whilst the output of coefficient A and $B^{(i-1, j-1)}$ at the (i, j) cell are to be propagated into $(i+1, j-2)$ and $(i+1, j+1)$ cell, when the current cell is located at cell (i, j) . Obeying this instructed connections, and then the multiplier circuits in Fig. 5.1 can be connected as shown in Fig. 5.8 and Fig. 5.9 for A-cell and B, C-cell circuit, respectively. Operation function of the output signals o these two multiplier are same, the only different is that the B, C-cell has less computation time per cell, such as T_{XOR} than the latency of A-cell multiplier that

includes $T_{AND}+T_{XOR}$.

Output Signal of the Multiplier Circuits

The SPICE simulation all multiplier logic circuits are depicted in Figs. 5.10–5.18, where the circuit used can be confirmed in each figure caption. The author presents here the CSSAL output signals of the multiplier circuits using A-cell and B, C-cells for individual logic AND and XOR with and without C_x transistors. Figure 5.10 is the output function of the CSSAL multiplier using A-Cell with C_x transistors. Similarly, the Fig. 5.12 shows the result using B, C-Cell with C_x transistors. Figures 5.11, 5.13 are the output results without C_x transistors for both A-cell and B, C-cells multiplier circuits, respectively. These figures depicted that the CSSAL multiplier using C_x transistors produces dynamic hazards that surpass the V_{thn} level. Consequently, the circuit performances, such as energy dissipation and balancing peak current transitions are drastically decreased. Furthermore, if we observe the Fig. 5.11 and Fig. 5.13, the multiplier using A-cell has smooth and constant low level signal compare to the B, C-cell multiplier. Accurate investigation result of both multiplier circuits are tabulated in Table 5.1. The NSD result in this table have shown that CSSAL multiplier using A-cell is smaller than the one that using B, C-cells. Therefore, further investigations and LSI implementation will utilize multiplier with inner A-cell circuit and fundamental logic circuit CSSAL ver.2.

Figure 5.3: 4-bit input signals of the CSSAL multiplier over $GF(2^4)$.Figure 5.4: 4-bit input signals of the SyAL multiplier over $GF(2^4)$.

Figure 5.5: 4-bit input signals of the SAL multiplier over $GF(2^4)$.Figure 5.6: 4-bit input signals of the TDPL multiplier over $GF(2^4)$.

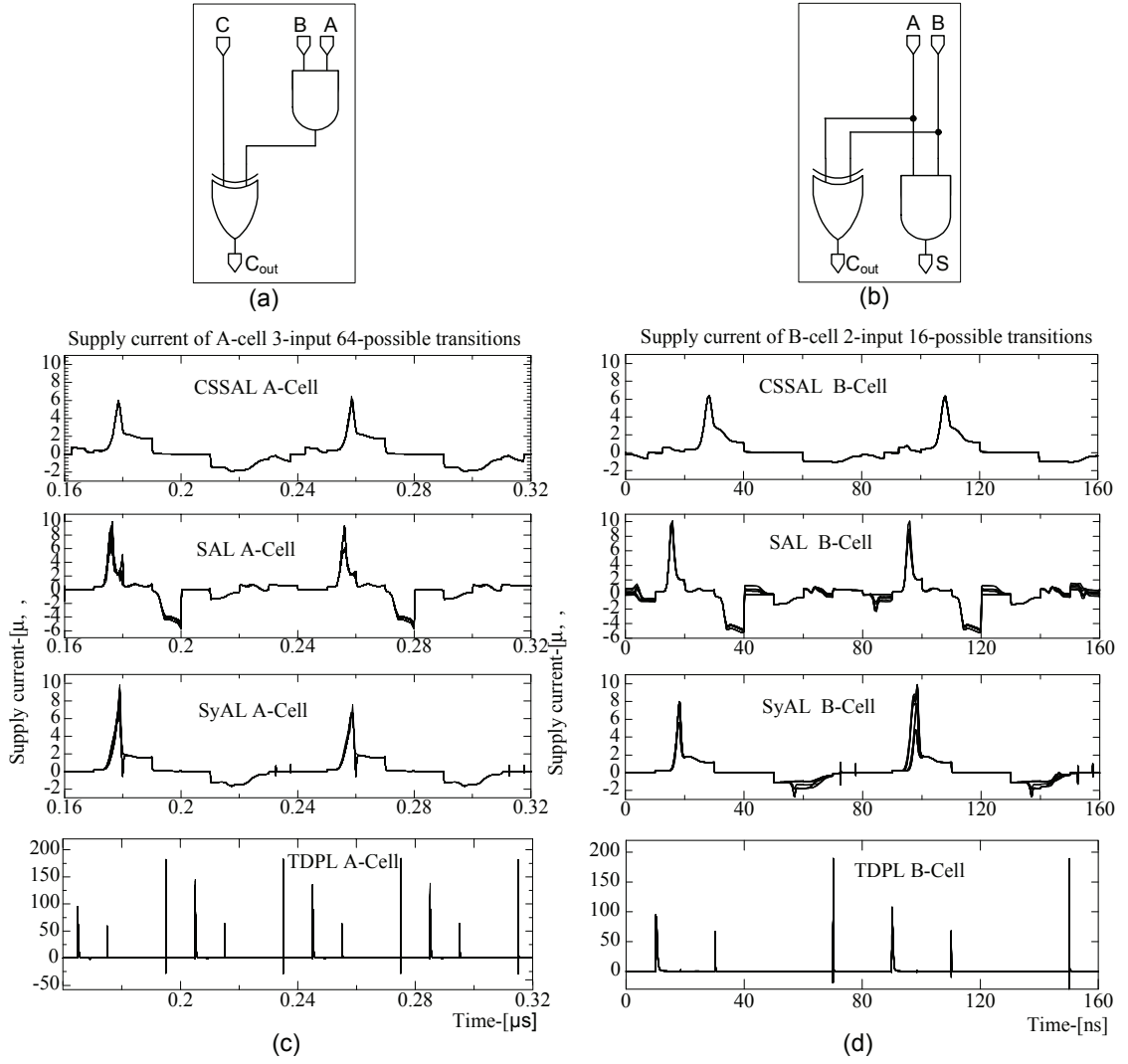


Figure 5.7: Investigation of A-cell and B-cell supply current transitions. CSSAL individual logic without C_x pass-transistors.

Supplemental remarks of Fig. 5.7 : Observing the Fig. 5.7(b) logic diagram shows that the parallel computation of AND and XOR logic that propagated by the same input signals and power clock supply, and hence the peak current transition is very similar to the one of the individual logics in Fig. 4.27. On the other side, Fig. 5.7(a) shows that the XOR cell has different input arrival time, which cause small visible different signal transitions of the CSSAL A-Cell in Fig. 5.7(c).

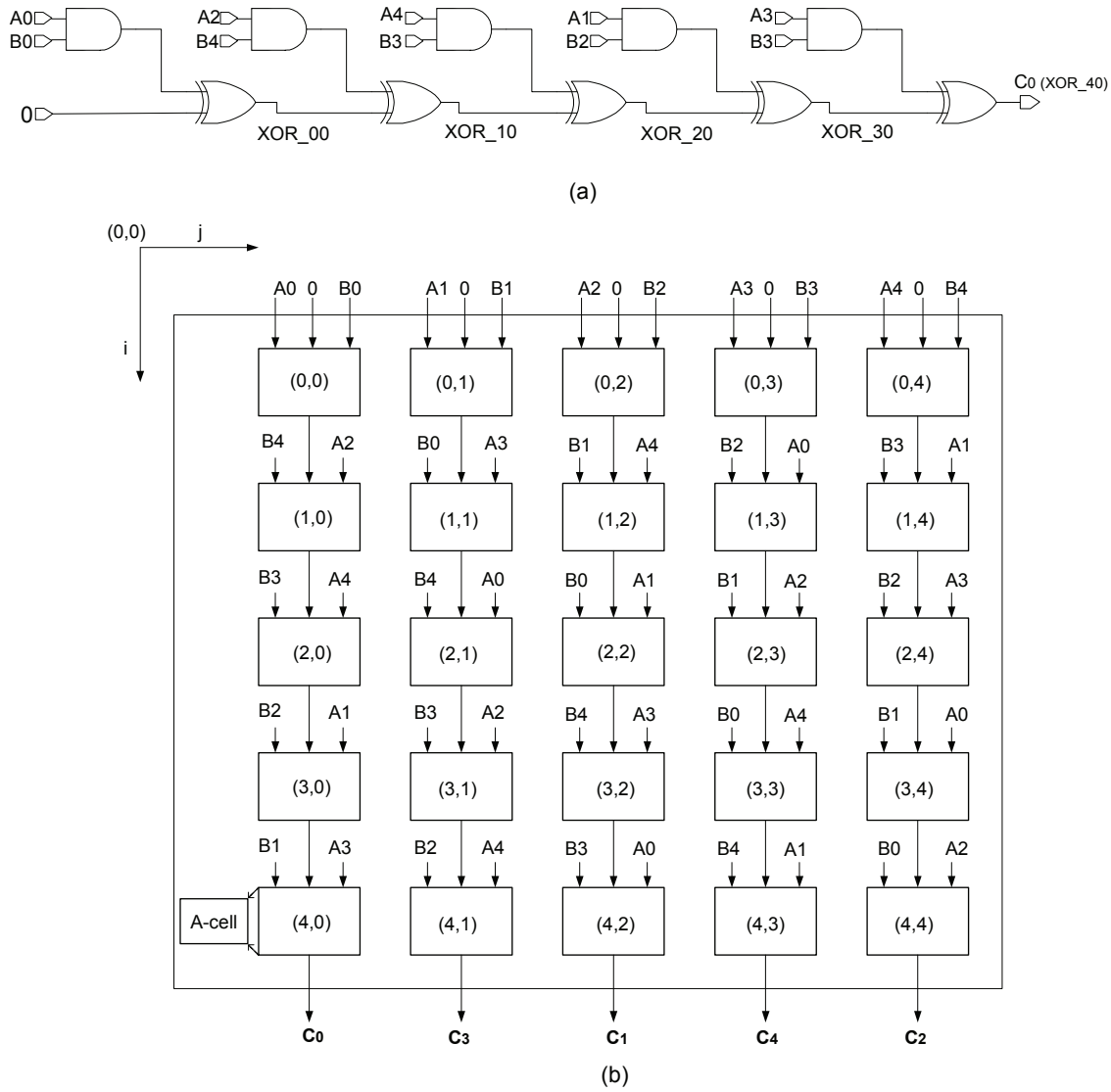


Figure 5.8: Input signals connection of inner A-cell in bit-parallel multiplier over $GF(2^4)$: (a) Logic diagram of the first column, and (b) Block diagram.

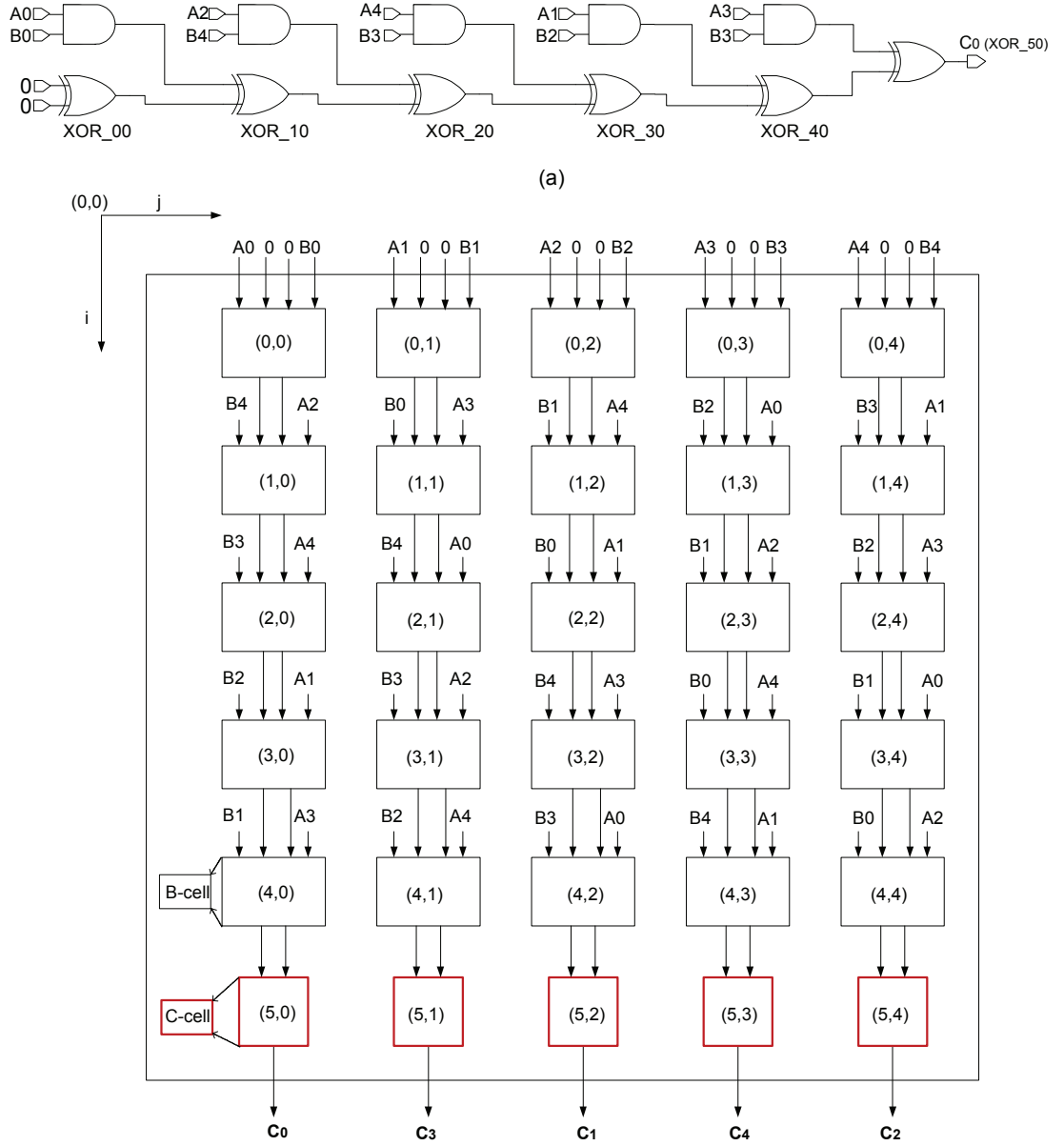


Figure 5.9: Input signals connection of inner B, C-cells in bit-parallel multiplier over $GF(2^4)$: (a) Logic diagram of the first column, and (b) Block diagram.

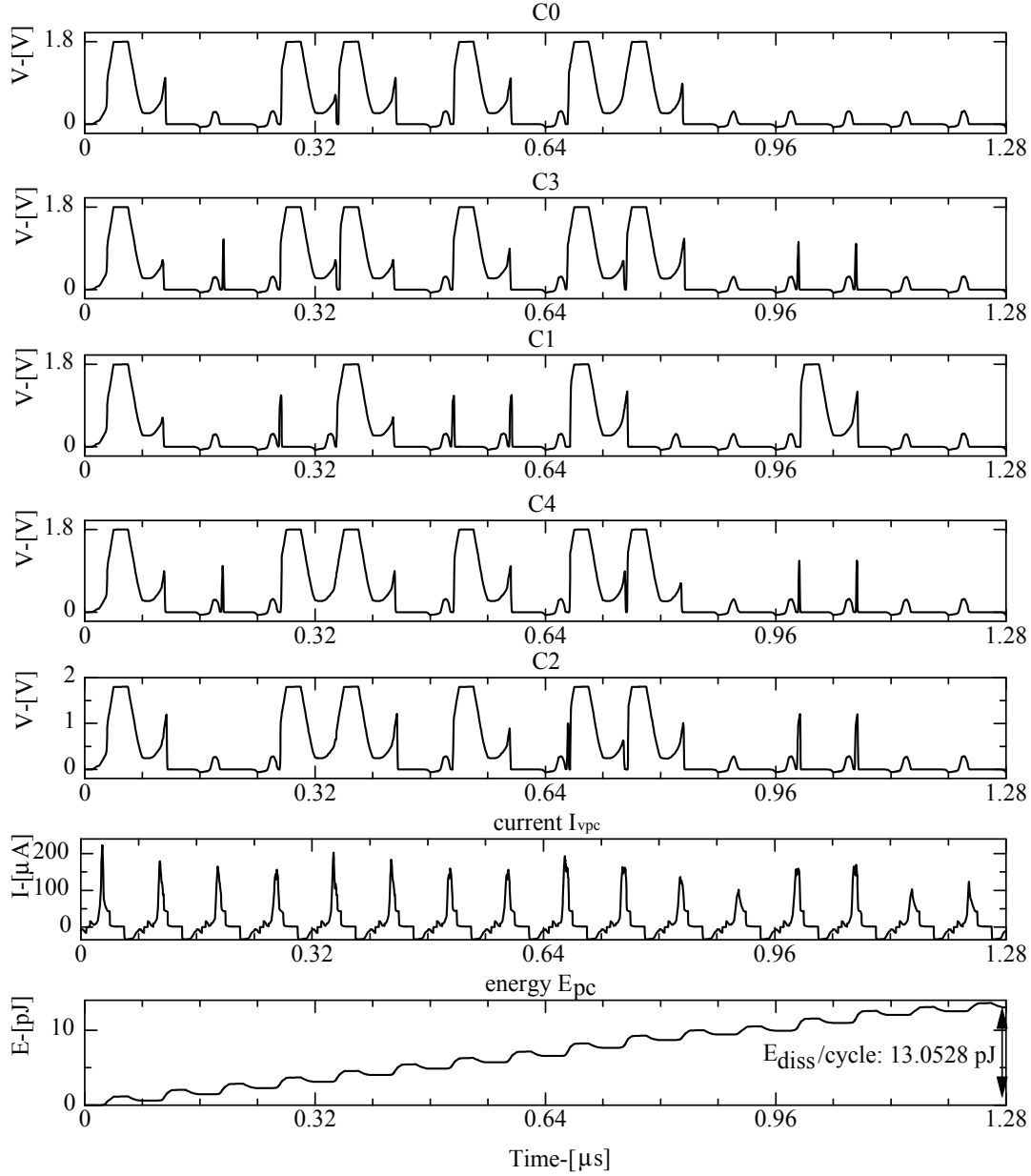


Figure 5.10: Output signals of CSSAL bit-parallel multiplier over $GF(2^4)$ using A-Cell with C_x pass-transistors.

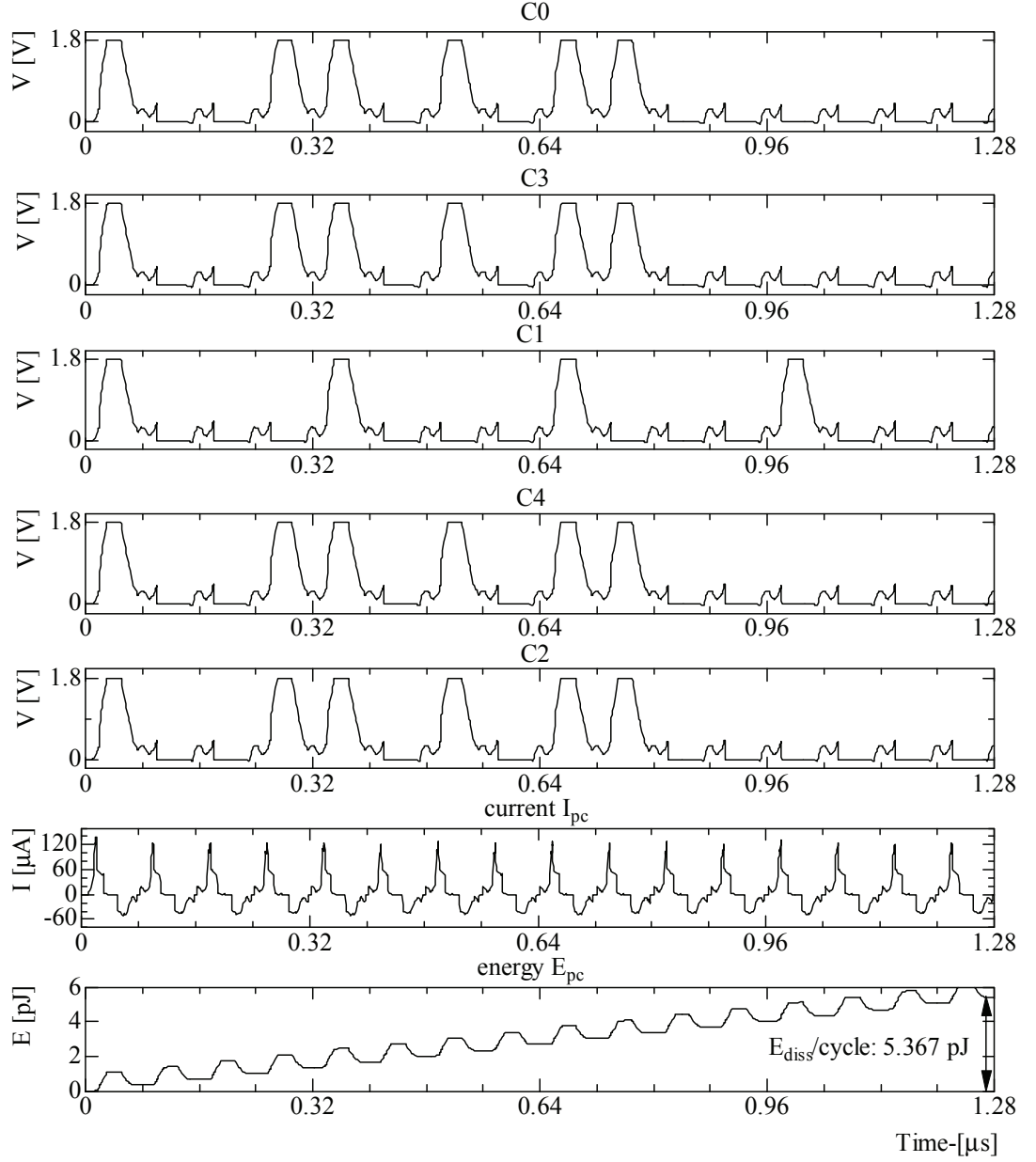


Figure 5.11: Output signals of CSSAL bit-parallel multiplier over $GF(2^4)$ using A-Cell without C_x pass-transistors.

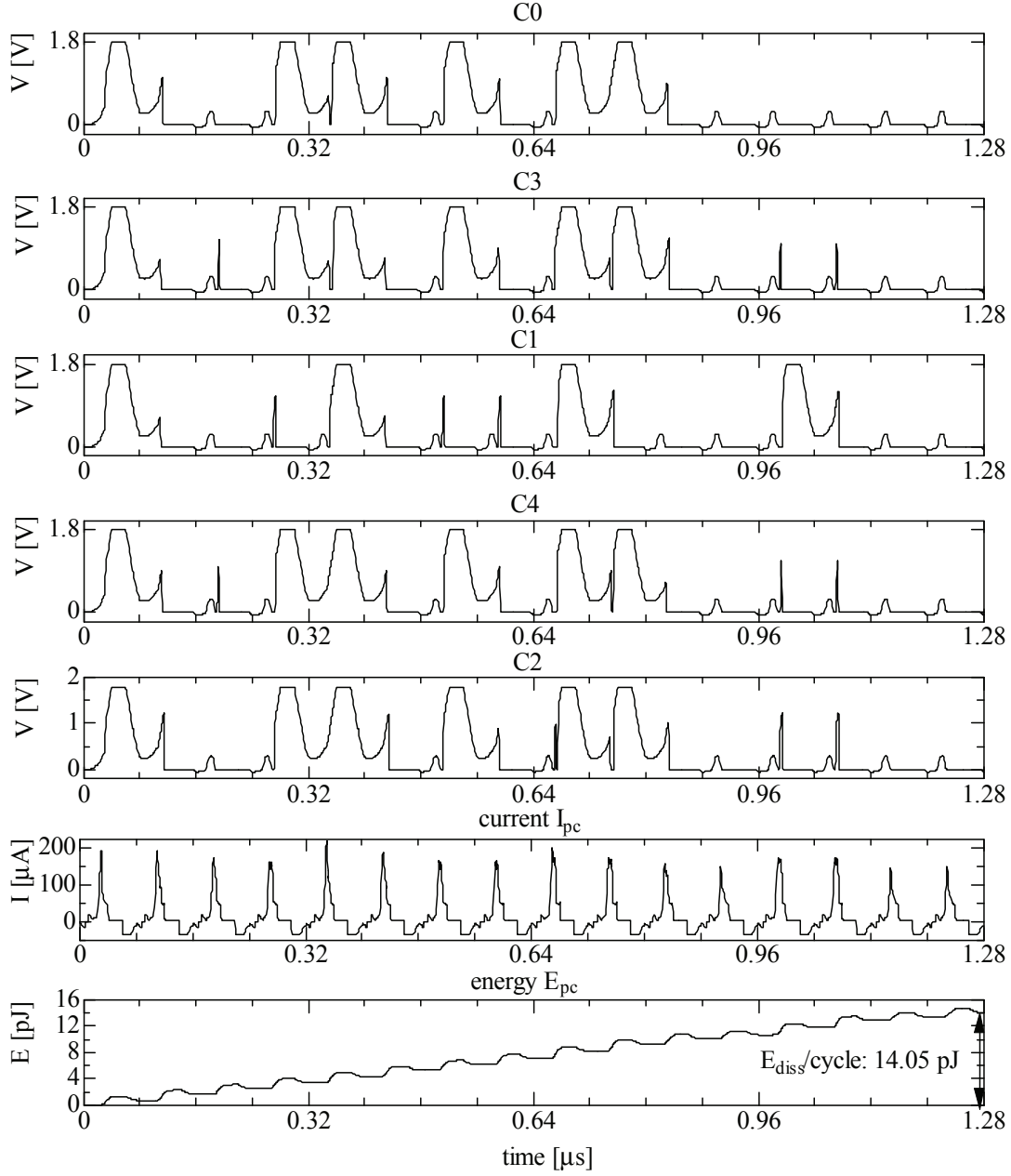


Figure 5.12: Output signals of CSSAL bit-parallel multiplier over $GF(2^4)$ using B, C-Cell with C_x pass-transistors.

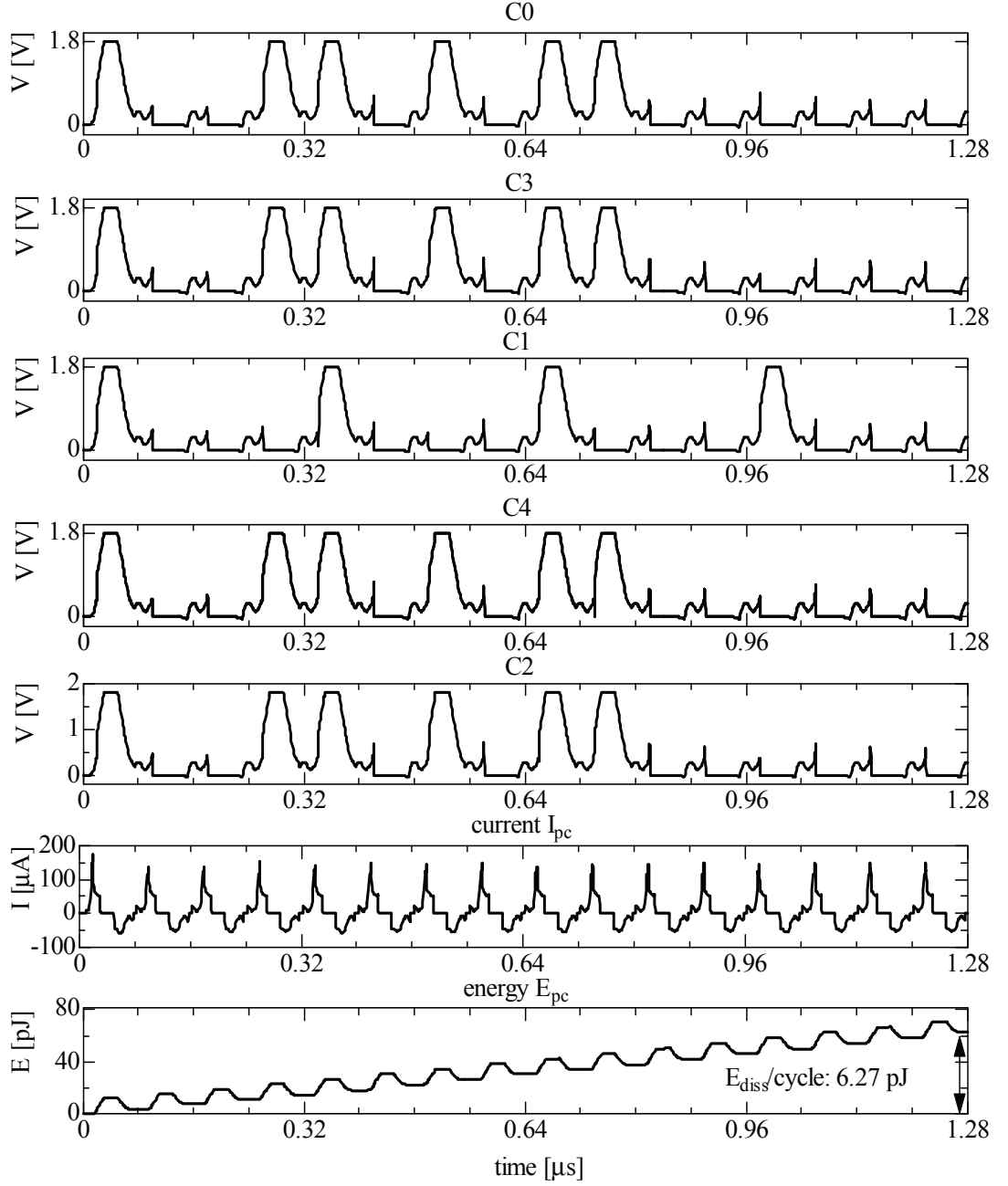


Figure 5.13: Output signals of CSSAL bit-parallel multiplier over $GF(2^4)$ using B, C-Cell without C_x pass-transistors.

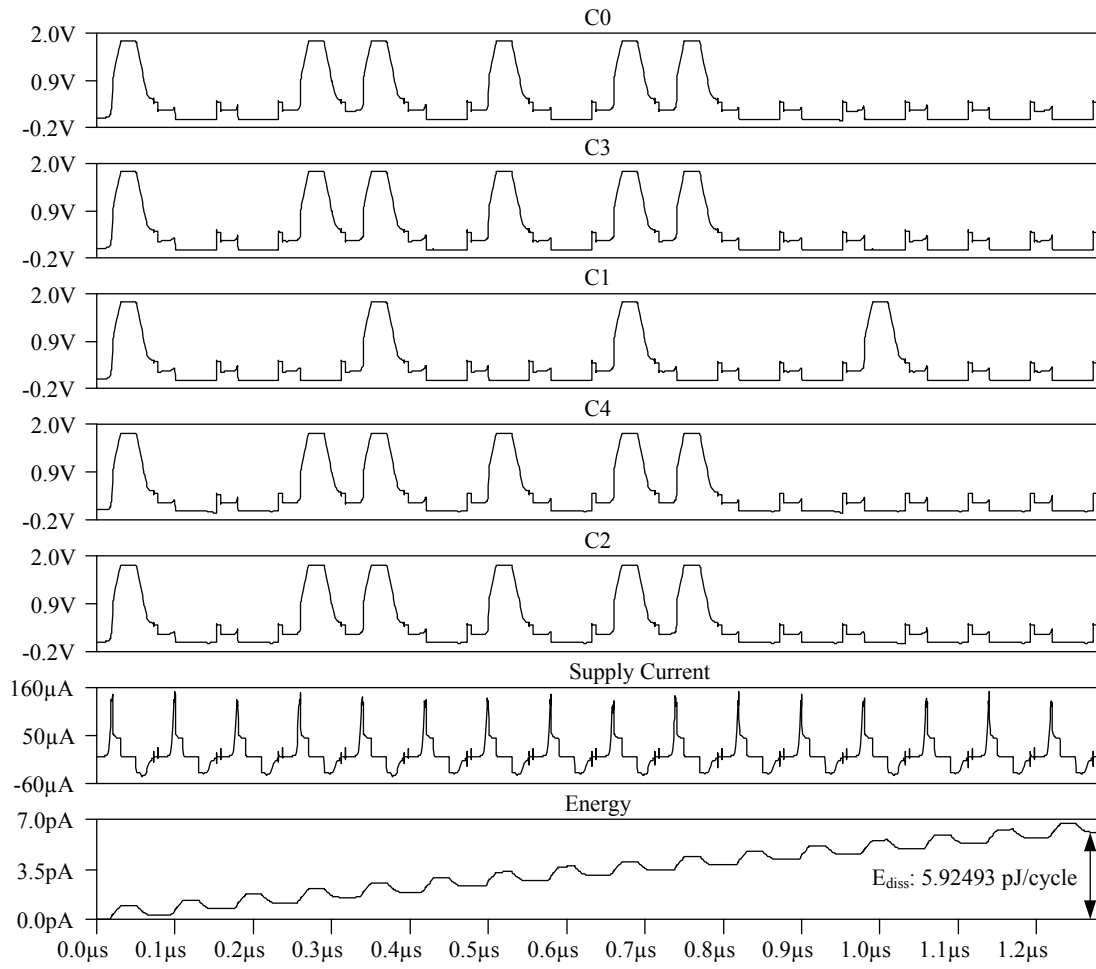
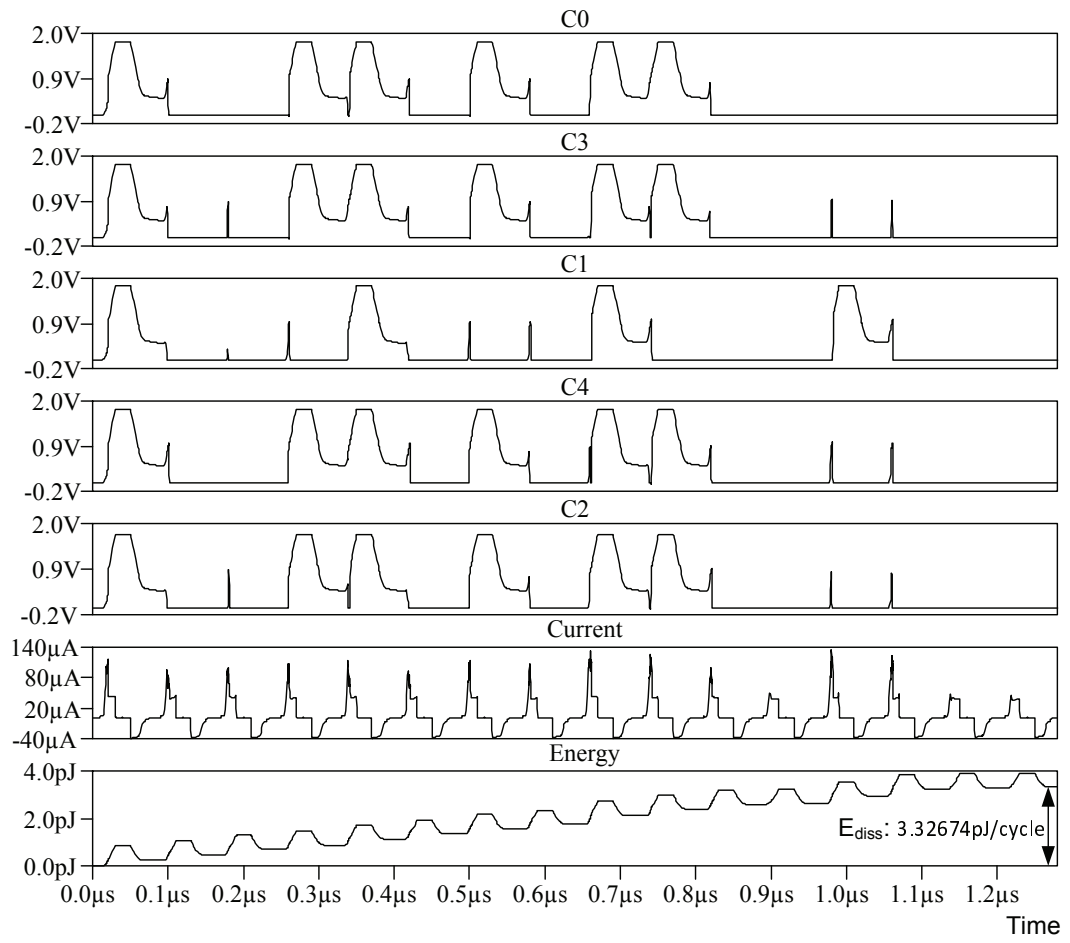


Figure 5.14: Output signals of SyAL bit-parallel multiplier over $GF(2^4)$.

Figure 5.15: Output signals of 2N-2N2P bit-parallel multiplier over $GF(2^4)$.

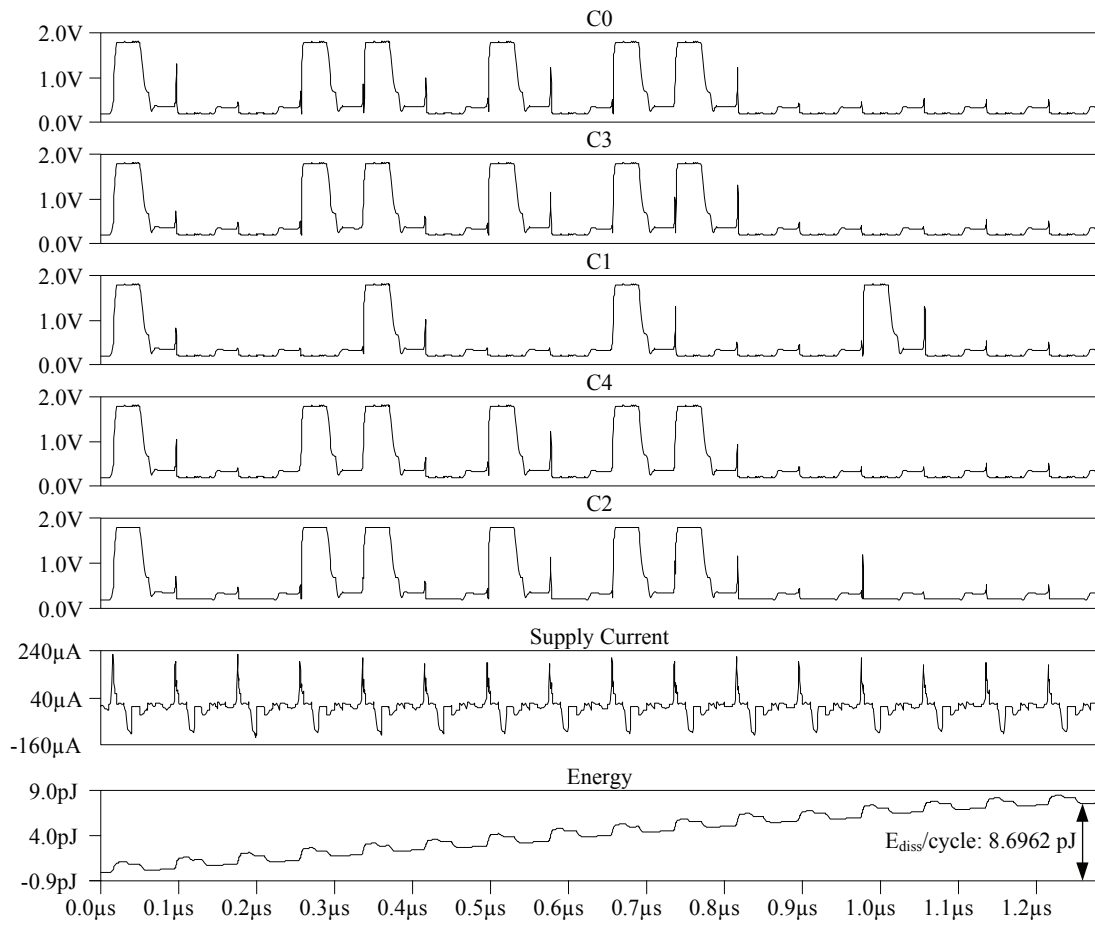


Figure 5.16: Output signals of SAL bit-parallel multiplier over $GF(2^4)$.

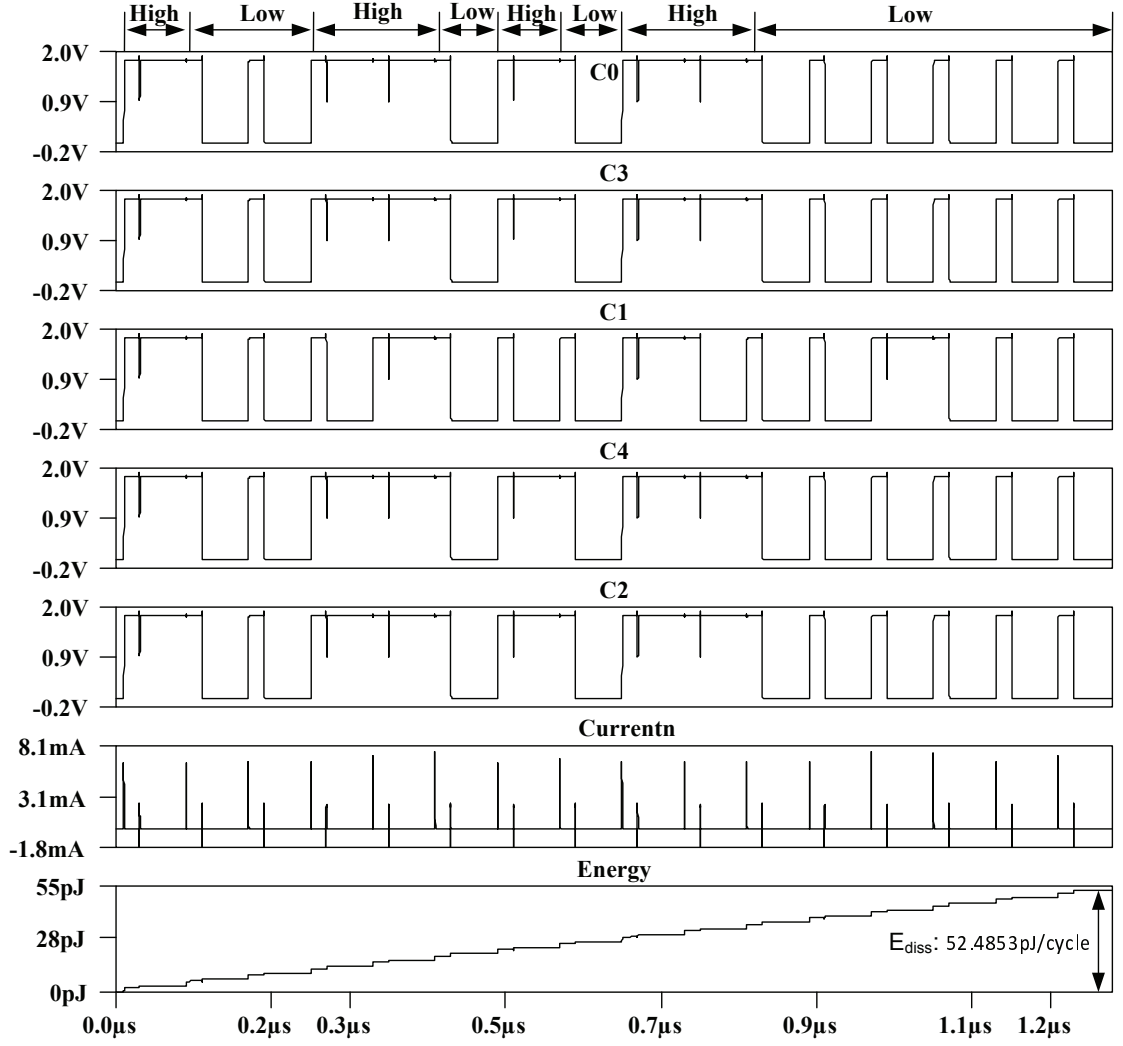


Figure 5.17: Output signals of SABL bit-parallel multiplier over $GF(2^4)$.

Supplemental remarks of Fig. 5.17 : On the top of this figure describes the High and the Low labels. This aims to indicate the logical 1 level and the logical 0 level of each output signals. In the logical 0 level, we can see some pulses high to Vdd level; this signals are the behavior of the pre-charge logic style, to initially charge both output wires to high level before the logics are evaluated.

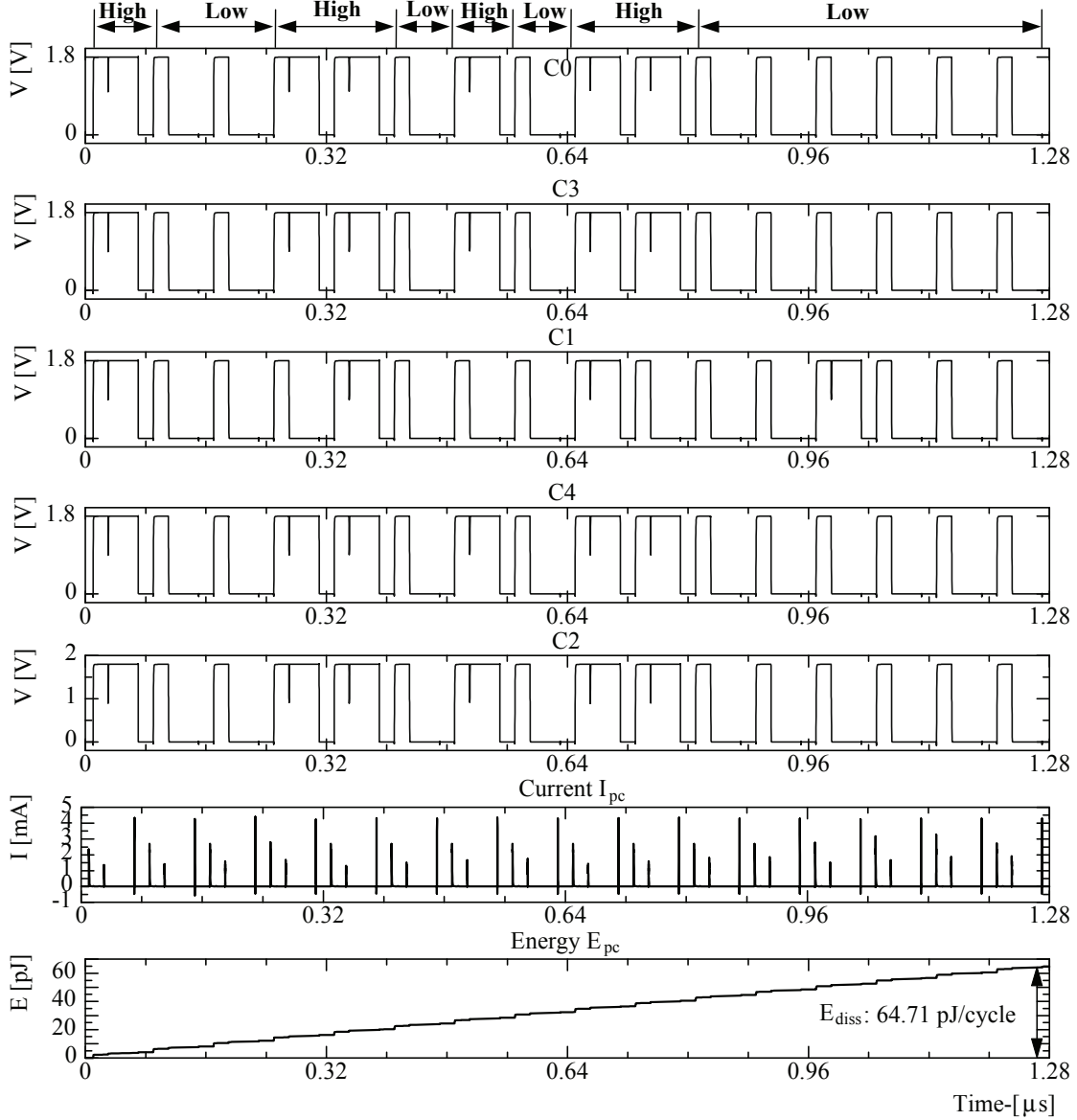


Figure 5.18: Output signals of TDPL bit-parallel multiplier over $GF(2^4)$.

Supplemental remarks of Fig. 5.18 : The TDPL output signals of the multiplier are similar to the one of the SABL in the previous page. The only different is that the TDPL has both pre-charge and discharge operations, in which we can see that, in the logical high level condition, both complementary signals are simultaneously at logical high and low levels. Discharging both output wires to logical low level is the unique property of the TDPL.

Table 5.1: Simulation and calculation results for A-cell and B, C-cells into bit-parallel cellular multiplier over $GF(2^4)$.

$GF(2^4)$ using A-cell @ 12.5 MHz				
Circuit	SAL	SyAL	CSSAL	TDPL
E_{min} [fJ]	337.15	311.64	310.28	3957.84
E_{max} [fJ]	628.92	418.25	350.74	4112.35
\overline{E} [fJ]	506.78	382.02	333.86	4042.54
σ_E [fJ]	95.42	31.02	11.15	40.77
NED [%]	50.65	25.49	11.54	3.76
NSD [%]	18.83	8.12	3.32	1.01
$GF(2^4)$ using B, C-cells @ 12.5 MHz				
E_{min} [fJ]	345.70	318.12	342.01	3892.87
E_{max} [fJ]	842.90	427.27	435.56	4053.52
\overline{E} [fJ]	592.63	396.04	392.11	3943.42
σ_E [fJ]	142.56	34.25	32.21	44.47
NED [%]	58.98	25.55	21.48	3.96
NSD [%]	24.06	8.65	8.47	1.12

5.2.3 Frequency Spectrum Analysis Result: Multiplier Circuits

This section verifies the logic's behavioral in consuming instantaneous power for all possible input transitions using FFT from SPICE simulation. The vital information to be observed in this FFT SPICE simulation is that, ideally, only a single peak of a certain frequency spectrum (f_0) should be plotted if the logic circuit is claimed to consume constant power for every input transitions. For instance, in this investigation, the power clock frequency is set to 1.25 MHz, which denoted as fundamental frequency f_0 . The harmonic frequencies below f_0 are denoted as $f_0/2$, $f_0/3$, $f_0/4$, etc. The relevant quantities of spectral analysis are listed in Table 5.2. As shown in this table, the stop-time of SPICE simulation was set to 640 μ s, which resulting the maximum frequency f_{max} . The FFT frequency range was expanded from 0–1.4 MHz as shown in Figs. 5.19–5.23 for each respective circuit. Observing Figs. 5.19(b)–5.23(b), the CSSAL multiplier has very low signals before f_0 and noiseless compare to the other logic styles. This signal spectrum of FFT results validate the effectiveness of the CSSAL to diminish information leakage in side-channel attacks when implementing the differential frequency analysis [108].

Table 5.2: FFT parameters and calculation results of the CSSAL, 2N-2N2P, SyAL, and TDPL multipliers circuits @ 1.25 MHz input clock frequency.

Parameter	CSSAL	2N-2N2P	SyAL	SAL	TDPL
M data (time)	89981	84554	118527	73927	145772
N data (freq.)	131072	131072	131072	131072	131072
t_1 [μ s]	0	0	0	0	0
t_2 [μ s]	640	640	640	640	640
Δt [ns] (by Eq. (2.28))	3.56	3.78	2.70	4.33	2.20
f_s [MHz] (by Eq. (2.32))	409.60	409.60	409.60	409.60	409.60
f_{max} [MHz]	204.798	204.798	204.798	204.798	204.798
Δf [KHz]	1.5625	1.5625	1.5625	1.5625	1.5625

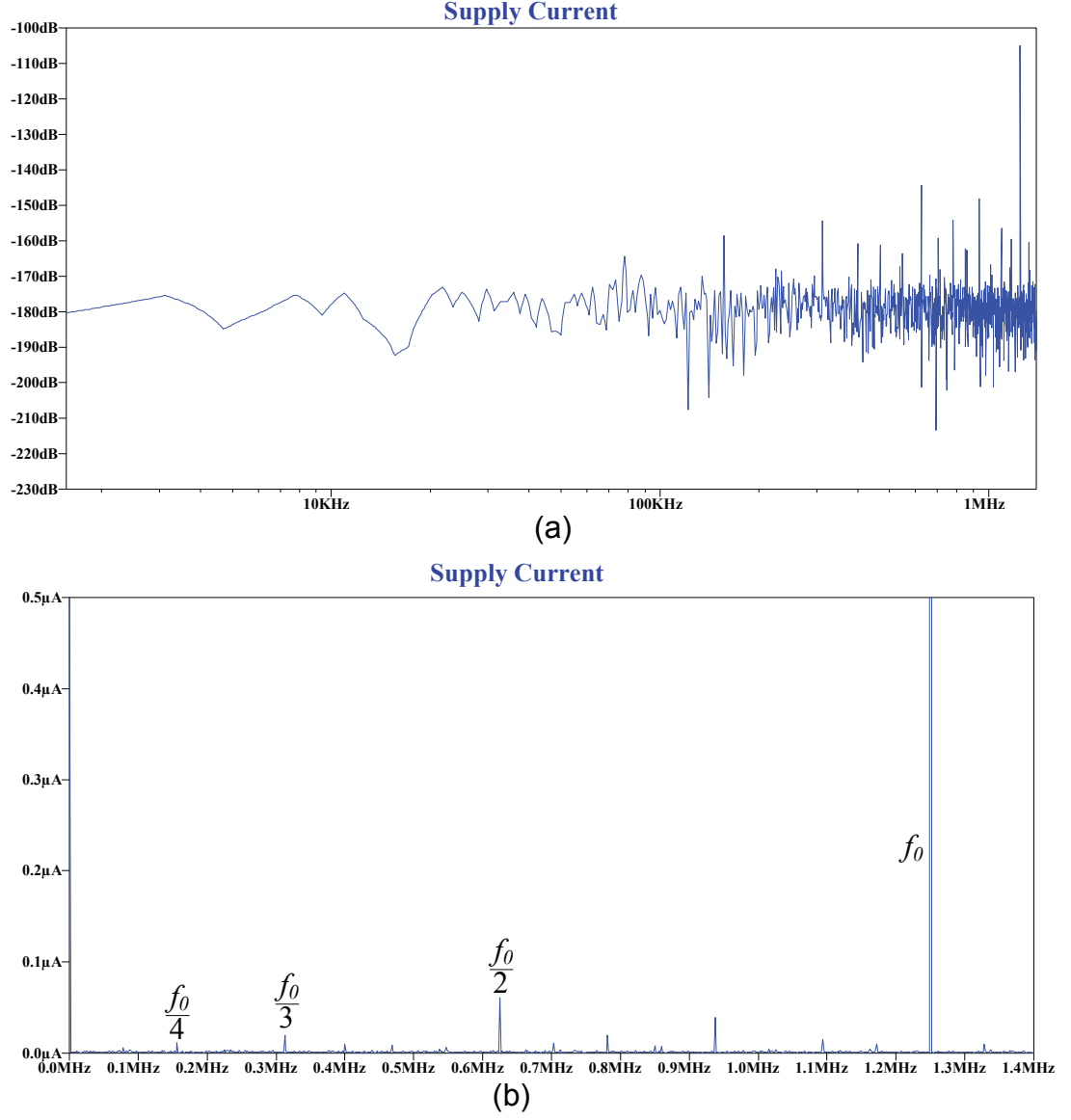


Figure 5.19: Signal spectrum of the CSSAL multiplier @ 1.25 MHz: (a) Vertical y -axis decibel and horizontal x -axis logarithmic, (b) Both horizontal and vertical axis are set to linear.

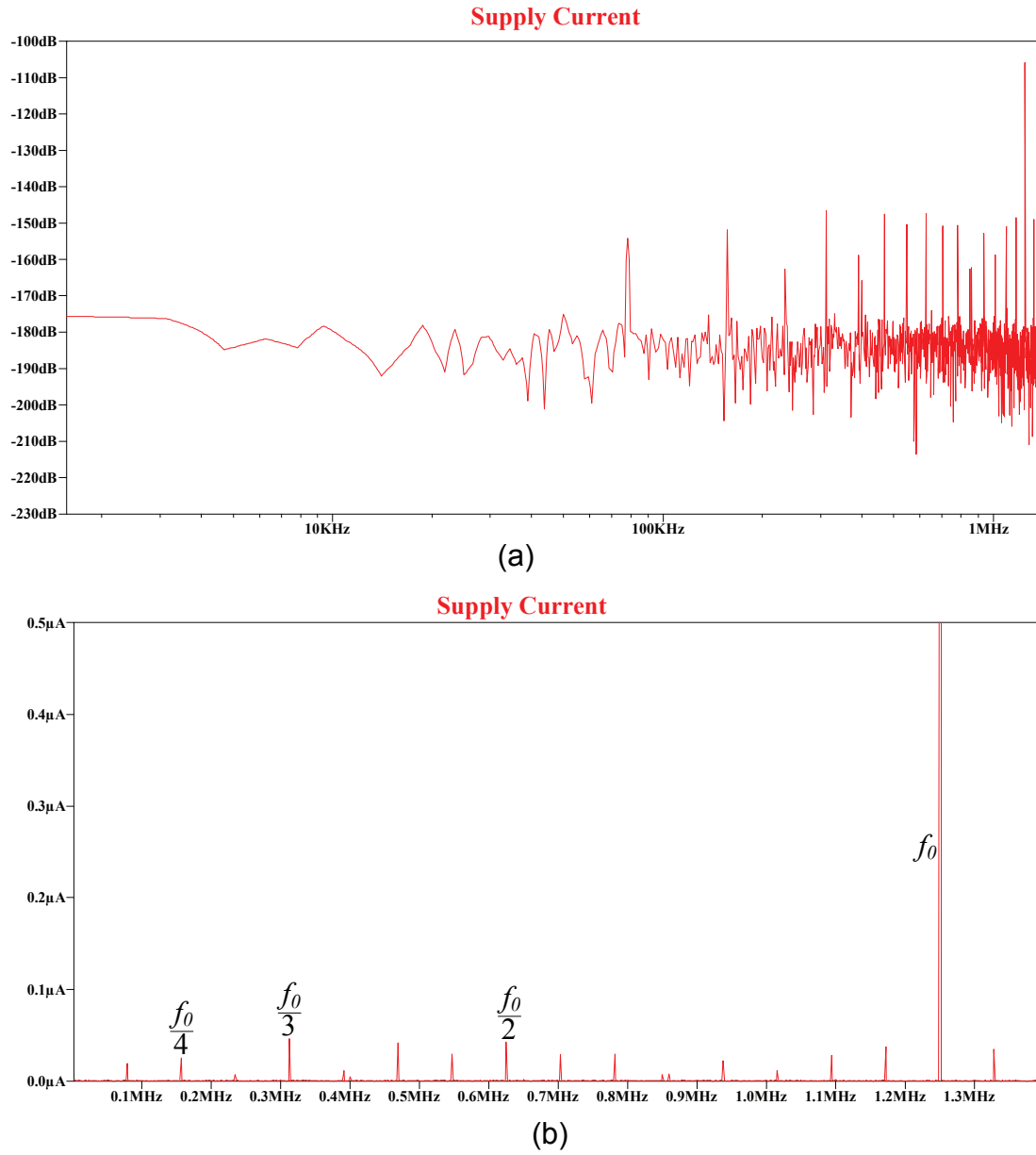


Figure 5.20: Signal spectrum of the SyAL multiplier @ 1.25 MHz: (a) Vertical y -axis decibel and horizontal x -axis logarithmic, (b) Both horizontal and vertical axis are set to linear.

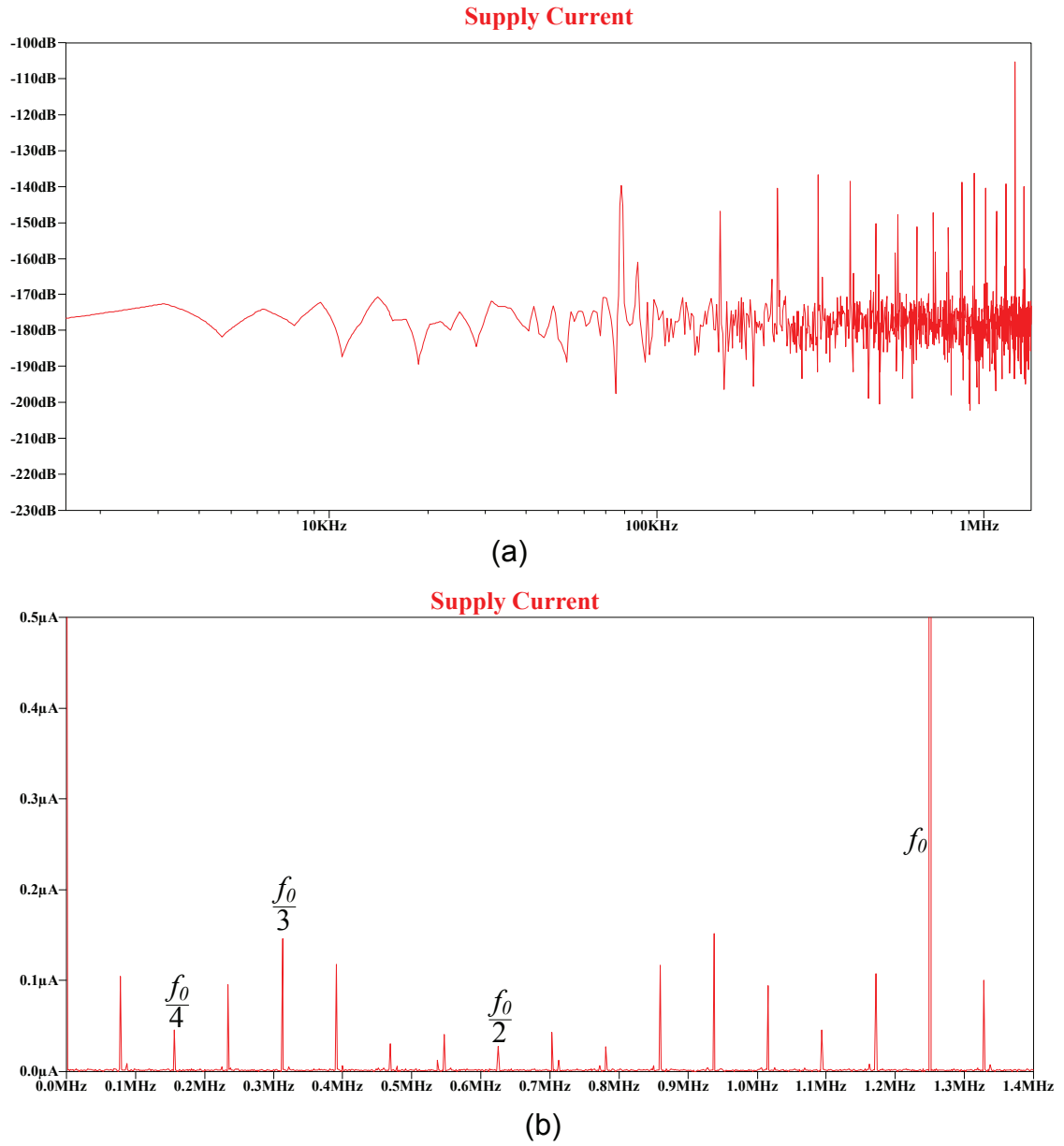


Figure 5.21: Signal spectrum of the SAL multiplier @ 1.25 MHz: (a) Vertical y -axis decibel and horizontal x -axis logarithmic, (b) Both horizontal and vertical axis are set to linear.

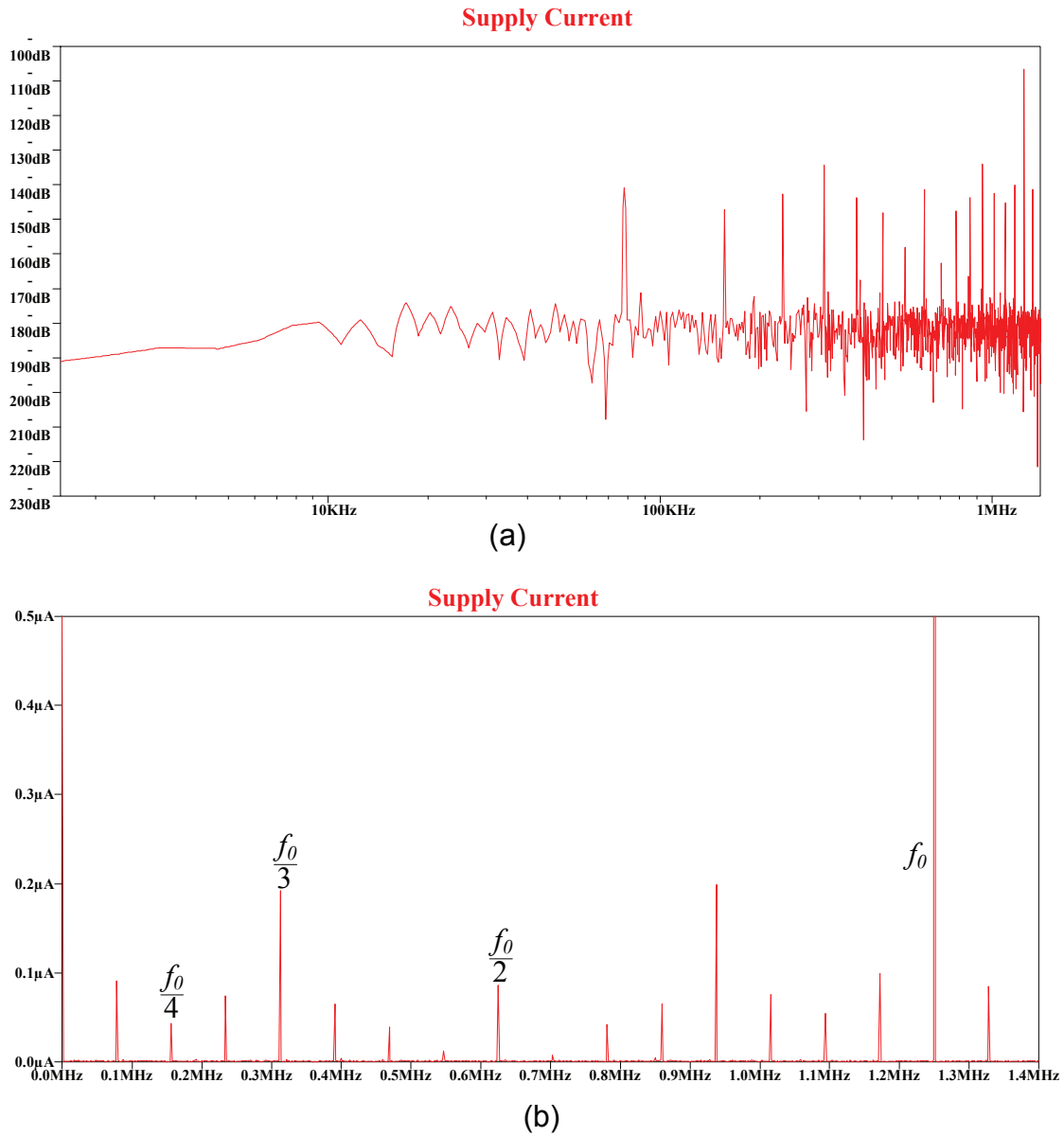


Figure 5.22: Signal spectrum of the 2N-2N2P multiplier @ 1.25 MHz: (a) Vertical y -axis decibel and horizontal x -axis logarithmic, (b) Both horizontal and vertical axis are set to linear.

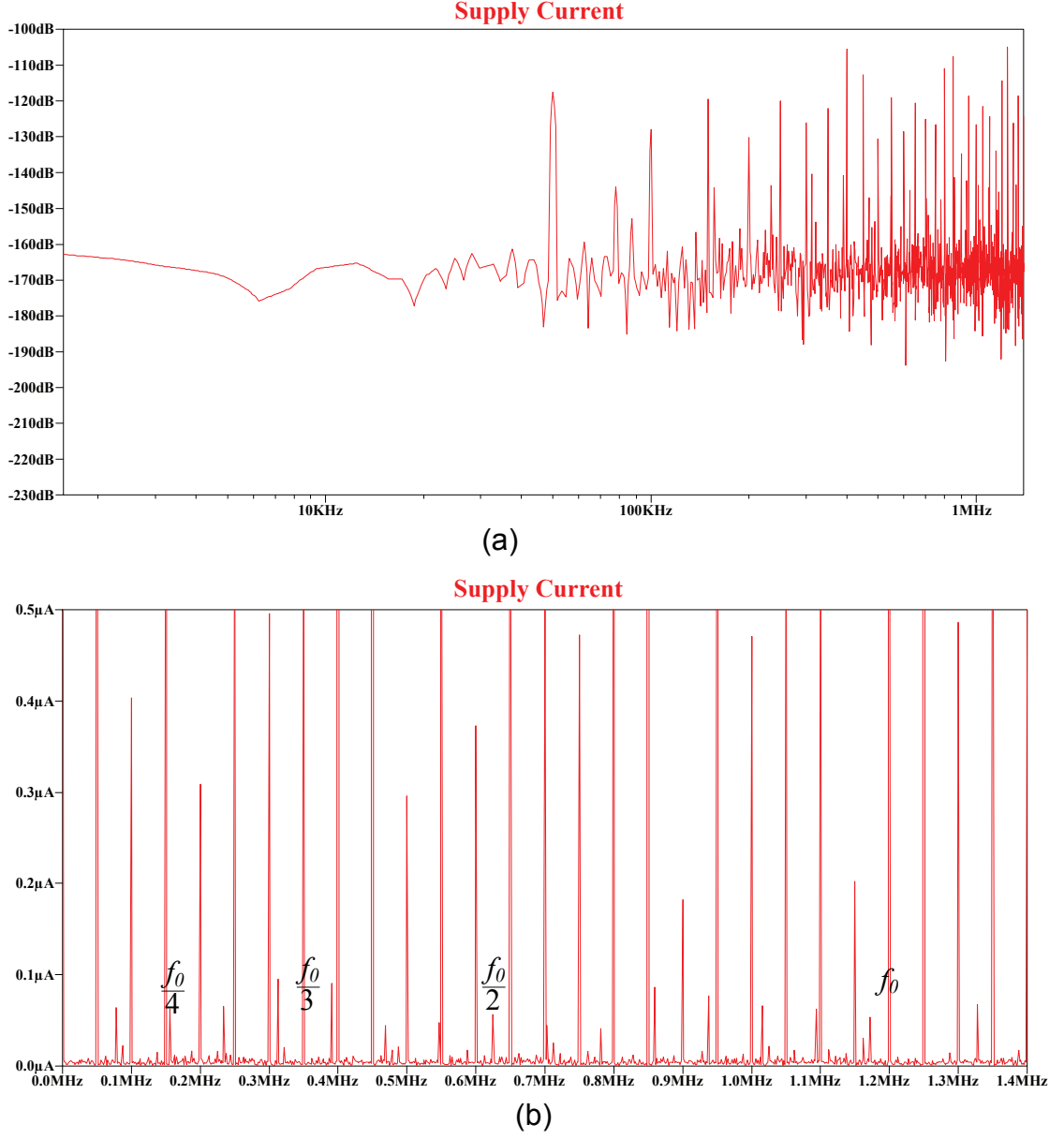


Figure 5.23: Signal spectrum of the TDPL multiplier @ 1.25 MHz: (a) Vertical y -axis decibel and horizontal x -axis logarithmic, (b) Both horizontal and vertical axis are set to linear.

5.2.4 Analysis and Comparison: Multiplier Circuits

Security View Point

Verification of Supply Current Transition : Figure of merit to measure resistance of the proposed CSSAL multiplier against power analysis attack will be discussed and compared here. Similar to the individual logic investigation, the multiplier circuits also tested using SPICE simulator, plotting each respective supply current transitions and each power variation calculation. The peak current traces are shown in Fig. 5.24, which includes 8-transitions represent 256-possible transition made up by 4-bit input patterns. Although this transitions is just about 3% of the total transitions, the author assures that the peak current will not show any significant change compare to the SAL, SyAL and the 2N-2N2P multiplier logic styles. The SABL or TDPL peak current are not exist in this figure, because they have very high peak current, such as TDPL has 5.6 mA and SABL has 8.1 mA, and if we compare to the CSSAL, it decreases the peak current value about 22 times lower. Please note that, these results in Fig. 5.24 are plotted from SPICE simulation with 10-fF nominal capacitance at each output node of individual logic (AND/NAND and XOR/XNOR output wires).

Verification of Energy Fluctuation : Further investigation results are summarized in Table 5.3. The NSD values in this table illustrate that the circuit operation at 12.5 MHz power clock frequency (the speed of application approach, such as 13.56 MHz carrier frequency of contactless smart card), the CSSAL has smallest value within adiabatic multiplier circuits. The conventional TDPL remains strongest in this comparison table. The SAL multiplier circuit does not work in higher frequency, thus the data are not available (NA) at 125 MHz. From this data, we may judge the parameter list under the frequency column of Table 5.3 show that their results are well-compared within the same logic technique. It means that not very precise data comparison can be done between the conventional CMOS logic style and adiabatic logic technique. For example, the TDPL energy variance of σ_E is higher than the CSSAL at 12.5 MHz, but the means of energy (\bar{E}) of the TDPL is much higher, and hence the NSD value (σ_E/\bar{E}) surely becomes small.

For reasonable comparison parameter, the author of this dissertation has defined the figure of merit in Equation (2.27), which performs how the proposed logic consumes constant and low power than the other conventional logic styles. And therefore, the FoM results in Table 5.3 show that proposed logic has smallest values at low frequency band.

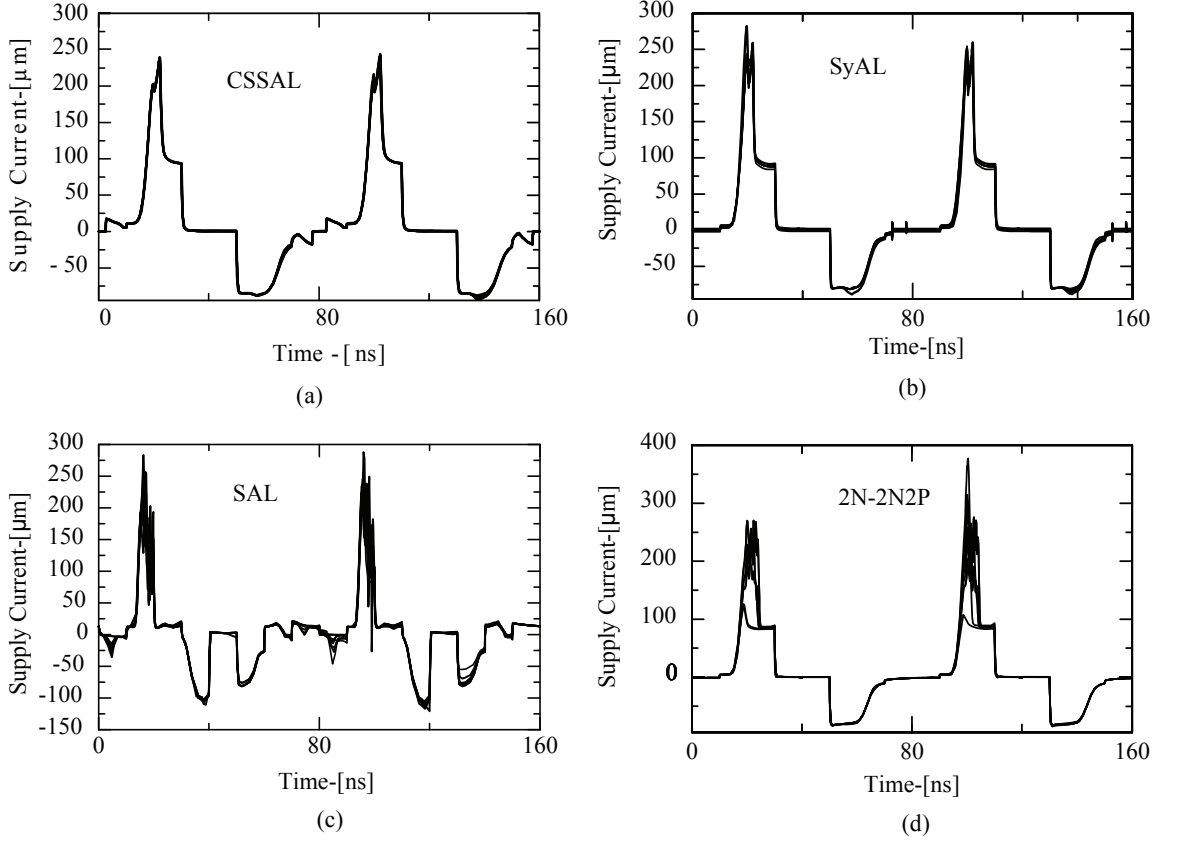


Figure 5.24: Instantaneous supply current transition of 8 input patterns representation of each adiabatic multiplier circuits; (a) CSSAL Multiplier; (b) SyAL Multiplier; (c) SAL Multiplier; (d) 2N-2N2P Multiplier.

Verification of CMOS Process Variation : Integrated circuit design in the current CMOS technology model (nano technology), designers face major challenges of systematic and random variation of process, supply voltage and the temperature fluctuation (PVT [109]). The process variation is caused by the inability to precisely the fabrication process at down scaling technology. Voltage variation can be caused by IR drops in the supply distribution network or LdI/dt noise under changing load [110]. The temperature variation is caused by the spatially and temporally varying factors that can negatively contribute to the performance degradation. All these variations affect the maximum clock frequency distribution of the LSI design [111]. Therefore, regarding to process variation, such as channel length (L), doping concentration, gate-oxide thickness (T_{ox}) and the variation of transistor threshold voltage (V_T) which affect the stability of circuit's power consumption in its role for counteracting power analysis attack; and hence the effectiveness of the pro-

posed logic with other adiabatic logic style in the multiplier circuit were investigated further in this part. In the SPICE simulation, it was assumed that the fluctuations in threshold voltage, doping concentration, and gate oxide thickness followed normal distributions, and a sigma variation of 10% were considered for each logic circuit. The Monte Carlo simulation was conducted with simulation iterations set to 100-repetition. This means, we assume to test 100 LSI multipliers using the same logic, where the parameters to measure the energy variations of each LSI were considered. The calculation results are shown in Figs. 5.25– 5.28. Observing this figures, the proposed CSSAL has smallest variations of NED or NSD, and has lowest NED [%] and NSD-[%] range compare to the SyAL, 2N-2N2P and the SAL circuit styles. For example, in Fig. 5.26, CSSAL has 100-sample distribution out of 100 data samples between 0.8–1.8 NSD percentage, which is very small range of variations (high stability) and at the lowest scale of NSD [%] (high resistivity against power analysis attack) than the one of SyAL. On the other hand, the SAL and 2N-2N2P multiplier results in Figs. 5.27, 5.28 show that, the they have vulnerability in terms of circuit's stability and resistivity under 0.18 μm CMOS process variation in this comparison study.

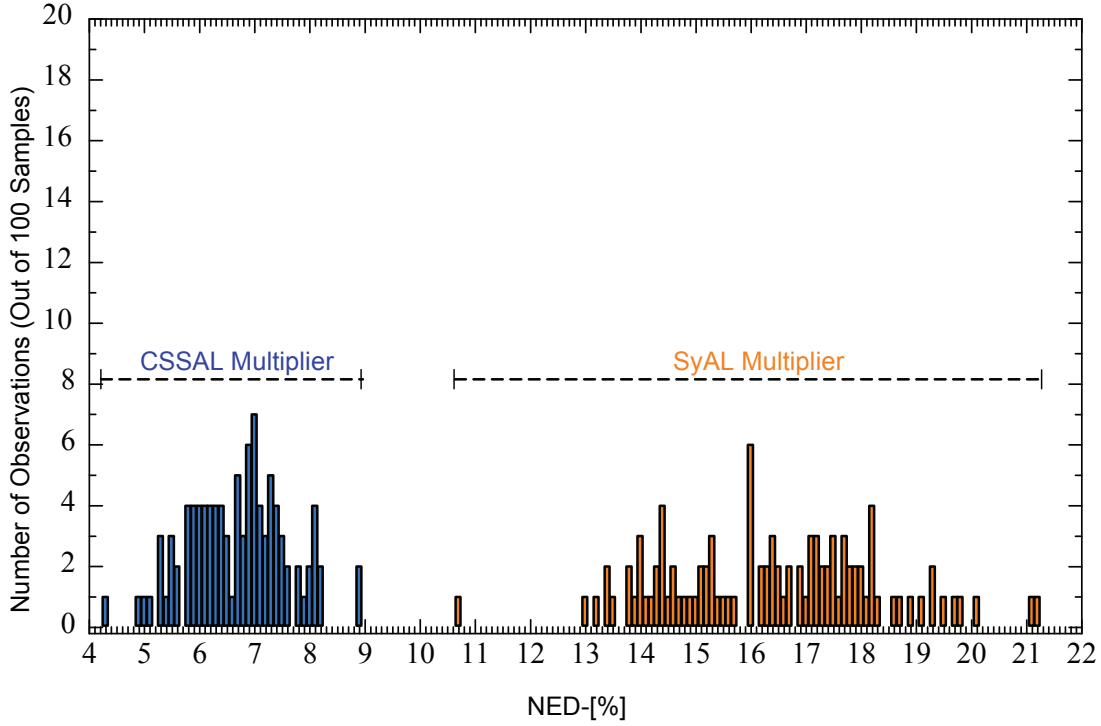


Figure 5.25: NED occurrences of the CSSAL and SyAL multipliers using Monte Carlo simulation for process variation verification.

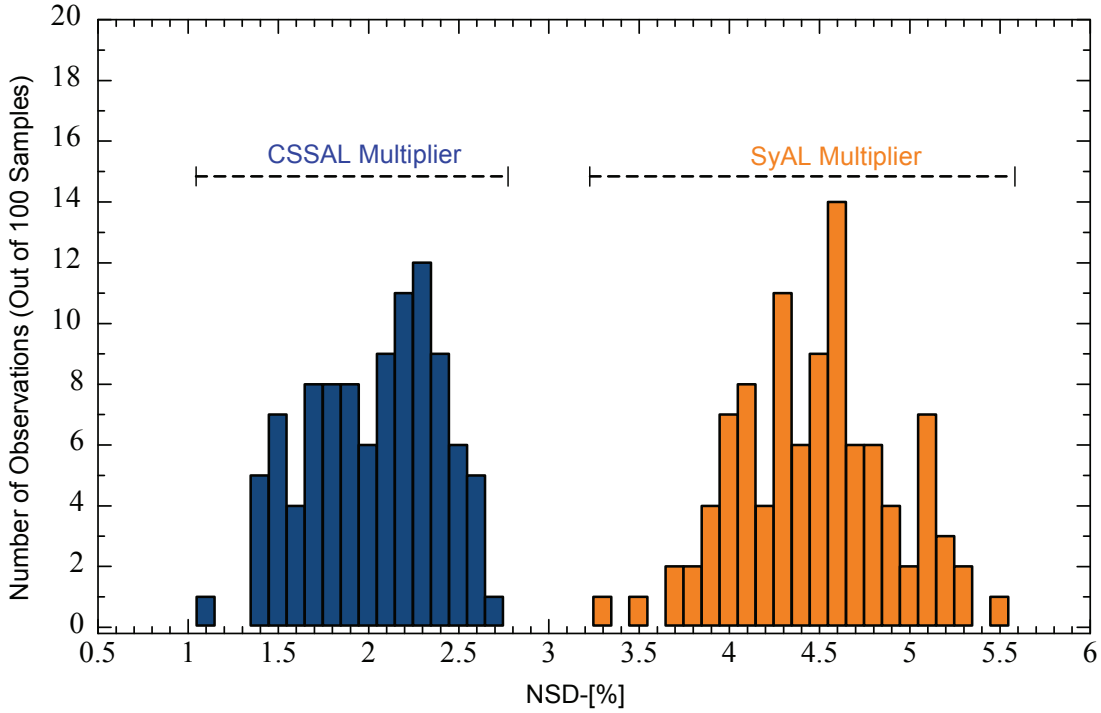


Figure 5.26: NSD occurrences of the CSSAL and SyAL multipliers using Monte Carlo simulation for process variation verification.

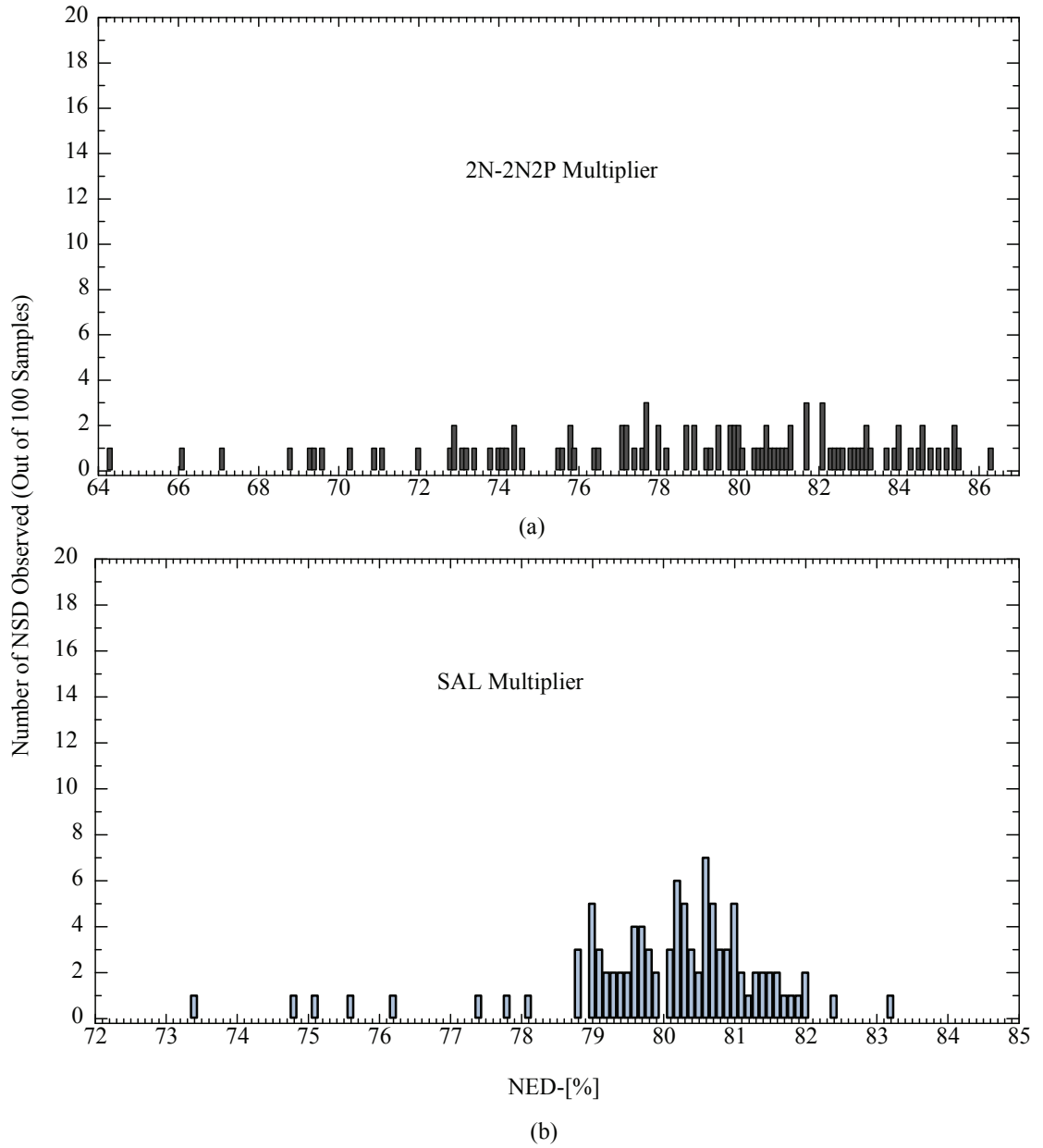


Figure 5.27: NED occurrences of the 2N-2N2P and SAL multipliers using Monte Carlo simulation for process variation verification.

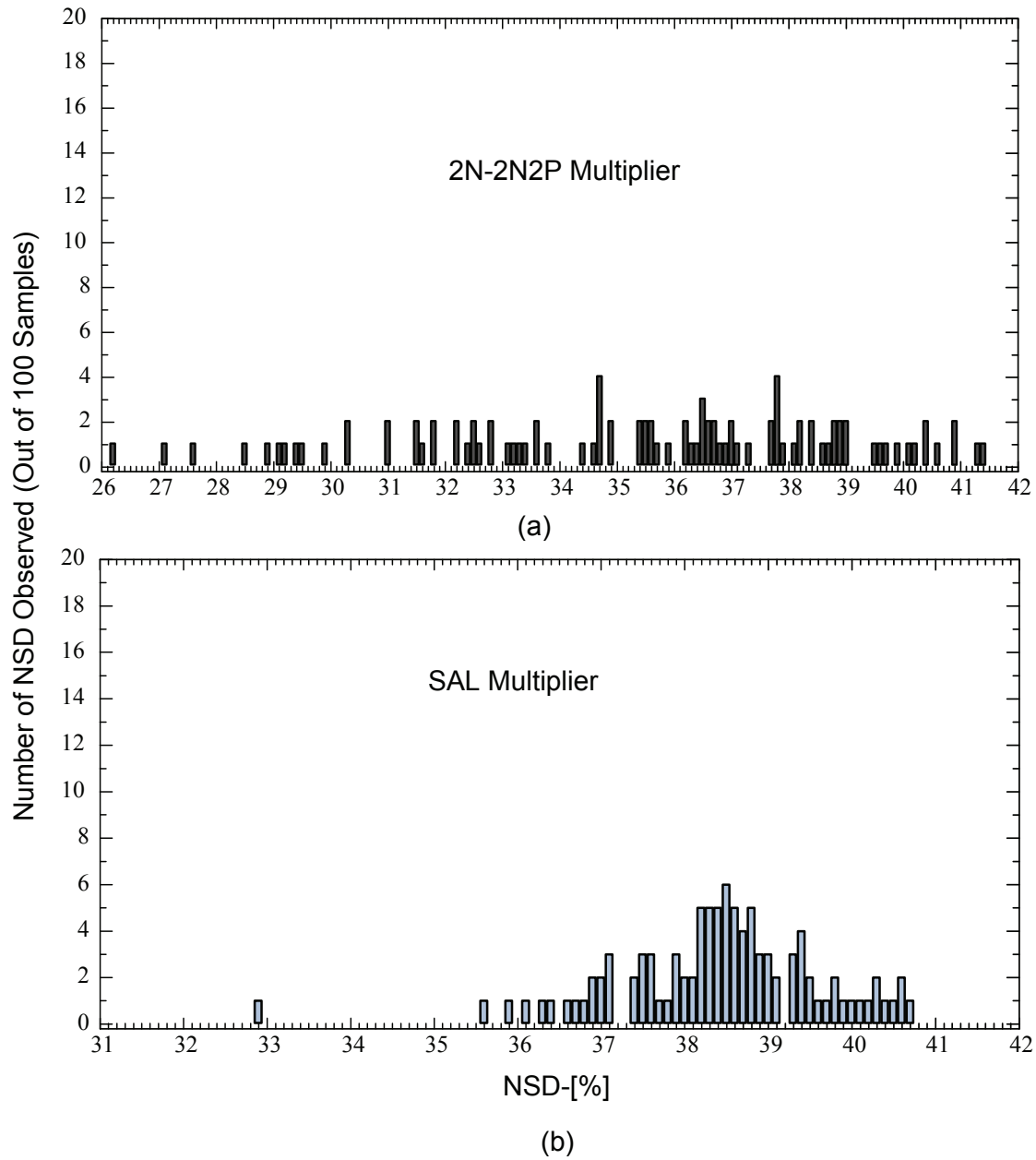


Figure 5.28: NSD occurrences of the 2N-2N2P and SAL multipliers using Monte Carlo simulation for process variation verification.

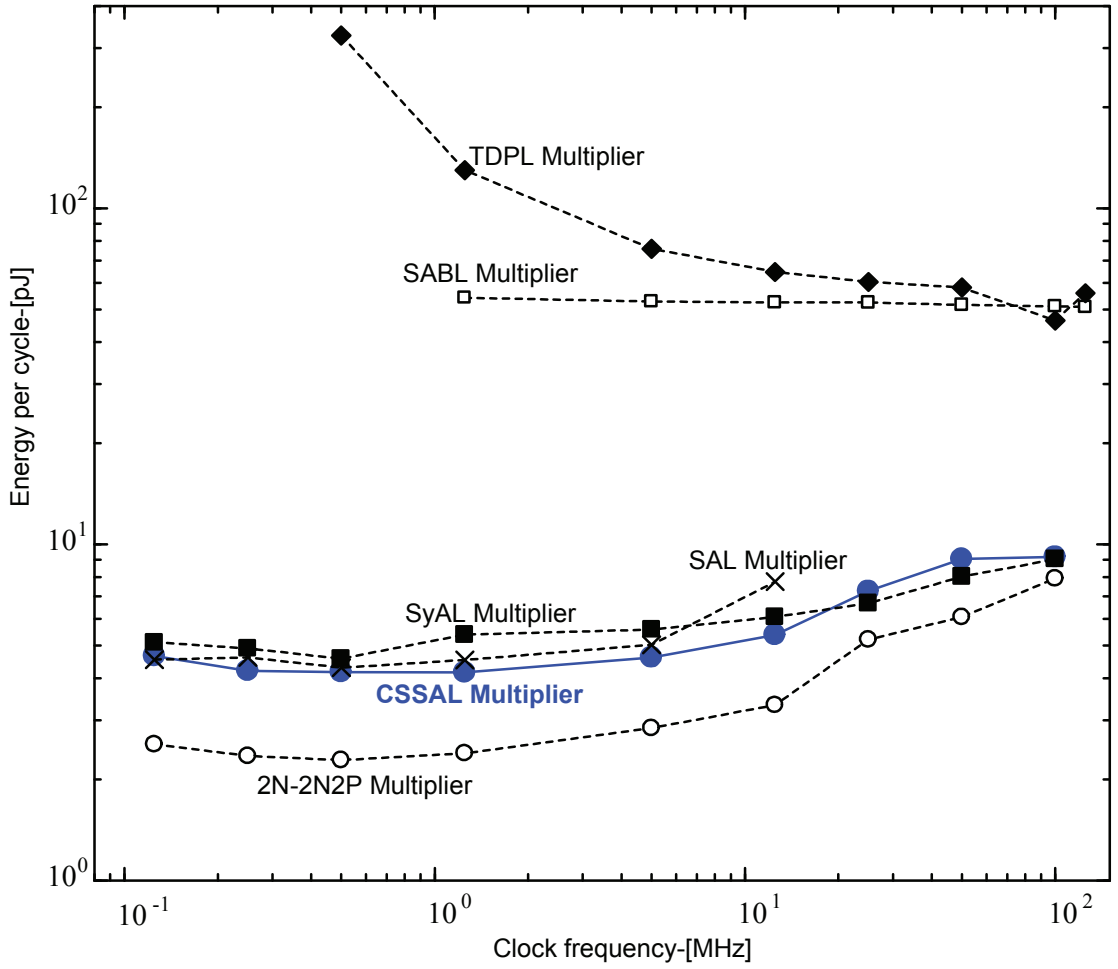


Figure 5.29: Energy dissipation comparison of the bit-parallel multiplier over $GF(2^4)$ using different logic styles.

Energy Dissipation Comparison

Figure 5.29 shows the cyclical energy dissipation of all multiplier circuits with dynamic frequency range from 125 KHz–125 MHz. The results in this graph are obtained from the pre-layout SPICE simulation with nominal capacitance at output nodes are in default, meaning without parasitic capacitance attached to individual logic of inner cell circuit. From this graphical information, we can say that the proposed logic consumes comparable energy to the other secure adiabatic logic styles. The 2N-2N2P has the lowest energy dissipation along the frequency ranges, because it has low gate complexity than the other. Adiabatic logic styles have ultra-low power compared to the conventional CMOS logic styles, for example, CSSAL reduces energy about 13 and 31 times from the SABL and TDPL, respectively at 1.25 MHz. From the security perspective, the CSSAL lowers the signal size, and hence the power and electromagnetic analysis may find some difficulties on CSSAL.

5.3 An 8-Bit AES S-Box Circuit using PPRM Representation

5.3.1 Overview of Advanced Encryption Standard

The advanced encryption standard (AES) algorithm is known as Rijndael algorithm, a symmetric block cipher that can process data block of 128 bits, using cipher key with key length 128, 194, and 256 bits. The basic unit for processing the AES algorithm is a Byte, a sequence of 8-bit treated as a single entity. The all byte values in the AES algorithm will be presented as an concatenation of its individual bit values (0 or 1) between braces in the order $\{b_7, b_6, b_5, b_4, b_3, b_2, b_1, b_0\}$. These bytes are interpreted as finite field element using polynomial representation:

$$\begin{aligned} & b_7x^7 + b_6x^6 + b_5x^5 + b_4x^4 + b_3x^3 + b_2x^2 + b_1x + b_0 \\ = & \sum_{i=0}^7 b_i x^i \end{aligned} \quad (5.15)$$

Finite field element can be added as “XOR” operation, or multiplied as “AND” operation. In the polynomial representation, multiplication in $GF(2^8)$ correspond with the multiplication of polynomial modulo an irreducible polynomial of degree 8. A polynomial is irreducible if its only divisors are one and itself. For AES algorithm, this irreducible polynomial is

$$x^8 + x^4 + x^3 + x + 1 \quad (5.16)$$

General AES encryption process is depicted in Fig. 5.30. There are eleven rounds compose of initial round till the final round. Initial round only AddRoundKey process, which to add the input plaintext with the initial secret-key. For the next nine rounds are repetitions of previous round with the new generated key. The last round does not make use of MixColumn transformation, which make encryption and decryption scheme symmetric. The following is the brief review of the AES round transformation:

SubByte () Transformation

The **SubByte()** transformation is a non-linear byte substitution that operates independently on each byte of the state using substitution table (S-box). There are

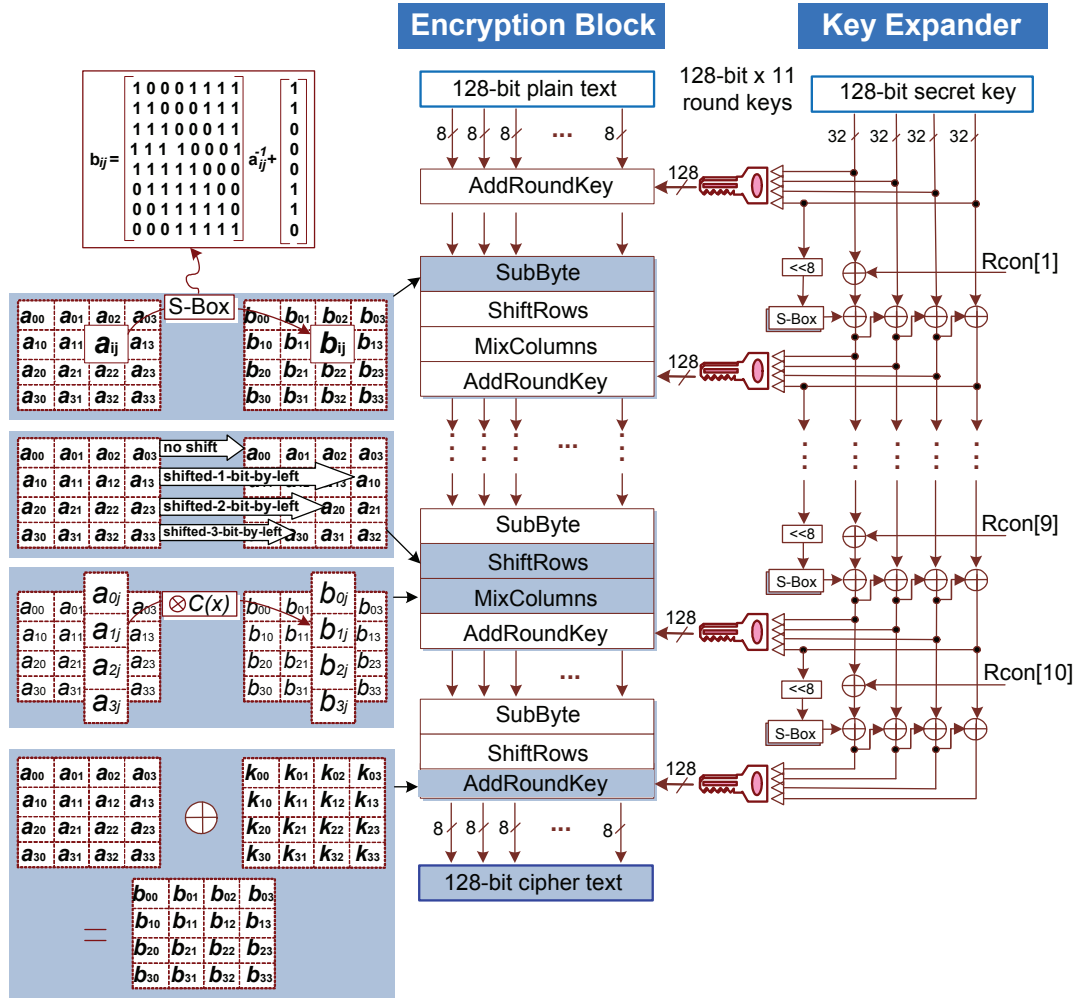


Figure 5.30: Encryption process of AES algorithm (adopted from original article [101]).

16 parallel S-boxes, each with 8 input and output bits as can be seen in Fig. 5.30. In the AES hardware design, the most difficult, high cost, and big area requirement in AES IC chip is SubByte block diagram.

ShiftRows () Transformation

In the **ShiftRows()** transformation, the bytes in the last three rows of the state are cyclically shifted over different numbers of bytes (offsets). The first row, $r = 0$, is not shifted. The second row is shifted 1-bit-to-the-left. The third row is shifted 2-bit-to-the-left. The fourth row is shifted 3-bit-to-the-left.

MixColumn () Transformation

MixColumn() treats four byte data block in each column as coefficients of a 4-term polynomial, and multiplies the data modulo $x^4 + 1$ with a fixed polynomial $c(x)$, as following:

$$a(x) = x^3 + x^2 + x^1 + 1 \quad (5.17)$$

$$c(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\} \quad (5.18)$$

And then, the $b(x) = c(x) \bullet a(x)$ can be calculated as:

$$\begin{aligned} b(x) = & \{03\}x^6 + (\{01\} \oplus \{03\})x^5 + (\{01\} \oplus \{01\} \oplus \{03\})x^4 \\ & + (\{02\} \oplus \{01\} \oplus \{01\} \oplus \{03\})x^3 + (\{02\} \oplus \{01\} \oplus \{01\})x^2 \\ & + (\{01\} \oplus \{03\})x + \{02\} \end{aligned} \quad (5.19)$$

The result, $b(x)$, does not represent a four-byte word. Therefore, the second step of the multiplication is to reduce $b(x)$ modulo a polynomial of degree 4; the result can be reduced to a polynomial of degree less than 4, as formulated following

$$x^i \bmod (x^4 + 1) = x^{i \bmod 4}, \quad (5.20)$$

then, the four term $b(x)$ polynomial can be written as following:

$$\begin{aligned} b(x) = & (\{02\} \oplus \{01\} \oplus \{01\} \oplus \{03\})x^3 \\ & + (\{03\} \oplus \{02\} \oplus \{01\} \oplus \{01\})x^2 \\ & + (\{01\} \oplus \{03\} \oplus \{02\} \oplus \{01\})x \\ & + (\{01\} \oplus \{01\} \oplus \{03\} \oplus \{02\}), \end{aligned} \quad (5.21)$$

or simply written in matrix equation as follows,

$$\begin{bmatrix} b_{0j} \\ b_{1j} \\ b_{2j} \\ b_{3j} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} a_{0j} \\ a_{1j} \\ a_{2j} \\ a_{3j} \end{bmatrix} \quad (5.22)$$

for $0 \leq j < Nb$, where Nb is the number of block as shown in Fig. 2.

AddRoundKey () Transformation

The round key is added to the state by a simple bitwise XOR operation, as indicated in Fig. 5.30.

Table 5.4: AES standard key-block-round combination

	Key Length (Nk Words)	Block Size (Nb Words)	Round Number (Nr)
AES-128	4	4	10
AES-192	6	4	12
AES-256	8	4	14

Key Expansion

The AES algorithm takes the cipher key, \mathbf{K} , and performs a key expansion routine to generate a key schedule. The key expansion generates a total of $Nb(Nr + 1)$ words: the algorithm requires an initial set of Nb words, and each of the Nr rounds requires Nb words of key data. The resulting key schedule consists of a linear array of 4-byte words, denoted $[w_i]$, with i in the range $0 \leq i < Nb(Nr + 1)$.

5.3.2 AES PPRM S-Box Circuit

The targeting S-box circuit of the multi-stage positive polarity Reed-Muller (PPRM) architecture has been proposed by Morioka and Satoh in [102]. The PPRM is the representation of Boolean function as a single algebraic sum (XOR) of one or more conjunction. Three sub-components of the conventional composite field S-box circuit were converted into the PPRM form: the pre-inversion section, the inversion section, and the post-inversion section, as depicted in Fig. 5.31. In the proposed CSSAL S-box circuit, the author applies three power clock supplies for each section, which completely avoid the glitch current, consume uniform transitional energy and ensure significant energy reduction in these comparative results, even though the transistor counts much higher than other adiabatic logics investigated, as shown in Table 5.5. Triple power clock Vpc0, Vpc1 and Vpc2 signals are depicted in Fig. 5.32. The primary input signal (X0, X0-) and Vpc2 shown in Fig. 5.32, and the input signal with Vpc signal shown in Fig. 4.13(b) are identical. The S-box circuit structure presented in the appendix of [102] describes that variables $x_7 - x_0$ denote the primary inputs of an S-box and $y_7 - y_0$ denote primary outputs. The other variables such as a , b , c , and d denote the internal wires.

Moreover, the author utilizes a logic sharing method instead of a multiple logic

Table 5.5: Gate size, transistor counts, layout area and delay of an 8-bit S-box circuit (0.18- μm 1.8-V CMOS standard cell @ 12.5 MHz)

Gate Counts			
Circuit	Buffer	AND	XOR
Figure 5.31(b)	148	141	216

Transistor Counts and Delay					
Circuit	Buffer	AND	XOR	S-box	Delay (ns)
CSSAL	9×148	19×141	19×216	8,115	22.41
SyAL	5×148	15×141	15×216	6,095	12.29
2N-2N2P	6×148	8×141	10×216	4,176	15
PPRM [102]	–	8×203	16×228	5,272	3

recurrence method that uses the same input signals; hence, the dual-input logic complexity is reduced to approximately 17% that of Morioka's design in [102]. An example of Boolean function is shown in appendix B.1.2. Comparison of the transistor counts shown in Table 5.5 is calculated as following example: CSSAL S-box: $(9 \times 148) + (19 \times 141) + (19 \times 216) = 8,115T$, where T = Transistors, and the number of 148, 141, 216 are the gate counts of Buffer, AND and XOR respectively.

Furthermore, the proposed CSSAL S-box timing diagram in Fig. 5.32 shows that the phase delay time of V_{pc2} is 20 ns from primary input signal at 12.5 MHz power clock frequency; and therefore the logic speed is slow in comparison with the other adiabatic logic styles as summarized in Table 5.5. Definitions of adiabatic delay time is derived as

$$Td_{Adiabatic} = \text{Phase delay time} + \text{Propagation delay time} \quad (5.23)$$

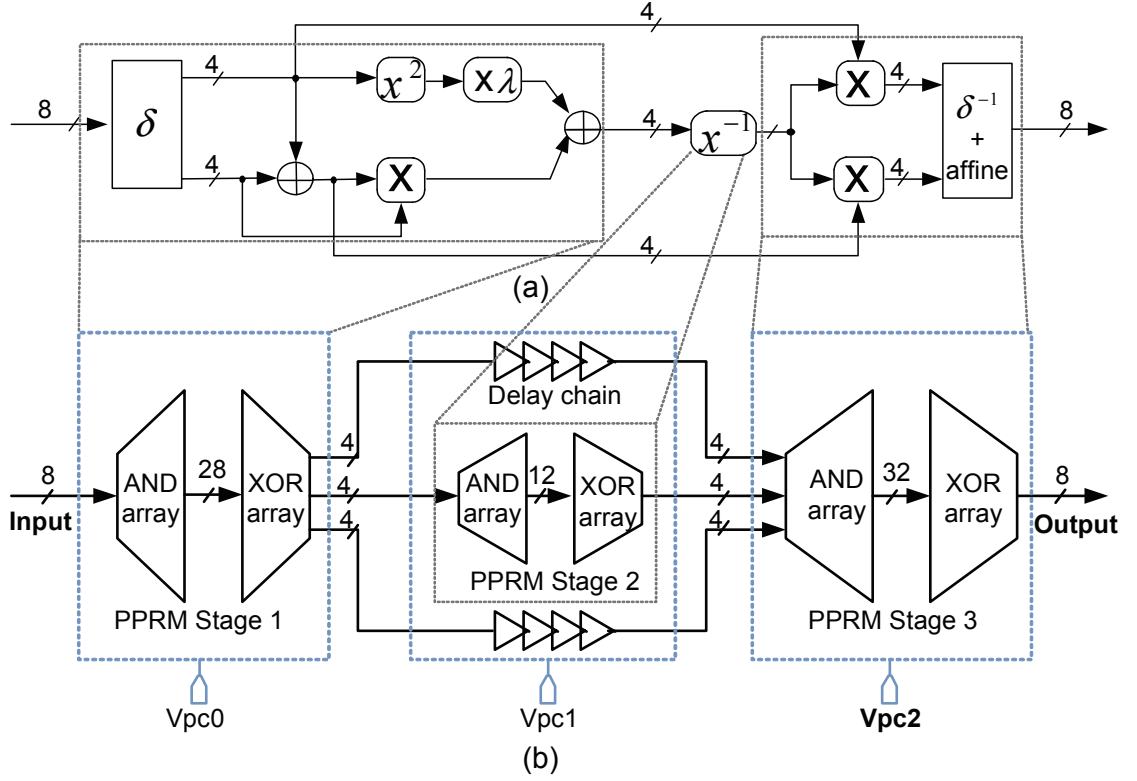


Figure 5.31: (a) Conventional composite field AES S-box architecture; (b) multi-stage PPRM representation with the implementation of the proposed triple V_{pcs} signals in the CSSAL 8-bit S-box circuit.

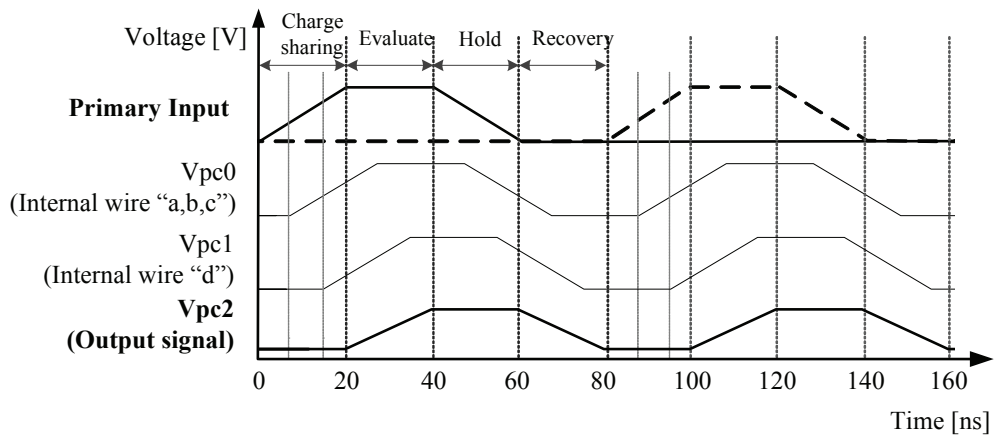


Figure 5.32: Triple power clock signals for CSSAL S-box circuit shown in Fig. 5.31(b).

Dynamic Hazard Effect in Secure Logic Design

The most annoyingly issue for logic designers is how to avoid dynamic hazard in the complex digital circuit, especially for essential and secure data processing. In this section, the author wants to explain how he removes the hazard voltage or glitch current in adiabatic operation. As you may know that the glitch current happens when inputs arrival time are different in a single logic or a block diagram. There are several approaches for avoiding dynamic hazards, such as balancing the logic path, shorting the critical path, parallelism, etc. In the case of adiabatic logic operation, the violation of phase delay cause the huge dynamic hazard. As indicated in Fig. 5.33(a) that, there are three-stages with single power clock supply; where the first stage with primary input signals, second stage where the input signals come from the output of first-stage. Then the output-stage where the inputs are coming from the outputs of stage-1 and stage-2. This design results the huge hazard voltage or high glitch current, as indicated with red-circle in Fig. 5.33(a).

To avoid this hazard voltage, the author uses three power clock signals for each-stage as shown in Fig. 5.33(b). With this technique, the dynamic hazard voltage is totally removed, and peak supply current is lowered about 74%. Simulation result that explains aforementioned issue is shown in Fig. 5.34.

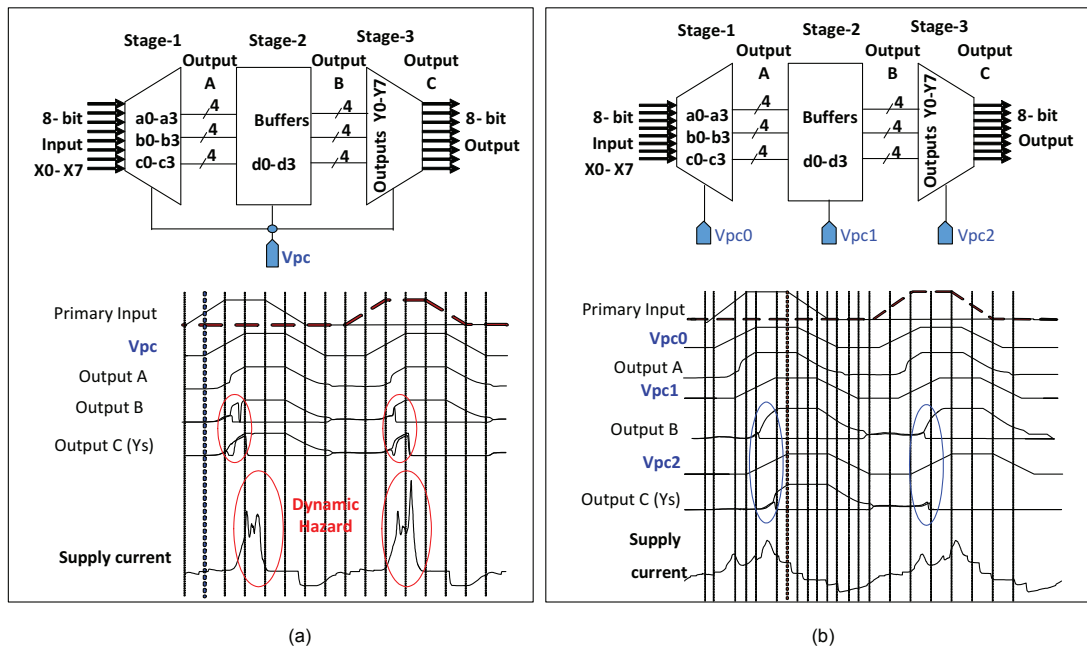


Figure 5.33: Dynamic hazard voltage removal using triple power clock supplies.

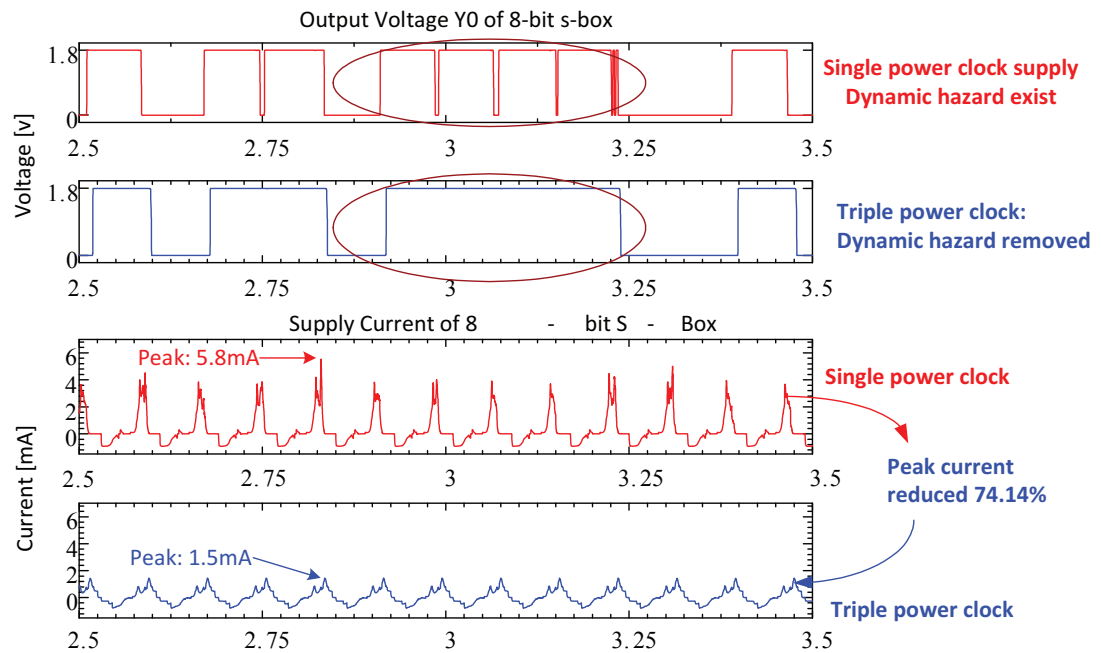


Figure 5.34: Simulation result of single power clock signals versus triple power clock signal.

5.3.3 Simulation Result

Input signals in SPICE simulation is depicted in Fig. 5.35. These input signals $a_0 - a_7$ are also supplied for 2N-2N2P and SyAL S-boxes. Respective output signals of the 8-bit S-box circuits are shown in Figs. 5.36–5.38. In these figures, we can observe that there are two signals (black and red) in each signals (exp. the Y0 signal). The red signal trace is the 8-bit S-box output of adiabatic signal and the black signal trace is plotted after using CMOS buffer circuit. This intends to show that whether the investigated S-box circuit produces hazard voltage or not after converting the adiabatic signal into normal pulse signal. Therefore, observing Figs. 5.36–5.38, the output signals of the CSSAL S-box completely removes the dynamic hazard voltage, if compare with the 2N-2N2P and SyAL S-boxes. For example, let us look at CSSAL Y0, 2N-2N2P Y0 and SyAL Y0 of Figs. 5.36–5.38 at time range of 4–6 μs , the CSSAL produces only the real state (logical 0 or 1), while the others produce also unwanted signals (dynamic hazard).

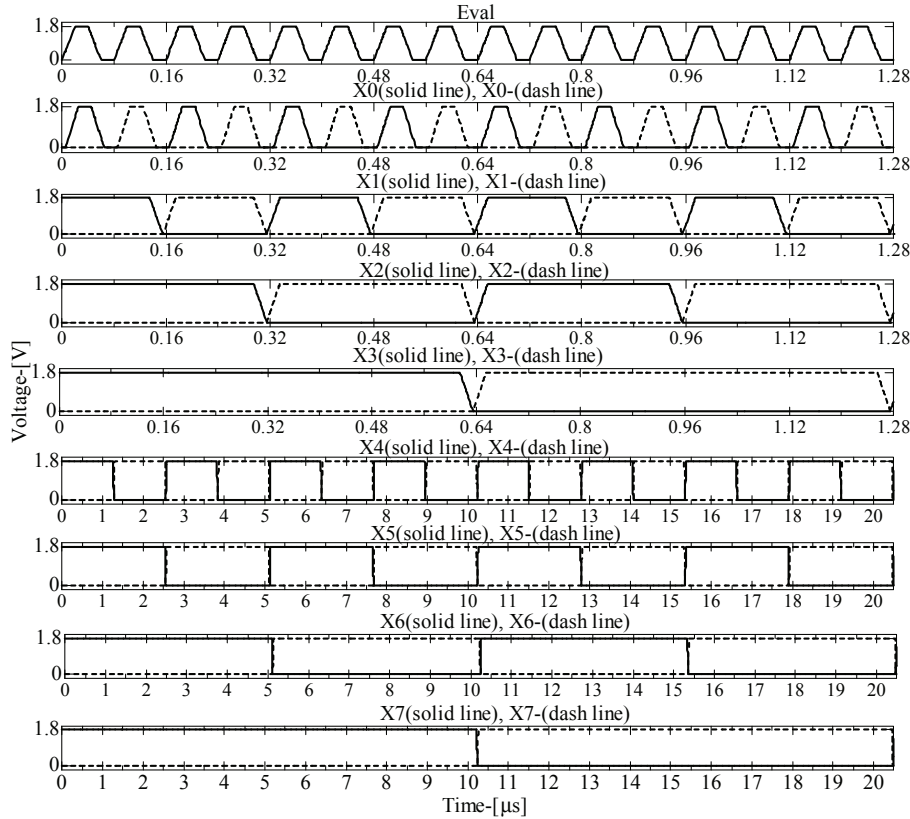


Figure 5.35: 8-bit input signals of the CSSAL S-box. Input X0–X7 also supplied to SyAL and 2N-2N2P S-boxes.

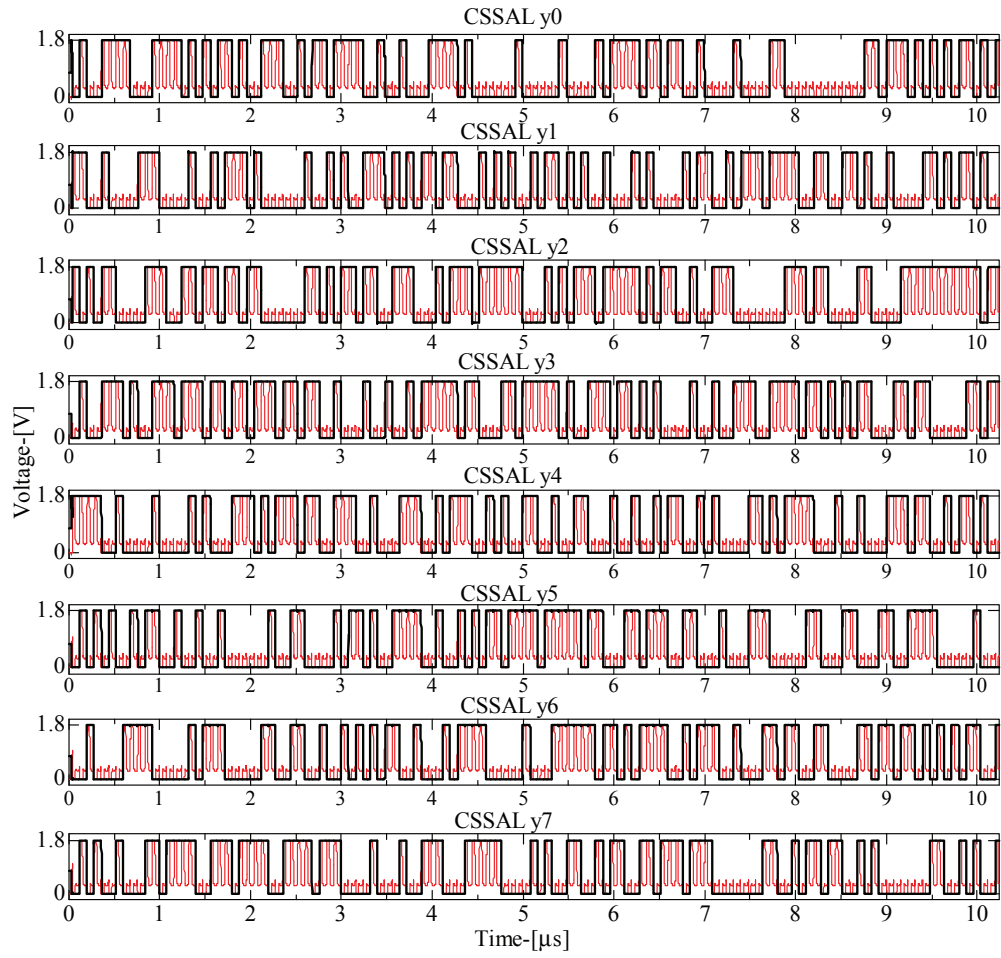


Figure 5.36: Output voltage of the proposed CSSAL S-box at 12.5 MHz power clock frequency.

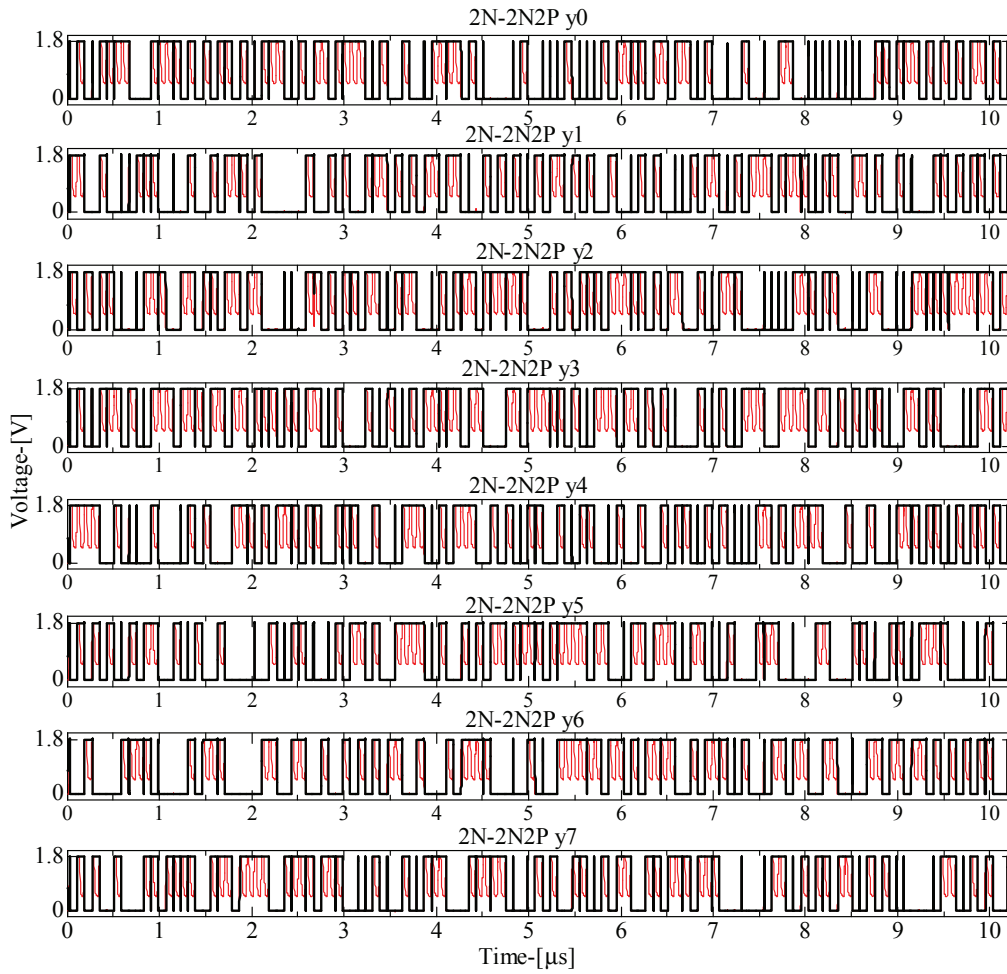


Figure 5.37: Output voltage of the 2N-2N2P S-box at 12.5 MHz power clock frequency.



Figure 5.38: Output voltage of the SyAL S-box at 12.5 MHz power clock frequency.

5.3.4 Frequency Spectrum Analysis Result: AES S-box Circuits

Similar explanation to the section 5.2.3 that we observe the harmonic frequency below the fundamental frequency f_0 . In this case, the circuit was investigated at power clock frequency of 12.5 MHz, and therefore the $f_0 = 12.5$ Mhz. Observing Fig. 5.39, the CSSAL S-box circuit has more balance peak current than the other investigating logic styles.

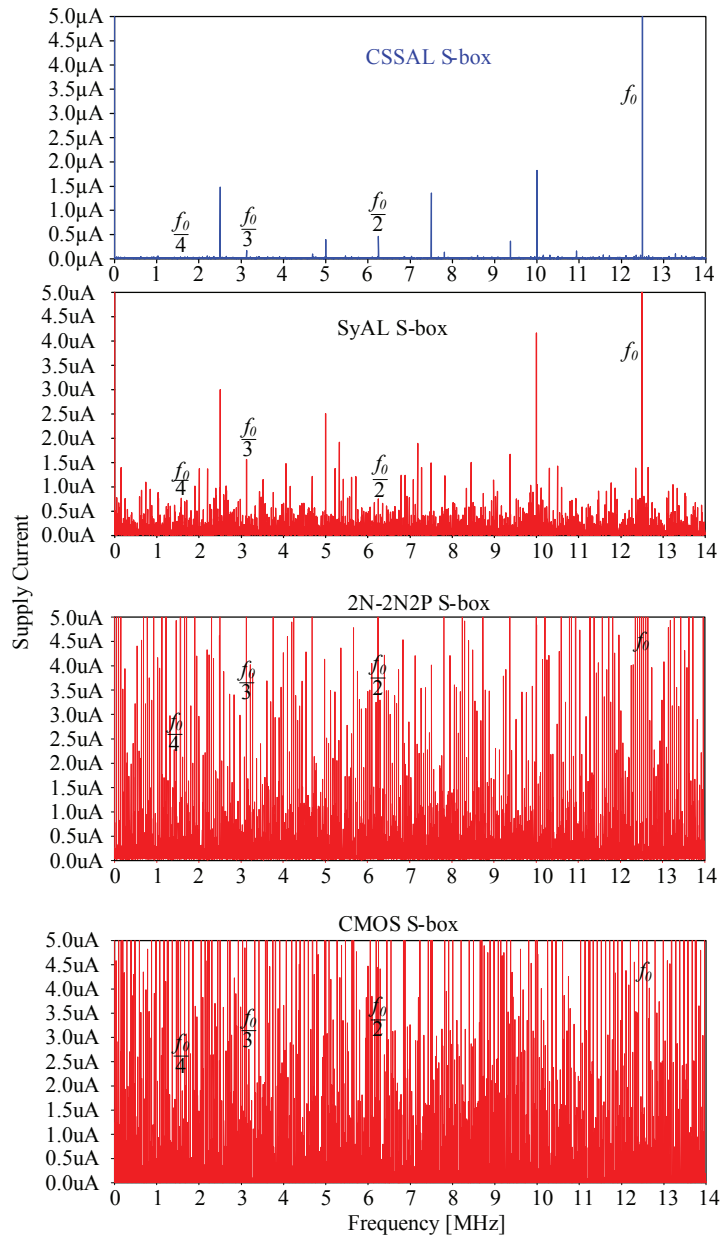


Figure 5.39: Signal spectrum of harmonic frequency of the AES S-box circuits.

5.3.5 Analysis and Comparison: AES S-box Circuits

Security View Point

Verification of Supply Current Traces : The supply current traces of all investigated S-box circuits are depicted in Fig. 5.40. This figure shows that the CSSAL S-box has the lowest and uniform peak supply current (about 1.5 mA order for all instantaneous peak supply current transitions by SPICE simulation result).

Verification of Energy Fluctuation : Simulation and calculation of the figure of merit to measure the resistance of the S-box circuit are summarized in Table 5.6. By observing this figure, we can conclude that the proposed CSSAL S-box circuit has an unique ability to withstand the DPA attack, because it has smallest values of NSD and FoM for all the active frequency bands.

Furthermore, a histogram of the observed energy per power clock cycle in Fig. 5.41 indicates that the CSSAL S-box circuit guarantees balanced energy consumption, independent of the processed data. For instance, the variation of transitional energy dissipation for the CSSAL S-box fluctuates between 6.5–6.8 pJ (6.5 pJ: E_{min} , 6.8 pJ: E_{max}) for 256 cyclical energy data samples; the CSSAL S-box has 184 numbers of observation at 6.7 pJ/cycle. On the other hand, the SyAL S-box gives a distribution of 18 numbers at 13.4 pJ/cycle (between 10.8–15.2 pJ), the 2N-2N2P S-box only gives a distribution of 5 numbers at 20 pJ/cycle (between 3.5–33.0 pJ), and the CMOS S-box [102] has only a distribution of 4 numbers at 25.3 pJ/cycle (between 3.5–44.4 pJ). In addition, it has been recognized that the conventional CMOS logic styles such as TDPL and SABL logics are stronger to DPA attacks in secure logic implementation; however, comparative data are not available in this work because we have found out in our thoroughly SPICE simulation results that they are not suitable to operate in the 8-bit S-box using the PPRM representation.

Verification of CMOS Process Variation : The results in Fig. 5.42 show that, the proposed CSSAL S-box circuit has less variation and always in the smallest scale of NSD percentage, in which the author assures that proposed CSSAL is stable and resistive against power analysis at any condition.

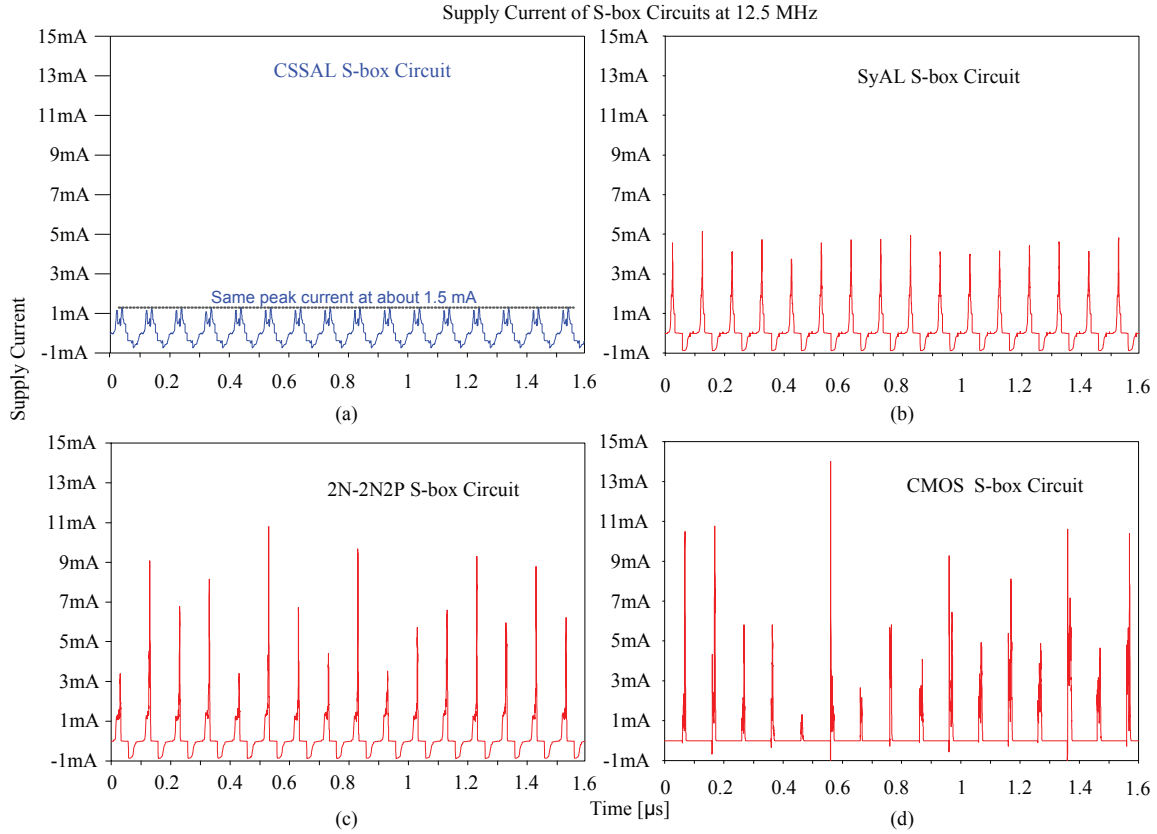


Figure 5.40: Comparison of peak supply current traces of the multi-stage PPRM 8-bit S-box circuit at an operating frequency of 12.5 MHz (at the worst case of the current-to-data dependency) using different logic styles; (a) S-box circuit using CSSAL, (b) S-box circuit using SyAL, (c) S-box circuit using 2N-2N2P and (d) S-box circuit using Morioka's circuit.

Table 5.6: Simulation and calculation results of 8-bit S-box circuit in PPRM representation using proposed CSSAL, SyAL, 2N-2N2P, and Morioka's circuit [102] respectively at 1.25 MHz, 12.5 MHz, and 50 MHz input power clock frequencies

Power variation of 8-bit S-Box circuit using PPRM representation												
Logic	Proposed CSSAL			SyAL			2N-2N2P			Morioka [102]		
Freq.-[MHz]	1.25	12.5	50	1.25	12.5	50	1.25	12.5	50	1.25	12.5	50
E_{min} [pJ]	3.11	6.5	13.52	4.85	10.78	24.87	1.83	3.49	6.48	3.50	3.50	2.80
E_{max} [pJ]	3.42	6.76	14.04	5.87	15.21	36.64	12.66	33.04	51.79	44.39	43.91	44.33
\overline{E} [pJ]	3.35	6.67	13.87	5.12	13.29	30.34	6.94	18.86	27.53	25.50	22.20	24.40
σ_E [pJ]	0.05	0.04	0.09	0.13	0.79	1.85	1.88	5.70	8.34	8.18	8.07	8.22
NED [%]	9.06	3.84	3.70	17.38	29.13	32.12	85.55	89.44	87.49	92.11	92.03	93.68
NSD [%]	1.49	0.60	0.65	2.54	5.94	6.10	27.09	30.22	30.29	36.37	36.36	36.62
$F_oM (\sigma_E \overline{E})$ [pJ ²]	0.17	0.27	1.25	0.67	10.50	56.13	13.05	107.50	229.60	208.59	179.15	200.57

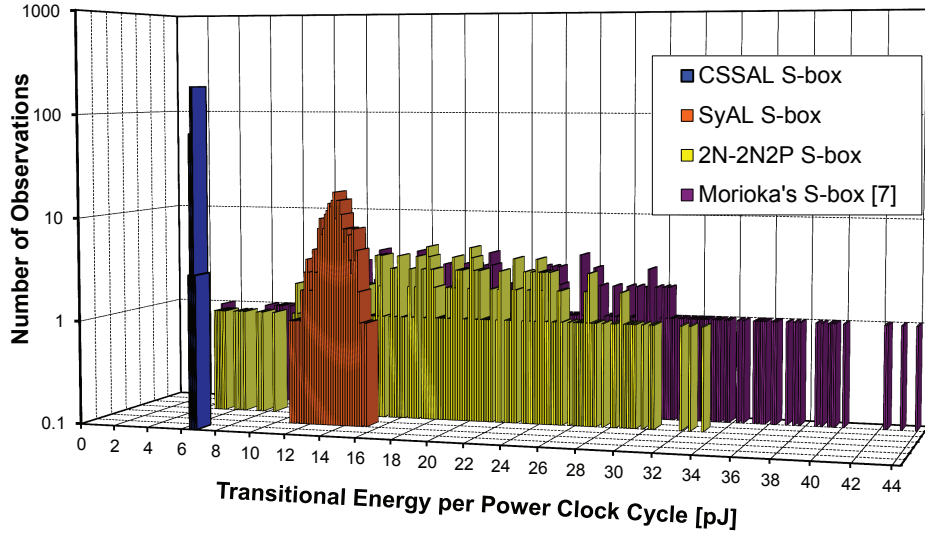


Figure 5.41: Observed amount of energy consumed per input transition: proposed CSSAL, SyAL, 2N-2N2P Morioka's circuit [102] implementation in the 8-bit S-box for 256 energy data sample.

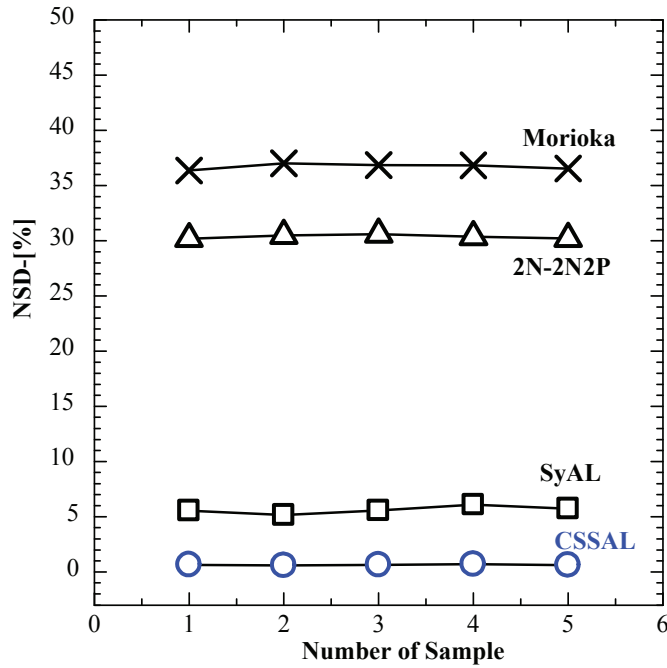


Figure 5.42: Simulation and calculation results of NSD using 0.18- μm CMOS process variation for five samples of each circuit.

Energy Dissipation

The graphical information shown in Fig. 5.43 evidently shows that our proposed CSSAL S-box has significant energy reduction as much as about a half of the other adiabatic S-box circuits investigated in this work. Moreover, the CSSAL S-box reduces energy about 3.3 times smaller than the conventional CMOS logic style in Morioka's work [102] at 12.5 MHz operating frequency.

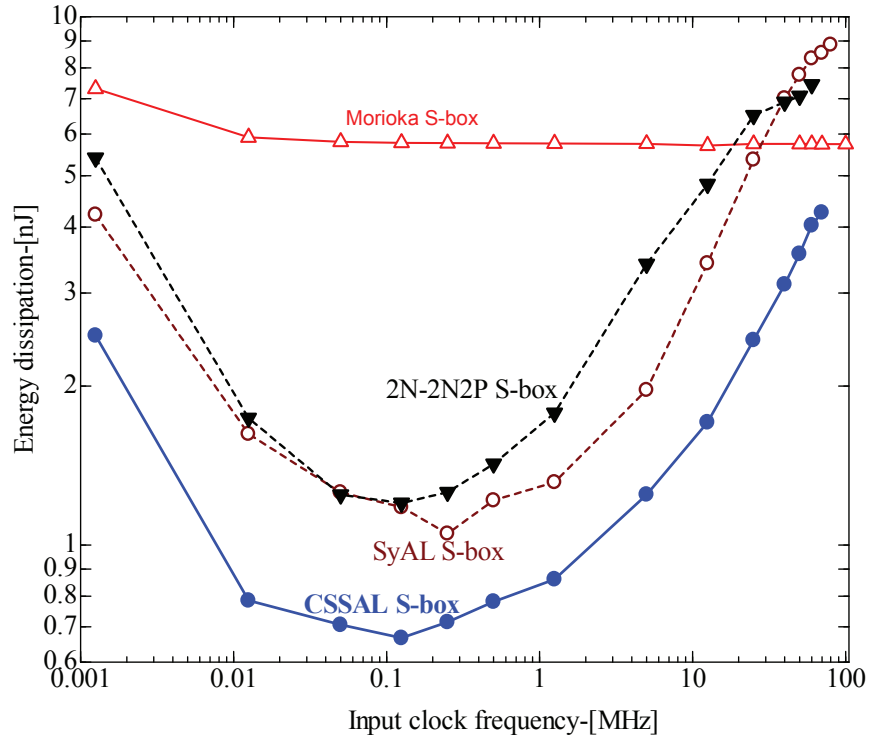


Figure 5.43: Simulated energy dissipation comparison of all the investigated adiabatic logics: CSSAL, SyAL, 2N-2N2P, and Morioka's circuit [102] in the multi-stage PPRM 8-bit S-box circuit at each operating frequency ranges.

5.4 Summary

In this chapter, the proposed CSSAL implementation in an existing bit-parallel cellular multiplier over $GF(2^4)$ and 8-bit AES S-box circuit have been thoroughly investigated. Based on the overall SPICE simulation results of the logic implementation that investigated in this chapter, the author emphasizes several points as summary, as follows:

1. CSSAL multiplier and S-box circuits have performed uniformly supply current transitions at low frequency bands if compared to the other low-power secure adiabatic logic styles.
2. CSSAL multiplier and S-box circuits have ultra-low signals in comparison with the conventional secure logic styles, such as the TDPL and the SABL. Practically, the power analysis attacker measures the overall power traces, for instance between the Vdd and the device under attacks; consequently, the supply current signals of the TDPL will be easily to identify its variances, whereas the CSSAL will effectively confront the power and electromagnetic analysis attacks.
3. Investigation results from frequency spectrum analysis, the CSSAL has low harmonic frequency below the fundamental frequency (f_0), thus it has confirmed that the proposed logic exhibits more uniform current traces than the other logic styles.
4. The effectiveness and the stability of the proposed CSSAL has been tested under CMOS process variations using Monte Carlo simulation which the results have shown that the proposed multiplier and the S-box circuits have very small range of variations (high stability) and always at the lowest scale of NSD [%] (high resistivity against power analysis attack) than the other secure adiabatic logic styles.

Chapter 6

LSI Implementation

6.1 Introduction

This chapter covers the large scale integrated (LSI) circuit implementation. Firstly, the author will describe the full custom layout design of ASIC-based design implementation in Section 6.2. The post-layout simulation will be conducted in order to check the circuit behavioral, whether the results meet the design specifications or not. Further investigation of the logic's merit, *i.e.*, from security aspect and from the dynamic power consumption corresponds to the dynamic operating frequency. Section 6.3 will present the fabricated LSIs measurement results. In the end of this chapter, the author will highlight the LSI features and some design challenges to be considered which will be oriented for future works.

6.2 Full-Custom LSI Layout Design

The fabricated LSI discusses in this section was implemented using full-custom layout design methodology. Although the full-custom layout design requires extensive work which consuming extra time of design, it enables the designer to fully control the performance of the chip, minimizes chip area, and/or able to reduce the power dissipation. We can achieve a high specification performance by trial-and-error during design process. The steps of the full-custom design flow during design process started from front-end to back-end verifications, tape-out and measurement are illustrated in Fig. 6.2. As shown in this diagram, the design process are grouped in four steps, as follows:

1. *Logic Design* : In this step, the definition of design specification, such as intended application purpose, clock frequency, input-output timing, function

mapping (*i.e.*, logic sharing), process selection (*i.e.*, 0.18 μm CMOS process), circuit architecture (in this work, the symmetric dual-rail logic) are all defined. The functional of the logic design, input-out timing verification, and the performance of the design specification are validated using analog circuit simulation. In this case, the LTspice analog simulator (LTspice IV version 4.20) and Hspice (Cadence Virtuoso Schematic Editor IC6.1.4.500-1) were used. This is the deterministic step which was conducted and verified using LTspice, and has been widely discussed in Chapter 4. Simulation results of this step is called as pre-layout simulation. This step also prerequisites of the consecutive design step, in which the CMOS circuit schematic, symbol creation and simulation verification were confirmed using Hspice schematic editor. Whole procedures and design methodology in this step is generally known as front-end verification (or so-called back-annotation).

2. *Layout Design*: If the pre-layout results meet the design specification, and then the layout is designed. In this step, the gate size of L/W , the size and distance between metal wires are freely decided, however must not below minimum sizes of design rule. Interestingly, the design rule enables us to check the cell by cell by extracting the design rule checking (DRC). If the DRC checking is met, then the layout versus schematic (LVS) check is extracted. The schematic mentioned here is referred to step 1. And, if both DRC and LVS perform some errors, then we have to step back to the layout design, and otherwise continue for the next step.
3. *Post-Layout Simulation*: This step is conducted using transistor netlist after the internal parasitic resistance and capacitance extraction (or shortly called as RCXT) are done, which is known as standard parasitic format file (.spf extension file). The parasitic RC values are depend on the layout design skills. For the new beginners, the trail-and-error could be an option to obtain a high performance in the end of this step. The accuracy of the post-layout simulation result is very high and close to the real chip, and therefore we are sometimes faced to mismatching of design performance between pre and post layout results. From the step of layout to post-layout simulation design flow is so-called as back-end verification. According to the diagram in Fig. 6.2, if the output result of this step different with the one of step 1 result, then we have to turn back to layout design; and otherwise go further to the next step.

4. *Fabrication and LSI Measurement* : When all design constraints are met, a file containing a circuit layout is generated and sent to the semiconductor foundry for manufacturing process. After getting the fabricated chip, then the measurement of electrical function, I/O function and LSI performance for designated specific purpose are verified.

The design steps described above were strictly obeyed during design process. The most time the author has spent was on the logic behavioral design and specification. Throughout these design procedures, the author was faced a lot of challenges, and specifically in layout design, in which the countless errors were occurred, and hence, the author has overcome with a great learning process.

6.2.1 Individual Logic Layout

The layout design of the fundamental logic styles are based on the logic implementation in Chapter 5. Therefore, the author has designed the AND and XOR logic circuits of the CSSAL, 2N-2N2P, SyAL and the TDPL. Some inverter logic circuits were also designed for critical path balancing analysis. The layout diagrams are shown in Figs. 6.2.1–6.7 for each respected circuit as mentioned in each caption.

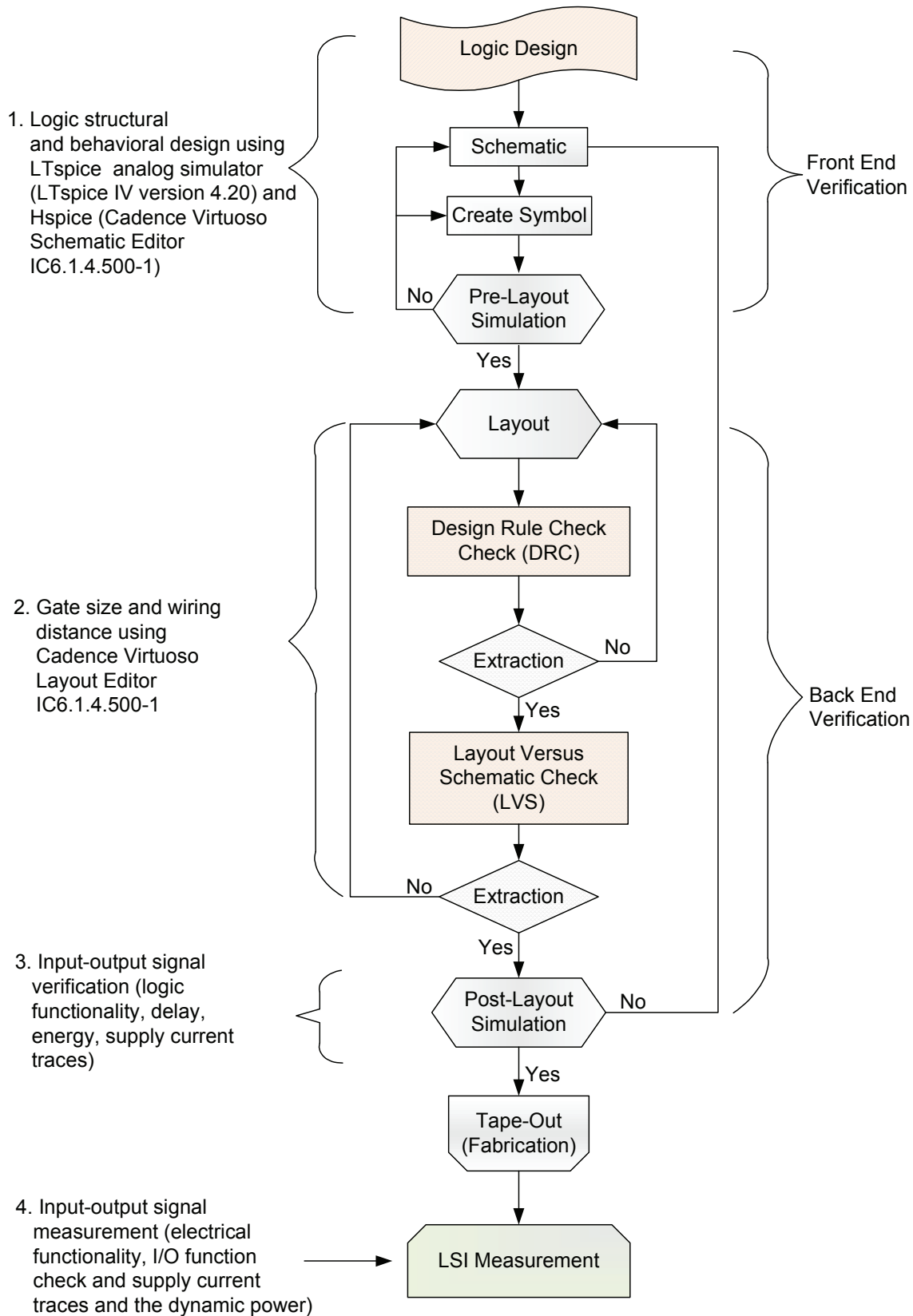


Figure 6.1: Full-custom LSI design flow.

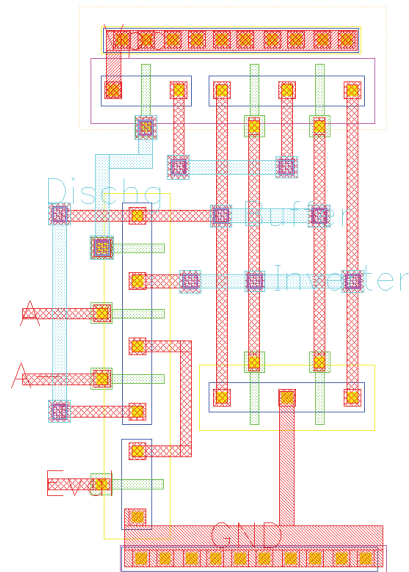


Figure 6.2: Layout of the CSSAL inverter/buffer of the circuit schematic diagram in Fig. 4.16(a).

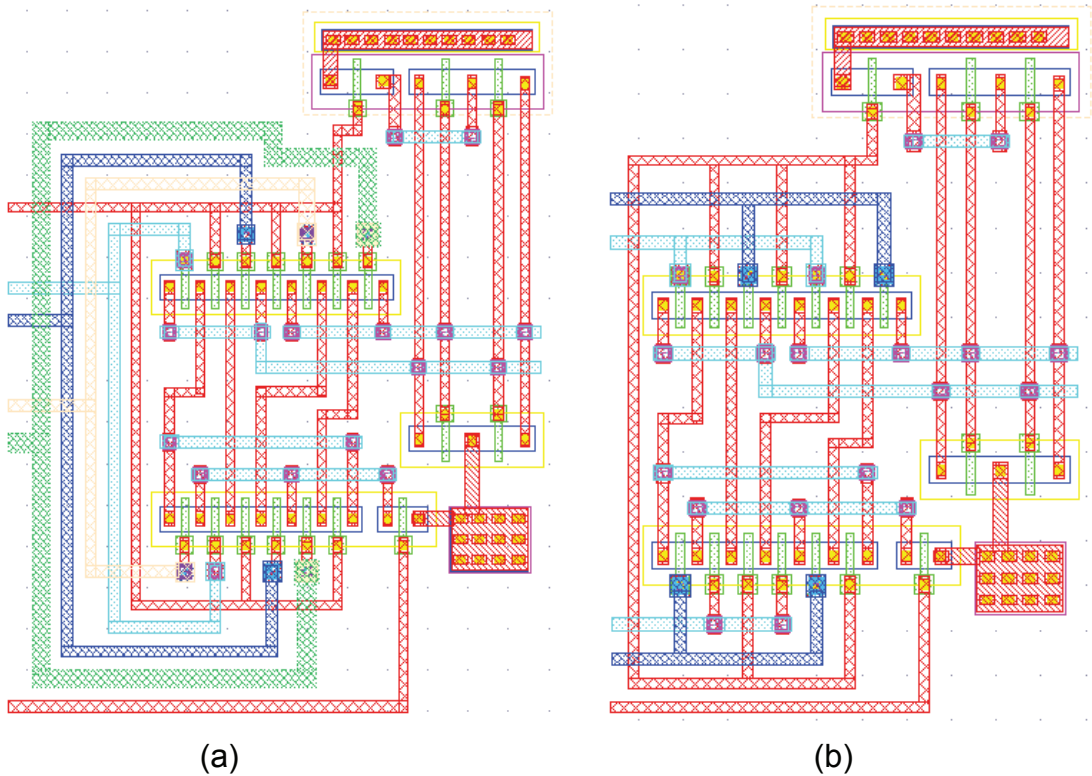


Figure 6.3: Layout of the CSSAL ver.2 circuit; (a) NAND/AND, (b) XNOR/XOR.

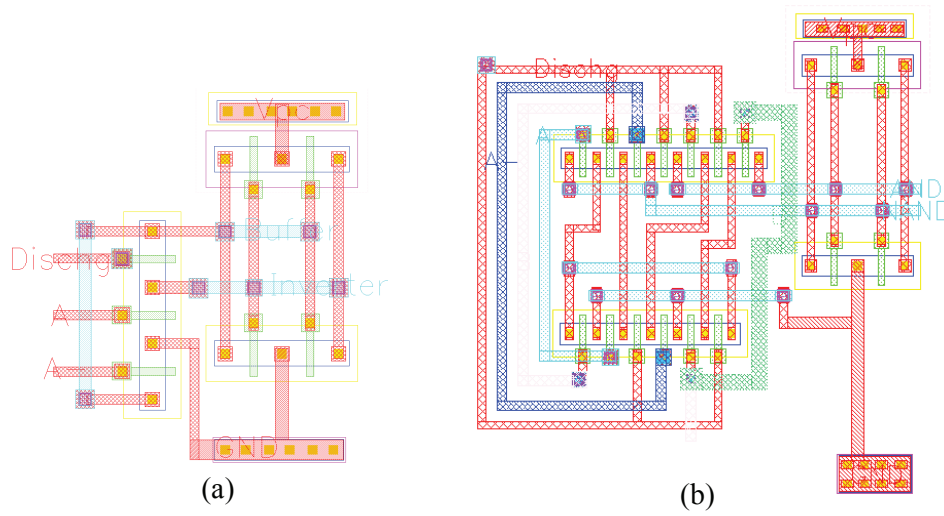


Figure 6.4: Layout of the CSSAL ver.4 circuit; (a) NAND/AND, (b) XNOR/XOR.

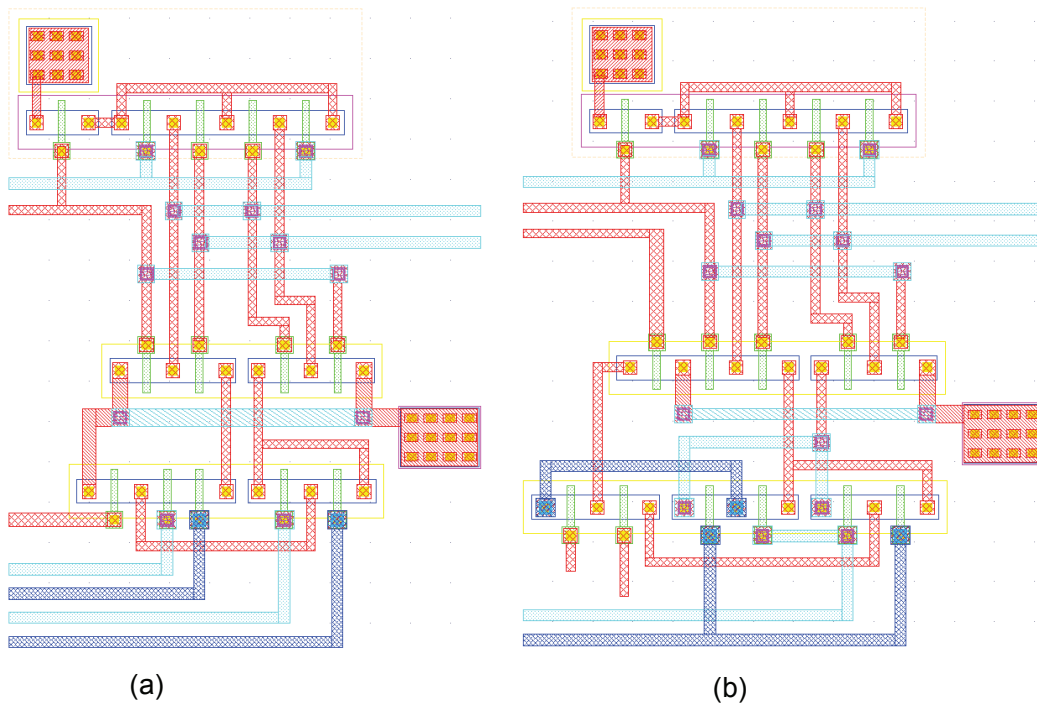


Figure 6.5: Layout of the TDPL circuit; (a) NAND/AND, (b) XNOR/XOR.

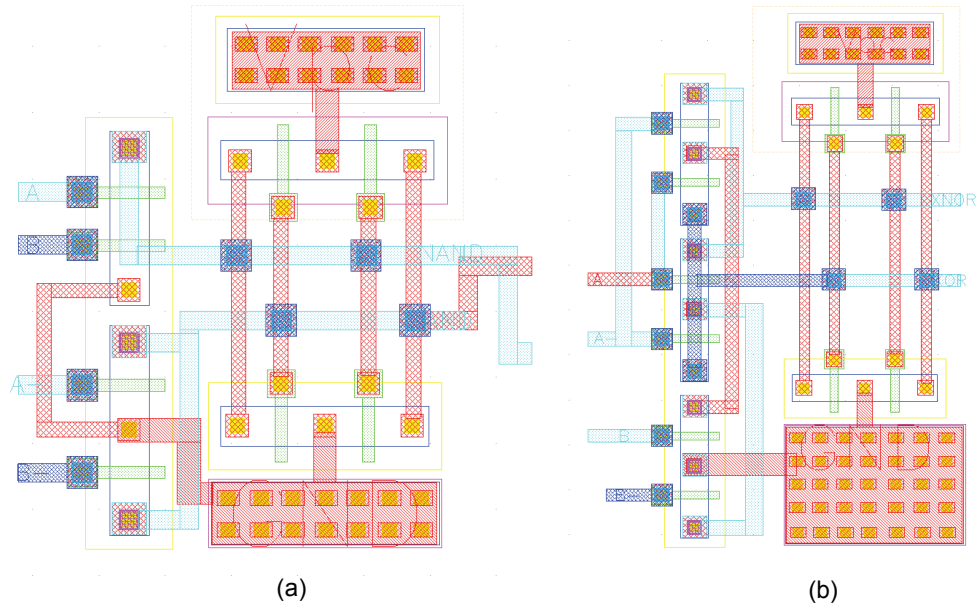


Figure 6.6: Layout of the 2N-2N2P circuit; (a) NAND/AND, (b) XNOR/XOR.

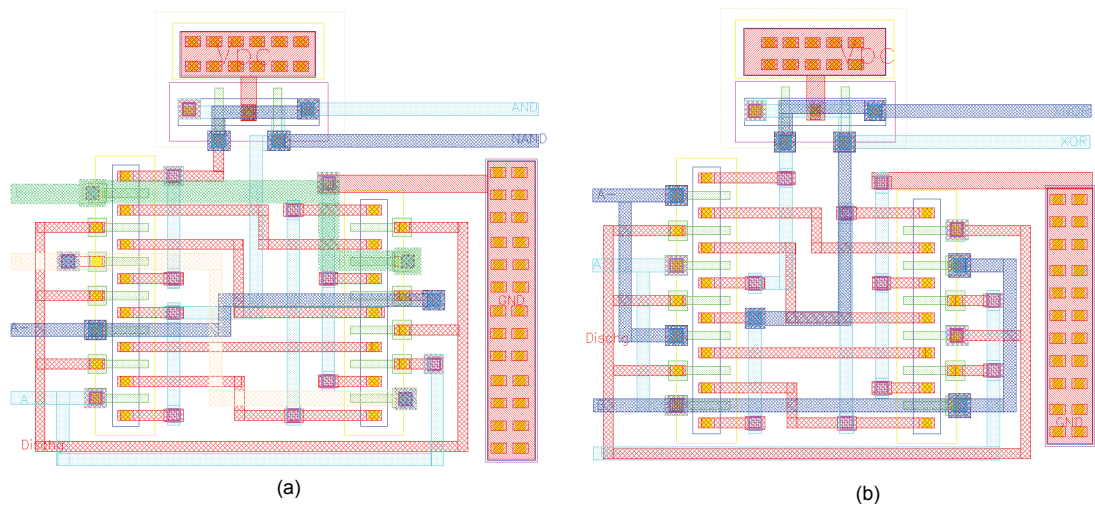


Figure 6.7: Layout of the SyAL circuit; (a) NAND/AND, (b) XNOR/XOR.

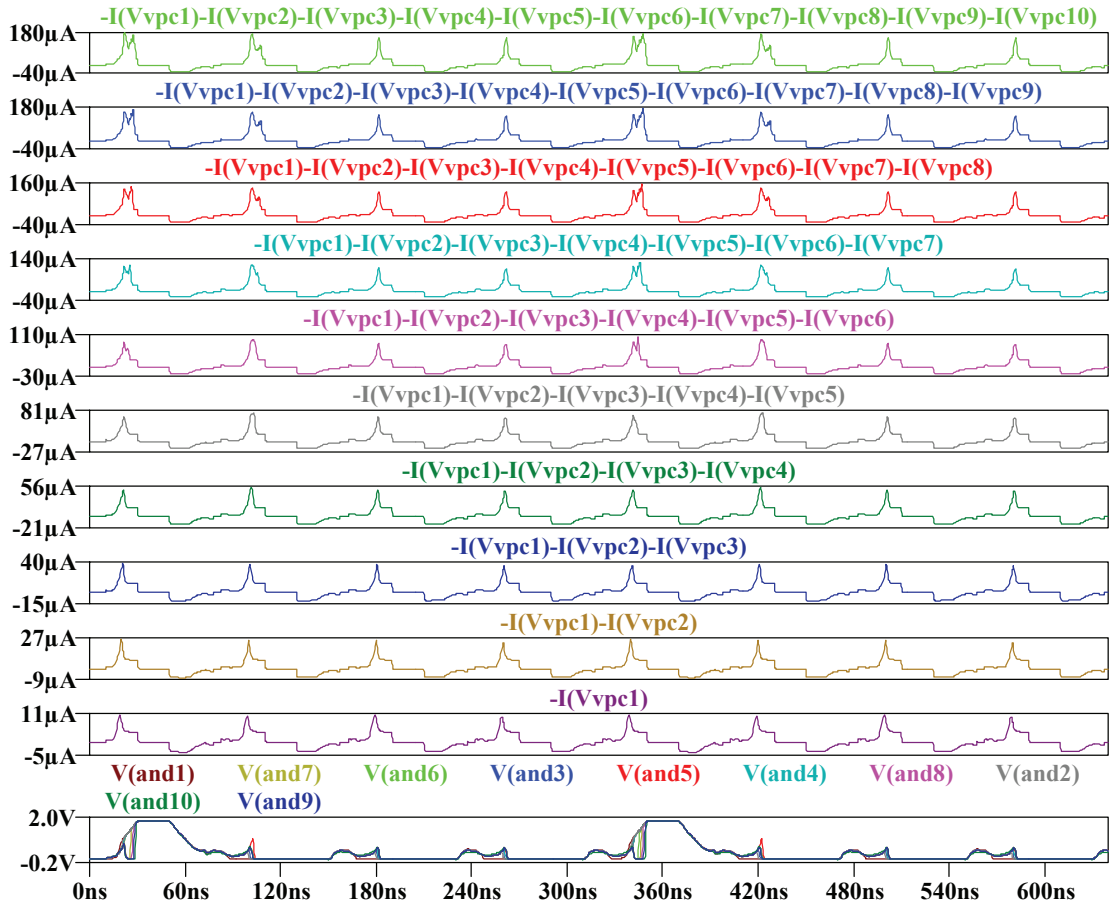


Figure 6.9: Supply current traces of the CSSAL ver.2 : Post-layout simulation result of the 10-stages NAND/AND chain using 10-Vpc Power supplies with normal logic operation at 12.5 MHz.

of AND chain, however, these stage numbers are decreased due to the effects of parasitic RC values of the designed layout.

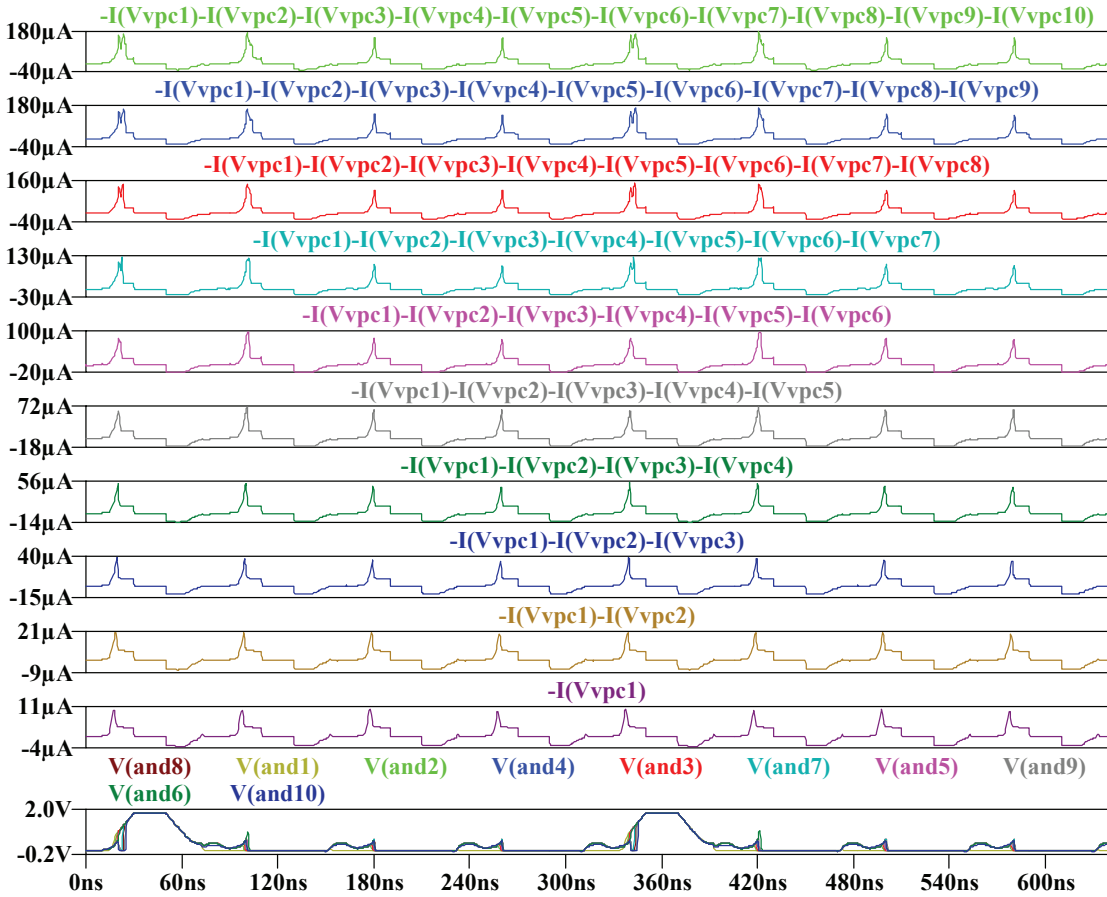


Figure 6.10: Supply current traces of the CSSAL ver.4 : Post-layout simulation result of the 10-stages NAND/AND chain using 10-Vpc Power supplies with normal logic operation at 12.5 MHz.

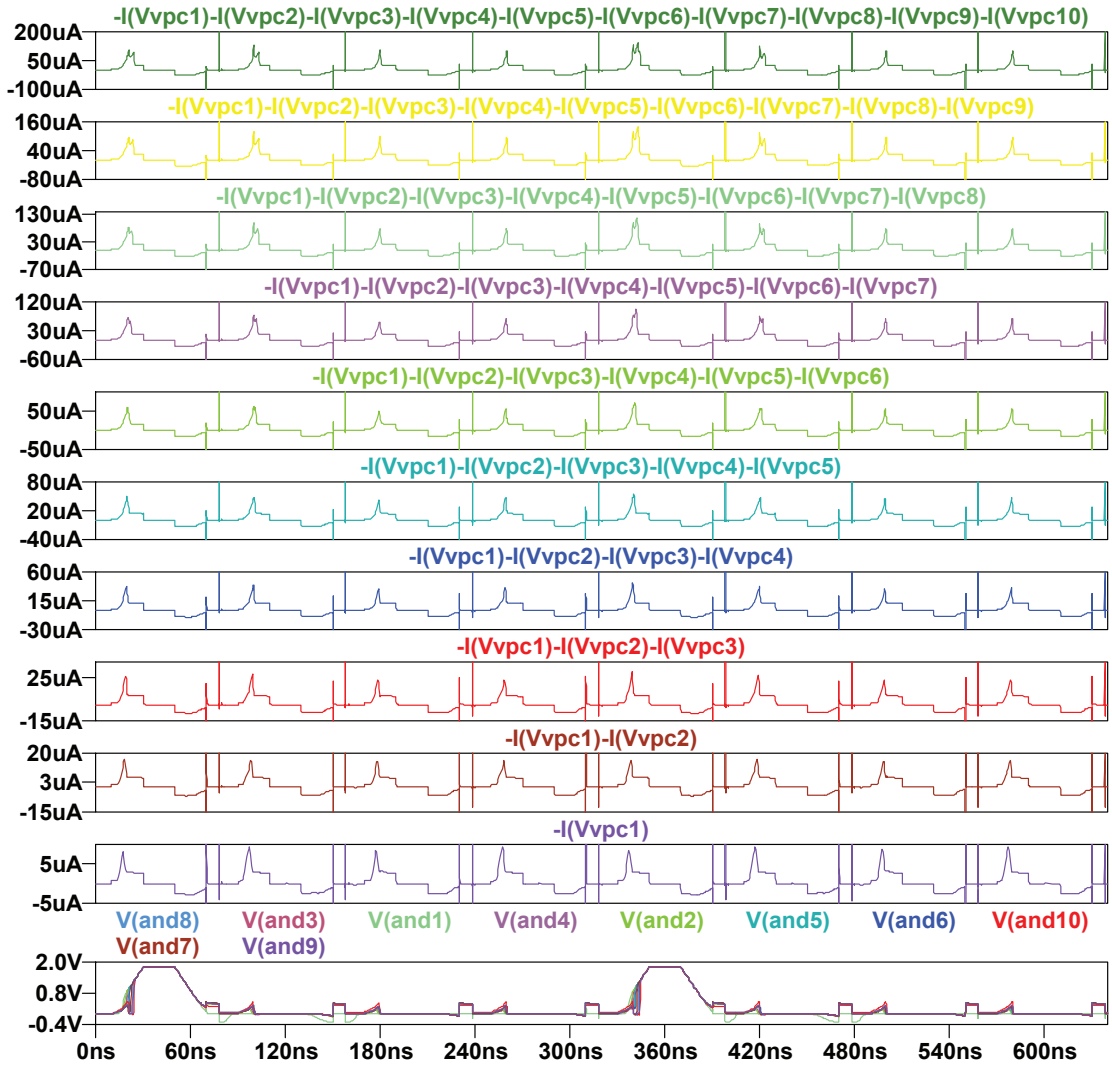


Figure 6.11: Supply current traces of the SyAL : Post-layout simulation result of the 10-stages NAND/AND chain using 10-Vpc Power supplies with normal logic operation at 12.5 MHz.

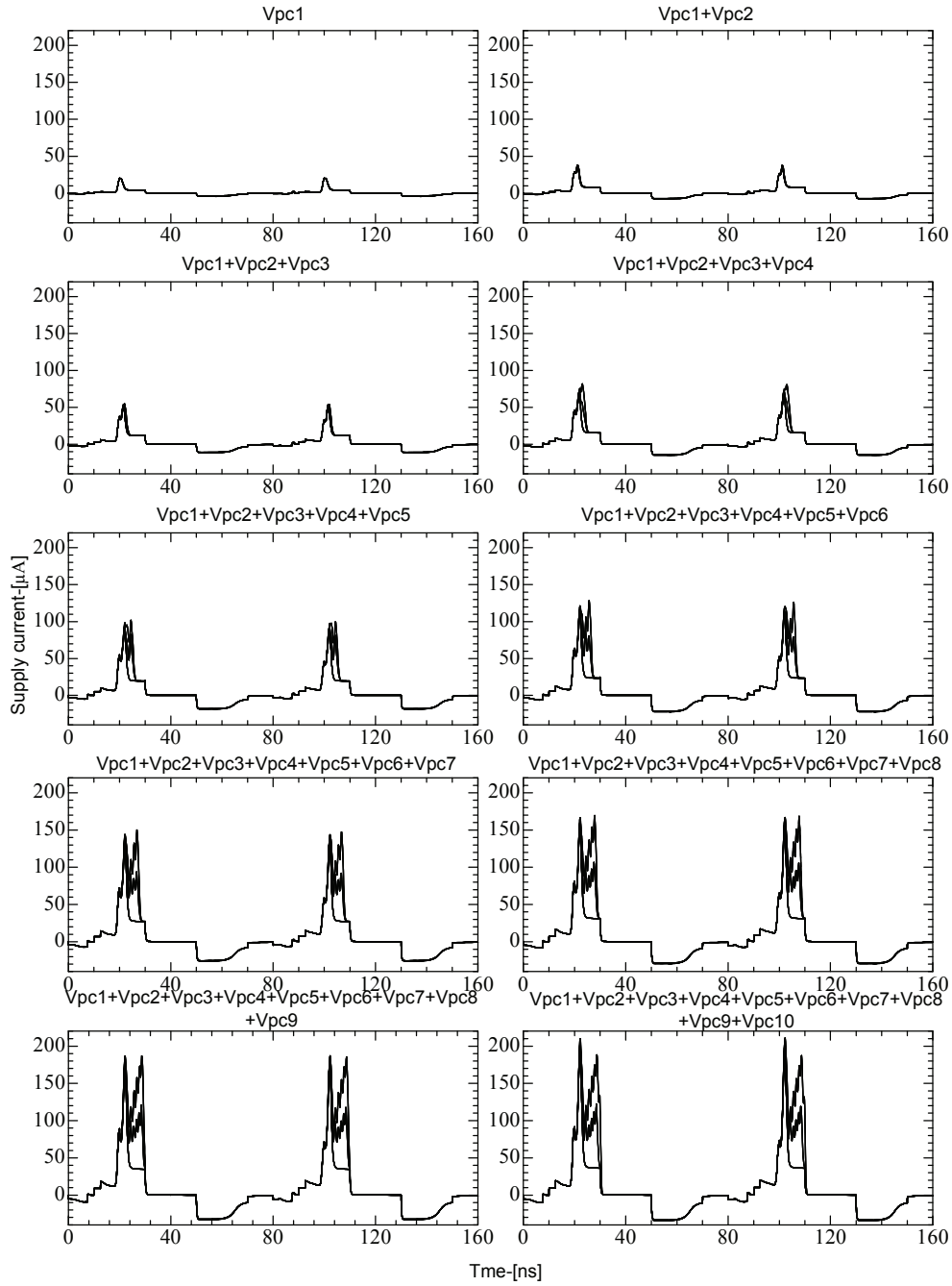


Figure 6.12: CSSAL ver.2: 16-possible supply current transition from post-layout simulation of the 10-stages NAND/AND chain using 10-Vpc Power supplies. The results indicate that the possible stage of the CSSAL ver.1 is 4 or 5 stages of AND chain.

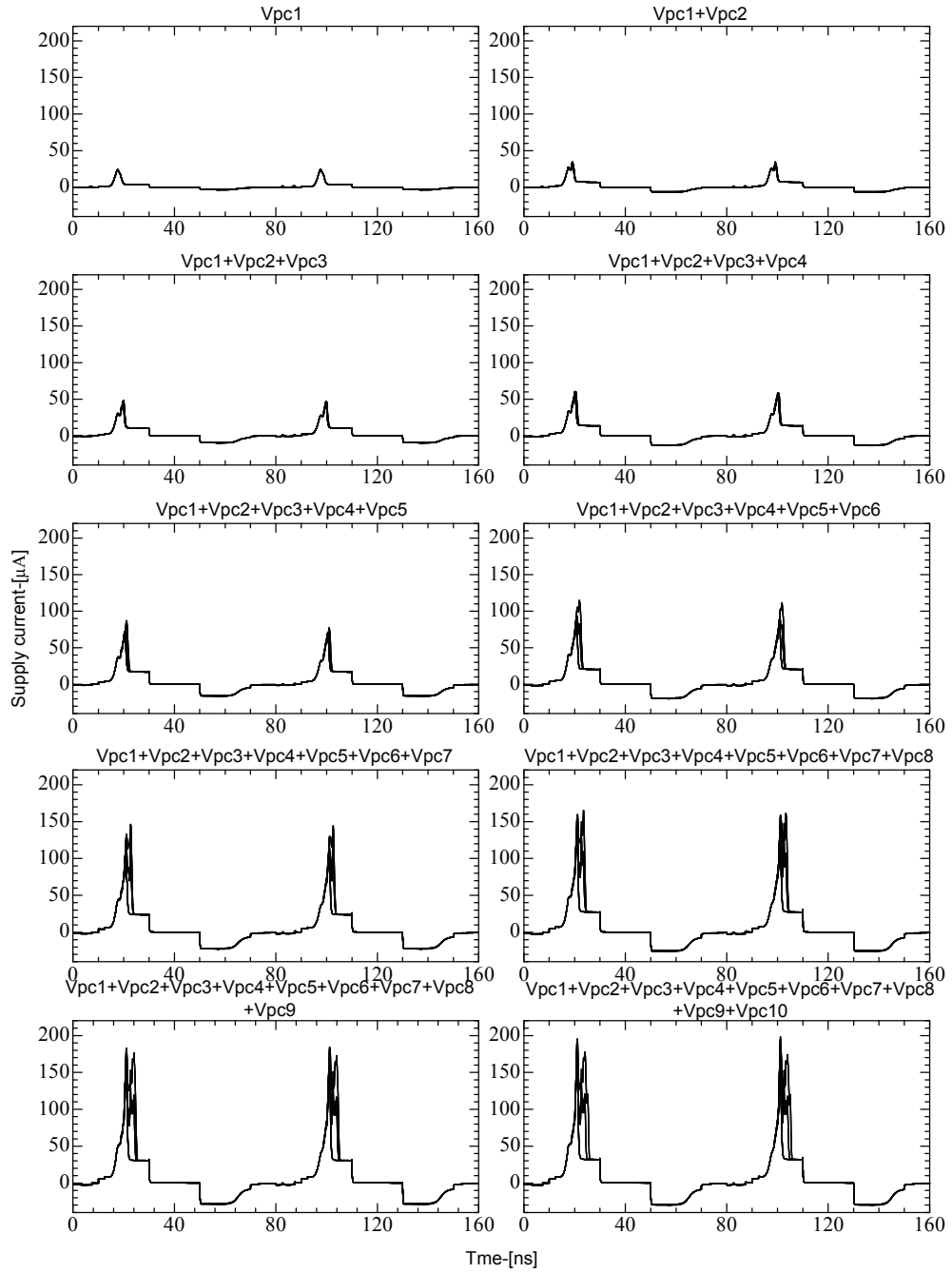


Figure 6.13: CSSAL ver.4: 16-possible supply current transition from post-layout simulation of the 10-stages NAND/AND chain using 10-Vpc Power supplies. The results indicate that the possible stage of the CSSAL ver.1 is 6 or 7 stages of AND chain.

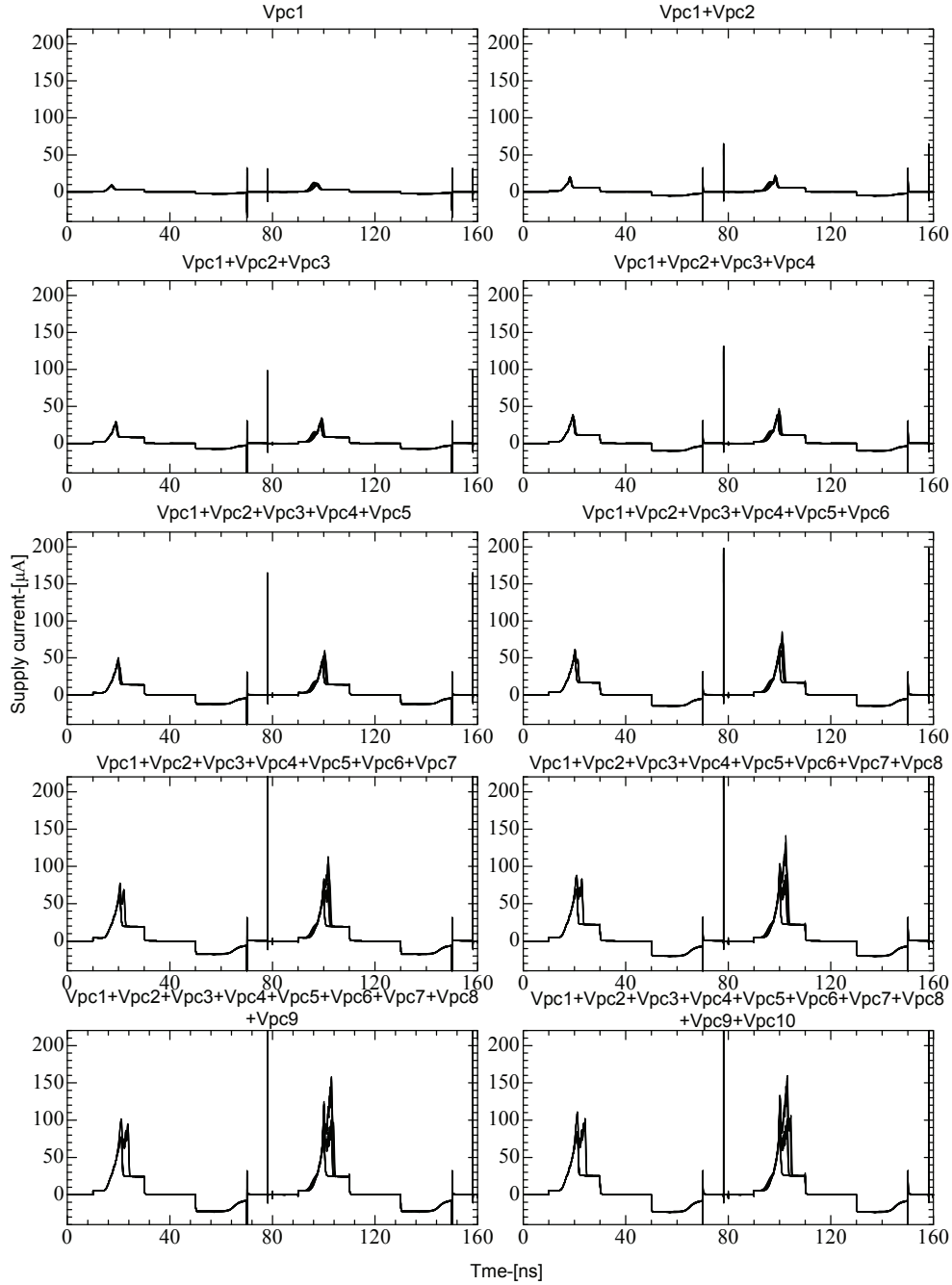


Figure 6.14: SyAL: 16-possible supply current transition from post-layout simulation of the 10-stages NAND/AND chain using 10-Vpc Power supplies. The results indicate that the possible stage of the SyAL ver.1 is 6 or 7 stages of AND chain.

6.2.2 Multiplier Circuit Layout

Full-custom layout of the multiplier circuit are shown in Fig. 6.15. The layout of four logic circuit styles were designed with the chip area are summarized in Table 6.1. The gate numbers listed in this table show that the CSSAL is much higher compared with the others, however with effective handicraft design, the chip size is smaller than the TDPL. The symmetric cell structure is easy to be implemented than that of the asymmetric logic structure, and hence the SyAL has lowest area in this comparison table.

Post-Layout Simulation Result

The author was repetitively conducting the same tedious works and similar results of tables and graphical information have been presenting in each step (*i.e.*, pre-layout and post-layout). The motive is to carefully investigate and identify the merit of the proposed logic with identical parameters are utilized. And therefore, the figure of merit of NED and NSD are employed over and over in this dissertation works. In this part, the circuit netlist with parasitic RC inclusion are simulated in LTspice, where the circuit resistance of each logic styles are re-examined. Hence, by observing the result in Table 6.2, the proposed CSSAL and the conventional SyAL exhibit their ability at low frequency range (1.25MHz). Conversely, the TDPL multiplier is performing its ability at high frequency band. Moreover, the data in Table 6.2 demonstrate that the 2N-2N2P has drawback in consuming various energy for every input transition. To the best of the author knowledge, the DPA and DEMA attacks reveal the secret information by statistically analyzing the power fluctuations and the current amplitude of attacked hardware, such as smart card. In this paradigm, the proposed CSSAL is stronger to thwart DEMA attack because the peak supply current comparison in Fig. 6.16 shows that CSSAL has low and uniform peak current than the conventional SyAL, 2N-2N2P and TDPL.

Table 6.1: Comparison of gate numbers and chip area of all investigating logic styles.

Gate Counts and Chip Area				
Multiplier	CSSAL ver.2	2N-2N2P	SyAL	TDPL
Gate Count	950	450	750	750
Layout Size (μm^2)	172 \times 155	150 \times 140	110 \times 158	183 \times 173

Apart from the logic ability for resistance against SCA attacks, the power reduction is also one of the design specification. The author has checked the post-layout energy dissipation, which is obviously described by the graphical information in Fig. 6.17 that the proposed CSSAL multiplier has significant energy reduction at lower frequencies, such as at the minimum energy point of the CSSAL multiplier at 0.5 MHz reduced energy about 27 times lower than that of the TDPL multiplier. On the other hand, in comparison with SyAL and 2N-2N2P, the proposed logic performs high energy efficiency at dynamic frequency ranges under 1.25 MHz. Furthermore, if we recall Fig. 5.29, the maximum clock speed degraded from 125 MHz to 50 MHz in post-layout result. The 2N-2N2P consumed lower energy at lower frequencies in pre-layout result, but the opposite result occurred after the post-layout simulation. The author has noticed that the layout wiring violations may affect the design performance.

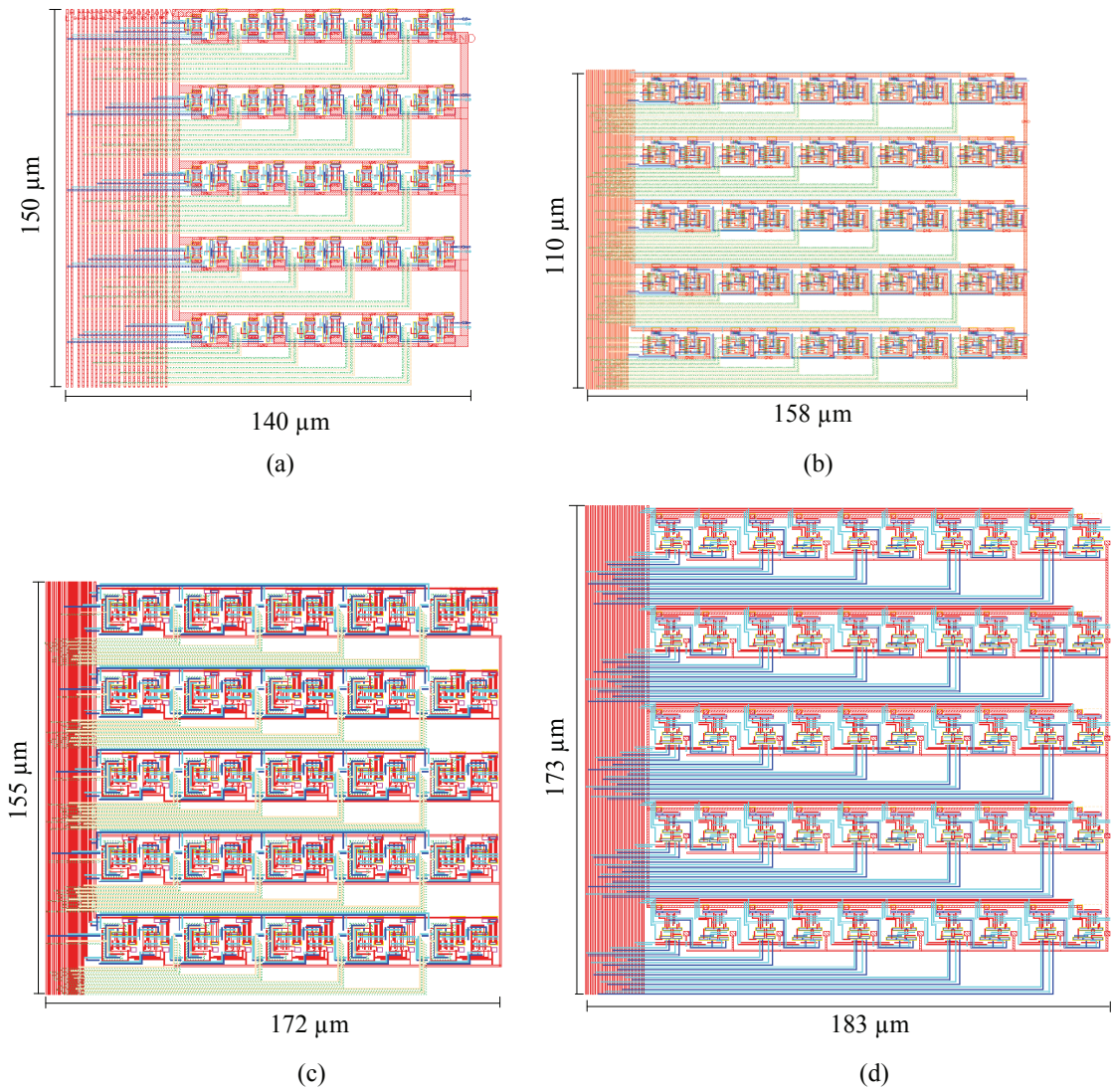


Figure 6.15: Layout of the multiplier circuits; (a) 2N-2N2P multiplier, (b) SyAL multiplier (c) CSSAL multiplier and (d) TDPL multiplier.

Table 6.2: Simulation and calculation results of the bit-parallel cellular multiplier over $GF(2^4)$ at 1.25MHz, 12.5MHz, 50MHz power clock frequency for the CSSAL, SyAL, 2N-2N2P, and the TDPL multiplier, respectively.

Power variation of cellular multiplier over $GF(2^4)$				
1.25 MHz				
	Proposed	SyAL	2N-2N2P	TDPL
E_{min} [pJ]	0.4	0.39	0.045	7.04
E_{max} [pJ]	0.46	0.46	0.81	35.44
\overline{E} [pJ]	0.43	0.43	0.47	14.31
σ_E [pJ]	0.015	0.02	0.23	8.4
NED [%]	12.04	14.17	94.45	80.13
NSD [%]	3.49	4.69	49.08	58.71
12.5 MHz				
	Proposed	SyAL	2N-2N2P	TDPL
E_{min} [pJ]	0.65	0.50	0.08	6.9
E_{max} [pJ]	1.24	0.68	0.98	9.61
\overline{E} [pJ]	0.88	0.58	0.57	7.67
σ_E [pJ]	0.21	0.04	0.79	0.27
NED [%]	47.33	26.38	92.13	28.18
NSD [%]	24.48	6.93	48.05	10.27
50 MHz				
	Proposed	SyAL	2N-2N2P	TDPL
E_{min} [pJ]	0.94	0.27	0.14	6.97
E_{max} [pJ]	2.63	1.77	1.91	7.44
\overline{E} [pJ]	1.04	1.52	1.09	7.04
σ_E [pJ]	0.58	0.36	0.55	0.18
NED [%]	64.19	84.51	92.57	8.7
NSD [%]	38.06	34.60	50.45	2.49

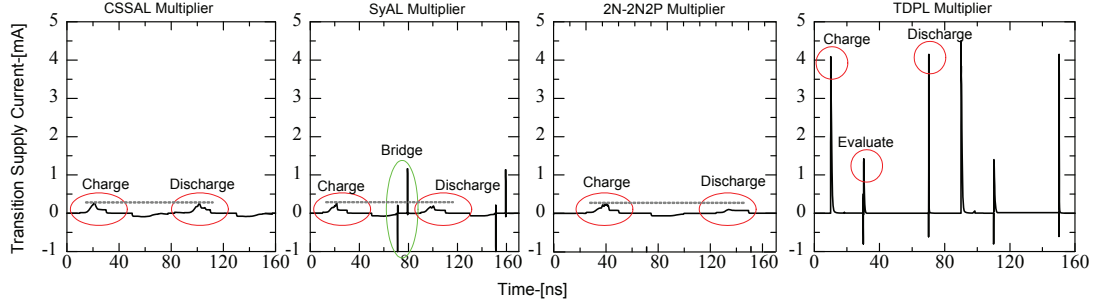


Figure 6.16: The comparison of peak supply current of the CSSAL, SyAL, 2N-2N2P and the TDPL multiplier when complementary dual output states are shifted for charging and discharging process. The proposed CSSAL has low and same peak current during charging and discharging processes, while the SyAL, 2N-2N2P have low and different peak current, and the TDPL has very high peak current traces.

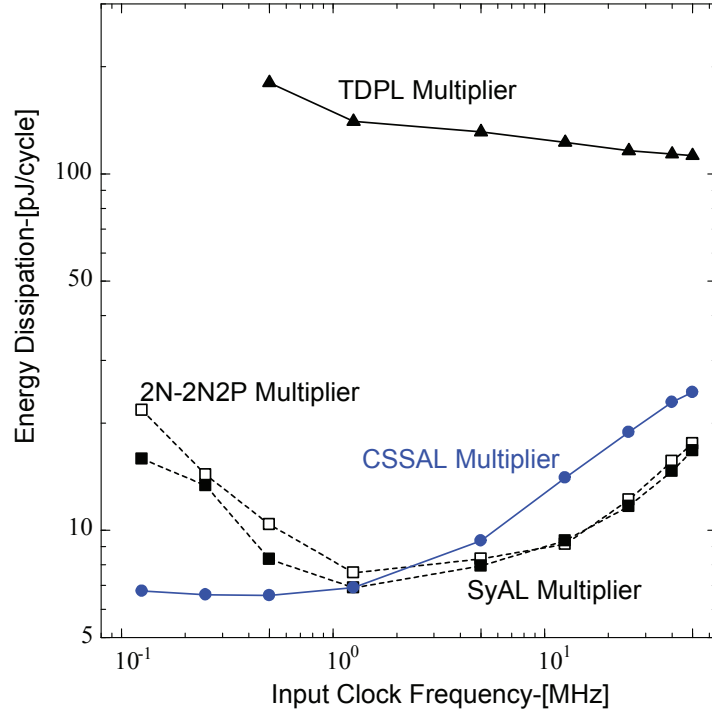


Figure 6.17: Post-layout energy dissipation comparison of the bit-parallel cellular multiplier over $GF(2^4)$ in respect to the different input clock frequency.

6.2.3 CSSAL S-box Circuit Layout

The layout of the CSSAL S-box circuit is shown in Fig. 6.18. The fundamental logic of Inverter/Buffer, AND/NAND, and XOR/XNOR circuit layout in Figs. 6.2.1, 6.2.1 were used in this S-box layout. The layout area can be seen in Fig. 6.18.

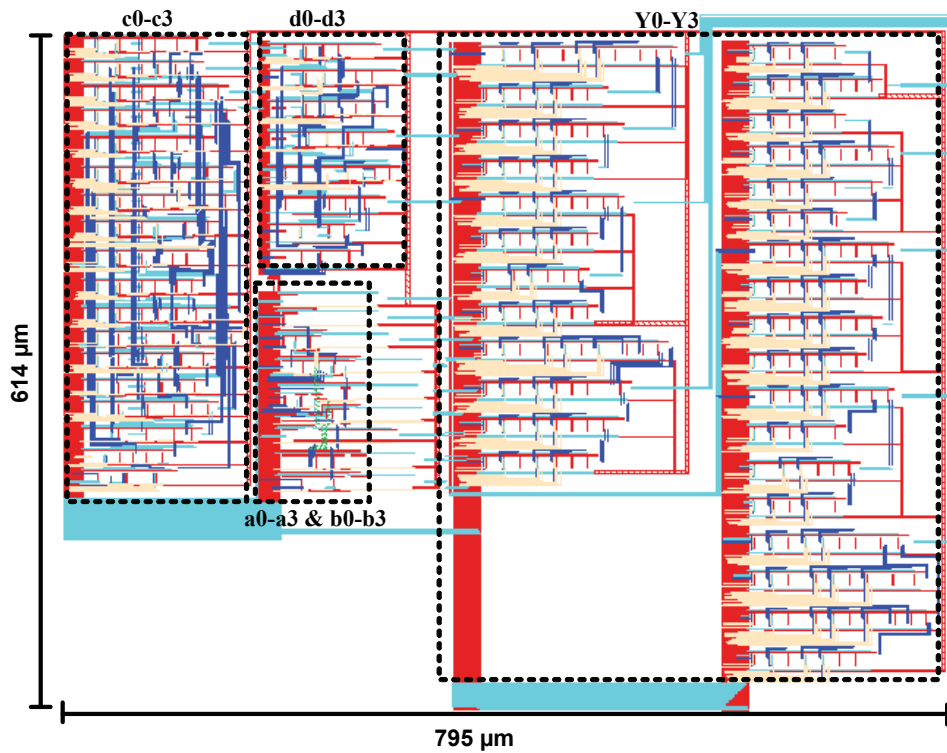


Figure 6.18: Layout of the CSSAL 8-bit AES S-box circuit.

6.3 Fabricated LSI Measurement

6.3.1 Measurement Equipment

A multiplier LSI was fabricated through the chip fabrication program of VLSI Design and Education Center (VDEC), the University of Tokyo in collaboration with the Rohm Corporation. The top level layout frame and the captured microphotograph of LSI chips are depicted in Figs. 6.19 and 6.20 for both multiplier and S-box circuits, respectively. From these figures, we can easily observe how the place and route were conducted. The microphotograph in Figs. 6.19(b) and 6.20(b) were captured using Leica light microscope in the division of instrumental analysis of Gifu university, as shown in Fig. 6.21. Figure 6.22 shows the LSI chip image from bottom and top view, and the same figure is also showing an unit under test. The measurement equipment are depicted in Fig. 6.23, such as NF function generator (FG) with 30 MHz of internal clock speed. The FG are used to generate the power clock signals for LSI measurement. The measured outputs signals are displayed using LeCroy digital oscilloscope with 2.5 GS/s (Giga sample per second), and its internal clock speed is 400 MHz. Kenwood DC power supply is used to generated the I/O buffer circuit. Power measurement was conducted using Keythley picoamperemeter. In addition, the AVO meter was also utilized to check the connection between the wires and the DC power supply accuracy test.

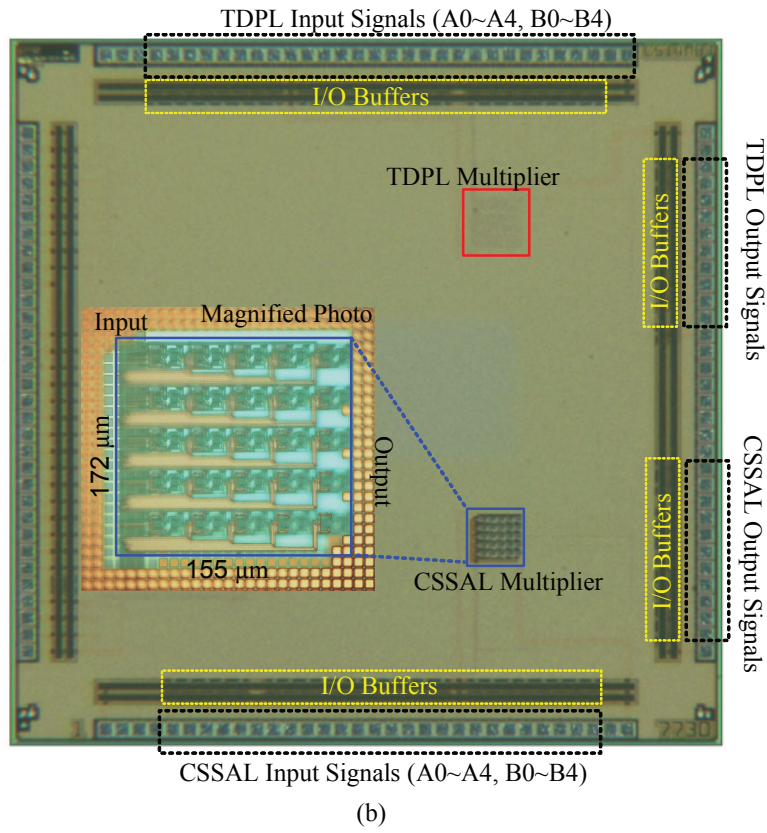
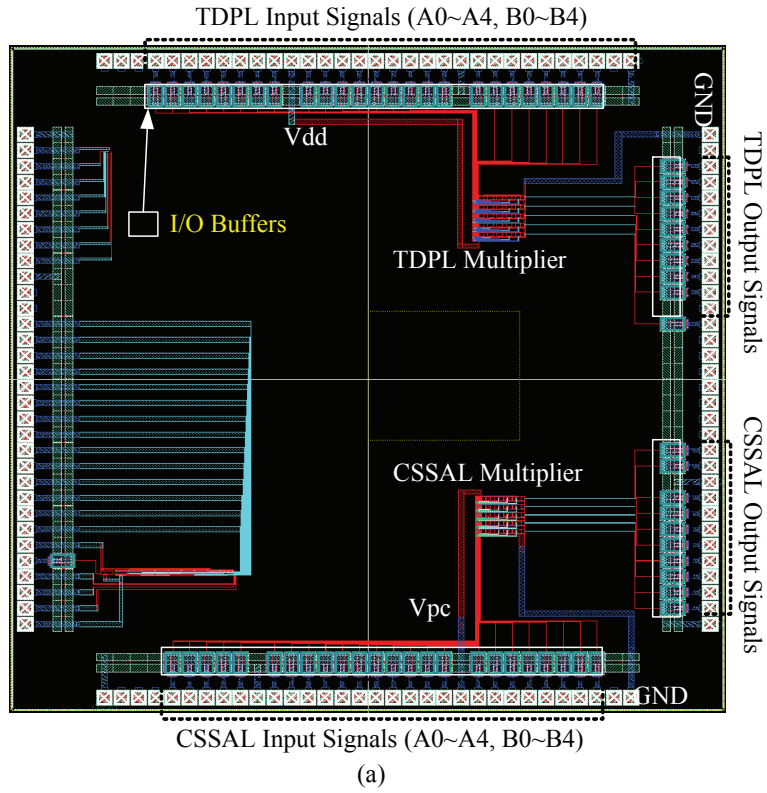


Figure 6.19: Multiplier over $GF(2^4)$: (a) LSI frame from the layout view, (b) Fabricated LSI's photomicrograph.

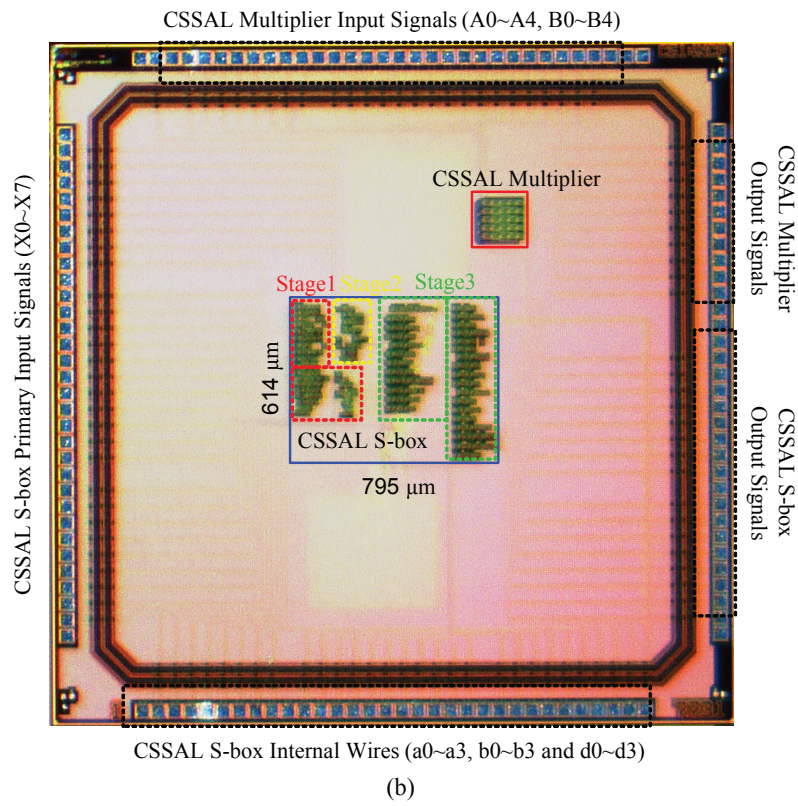
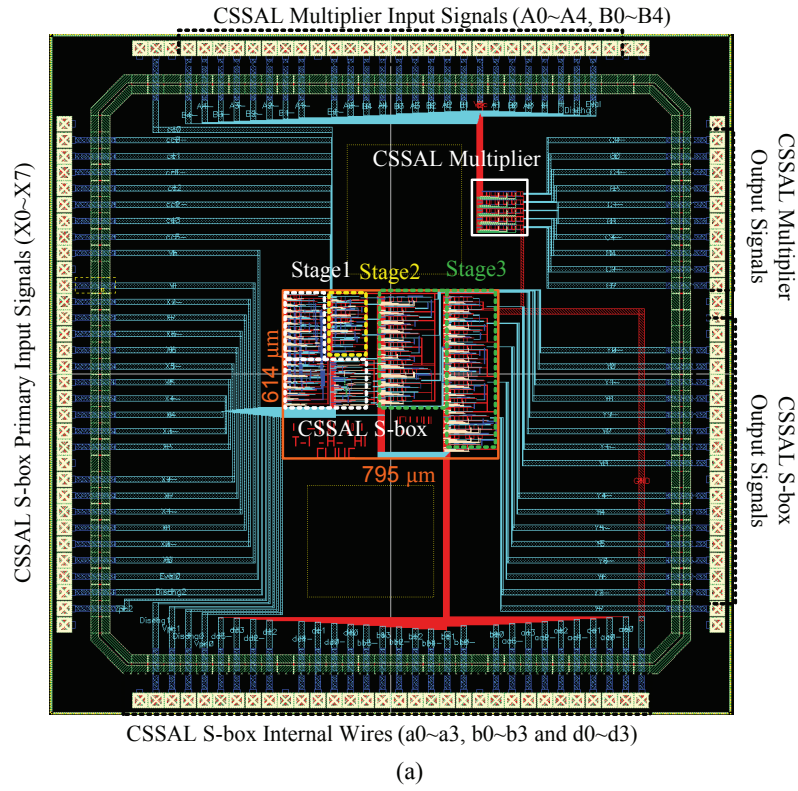


Figure 6.20: 8-bit AES S-box: (a) LSI frame from the layout view, (b) Fabricated LSIs photomicrograph.

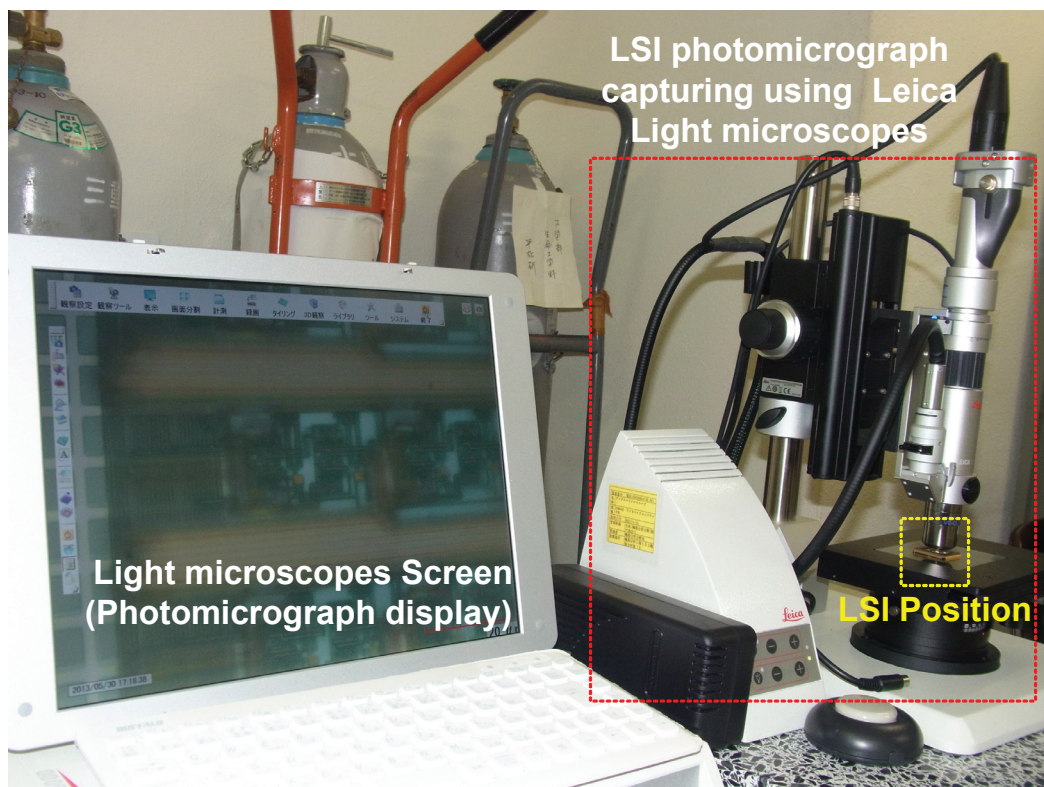
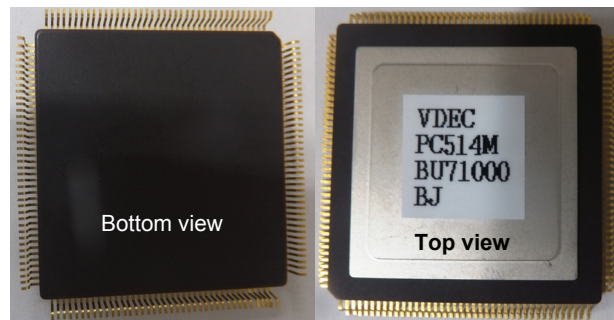
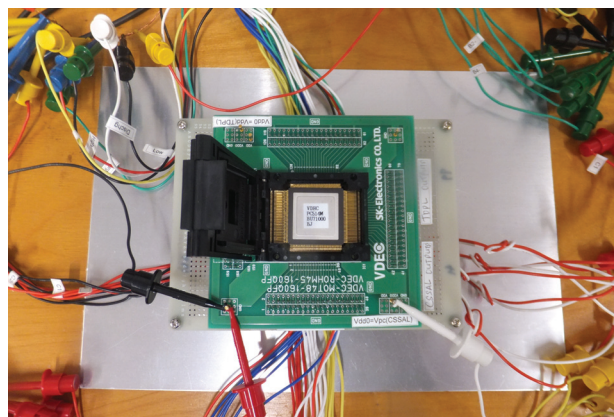


Figure 6.21: Capturing the LSI photomicrograph using Leica Light microscopes in the division of instrumental analysis, Gifu University.



(a) LSI chip image



(b) Unit under test

Figure 6.22: (a) LSI image and (b) Unit under test.

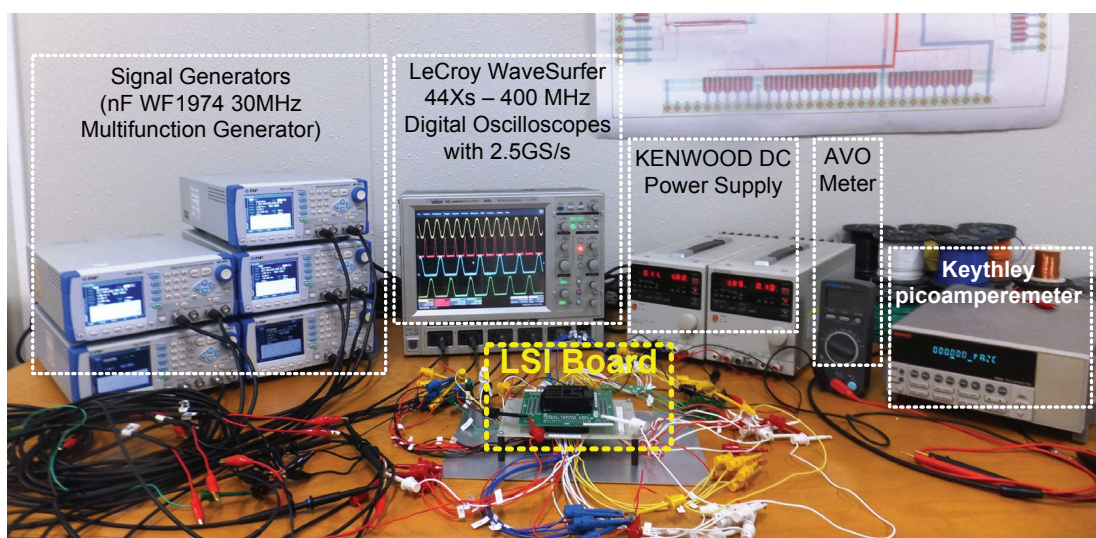


Figure 6.23: Measurement photo including the used equipment.

6.3.2 Multiplier Measurement Results

The LSI chip measurement diagram in this work is depicted in Fig. 6.24. The measurement results of the input and output signals at 1.25 MHz power clock frequency are shown in Fig. 6.25 and Fig. 6.26 for TDPL and CSSAL multiplier, respectively. In this measurement technique, the connections of the input signals were combined, such as $In1 = \{A0, A1, A2, A3, A4\}$, $In1 = \{\bar{A}0, \bar{A}1, \bar{A}2, \bar{A}3, \bar{A}4\}$, $In2 = \{B0, B1, B2, B3, B4\} = 1$ (constant V_{dd}) and $\{\bar{B}0, \bar{B}1, \bar{B}2, \bar{B}3, \bar{B}4\} = 0$ (connected to ground); accordingly, the output voltage of a cellular multiplier $\{C0, C1, C2, C3, C4\}$ are correctly produced as $In1 \times In2 = Out = 1$, if all input signals are at logical high level. In this measurement, the supply current traces were plotted by inserting a small shunt resistance $R_s = 1 \Omega$ between V_{ss} pin of the multiplier chip and the true source (ground), as shown in Fig. 6.24. Then, the peak differences of the supply current in four representation input transitions as depicted in Fig. 6.27. The current traces in this figure were plotted as follows: trace (a) obtained from input condition of $In1$ equal $0 \rightarrow 1$ and $1 \rightarrow 0$ transitions while $In2$ stable at $0 \rightarrow 0$ transition; trace (b) was plotted at conditions of $In1$ equal $0 \rightarrow 1$ and $1 \rightarrow 0$ transition while $In2$ stable at $1 \rightarrow 1$ transitions; And in opposite conditions, the trace (c) and (d) are obtained. These measurement traces show that current traces of the CSSAL in Fig. 6.27(a) performs similar and uniform peak current trances regardless of what is the input pattern, and hence it is resistive to side-channel analysis attacks. On the other hand, the TDPL in Fig. 6.27(b) exhibits different peak current traces.

Furthermore, the CSSAL LSI chip was also tested using dual-input complementary signals, which perfectly work like NAND/AND function as depicted in Fig. 6.28.

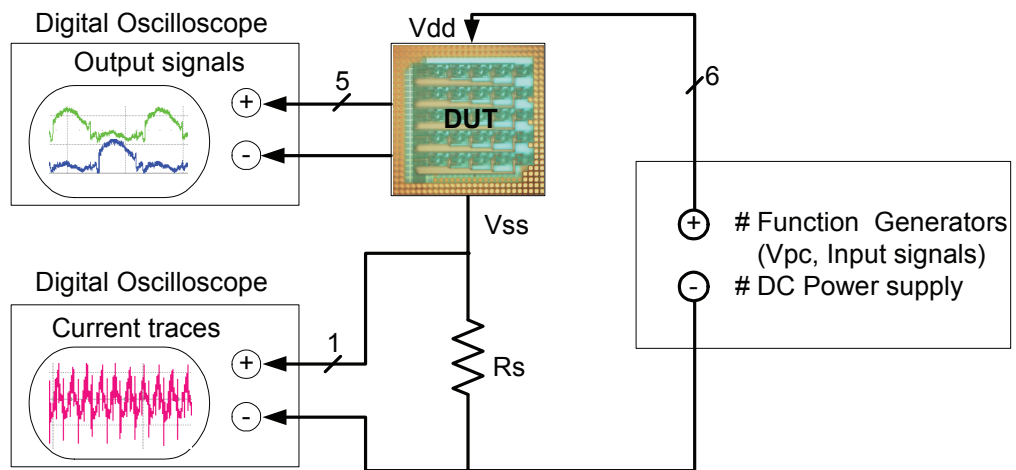


Figure 6.24: Measurement diagram.

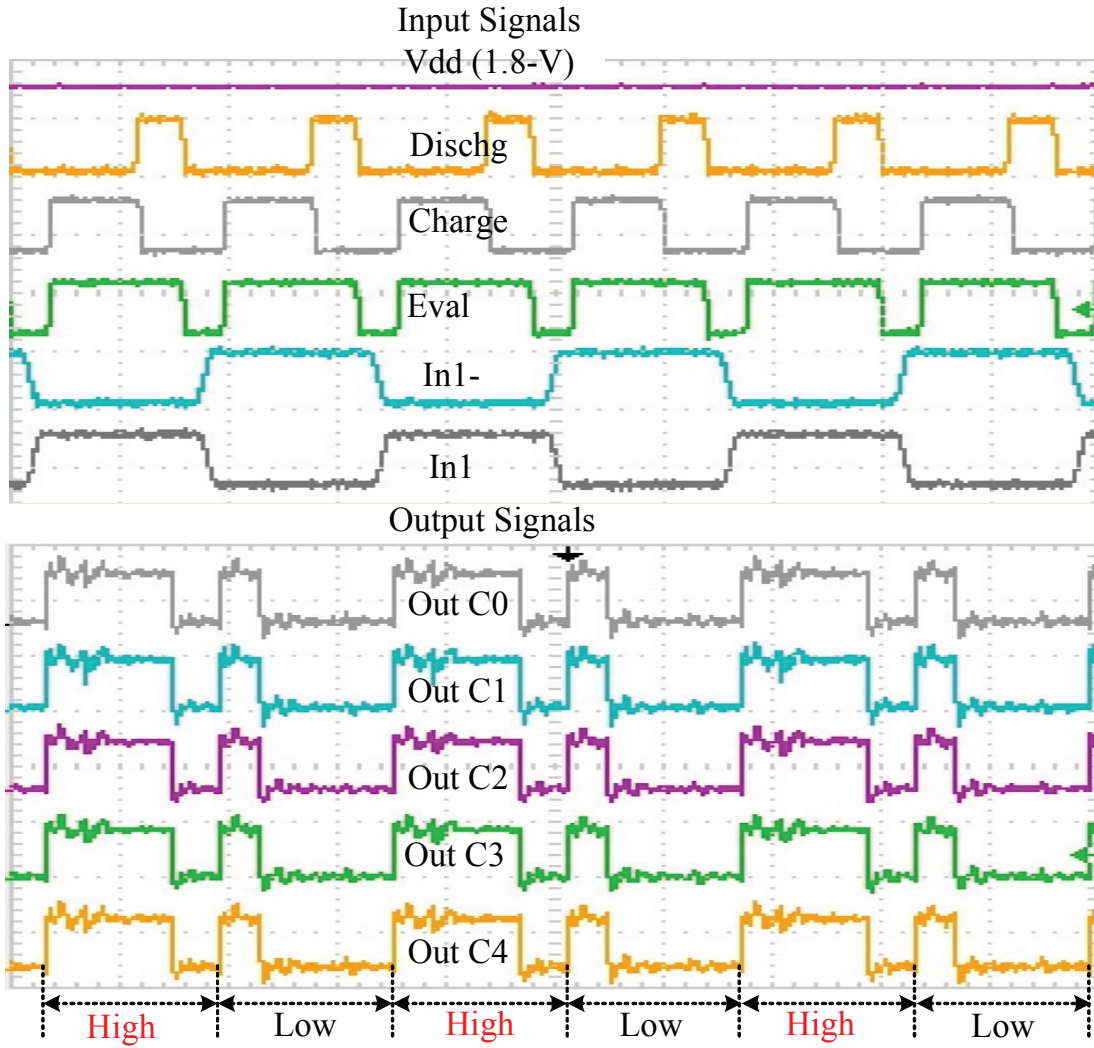


Figure 6.25: Input-output measurement signals of the TDPL bit-parallel cellular multiplier over $GF(2^4)$ at 1.25 MHz power clock frequency. Vertical scale: 2 V/div. Horizontal scale: 1 μ s/div.

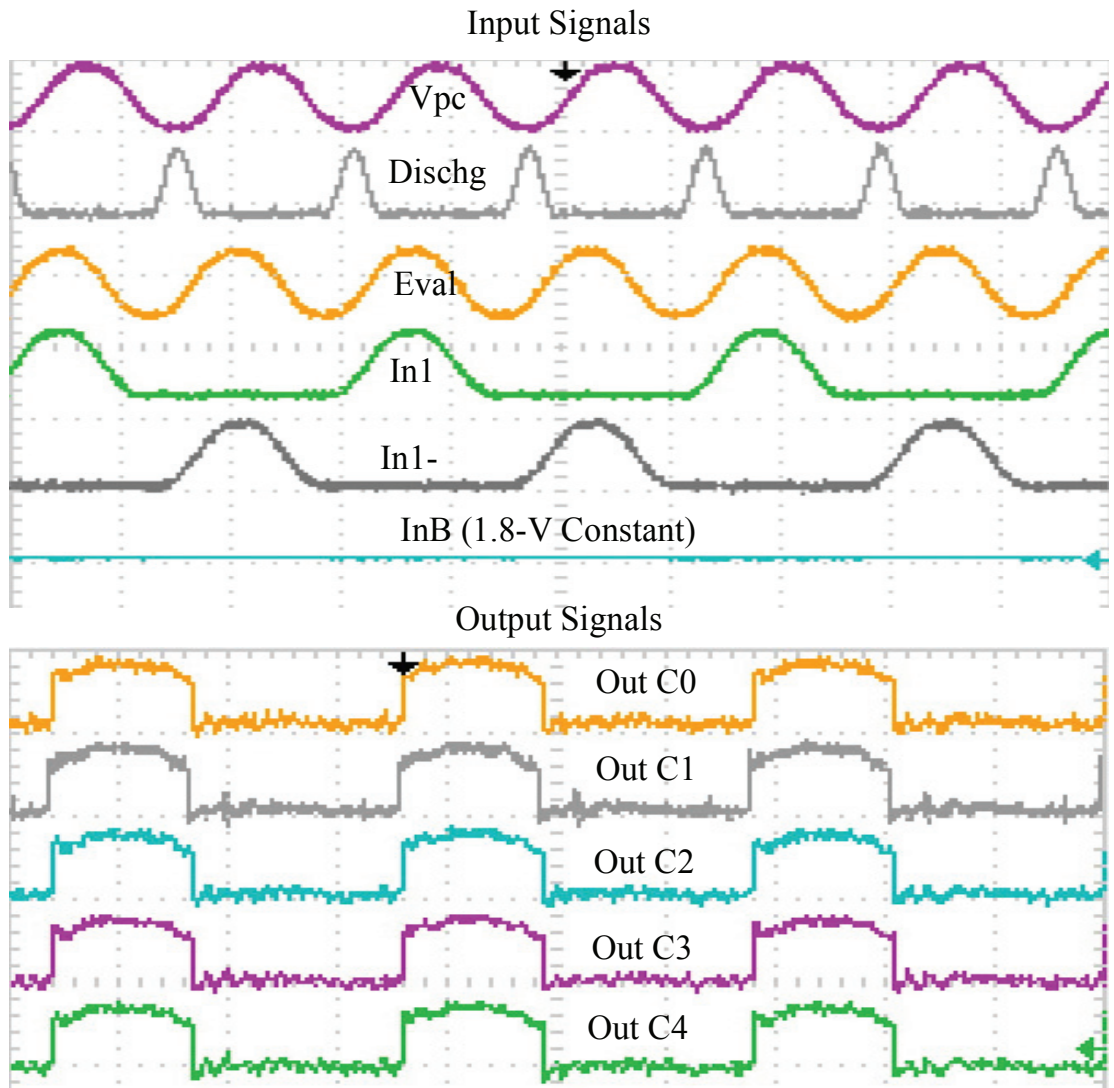
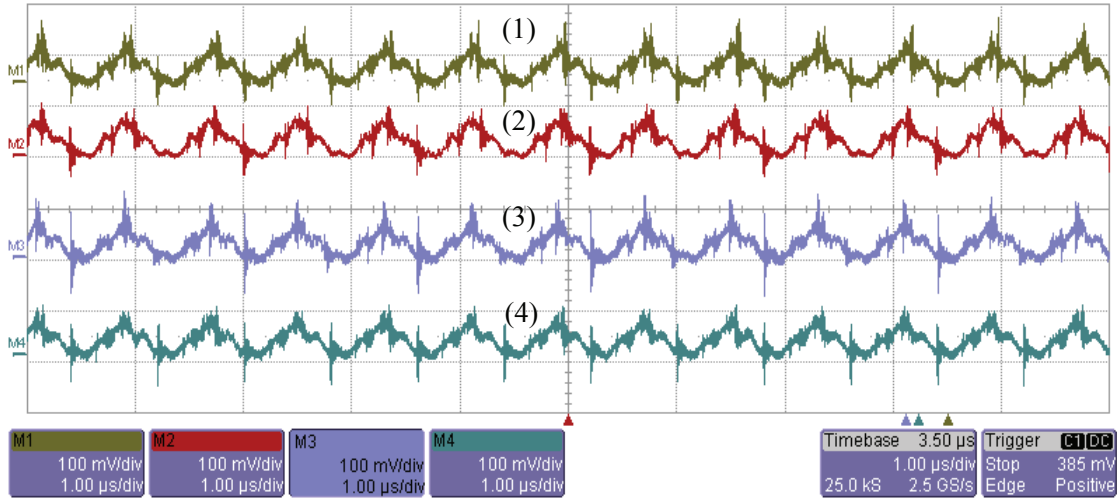
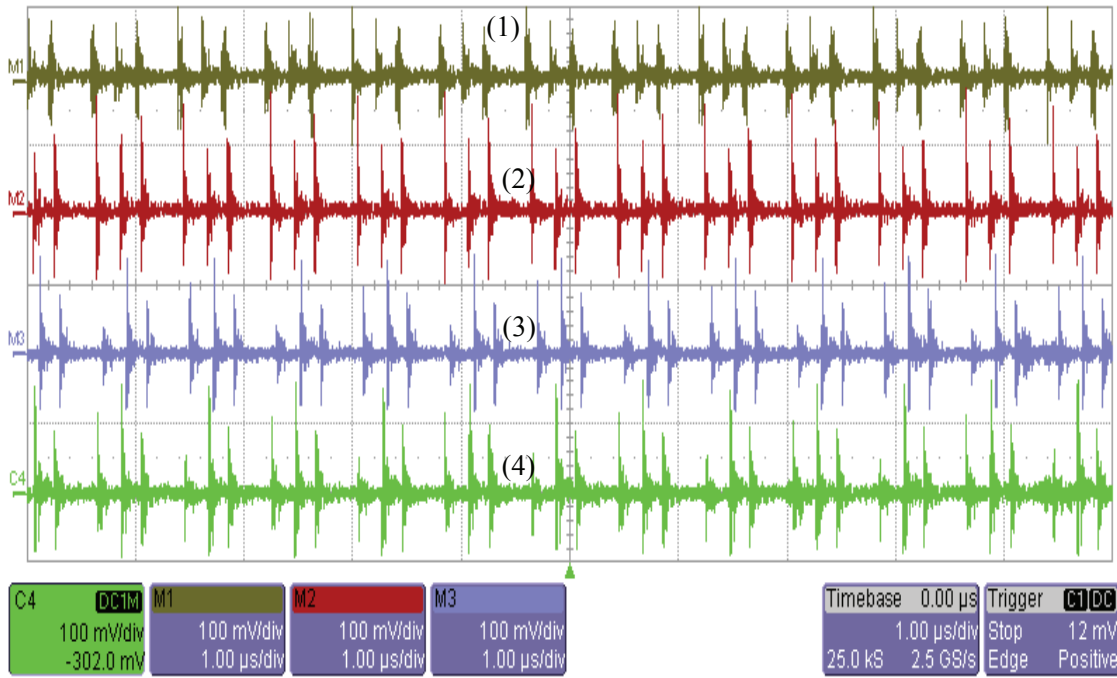


Figure 6.26: Input-output measurement signals of the CSSAL bit-parallel cellular multiplier over $GF(2^4)$ at 1.25 MHz power clock frequency. Vertical scale: 2 V/div. Horizontal scale: 1 μs /div.



(a)



(b)

Figure 6.27: Supply current traces of (a) CSSAL multiplier, (b) TDPL multiplier. Measurement is conducted as: (1) Transition of $In1 = 0 \rightarrow 1$, $1 \rightarrow 0$ when $In2 = 0 \rightarrow 0$; (2) Transition of $In1 = 0 \rightarrow 1$, $1 \rightarrow 0$ when $In2 = 1 \rightarrow 1$; (3) Transition of $In2 = 0 \rightarrow 1$, $1 \rightarrow 0$ when $In1 = 0 \rightarrow 0$; (4) Transition of $In2 = 0 \rightarrow 1$, $1 \rightarrow 0$ when $In1 = 1 \rightarrow 1$. Vertical scale: 100 mV/div. Horizontal scale: 1 μs/div.

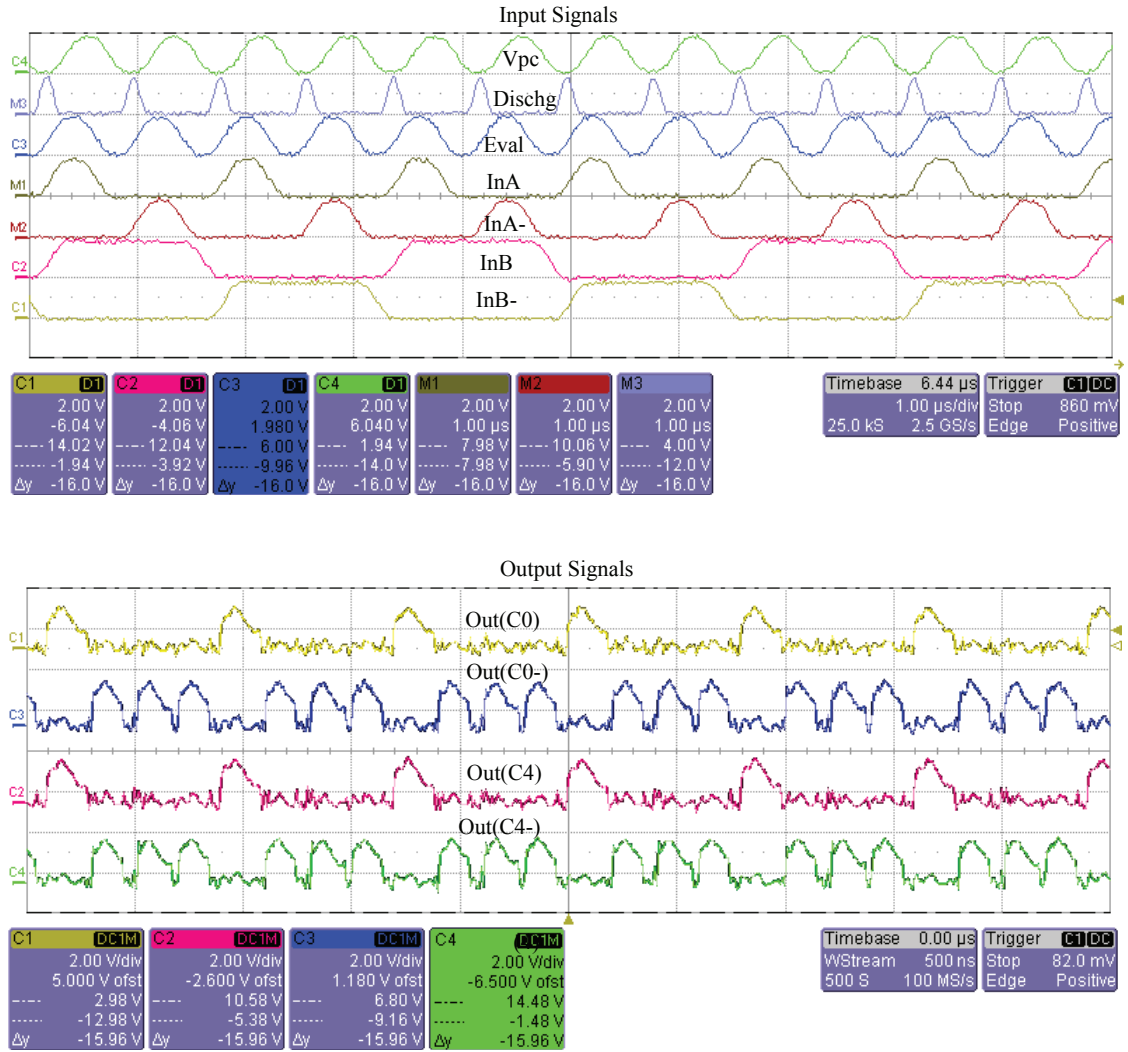


Figure 6.28: Dual-inputs (InA , InB) of the CSSAL bit-parallel cellular multiplier over $GF(2^4)$ at 1.25 MHz power clock frequency. Output signals show the correct AND/NAND logic function.

6.3.3 CSSAL S-box Measurement Results

The measurement results of the CSSAL 8-bit AES S-box are depicted in Figs. 6.29–6.35. Measurement diagram is similar to multiplier one in Fig. 6.24. Input signals for CSSAL S-box LSI measurement are shown in Fig. 6.29. In this measurement technique, the connection of the input signals were combined, such as $In1 = \{X0, X2, X4\}$, $In\bar{1} = \{\bar{X}0, \bar{X}2, \bar{X}4\}$, $In2 = \{X1, X3, X5\}$, $In\bar{2} = \{\bar{X}1, \bar{X}3, \bar{X}5\}$, $In3 = \{X6, X7\} = 1$ (Constant Vdd), and $In\bar{3} = \{\bar{X}6, \bar{X}7\} = 0$ (connected to ground). The measurement results presented in this section were measured at power clock frequency set to 125 kHz. The measurement was also tested at 0.5 MHz and above, but results were not correct as the simulation results. The pins of internal wires of $a0 - a3$, $b0 - b3$, $c0 - c3$, and $d0 - d3$ were also provided during the layout design; and hence, the author also has measured each wire as shown in Figs. 6.30–6.33, respectively. The SPICE simulation results that were using same input pattern are also placed on the top of each figure, in order to proof the correctness of the measurement signals. Simulation and measurement signals of final output Y0–Y7 of the CSSAL 8-bit S-box are depicted in Fig. 6.34.

The measurement results of the supply current traces of I_{Vpc0} , I_{Vpc1} , I_{Vpc2} are plotted in Fig. 6.35. These supply current traces were measured with $R_s = 100 \Omega$ (see Fig. 6.24). The supply current traces in Fig. 6.35 perform uniform peak current traces, which could be resistive to side-channel attacks at this operating speed (125 kHz).

The CSSAL S-box LSI's power consumption were measured and calculated as: $Power(Watt) = V(I_{Vpc0} + I_{Vpc1} + I_{Vpc2})$, where V is the peak voltage value of Vpcs signals (1.8 V), I_{Vpc0} , I_{Vpc1} , and I_{Vpc2} are each power clock supply current that were measured using pico-ampere-meter. Therefore, the CSSAL S-box LSI's power consumption is 364.14 μW at 125 kHz power clock frequency.

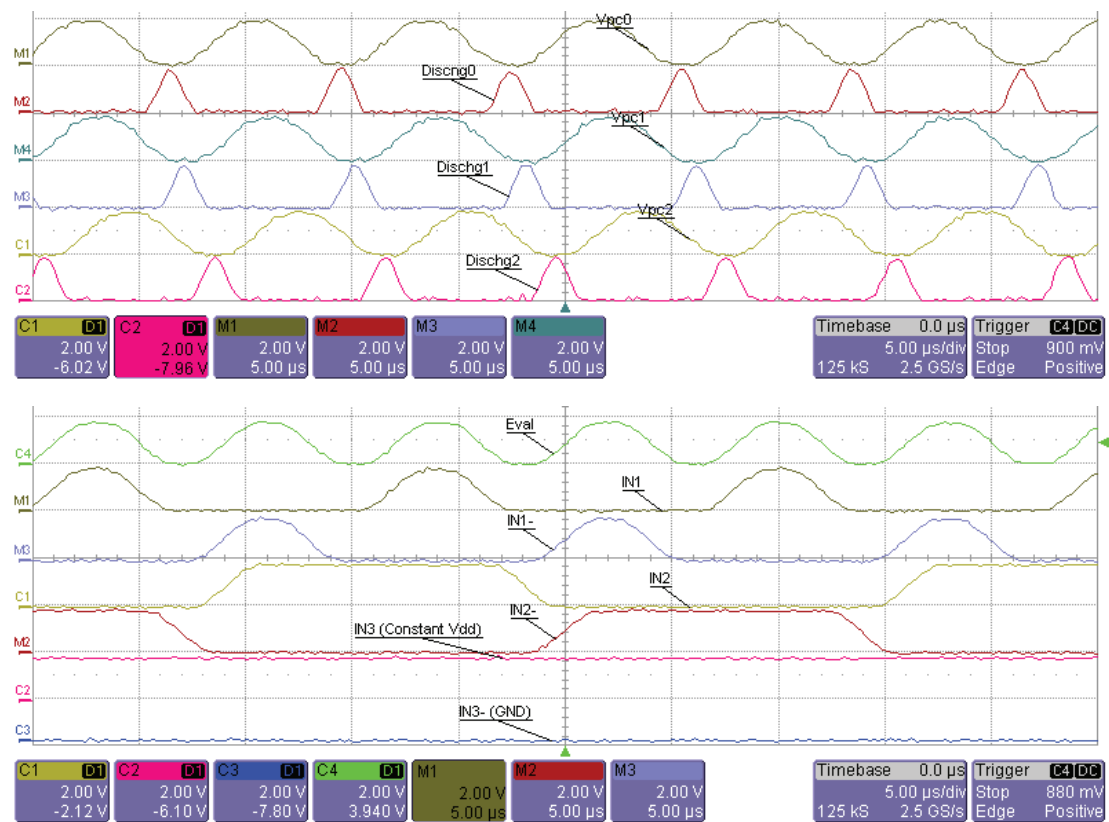


Figure 6.29: Input measurement signals of the CSSAL S-box LSI at 125 kHz power clock frequency. Vertical scale: 2 V/div. Horizontal scale: 5 μs/div.

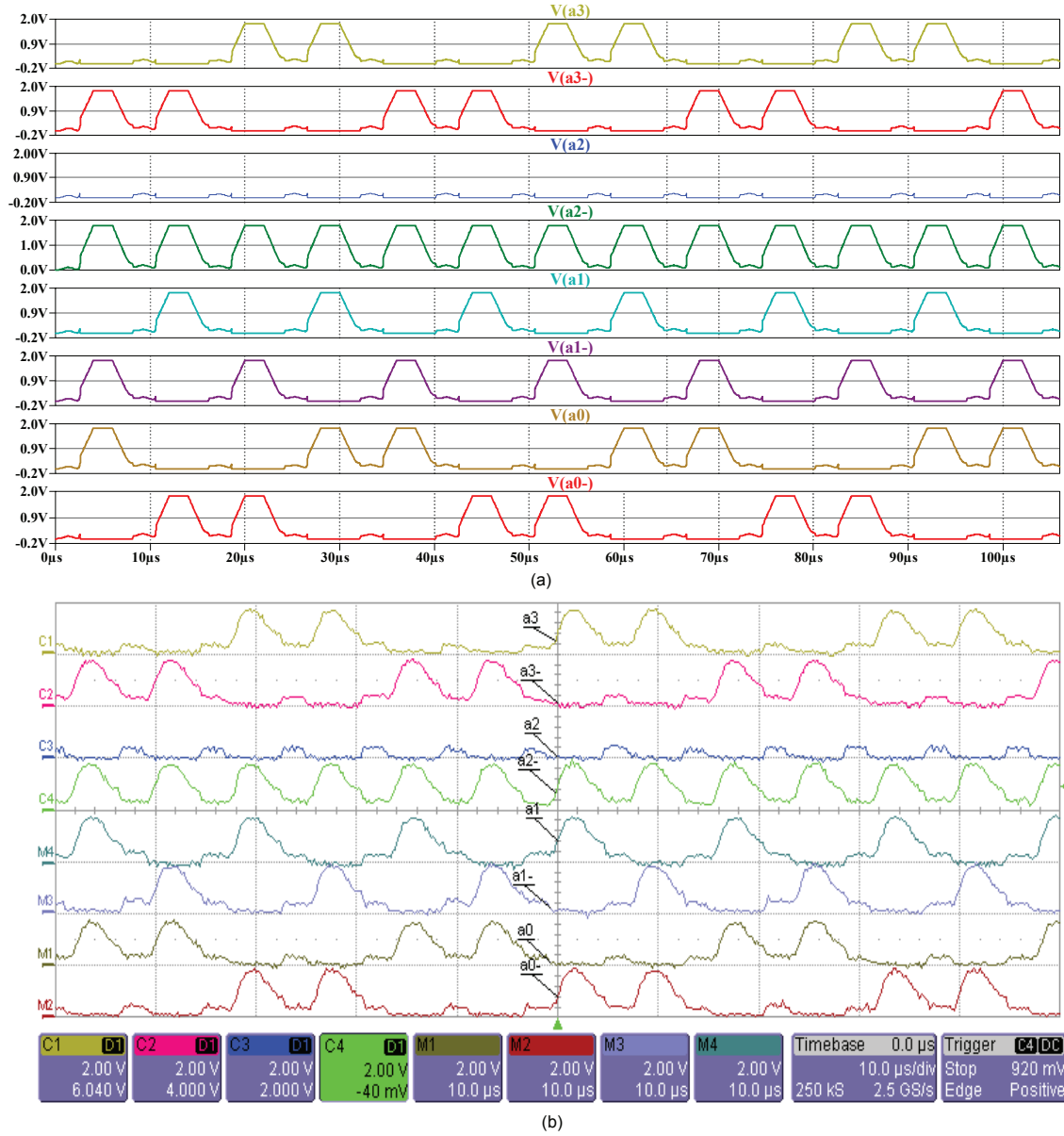


Figure 6.30: Output signals of internal wires a0-a3 of the CSSAL S-box LSI at 125 kHz power clock frequency. Top: Simulation signals. Bottom: Measurement signals with vertical scale: 2 V/div and horizontal scale: 10 μs/div.

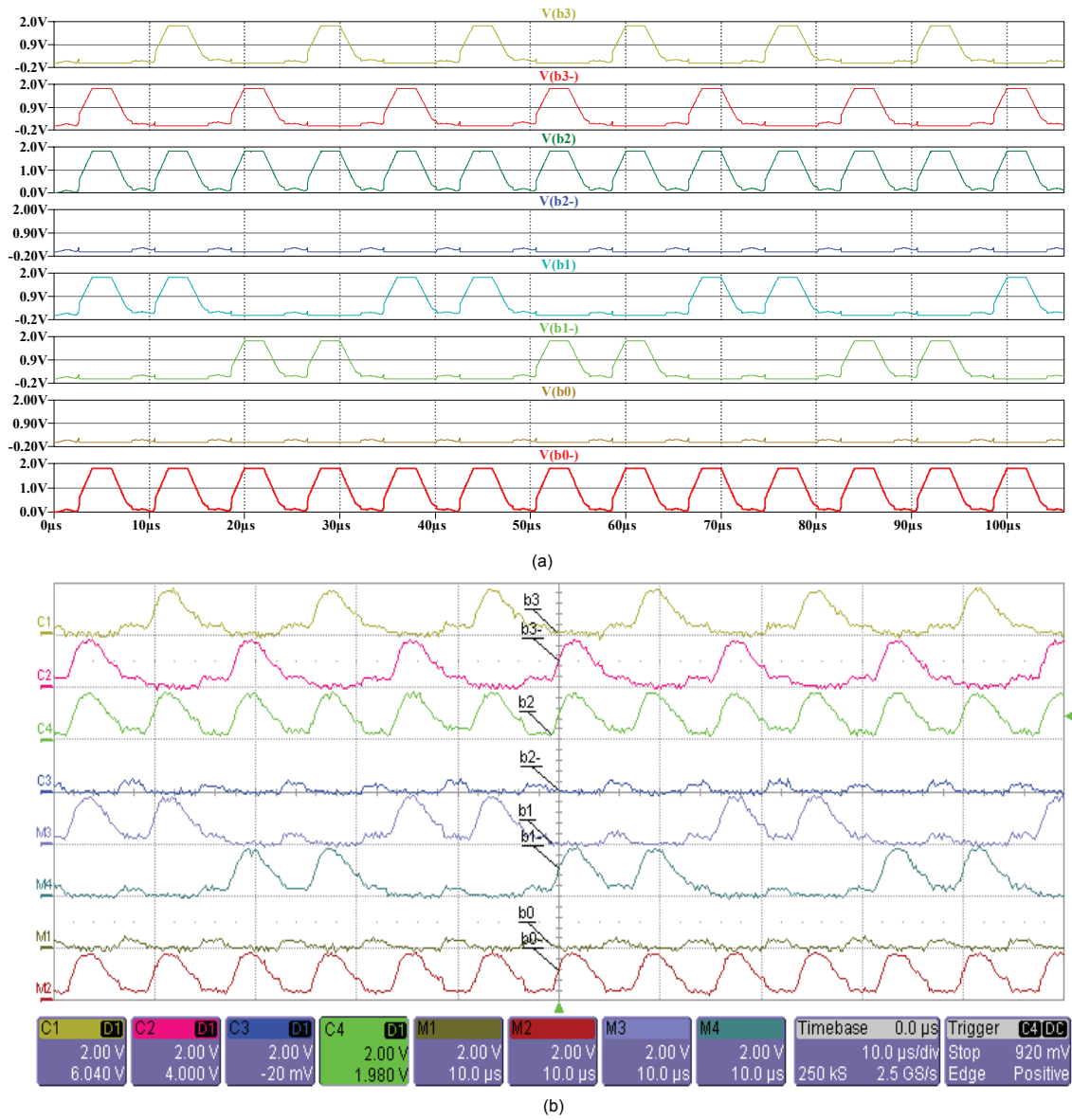


Figure 6.31: Output signals of internal wires b0–b3 of the CSSAL S-box LSI at 125 kHz power clock frequency. Top: Simulation signals. Bottom: Measurement signals with vertical scale: 2 V/div and horizontal scale: 10 μ s/div.

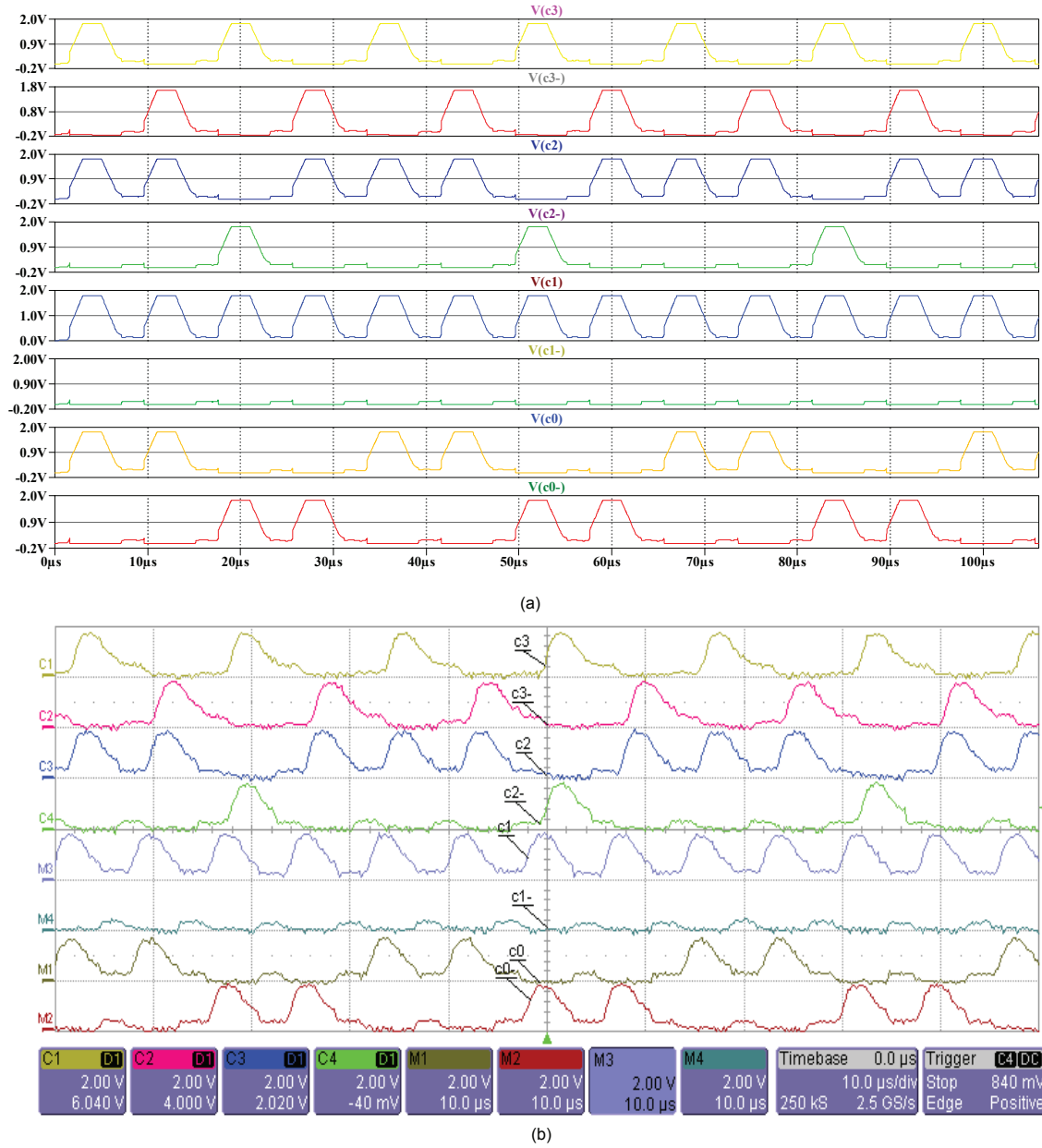


Figure 6.32: Output signals of internal wires c0–c3 of the CSSAL S-box LSI at 125 kHz power clock frequency. Top: Simulation signals. Bottom: Measurement signals with vertical scale: 2 V/div and horizontal scale: 10 μ s/div.

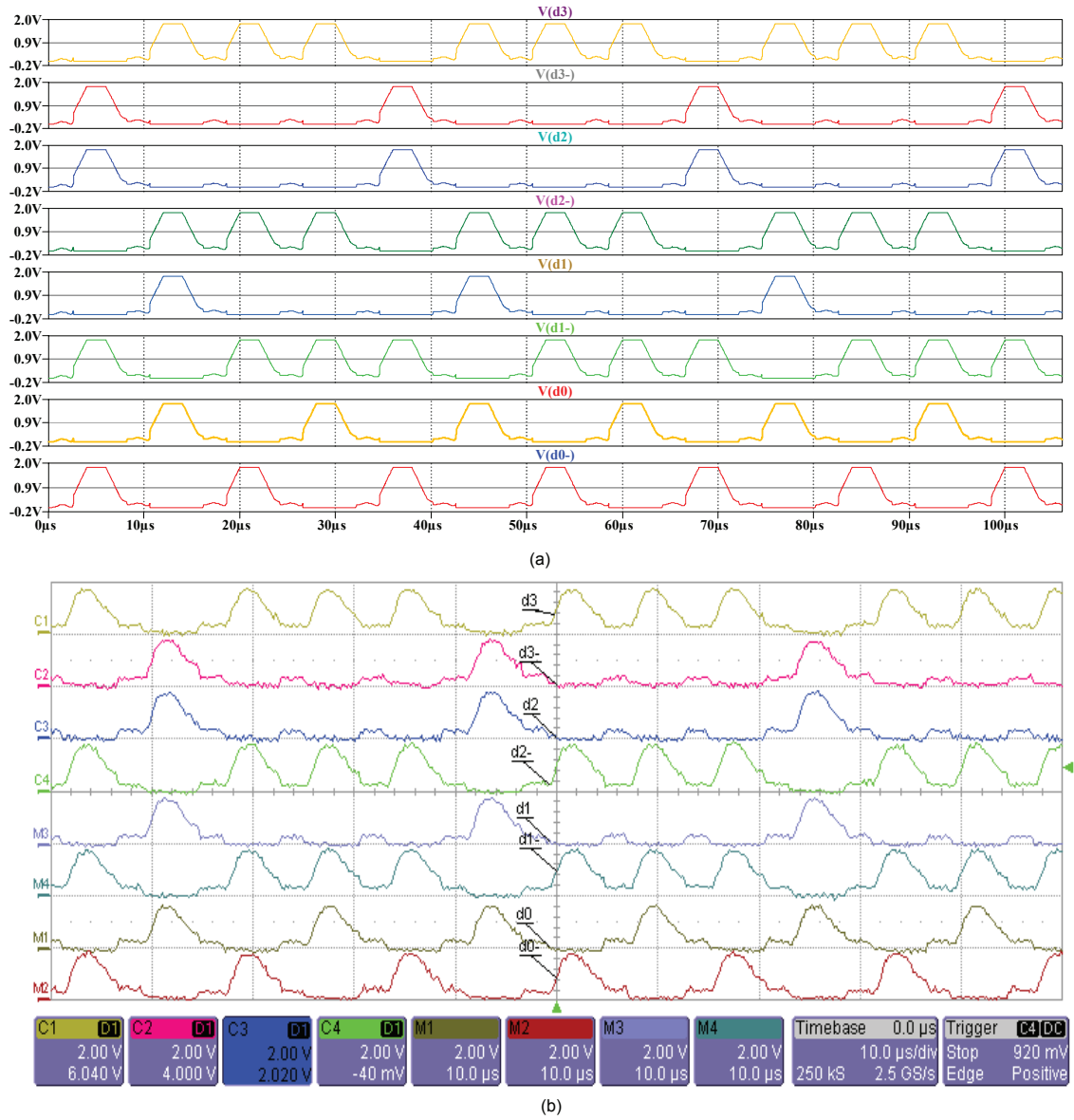


Figure 6.33: Output signals of internal wires d0–d3 of the CSSAL S-box LSI at 125 kHz power clock frequency. Top: Simulation signals. Bottom: Measurement signals with vertical scale: 2 V/div and horizontal scale: 10 μ s/div.

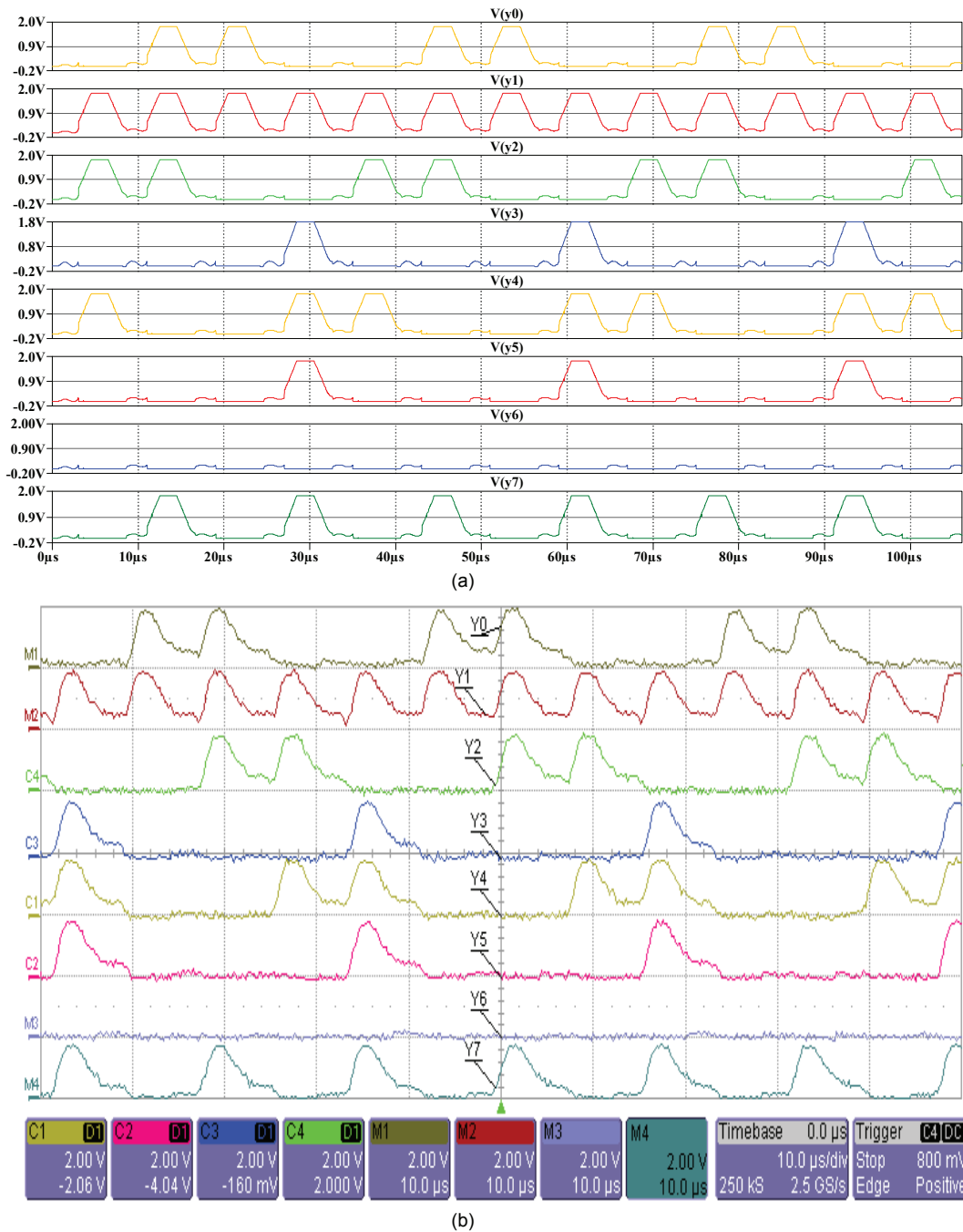


Figure 6.34: Output signals of Y0–Y7 of the CSSAL S-box LSI at 125 kHz power clock frequency. Top: Simulation signals. Bottom: Measurement signals with vertical scale: 2 V/div and horizontal scale: 10 μs/div.

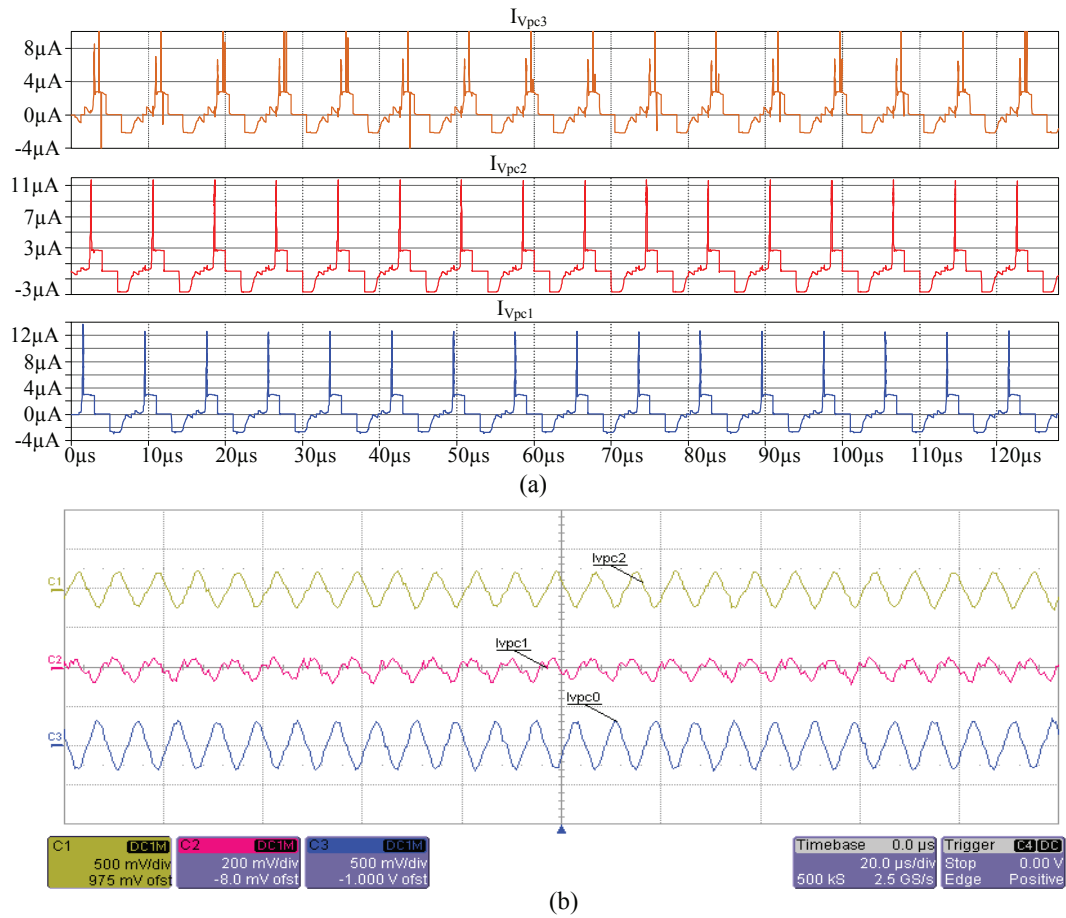


Figure 6.35: Supply current traces of the I_{Vpc0} , I_{Vpc1} , I_{Vpc2} at 125 kHz power clock frequency: (a) Simulation and (b) Measurement results with input signal in Fig. 6.29.

6.4 Summary

The author has verified the bit-parallel cellular multiplier and the 8-bit AES S-box logic functionality in the LSI chip measurement that, the output voltages are correctly measured same as the result in the SPICE simulation. The CSSAL multiplier and S-box chip features are summarized in Table 6.3. The operating frequency range of dynamic power clock frequencies for the multiplier chip measurement is from 0.125–5 MHz, where its power dissipation is smaller about 11 times than the measured TDPL multiplier at 1.25 MHz clock frequency. Furthermore, the CSSAL S-box were correctly measured at maximum speed of 125 kHz only. The CSSAL multiplier and S-box LSI chips perform an uniform power trace that are securely applicable for low-power and low frequency devices, such as smart cards, RFID tags and other low power cryptographic devices.

Table 6.3: Proposed CSSAL multiplier over $GF(2^4)$ and the 8-bit AES S-box chip features summary

Application Area	Smart card, RF-ID tag, other Cryptographic-circuits	
Feature	The supply peak current traces of the proposed LSIs chip are uniformly plotted which are resistive to DPA attacks	
Performance	Multiplier	S-box
Dynamic Ope. Freq.	0.125–5 MHz	125 KHz
P_{diss} .	CSSAL: 7.9 mW @ 1.25 MHz TDPL: 90 mW @ 1.25 MHz	CSSAL: 364.14 μ W @ 125 KHz
LSI area	CSSAL: 155(W) \times 172(H) μm^2 TDPL: 183(W) \times 173(H) μm^2	CSSAL: 795(W) \times 614(H) μm^2
Selling point	innovative/applicability approach	
Fab. run	Rohm 0.18 μm , Y2012, No. 5	Rohm 0.18 μm , Y2013, No. 1

Chapter 7

Conclusions and Future Works

7.1 Conclusions

In side-channel attacks, specifically the power analysis and the electromagnetic analysis attacks are possible because there exist correlation between the processed data and the power consumption when a crypto device executes encryption/decryption calculation. As a solution, there are two basic techniques have been widely developed to countermeasure against side-channel attacks; such as hiding and masking techniques. In this dissertation, the author has contributed to side-channel countermeasures at cell level using hiding technique, meaning that the logic circuit whose instantaneous power dissipation is constant along with the possible input/output signal transitions. The proposed CSSAL circuits were implemented using symmetric dual-rail logic style which operates in adiabatic switching principle. The validation and effectiveness of the proposed logic was tested under secure logic verification methods, such as power verification using statistical parameters and the frequency spectrum analysis. Several conventional secure logic styles were implemented under same the parameters and conditions, aimed to verify the validity and effectiveness of the proposed CSSAL to thwart side-channel attacks. Based on the investigation and evaluation result of the fundamental logics and LSI multiplier implementation, the author has summarized the important points of the merit and drawback, as follows:

1. The proposed CCSAL was implemented in such a way to initially discharge all internal node changes to ground level before the logic circuit is evaluated. By doing this, it has ability to consume uniform/constant power for every input transitions.
2. CSSAL significantly reduces the signal size compare to conventional secure logic styles (TDPL and SABL), either in the fundamental logics or in the

multiplier logic circuit. As a result, the attacker will find difficulties to detect the signal variances.

3. Simulation and measurement results of the CSSAL multiplier and CSSAL S-box LSIs have shown similar result from the security view point at low frequency speed.
4. The proposed CSSAL cannot be implemented in high speed digital circuit environment, and therefore, the application target devices should be considered. The overall results ensure that the CSSAL can be securely applicable for low-power and low frequency devices, such as smart cards, RFID tags and other battery-powered embedded systems.

7.2 Future Research Direction

The results of the current work claimed that low-power and high security measures to thwart side-channel attacks were achieved. However, the proposed work was area consuming, and hence further analysis and investigative study for gate size reduction will be addressed for future work. Moreover, the author will aim to implement full AES encryption and decryption hardware using up-to-date CMOS process technology (*i.e.*, 90, 65, 45 nm or below).

Bibliography

- [1] C. Paar and J. Pelzl, *Understanding Cryptography: A textbook for student and practitioners*, Springer, New York, 2010.
- [2] M. Borda, *Fundamental in information theory and coding*, Springer, Berlin, 2011.
- [3] “Egyptian hieroglyphs.” Internet: http://en.wikipedia.org/wiki/Egyptian_hieroglyphs, Modified on Sep. 3, 2014.
- [4] “Caesar cipher.” Internet: http://en.wikipedia.org/wiki/Caesar_cipher, Modified on Sep. 3, 2014.
- [5] P. Reuvers and M. Simons. “Enigma machine.” Internet: <http://www.cryptomuseum.com/crypto/enigma/index.htm>, Modified on Sep. 14, 2014.
- [6] “Universal serial bus (USB) token.” Internet: <https://www.novainfosec.com/2012/06/28/crypto-cracked-but-two-factor-authentication-remains-safe/>, Jun. 28, 2012.
- [7] “Smart Cards.” Internet: [ttp://www.shutterstock.com/s/%22smart+card%22/search.html](http://www.shutterstock.com/s/%22smart+card%22/search.html), 2003.
- [8] “RFID tags.” Internet: <http://www.csols.com/wordpress/rfid-labelling-solutions/>, Dec. 19, 2011.
- [9] “PLCC package.” Internet: <http://www2.hdl.co.jp/en/index.php/products/plcc68-series/ap68-04-tpm.html>, Updated on Jun. 12, 2014.
- [10] J. Ferrari, R. Mackinnon, S. Poh, and L. Yatawara, *Smart Cards: A Case Study*, IBM Corporation, International Technical Support Organization, 1998. [Online] Available at <http://www.redbooks.ibm.com/redbooks/pdfs/sg245239.pdf>

- [11] *Identification cards—Integrated circuit cards with contacts*, International Standard ISO/IEC 7816-15, 2004.
- [12] Y. Xhang and P. Kitsos, *Security in RFID and sensor networks*, Auerbach Publications, New York, 2009.
- [13] National Institute of Standards and Technology (NIST), “The Advanced Encryption Standard (AES),” FIPS Publication 197 (2001). [Online] Available at <http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf>.
- [14] B. Schneier, J. Kelsey, D. Whiting, D. Wagner, C. Hall, and N. Ferguson, “Twofish : A 128-Bit Block Chiper,” [Online] Avaliabale at <http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=A8A31BE3520F9B113ED3CCB9C4B8E57C?doi=10.1.1.35.1273&rep=rep1&type=pdf>.
- [15] R. Anderson, E. Biham, and L. Knudsen, “Serpent: A proposal for the advanced encryption standard,” [Online] Available at <http://www.cl.cam.ac.uk/~rja14/Papers/serpent.pdf>
- [16] B. Schneier, “Blowfish algorithm: A 64-bit Block Cipher,” Available at <https://www.schneier.com/blowfish.html>, 2004.
- [17] C. M. Adams, “Constructing symmetric ciphers using the CAST design procedure,” Kluwer Academic Publishers, Boston, 1997. [Online] Available at <http://tnlandforms.us/cs594-cns96/cast.pdf>.
- [18] R. L. Rivest, “The RC4 encryption algorithm,” RSA Data Security, Inc., Mar. 1992.
- [19] NIST, “The Dat Encryption Standard (DES),” FIPS Publication 140-1 (1999). [Online] Available at <http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf>.
- [20] Skipjack encryption algorithm: NIST specification. [Online] Available at <http://csrc.nist.gov/groups/ST/toolkit/documents/skipjack/skipjack.pdf>.
- [21] P. Chaudhari and H. Diwanji, “Enhanced SAFER+ Algorithm for Bluetooth to Withstand Against Key Pairing Attack” *Advances in Intelligent Syst. and Computing*, Vol. 176, pp. 651-660, 2012,
- [22] International Data Encryption Algorithm (IDEA): Information available at http://en.wikipedia.org/wiki/International_Data_Encryption_Algorithm.

-
- [23] W. Diffie and M. E. Hellman, "New directions in cryptography," *IEEE Trans. on Information Theory*, Vol. IT-22, no. 6, pp. 644–654, Nov. 1976.
- [24] R. L. Rivest, A. Shamir, and L. M. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, Feb. 1978.
- [25] NIST, "Digital Signature Standard, Federal Information Processing Standards Publication 186, May 1994. [Online] Available at <http://csrc.nist.gov/publications/PubsFIPS.html#fips186-3>.
- [26] O. Kömmerling and M. G. Kuhn, "Design principle for tempter-resistant smartcard processors," *USENIX Workshop on Smartcard Technology (Smartcard'99)*, pp. 9–20, May 1999.
- [27] S. P. Skorobogatov, *Semi-invasive attacks—A new approach to hardware security analysis*, PhD thesis, University of Cambridge, 2005. [Online] Available at <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.228.2204&rep=rep1&type=pdf>.
- [28] H. Bar-El, H. Choukri, D. Naccache, M. Tunstall, and C. Whelan, "The Sorcerer's apprentice guide to fault attacks," *Cryptography ePrint* [Online] Archive at <https://eprint.iacr.org/2004/100.pdf>.
- [29] K. Nohl, D. Evans, Starbug, and H. Plötz, "Reverse-engineering a cryptographic RFID tag," *Symp. USENIX Security*, pp. 1–9, Jul. 2008.
- [30] C. E. Shannon, "Communication theory of secret system," *Bell System Technical Journal*, vol. 28, no. 4, pp. 656–715, 1946.
- [31] C. Monteiro, Takahashi Y., and Sekine T., "DPA Resistance of charge-sharing symmetric adiabatic logic," *Proc. of IEEE ISCAS'13*, pp. 2581–2581, May 2013.
- [32] C. Monteiro, Y. Takahashi, and T. Sekine, "Charge-sharing symmetric adiabatic logic in countermeasure against power analysis attacks at cell level," *Microelectronics Journal*, vol. 44, no. 6, pp. 496–503, Jun. 2013.
- [33] W. C. Athas, L. J. Svesson, J. G. Koller, N. Traztzanis, and E. Y. -C. Chuo, "Low power digital system based on adiabatic-switching principles," *IEEE*

- Trans. Very Large Scale Integration Syst.*, vol. 2, no. 4, pp. 398–406, Dec. 1994.
- [34] M. Alioto, L. Giancane, G. Scotti, and A. Trifiletti, “Leakage power analysis attacks: A novel class of attacks to nanometer cryptographic circuits,” *IEEE Trans. Circuit and Syts.-I*, vol. 57, no. 2, pp. 355–367, Feb. 2010.
- [35] M. Alioto, L. Giancane, G. Scotti, and A. Trifiletti, “Effectiveness of leakage power analysis attacks on DPA-resistant logic styles under process variation,” *IEEE Trans. Circuit and Syts.-I*, vol. 61, no. 2, pp. 429–442, Jan. 2014.
- [36] H. J. M. Veendrick, “Short-circuit dissipation of static CMOS circuitry and its impact on the design of buffer circuits,” *IEEE J. Solid-State Circuits*, vol. 19, no. 4, pp. 468–473, Aug. 1984.
- [37] V. De, Y. Ye, A. Keshavarzi, S. Narendra, J. Kao, D. Somasekhar, R. Nair, and S. Boskar, “Techniques for leakage power reduction,” *Design of High-Performance Microprocessor Circuits*, IEEE Press, chap. 3, pp. 46–62, 2001.
- [38] P. R. Panda, A. Shrivastava, B. V. N. Silpa, and K. Gummidipudi *Power-efficient System Design*, Springer, New York, 2010.
- [39] Y. Lu and V. D. Agraval, “CMOS leakage and glitch minimization for power-performance tradeoff,” *J. Low Power Electronics*, vol. 2, no. 3, pp. 378–387, 2006.
- [40] C. Monteiro, Y. Takahashi, and T. Sekine, “An LSI Implementation of a Bit-Parallel Cellular Multiplier over $GF(2^4)$ using Secure Charge-Sharing Symmetric Adiabatic Logic,” *Proc. IEEE ISCAS 2014*, pp. 826–829, Jun. 2014.
- [41] P. Kocher, Timing attacks on implementation of Diffie-Hellman, RSA, DSS and other system,” *Proc. of the 16th Annual International Cryptology Conference on Advances in Cryptology*, LNCS 1109, pp. 104–113, Springer-Verlag, Aug. 1996.
- [42] J. -F. Dhem, F. Koeune, P. -A. Leroux, P. Mestre, J. -J. Quisquater, and J. -L. Willems, “A practical implementation of the timing attack.” *Proc. CARDIS-1998, Smart Card Research and Advanced Applications*, LNCS 1820, pp. 167–182, Springer-Verlag, 1998.

-
- [43] D. Brumley and D. Boneh, "Remote timing attacks are practical," *J. Computer Networks*, vol. 48, no. 5, pp. 701–716, Aug. 2005.
- [44] B. B. Brumley and N. Tuveri, "Remote timing attacks are still practical," *Proc. Computer Security–ESORICS*, LNCS, vol. 6879, pp 355–371, 2011.
- [45] A. K. Lenstra, H. W. Lenstra Jr., and L. Lovasz, "Factoring polynomials with rational coefficients," *Mathematische Annalen*, vol. 261, no. 4, pp. 515–534, Dec. 1982.
- [46] D. Poulakis, "Some lattice attacks on DSA and ECDSA," *J. Applicable Algebra in Engineering, Communication and Computing*, vol. 22, no. 5–6, pp. 347–358, Dec. 2011.
- [47] P. C. Kocher, J. Jaffe, B. Jun, "Differential power analysis," *Proc. of 19th Int’L Advances in Cryptology Conference (CRYPTO’99)*, LNCS 1666, pp. 388–397, Springer-Verlag, 1999.
- [48] T. Messerges, E. Dabbish, and R. Sloan, "Investigations of power analysis attacks on smartcards," *Proc. USENIX Workshop on Smartcard Technology*, pp. 151–162, May 1999.
- [49] S. Mangard, "A simple power analysis (SPA) attack on implementation of the AES key expansion," *Proc. Information Security and Cryptology–ICISC*, LNCS 2587, pp. 343–358, Springer-Verlag, 2003.
- [50] S. Mangard, E. Oswald, and T. Popp *Power Analysis Attacks: Revealing the Secret of Smart Cards*, Springer, New York, 2007.
- [51] T. Messerges, E. Dabbish, and R. Sloan, "Examining smart-card security under the threat of power analysis attacks," *IEEE Trans. Computers*, vol. 51, no. 5, pp. 541–552, May 2002.
- [52] M. Alioto, M. Poli, and S. Rocchi, "Differential power analysis attacks to precharged buses: A general analysis for symmetric-key cryptographic algorithms," *IEEE Trans. Dependable and Secure Computing*, vol. 7, no. 3, pp. 226–239, Aug. 2010.
- [53] N. Homma, A. Miyamoto, T. Aoki, A. Satoh, and A. Samir, "Comparative power analysis of modular exponentiation algorithms," *IEEE Trans. Computers*, vol. 59, no. 6, pp. 795–807, Jun. 2010.

- [54] J. Wu, Y. Shi, and M. Choi, "Measurement and evaluation of power analysis attacks on asynchronous S-box," *IEEE Trans. Instrumentation and Measurement*, vol. 61, n. 10, pp. 2765–2775, Oct. 2012.
- [55] H. Marzouqi, M. Al-Qutayri, and K. Salah, "Review of gate-level differential power analysis and fault analysis countermeasures," *IET J. Information Security*, vol. 8, no. 1, pp. 51–66, 2013.
- [56] E. Brier, C. Clavier, and F. Olivier, "Correlation Power Analysis with a Leakage Model," *Proc. of CHES-2004*, LNCS 3156, pp. 16–29, Springer-Verlag, 2004.
- [57] F. Zhang and Z. J. Shi, "Differential and correlation power analysis attacks on HMAC-Whirlpool," *Proc. IEEE 8th Int’L Conf. Information Technology: New Generations (ITNG)*, pp. 359–365, Apr. 2011. ,
- [58] J. Wu, Y. Shi, and M. Choi, "Measurement and evaluation of power analysis attacks on asynchronous S-box," *IEEE Trans. Instrumentation and Measurement*, vol. 61, no. 10, pp. 2765–2775, Oct. 2012.
- [59] E. D. Mulder, P. Buysschaert, S. B. Ors, P. Delmotte, B. Preneel, G. Vandenbosch, and I. Verbauwhede, "Electromagnetic analysis attack on an FPGA implementation of an elliptic curve cryptosystem," *Proc. IEEE Conf. Computer as a Tool-EUROCON’05*, pp. 1879–1882, Nov. 2005.
- [60] J.-J. Quisquater and D. Samyde, "Electromagnetic analysis (EMA): Measures and counter-measures for smart cards," *Proc. Int’L Conf. on Research in Smart Cards, E-smart*, LNCS 2140, pp. 200–210, Springer-Verlag, Sep. 2001.
- [61] K. Gandolfi, C. Mourtel, and F. Olivier, "Electromagnetic analysis: Concrete results," *Proc. Third Inte’L Workshop on Cryptographic Hardware and Embedded Systems-CHES*, LNCS 2162, pp. 251–261, Springer-Verlag, May 2001.
- [62] H. Li, A. T. Marketos, and S. Moore, "Security evaluation against electromagnetic analysis at design time," *Proc. IEEE High-Level Design Validation and Test Workshop*, pp. 211–218, Nov. 2005.
- [63] E. De Mulder, S.B. Ors, B. Preneel, and I. Verbauwhede, "Differential electromagnetic attack on an FPGA implementation of elliptic curve cryptosystems," *Proc. IEEE World Automation Congress-WAC’06*, pp. 1–6, Jul. 2006.

-
- [64] A. Dehbaoui, S. Ordas, L. Torres, M. Robert, and P. Maurine, "Implementation and efficiency evaluation of construction-based countermeasures against electromagnetic analysis," *Proc. IEEE 6th Int'L Conf. on Design and Technology of Integrated Syst. in Nanoscale Era*, pp. 1–6, App. 2011.
- [65] V. Lomná, A. Dehbaoui, T. Ordas, P. Maurine, L. Torres, M. Robert, R. Soares, N. Calazans, and F. Moraes, "Secure triple track logic robustness against differential power and electromagnetic analyses," *J. Integrated Circuits and Systems*, vol. 4, no. 1, pp. 20–28, 2009.
- [66] K. Gu, L. Wu, X. Li, and X. Zhang, "Design and implementation of an electromagnetic analysis system for smart cards," *Proc. IEEE 7th Conf. Computational Intelligence and Security*, pp. 653–656, Dec. 2011.
- [67] J. Rabaey, *Digital Integrated Circuits: A Design Perspective*, Prentice Hall, New York, 1996.
- [68] D. J. Tran and M. J. Acuff, "Dynamic logic circuit", U.S. Patent No.: 5 859 547, Jan. 12, 1999.
- [69] K. Tiri, M. Akmal, and I. Verbauwhede, "A dynamic and differential CMOS logic with signal independent power consumption to withstand differential power analysis on smart cards," *Proc. 28th European Conference on Solid-State Circuits (ESSCIRC'02)*, pp. 403–406, 2002.
- [70] K. Tiri and I. Verbauwhede, "Securing encryption algorithms against DPA at the logic level: Next generation smart card technology," *Proc. Cryptographic Hardware and Embedded Systems (CHES)*, LNCS 2779, pp. 125–136, Springer-Verlag, 2003.
- [71] K. Tiri and I. Verbauwhede, "Charge recycling sense amplifier based logic: securing low power security IC's against DPA," *Proc. ESSCIRC'04*, pp. 179–182, 2004.
- [72] K. Tiri, I. Verbauwhede, "A logic level design methodology for a secure DPA resistant ASIC or FPGA implementation," *Proc. Design, Automation and Test in Europe Conference and Exhibition*, pp. 246–251, 2004.
- [73] J. S. Lee, J. W. Lee, and Y. H. Kim, "Symmetric discharge logic against differential power analysis," *IEICE Trans. Fundamental*, vol. E90–A, no. 1, pp. 234–240, Jan. 2007.

- [74] M. Bucci, L. Giancane, R. Luzzi, and A. Trifiletti, “Three-phase dual-rail pre-charge logic”, *Proc. Workshop Cryptographic Hardware and Embedded Systems*, pp. 232–241, Oct. 2006.
- [75] K.J. Kulikowski, V. Venkataraman, Z. Wang, A. Taubin, M. Karpovsky, “Asynchronous balanced gates tolerant to interconnect variability,” *Proc. IS-CAS 2008*, pp. 3190–3193, May 2008.
- [76] J. S. Coron and L. Goubin, “On Boolean and arithmetic masking against differential power analysis,” *Proc. CHES’00*, LNCS 1956, pp. 231–237, Springer–Berlin, Jan. 2000.
- [77] L. Goubin, “A sound method for switching between Boolean and arithmetic masking,” *Proc. CHES’01*, LNCS 2162, pp. 3–15, Springer-Verlag, Sep. 2001.
- [78] J. Dj. Golic, “Multiplicative masking and power analysis of AES,” *Proc. CHES’02*, LNCS 2532, pp. 198–212, Springer-Verlag, 2002.
- [79] T. Popp, S. Mangard, “Masked dual-rail pre-charge logic: DPA-Resistance without routing constraints,” *Proc. of CHES’05*, LNCS 3659, pp. 172–186, Springer-Verlag, 2005.
- [80] I. Hassoune, F. Mace, D. Flandre, and J.-D. Laget, “Low-swing current mode logic (LSCML): A new locig style for secure and robust smart cards against power analysis attacks,” *J. Microelectronic*, vol. 37, pp. 997–1006, 2006.
- [81] S. Guilley, F. Flament, P. Hoogvorst, R. Pacalet, and Y. Mathieu, “Secured CAD Back-End Flow for Power-Analysis-Resistant Cryptoprocessors,” *IEEE Trans. Design & Test of Computers*, vol. 24, no. 6, pp. 546–555, Nov. 2007.
- [82] S. Rammohan, V. Sundaresan, and R. Vemuri, “Reduced complementary dynamic and differential logic: A CMOS logic Style for DPA-resistant secure IC design,” *Proc. 21st Int’L Conf. on VLSI Design (VLSID)*, pp. 699–705, Jan. 2008.
- [83] K. J. Kulikowski, V. Venkataraman, Z. Wang, A. Taubin, and M. Karpovsky, “Asynchronous balanced gates tolerant to interconnect variability,” *Proc. IS-CAS 2008*, pp. 3190–3193, May 2008.
- [84] S. Guilley, F. Plament, Y. Mathieu, and R. Pacalet, “Security Evaluation of a balanced quasi-delay insensitive library,” *IEEE Press. Conf. Design*

- of Circuits and Integrated Syst.*, (6pages), Nov. 2008. [Online] Available at <https://hal.archives-ouvertes.fr/file/index/docid/283405/filename/seclib.pdf>.
- [85] S. Guilley, L. Sauvage, F. Flament, V.-N. Vong, P. Hoogvorst, and R. Pacalet, "Evaluation of power constant dual-rail logics countermeasures against DPA with design time security metrics," *IEEE Trans. Computers*, vol. 59, no. 9, pp. 1250–1263, Sep. 2010.
 - [86] D. Kamel, M. Renauld, D. Bol, F.-X. Standaert, and D. Flandre, "Analysis of dynamic differential swing limited logic for low-power secure applications," *J. Low Power Electr. Appl.*, vol. 2, no. 1, pp. 98–126, Mar. 2012.
 - [87] L. G. Heller, W. R. Griffin, J. W. Davis, and N. G. Thomas, "Cascode voltage switch logic: A differential CMOS logic family," *Proc. IEEE ISSCC Dig. Tech. Papers*, vol. 27, pp. 16–17, Feb. 1984.
 - [88] K.-K. Mok, K.-H. Tsang, C.-F. Chan, C.-S. Choy, and K.-P. Pun, "Adiabatic smart card," *Proc. IEEE Asia Pacific Conf. Circuits and Syst. (APCCAS)*, pp. 287–290, Dec. 2006.
 - [89] K.-K. Mok and C.-F. Chan, "A 13.56 MHz adiabatic smart card/RFID" *Proc. IEEE ASICON'07*, pp. 874–877, Oct. 2007.
 - [90] Y. Takahashi, T. Sekine, and M. Yokoyama, "A comparison of adiabatic logic as a countermeasures against power analysis attacks," *Proc. IEEE Int'l Conf. Syst. Science and Engineering (ICSSE)* pp. 615–618, Jul. 2010.
 - [91] C. Monteiro, Y. Takahashi, and T. Sekine, "Resistance against power analysis attacks on adiabatic dynamic and adiabatic differential logics for smart card," *Proc. IEEE Intelligent signal processing and communication syst. (ISPACS)*, 5pages (DVD), Dec. 2011. (DOI: 10.1109/ISPACS.2011.6146067)
 - [92] Y. Moon and D.-K. Joeng, "An efficient charge recovery logic circuit", *IEEE J. Solid-State Circuit*, vol. 31, no. 4, pp. 415–522, Apr. 1996.
 - [93] A. Kramer, J. S. Denker, B. Flower and J. Moroney, "2nd order adiabatic computation 2N-2P and 2N-2N2P logic circuits," *Proc. of Int. Symp. on Low Power Design*, pp. 191–196, 1995.
 - [94] M. Khatir, and A. Moradi, "Secure adiabatic logic: A low-energy DPA-resistant logic style," *Cryptology ePrint Archive*, Report 2008/123, 2008. [Online] Available at <http://eprint.iacr.org/2008/123>.

- [95] B. -D. Choi, K. E. Kim, K. -S. Chung, and D. K. Kim, "Symmetric adiabatic logic circuits against differential power analysis," *ETRI Journal*, vol. 32, no. 1, pp. 166–168, Feb. 2010.
- [96] D. Maksimovic and V.G. Oklobdzija, "Clocked CMOS adiabatic logic with single AC power supply," *Proc. ESSCIRC'95*, pp. 370–373, 1995.
- [97] B. V. P. Vasantha Kumar, N. S. Murthy Sharma, K. Lal Kishore, and A. Rajakumari, "Selective glitch reduction technique for minimizing peak dynamic IR drop," *J. Microelectronics and Solid State Electronics*, vol. 2, no. 2A, pp. 27–32, 2013.
- [98] S. Mangard , N. Pramstaller, and E. Oswald, "Successfully attacking masked AES hardware implementations," *Proc. CHES'05*, LNCS 3659, pp. 157–171, Springer-Verlag , 2005.
- [99] S. Mangard, T. Popp, and B. M. Gammel, "Side-channel leakage of masked CMOS gates," *Proc. Int'L Conf. Topics in Cryptology–CT-RSA*, LNCS 3376, pp. 351–365, Springer-Verlag, 2005.
- [100] S. Nikova, V. Rijmen, M. Schlaffer, "Secure hardware implementation of non-linear functions in the presence of glitches," *Journal of Cryptology*, vol. 24. no. 2, pp. 292–321, 2011.
- [101] A. Satoh, S. Morioka, K. Takano, S. Munetoh, "A compact Rijndael hardware architecture with S-box optimization," *Proc. Advances in Cryptology–ASIACRYPT'01*, LNCS 2248, pp. 239–254, Springer-Verlag, Nov. 2001.
- [102] S. Morioka and A. Satoh, "An optimized S-box circuit architecture for low power AES design," *Proc. 4th Int'L Workshop on CHES'03*, LNCS 2523, pp. 172–186, Springer-Verlag, 2002.
- [103] J. -H. Chen, S. -J. Huang, W. -C. Lin, Y. -K. Lu, and M. -D. Shieh, "Exploration of low-cost configurable S-Box designs for AES applications," *Proc. Int'L Conf. Embedded Software and Syst. (ICESS)*, pp. 422–428, 2008.
- [104] P. V. S. Shastri, A. Agnihotri, D. Kachhwaha, and J. Singh, "A Combinational Logic Implementation of S-box of AES," *Proc. IEEE 54th Int'L Midwest Symposium on Circuits and Syst. (MWSCAS)*, pp. 1–4, 2011.

-
- [105] S. Talapatra, H. Rahaman, J. Mathew, “Low complexity digit serial systolic Montgomery multipliers for special class of $GF(2^m)$,” *IEEE Trans. VLSI Syst.*, vol. 18, no. 5, pp. 847–852, 2010.
- [106] B. A. Laws, Jr. and C. K. Rushforth, “A cellular-array multiplier for $GF(2^m)$,” *IEEE Trans. Computer*, vol. C-20, no. 12, pp 1573–1578, 1971.
- [107] C. -H. Liu, N. -F. Huang, and C. -Y. Lee, “Computation of AB^2 multiplier in $GF(2^m)$ using an efficient low-complexity cellular architecture,” *IEICE Trans. Fundamentals*, vol. E83-A, no. 12, pp. 2657–2663, Dec. 2000.
- [108] C. H. Gebotys, C. C. Tiu, and X. Chen, “A countermeasure for EM attack of a wireless PDA,” *Proc. IEEE Int’L Conf. Information Technology: Coding and Computing (ITCC’05)*, pp.544–549, Apr. 2005.
- [109] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De, “Parameter variations and impact on circuits and microarchitecture,” *Proc. Design Automation Conf.*, pp. 338–342, Jun. 2003.
- [110] K. Bowman, S. Duvall, and J. Meindl, “Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration,” *IEEE J. Solid State Circuits*, vol. 37, no. 2, pp. 183–190, Feb. 2002.
- [111] S. R. Sarangi, B. Greskamp, R. Teodorescu, J. Nakano, A. Tiwari, and J. Torrellas, “VARIUS: A Model of Process Variation and Resulting Timing Errors for Microarchitects,” *IEEE Trans. Semiconductor Manufacturing*, vol. 21, no. 1, pp. 3–13, Feb. 2008.
- [112] Nanoscale Integration and Modeling (NIMO) Group, “Predictive Technology Model (PTM),” Arizona State University. [Online] Available at <http://nimo.asu.edu/>
- [113] J. Hu, W. Zhang, X. Ye, and Y. Xia, “Low power adiabatic logic circuits with feedback structure using three-phase power supply,” *Proc. IEEE Int’L Conf. Communications, Circuits and Syst.*, pp. 1375–1379, May 2005.

Chapter

List of publications

Publications related to this thesis

Journal Papers:

1. Câncio Monteiro, Y. Takahashi, and T. Sekine, “Low-power secure S-Box circuit using CSSAL for AES hardware design,” *IET Circuits, Devices & Systems* (Accepted).
2. Câncio Monteiro, Y. Takahashi, and T. Sekine, “Low power bit-parallel cellular multiplier implementation in secure dual-rail adiabatic logic,” *IACSIT International J. Modeling and Optimization*, vol. 3, no. 4, pp. 329-332, August 2013.
3. Câncio Monteiro, Y. Takahashi, and T. Sekine, “Charge-sharing symmetric adiabatic logic in countermeasure against power analysis attacks at cell level,” *Microelectronics Journal* vol. 44, pp. 496-503, June 2013.

International conference papers:

1. Câncio Monteiro, Y. Takahashi, and T. Sekine, “Effectiveness of Dual-Rail CSSAL Against Power Analysis Attack Under CMOS Process Variation, *Proc. IEEE APC-CAS 2014*, pp. 121–124, Sep. 17-20, Okinawa, Japan.
2. Câncio Monteiro, Y. Takahashi, and T. Sekine, “Process variation verification of low-power secure CSSAL AES S-box,” *Proc. IEEE MWSCAS 2014*, pp. 21–24, August 3-6, College Station, TX, USA.

3. Câncio Monteiro, Y. Takahashi, and T. Sekine, "An LSI Implementation of a Bit-Parallel CellularMultiplier over $GF(2^4)$ using Secure Charge-Sharing Symmetric Adiabatic Logic," *Proc. IEEE ISCAS 2014*, pp. 826-829, June 1-5, Melbourne, Australia.
4. Câncio Monteiro, Y. Takahashi, and T. Sekine, "Low power secure CSSAL bit-parallel multiplier over $GF(2^4)$ in 0.18 μm CMOS technology," *Proc. IEEE ECCTD 2013*, Digital Circuit Design (USB), 4pages, September 8-12, Dresden, Germany.
5. Câncio Monteiro, Y. Takahashi, and T. Sekine, "Robust secure charge-sharing symmetric adiabatic logic against side-channel attacks," *Proc. IEEE TSP 2013*, pp. 736-739, July 2-4, Roma, Italy.
6. Câncio Monteiro, Y. Takahashi, and T. Sekine, "Low power secure AES S-box using adiabatic logic circuit," *Proc. IEEE FTFC 2013*, Regular session 3 (USB), 4pages, June 20-21, Paris, France.
7. Cancio Monteiro, Y. Takahashi, and T. Sekine, "DPA Resistance of charge-sharing symmetric adiabatic logic," *Proc. IEEE ISCAS 2013*, pp. 2581–2584, May 19-23, Beijing, China.
8. Câncio Monteiro, Y. Takahashi, and T. Sekine, "Low Power Bit-Parallel Multiplier over $GF(2^4)$ using CSSAL for Cryptographic Hardware Implementation," *Proc. IEEE Symposium on Low-Power and High-Speed Chips (COOL Chips XVI)*, Poster 9, April 17-19, Yakohama, Japan.
9. Câncio Monteiro, Y. Takahashi, and T. Sekine, "Secure charge-sharing symmetric adiabatic logic implementation in AES S-Box architecture for smart card," *Proc. IEEE ICEIC 2013*, pp. 304-305, Jan. 30-Feb. 2, Bali, Indonesia.
10. Câncio Monteiro, Y. Takahashi, and T. Sekine, "A comparison of cellular multiplier cell using secure adiabatic logics," *Proc. ITC-CSCC 2012*, E-M2-03 (CD-ROM), 4pages, July 15-18, Hokkaido, Japan.

Domestic conference papers:

1. Câncio Monteiro, Y. Takahashi, and T. Sekine, "Security evaluation of CSSAL countermeasure against side-channel attacks using frequency spectrum analysis," *IEICE Technical Report on EMCJ*, vol.114, no.381, EMCJ2014-82, pp.75–80, Dec. 2014.
2. Câncio Monteiro, Y. Takahashi, and T. Sekine, "Measurement of CSSAL Multiplier over $GF(2^4)$ LSI Implemented in 0.18 μm CMOS Technology," *Proc. IEICE General Conference 2014*, p. 2 (A-1-2), March 2014.
3. Câncio Monteiro, Y. Takahashi, and T. Sekine, "LSI implementation of a bit-Parallel cellular multiplier over $GF(2^4)$ using charge-sharing symmetric adiabatic logic," *2013 Proc. IEICE Society Conference 2013*, p. 101 (C-12-41), September 2013.
4. Câncio Monteiro, Y. Takahashi, and T. Sekine, "LSI implementation of a secure low-power CSSAL cellular multiplier," *IEICE Technical Report on Circuit and System*, vol. 113, no. 224, pp. 89-94, September 2013.
5. Y. Takahashi, Câncio Monteiro, and T. Sekine, "CSSAL: Charge sharing symmetric adiabatic logic—Case study of logic circuit design and cryptographic circuit design—," *IEICE Technical Report*, vol.113, no. 224, CAS2013-49, pp. 71–75, Sept. 2013.
6. Câncio Monteiro, Y. Takahashi, and T. Sekine, "Low power CSSAL bit-parallel multiplier over $GF(2^4)$ in 0.18 μm CMOS technology," *it IEICE Technical Report on EMCJ*, vol. 113, no. 2, pp. 13-18, April 2013.
7. Câncio Monteiro, Y. Takahashi, and T. Sekine, "Survey on secure adiabatic logic for countermeasure against side-channel attacks," *it IEICE Technical Report on EMCJ*, vol. 112, no. 361, pp. 95-100, December 2012.
8. Câncio Monteiro, Y. Takahashi, and T. Sekine, "Investigation study of inner-cell bit-parallel multiplier over $GF(2^m)$ using secure adiabatic logic style," *Proc. IEICE Society Conference 2012*, p. 116 (A-7-6), September 2012.
9. Câncio Monteiro, Y. Takahashi, and T. Sekine, "A comparison of cellular multiplier cell for finite field $GF(2^m)$ using secure adiabatic logics," *it IEICE*

Technical Report on IEEJ, vol. ECT-12, no. 3, pp. 73-77 (ETC-12-052), June 2012.

Appendix A

Simulation Result using 90 nm CMOS Process

A.1 Multiplier over $GF(2^4)$

The proposed CSSAL circuits implemented in the bit-parallel cellular multiplier over $GF(2^4)$ were also simulated using 90 nm predictive technology model [112]. Transistor sizes, drain-source areas, and the drain-source perimeters are listed in Table A.1. Investigation processes are same with the one using 0.18 μm CMOS process presented in the main document of this dissertation. Therefore, the result of supply current transitions in Fig. A.1 is similar to Fig. 5.24, and the process variation verification in Fig. A.3 is also same as the one in Fig. 5.26. For better understanding of these figures, the author suggests you to review the explanation in Section 5.2.4.

Comparing these graphical and histogram information, we can figure out that the proposed CSSAL multiplier remains strong even in nano meter scale process corner. Proposed CSSAL multiplier circuit becomes more stable under process variation when using 90 nm CMOS process, and the values of NSD are smaller as shown in Fig. 5.26. The results in this chapter has been accepted in proceeding of IEEE APCCAS-2014.

Table A.1: Simulation parameters

Parameters		
	PTM90P/N (90 nm)	ROHMN/P018typ (0.18 μm)
L/W	90 nm / 270 nm	0.18 μm / 0.6 μm
AD=AS	0.1215p	0.24p
PD=PS	1.44 μ	2 μ
AD=AS= $(1\lambda+2\lambda+2\lambda)\times W = 5\lambda\times W$		
PD=PS= $2\times(1\lambda+2\lambda+2\lambda) + 2W = 10\lambda + 2W$		
$\lambda = L$		

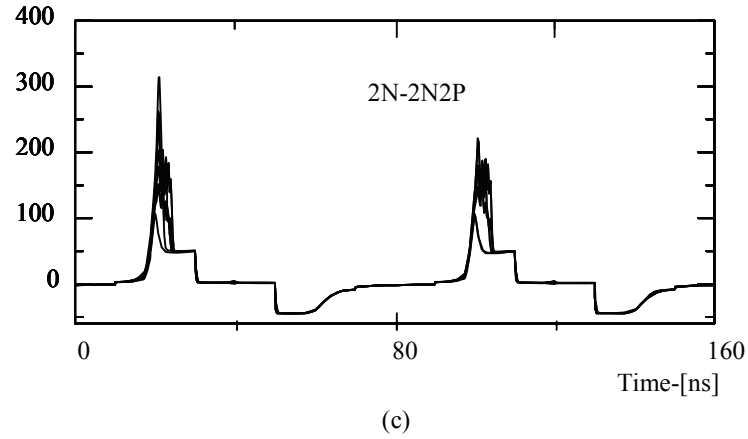
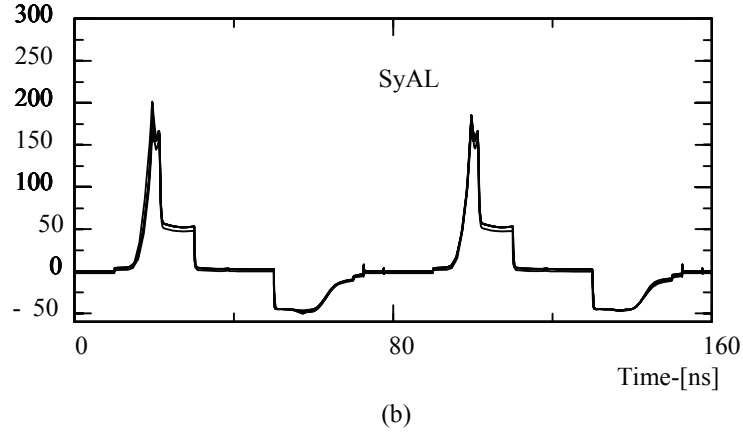
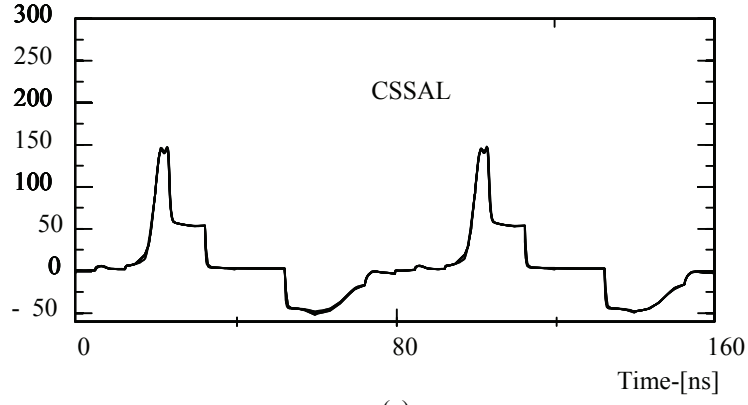


Figure A.1: Instantaneous supply current transition of 8 input patterns representation of each adiabatic multiplier circuits; (a) CSSAL Multiplier; (b) SyAL Multiplier; (c) 2N-2N2P Multiplier.

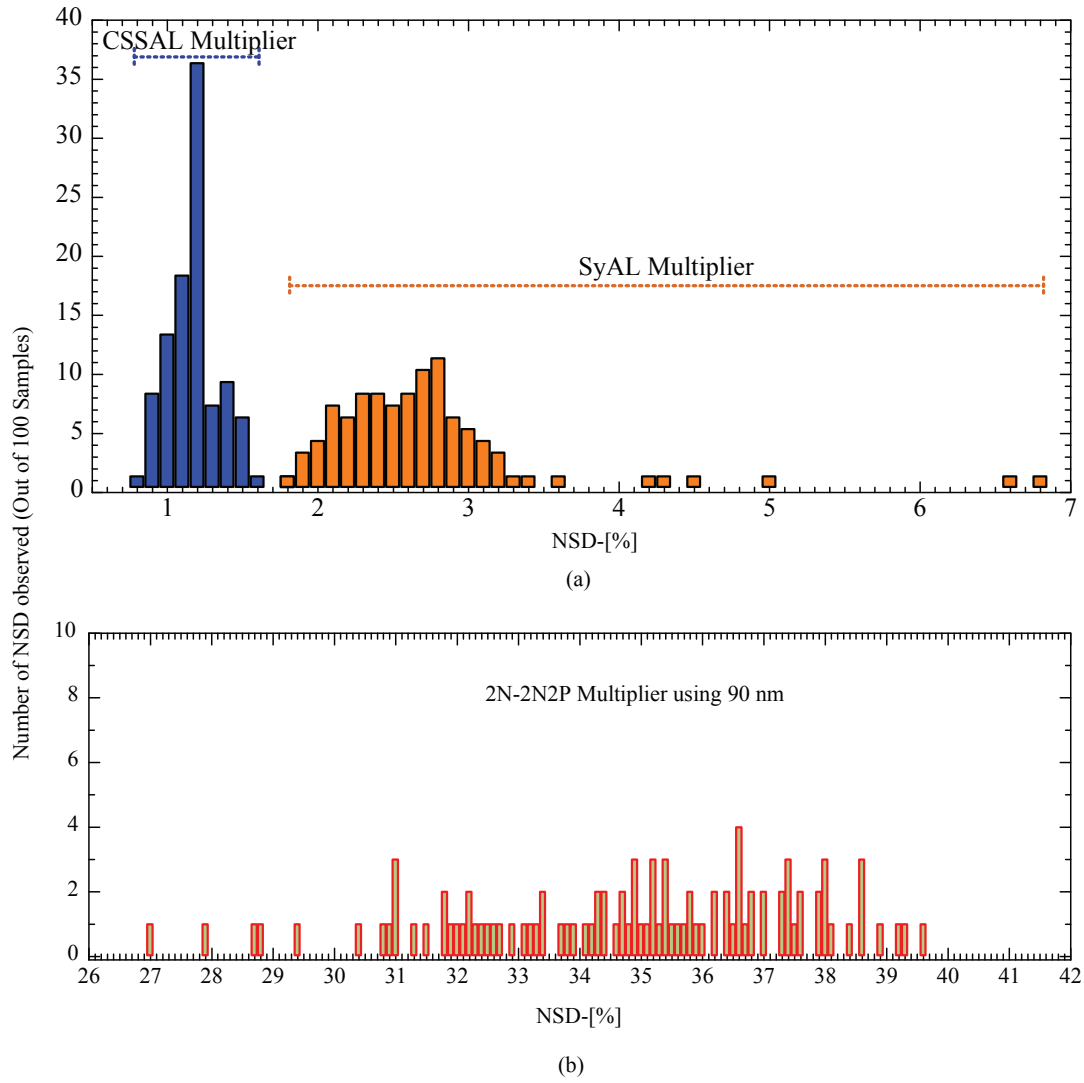


Figure A.2: NSD occurrences of the CSSAL and SyAL multipliers using Monte Carlo simulation for process variation verification: (a) CSSAL vs SyAL and (b) 2N-2N2P.

A.2 An 8-bit AES S-box Circuit

In this section, the Monte Carlo simulation results of the S-box circuit's stability using 90 nm PTM CMOS process are presented. The simulation and calculation result in Fig. A.3 shown the proposed CSSAL S-box circuit has less variation and always in the smallest scale of NSD percentage, which is same with the one using 180 nm CMOS process shown in Fig. 5.42.

In addition, energy dissipation comparison of the S-boxes using 90 nm process is depicted in Fig. A.4, where the CSSAL S-box consume less energy at 12.5 MHz than the others.

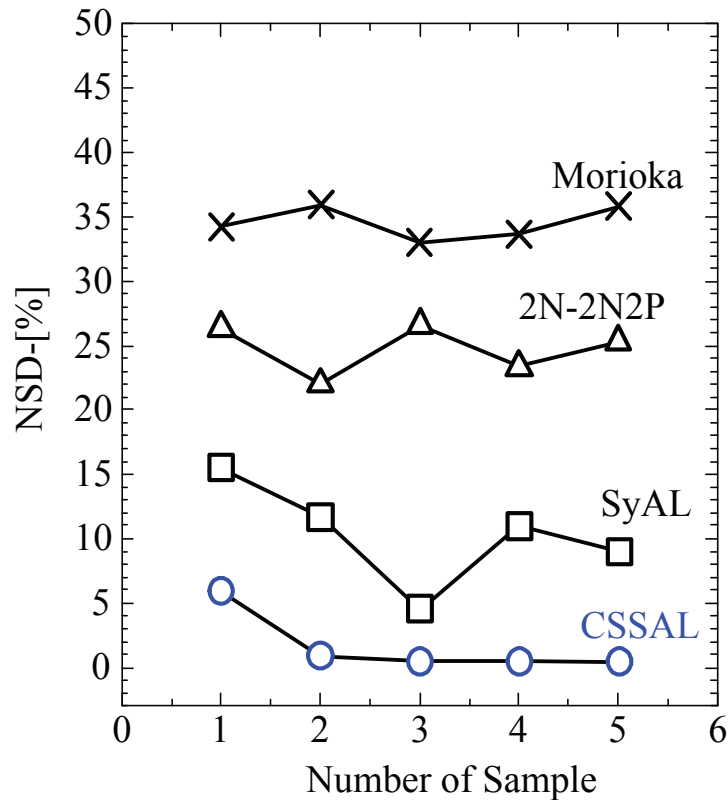


Figure A.3: Simulation and calculation results of NSD using 90 nm CMOS Process.

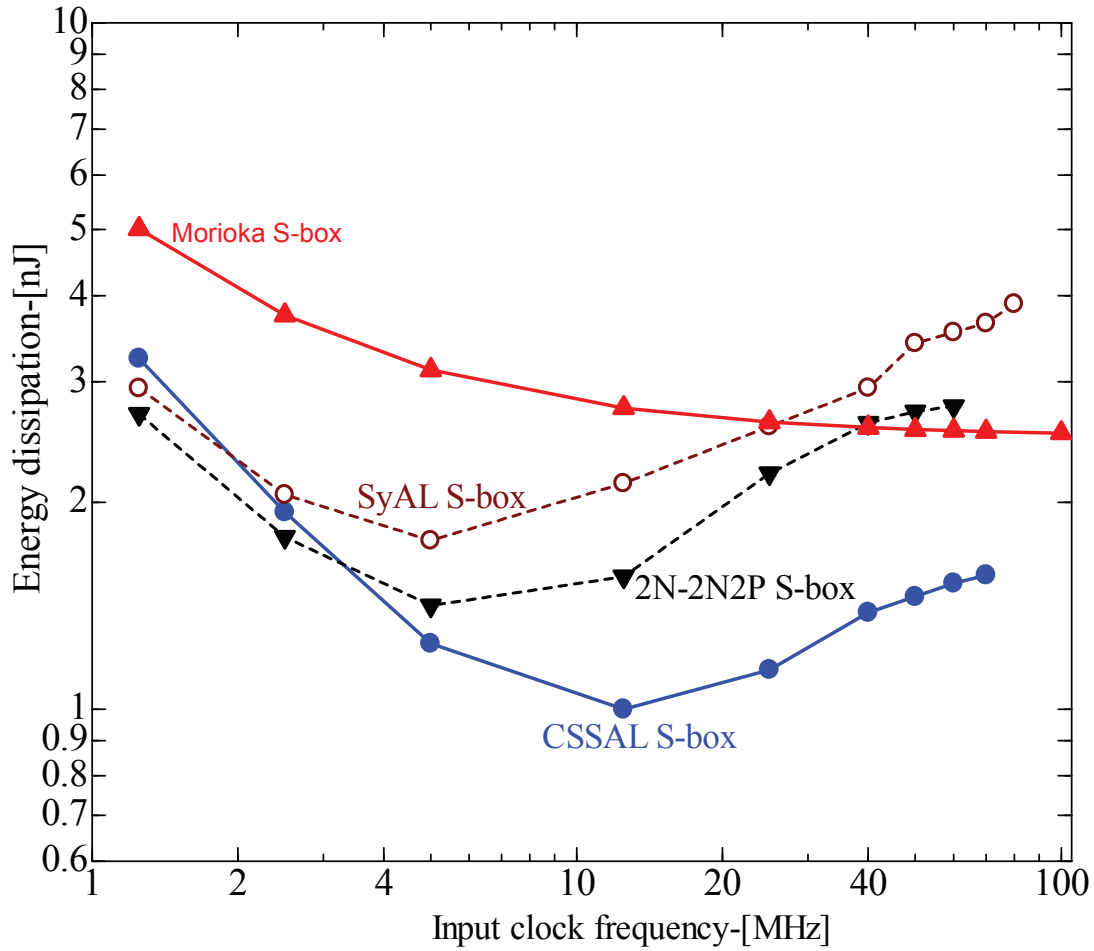


Figure A.4: Simulated energy dissipation comparison of all the investigated adiabatic logics: CSSAL, SyAL, 2N-2N2P, and Morioka's circuit [102] implementation in the multi-stage PPRM 8-bit S-box circuit at each operating frequency ranges.

Appendix B

PPRM Representation of AES

S-Box Internal Logic Circuit

In this chapter, the author discusses the logic sharing technique which adopted to optimize the gate size of PPRM S-box combinational logic style proposed in [102]. Both Boolean function and the corresponding logic diagram are included.

For comparison, the author of this dissertation classified the Boolean function as combinational logic version 1 (ver.1) which is same as the original design in the appendix of reference [102], and the combinational logic version 2 (ver.2), in which the gate size is optimized by using logic sharing technique.

B.1 Stage 1 of PPRM S-Box

B.1.1 Combinational Logic Version 1 (ver.1)

This version is same as originally written in Appendix PPRM representation in [102]. The variable x_7-x_0 denote primary inputs of S-box, and variables a and b denote internal wires. The logic designed in this version is that, all the XOR logics described for each internal wires of a_3-a_0 and b_3-b_0 are connected independent to each other, as shown in the following expression (see also Fig. B.1).

ver.1 :

$$\mathbf{a_3} = x_7 \oplus x_5$$

$$\mathbf{a_2} = x_7 \oplus x_6 \oplus x_4 \oplus x_3 \oplus x_2 \oplus x_1$$

$$\mathbf{a_1} = x_7 \oplus x_5 \oplus x_3 \oplus x_2$$

$$\mathbf{a_0} = x_7 \oplus x_5 \oplus x_3 \oplus x_2 \oplus x_1$$

$$\mathbf{b_3} = x_5 \oplus x_6 \oplus x_2 \oplus x_1$$

$$\mathbf{b}_2 = x_6$$

$$\mathbf{b}_1 = x_7 \oplus x_5 \oplus x_3 \oplus x_2 \oplus x_6 \oplus x_4 \oplus x_1$$

$$\mathbf{b}_0 = x_7 \oplus x_5 \oplus x_3 \oplus x_2 \oplus x_6 \oplus x_0$$

B.1.2 Logic Sharing Technique (Combinational Logic ver.2)

In this version of combinational logic based on first version is optimized using logic sharing technique in order to reduce the number of XOR logic counts. As described in Boolean function in ver.2 that by re-using the multiple XOR logic that appear in each internal wires, we can minimize the number of XOR of internal wires a and b . For example, in the ver.1 there are 27-XOR logics, however, by the logic sharing technique, we reduce the logic numbers to 15-XOR logics only, as shown in the following expression (see also Fig. B.2).

ver.2 :

$$\mathbf{x}_a = x_7 \oplus x_5, \mathbf{x}_b = x_3 \oplus x_2, \mathbf{x}_c = x_a \oplus x_b,$$

$$\mathbf{x}_d = x_6 \oplus x_4$$

$$\mathbf{a}_3 = x_a$$

$$\mathbf{a}_2 = x_d \oplus x_7 \oplus x_a \oplus x_1$$

$$\mathbf{a}_1 = x_c$$

$$\mathbf{a}_0 = x_c \oplus x_1$$

$$\mathbf{b}_3 = x_5 \oplus x_6 \oplus x_2 \oplus x_1$$

$$\mathbf{b}_2 = x_6$$

$$\mathbf{b}_1 = x_c \oplus x_d \oplus x_1$$

$$\mathbf{b}_0 = x_c \oplus x_6 \oplus x_0$$

The combinational logics shown in Fig. B.3 are designed based on the following Boolean function of c_0 - c_3 , which labeled as ver.1. There are 66-AND gates and 72-XOR gates in internal wires of c_0 - c_3 .

Figure B.4 is constructed after the gate reduction using same process as ver.2 aforementioned. In this figure, the AND gates are reduced from 66 to 28 gates.

ver.1 :

$$\mathbf{c}_3 = (x_5 \text{ and } x_1) \oplus (x_7 \text{ and } x_1) \oplus (x_5 \text{ and } x_2) \oplus (x_5 \text{ and } x_6) \oplus (x_5 \text{ and } x_7) \oplus (x_5 \text{ and } x_4) \oplus (x_7 \text{ and } x_4) \oplus (x_5 \text{ and } x_0) \oplus (x_7 \text{ and } x_0) \oplus (x_3 \text{ and } x_1) \oplus (x_4 \text{ and } x_1) \oplus (x_3 \text{ and } x_2) \oplus (x_2 \text{ and } x_4) \oplus (x_4 \text{ and } x_6) \oplus (x_2 \text{ and } x_1) \oplus (x_2 \text{ and } x_6) \oplus (x_6 \text{ and } x_1)$$

$$\mathbf{c}_2 = (x_6 \text{ and } x_1) \oplus (x_2 \text{ and } x_6) \oplus (x_3 \text{ and } x_6) \oplus (x_7 \text{ and } x_6) \oplus (x_1 \text{ and } x_0) \oplus (x_2 \text{ and } x_0) \oplus (x_3 \text{ and } x_0) \oplus (x_4 \text{ and } x_0) \oplus (x_6 \text{ and } x_0) \oplus (x_7 \text{ and } x_0) \oplus (x_5 \text{ and } x_2) \oplus (x_5 \text{ and } x_3) \oplus (x_2 \text{ and } x_4) \oplus (x_3 \text{ and } x_4) \oplus (x_5 \text{ and } x_7) \oplus (x_7 \text{ and } x_2) \oplus (x_5 \text{ and } x_6) \oplus (x_3 \text{ and } x_2) \oplus (x_7 \text{ and } x_3)$$

$$\mathbf{c}_1 = (x_2 \text{ and } x_1) \oplus (x_2 \text{ and } x_4) \oplus (x_5 \text{ and } x_4) \oplus (x_3 \text{ and } x_6) \oplus (x_5 \text{ and } x_6) \oplus (x_2 \text{ and } x_0) \oplus (x_3 \text{ and } x_0) \oplus (x_5 \text{ and } x_0) \oplus (x_7 \text{ and } x_0) \oplus x_1 \oplus (x_5 \text{ and } x_2) \oplus (x_7 \text{ and } x_2) \oplus (x_5 \oplus x_3) \oplus (x_5 \text{ and } x_7) \oplus x_7 \oplus x_2 \oplus (x_3 \text{ and } x_2) \oplus x_4 \oplus x_5$$

$$\mathbf{c}_0 = (x_1 \text{ and } x_0) \oplus (x_2 \text{ and } x_0) \oplus (x_3 \text{ and } x_0) \oplus (x_5 \text{ and } x_0) \oplus (x_7 \text{ and } x_0) \oplus (x_3 \text{ and } x_1) \oplus (x_6 \text{ and } x_1) \oplus (x_3 \text{ and } x_6) \oplus (x_5 \text{ and } x_6) \oplus (x_7 \text{ and } x_6) \oplus (x_3 \text{ and } x_4) \oplus (x_7 \text{ and } x_4) \oplus (x_5 \text{ and } x_3) \oplus (x_4 \text{ and } x_1) \oplus x_2 \oplus (x_3 \text{ and } x_2) \oplus (x_4 \text{ and } x_6) \oplus x_6 \oplus x_5 \oplus x_3 \oplus x_0$$

B.2 Stage 2 of PPRM S-Box

The combinational logics shown in Fig. B.5(a) are designed based on the following Boolean function of the d_0 – d_3 (labeled as ver.1). There are 25-AND gates and 21-XOR gates in ver.1. Input signals for internal wires d_0 – d_3 are from the output of c_0 – c_3 that the author has presented in Stage 1 of the PPRM S-Box. The combinational logic diagram of ver.2 is shown in Fig. B.5(b), where the AND gates are reduced from 25 to 9 gates.

$$\mathbf{d}_3 = (c_3 \text{ and } c_2 \text{ and } c_1) \oplus (c_3 \text{ and } c_0) \oplus c_3 \oplus c_2$$

$$\mathbf{d}_2 = (c_3 \text{ and } c_2 \text{ and } c_0) \oplus (c_3 \text{ and } c_0) \oplus (c_3 \text{ and } c_2 \text{ and } c_1) \oplus (c_2 \text{ and } c_1) \oplus c_2$$

$$\mathbf{d}_1 = (c_3 \text{ and } c_2 \text{ and } c_1) \oplus (c_3 \text{ and } c_1 \text{ and } c_0) \oplus c_3 \oplus (c_2 \text{ and } c_0) \oplus c_2 \oplus c_1$$

$$\mathbf{d}_0 = (c_3 \text{ and } c_2 \text{ and } c_0) \oplus (c_3 \text{ and } c_1 \text{ and } c_0) \oplus (c_3 \text{ and } c_2 \text{ and } c_1) \oplus (c_3 \text{ and } c_1) \oplus (c_3 \text{ and } c_0) \oplus (c_2 \text{ and } c_1 \text{ and } c_0) \oplus c_2 \oplus (c_2 \text{ and } c_1) \oplus c_1 \oplus c_0$$

B.3 Stage 3 of PPRM S-Box

The combinational logics shown in Figs .B.6 and B.7 are designed based on the following Boolean function of y_0 – y_7 , which is same as the one in the appendix of

reference [102].

ver.1 :

$$\mathbf{y_7} = (\text{d3 and a0}) \oplus (\text{d2 and a1}) \oplus (\text{d1 and a2}) \oplus (\text{d0 and a3}) \oplus (\text{b2 and d3}) \oplus (\text{b3 and d2}) \oplus (\text{b2 and d2}) \oplus (\text{d3 and a3}) \oplus (\text{d3 and a1}) \oplus (\text{d1 and a3}) \oplus (\text{b0 and d2}) \oplus (\text{b2 and d0}) \oplus (\text{d3 and a2}) \oplus (\text{d2 and a3}) \oplus (\text{b0 and d3}) \oplus (\text{b1 and d2}) \oplus (\text{b2 and d1}) \oplus (\text{b3 and d0})$$

$$\mathbf{y_6} = '1' \oplus (\text{a0 and d2}) \oplus (\text{a2 and d0}) \oplus (\text{d3 and a3}) \oplus (\text{a0 and d1}) \oplus (\text{a1 and d0}) \oplus (\text{d3 and a2}) \oplus (\text{d2 and a3}) \oplus (\text{a0 and d0}) \oplus (\text{d3 and a0}) \oplus (\text{d2 and a1}) \oplus (\text{d1 and a2}) \oplus (\text{d0 and a3})$$

$$\mathbf{y_5} = '1' \oplus (\text{d3 and a3}) \oplus (\text{d3 and a1}) \oplus (\text{d1 and a3}) \oplus (\text{d3 and a2}) \oplus (\text{d2 and a3}) \oplus (\text{b2 and d2}) \oplus (\text{b0 and d2}) \oplus (\text{b2 and d0}) \oplus (\text{b3 and d3}) \oplus (\text{b1 and d3}) \oplus (\text{b3 and d1}) \oplus (\text{d3 and a0}) \oplus (\text{d2 and a1}) \oplus (\text{d1 and a2}) \oplus (\text{d0 and a3})$$

$$\mathbf{y_4} = (\text{d3 and a1}) \oplus (\text{d1 and a3}) \oplus (\text{a0 and d0}) \oplus (\text{b3 and d3}) \oplus (\text{b0 and d1}) \oplus (\text{b1 and d0}) \oplus (\text{d3 and a0}) \oplus (\text{d2 and a1}) \oplus (\text{d1 and a2}) \oplus (\text{d0 and a3}) \oplus (\text{a1 and d1}) \oplus (\text{b2 and d2}) \oplus (\text{b0 and d0})$$

$$\mathbf{y_3} = (\text{b0 and d1}) \oplus (\text{b1 and d0}) \oplus (\text{b0 and d2}) \oplus (\text{b2 and d0}) \oplus (\text{b1 and d3}) \oplus (\text{b3 and d1}) \oplus (\text{b0 and d0})$$

$$\mathbf{y_2} = (\text{a0 and d2}) \oplus (\text{a2 and d0}) \oplus (\text{a0 and d1}) \oplus (\text{a1 and d0}) \oplus (\text{b1 and d1}) \oplus (\text{b2 and d2}) \oplus (\text{d3 and a1}) \oplus (\text{d1 and a3}) \oplus (\text{b0 and d2}) \oplus (\text{b2 and d0}) \oplus (\text{b3 and d3}) \oplus (\text{a0 and d0}) \oplus (\text{b0 and d3}) \oplus (\text{b1 and d2}) \oplus (\text{b2 and d1}) \oplus (\text{b3 and d0}) \oplus (\text{b0 and d0})$$

$$\mathbf{y_1} = '1' \oplus (\text{d3 and a0}) \oplus (\text{d2 and a1}) \oplus (\text{d1 and a2}) \oplus (\text{d0 and a3}) \oplus (\text{b1 and d1}) \oplus (\text{b2 and d3}) \oplus (\text{b3 and d2}) \oplus (\text{d3 and a3}) \oplus (\text{d3 and a1}) \oplus (\text{d1 and a3}) \oplus (\text{b3 and d3}) \oplus (\text{d3 and a2}) \oplus (\text{d2 and a3}) \oplus (\text{b0 and d0})$$

$$\mathbf{y_0} = '1' \oplus (\text{d3 and a0}) \oplus (\text{d2 and a1}) \oplus (\text{d1 and a2}) \oplus (\text{d0 and a3}) \oplus (\text{a0 and d2}) \oplus (\text{a2 and d0}) \oplus (\text{b0 and d1}) \oplus (\text{b1 and d0}) \oplus (\text{d2 and a2}) \oplus (\text{b0 and d2}) \oplus (\text{b2 and d0}) \oplus (\text{b1 and d3}) \oplus (\text{b3 and d1}) \oplus (\text{d3 and a2}) \oplus (\text{d2 and a3}) \oplus (\text{b0 and d0})$$

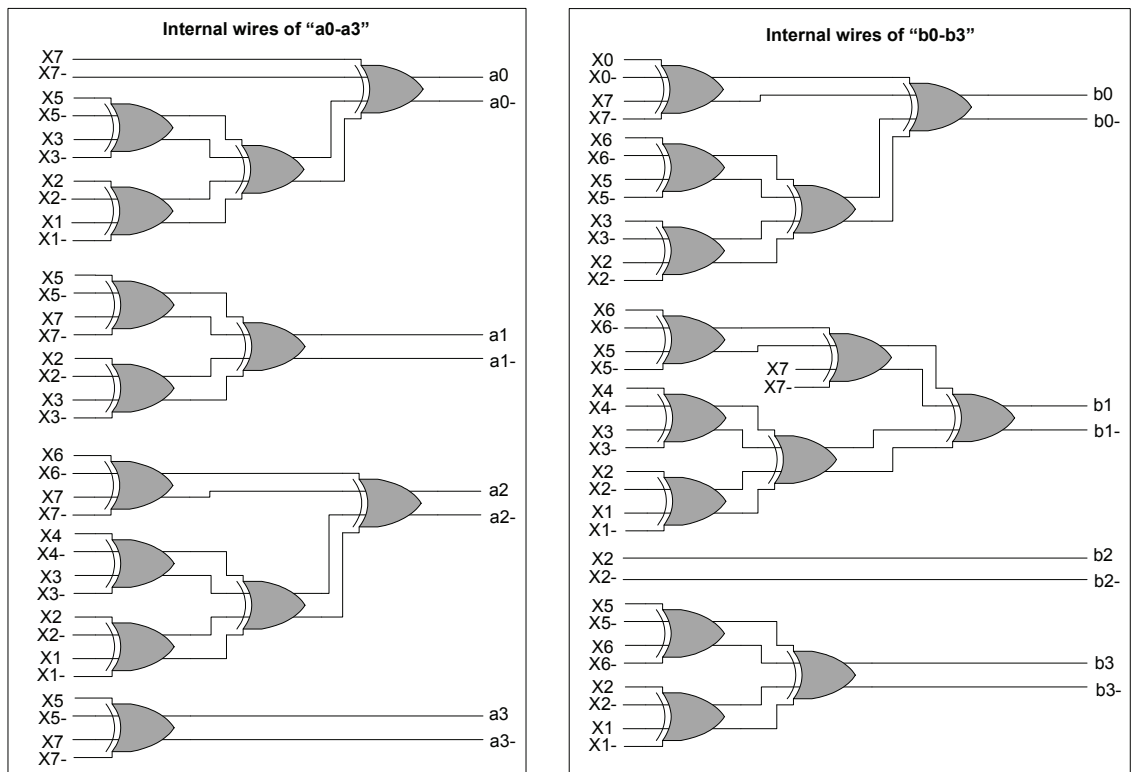


Figure B.1: Combinational logic of internal wires of a_0 – a_3 and b_0 – b_3 in appendix of [102].

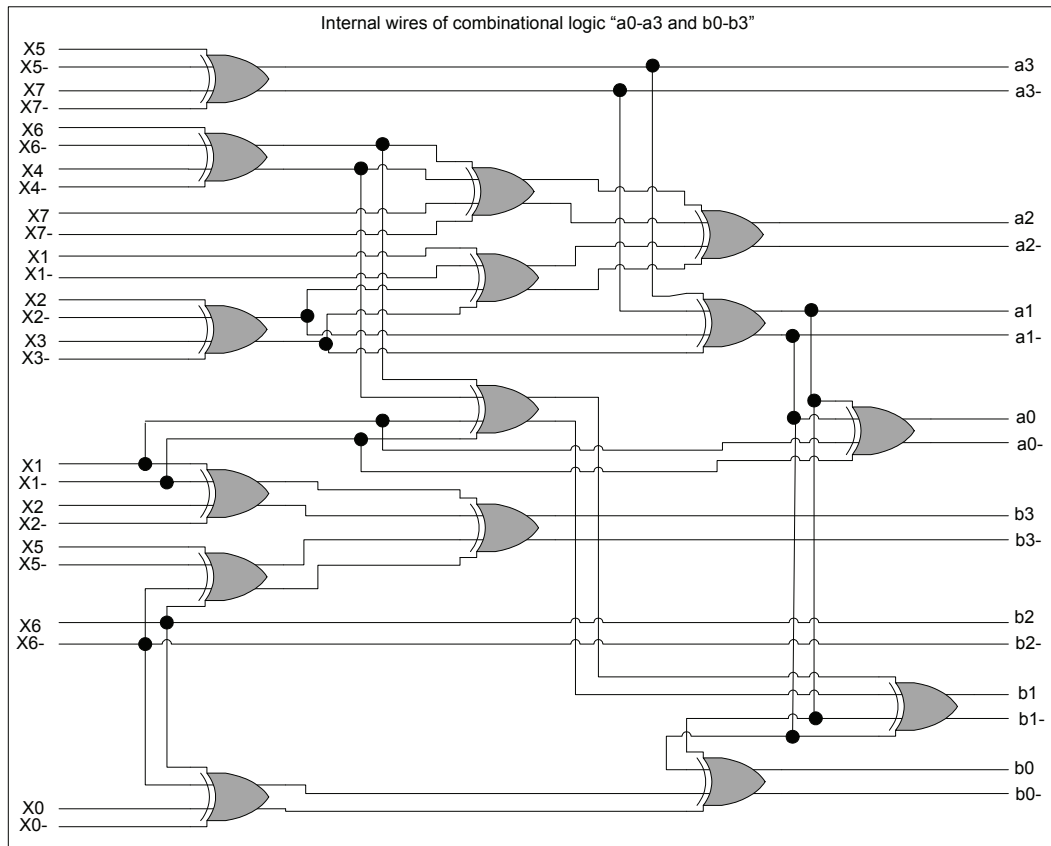
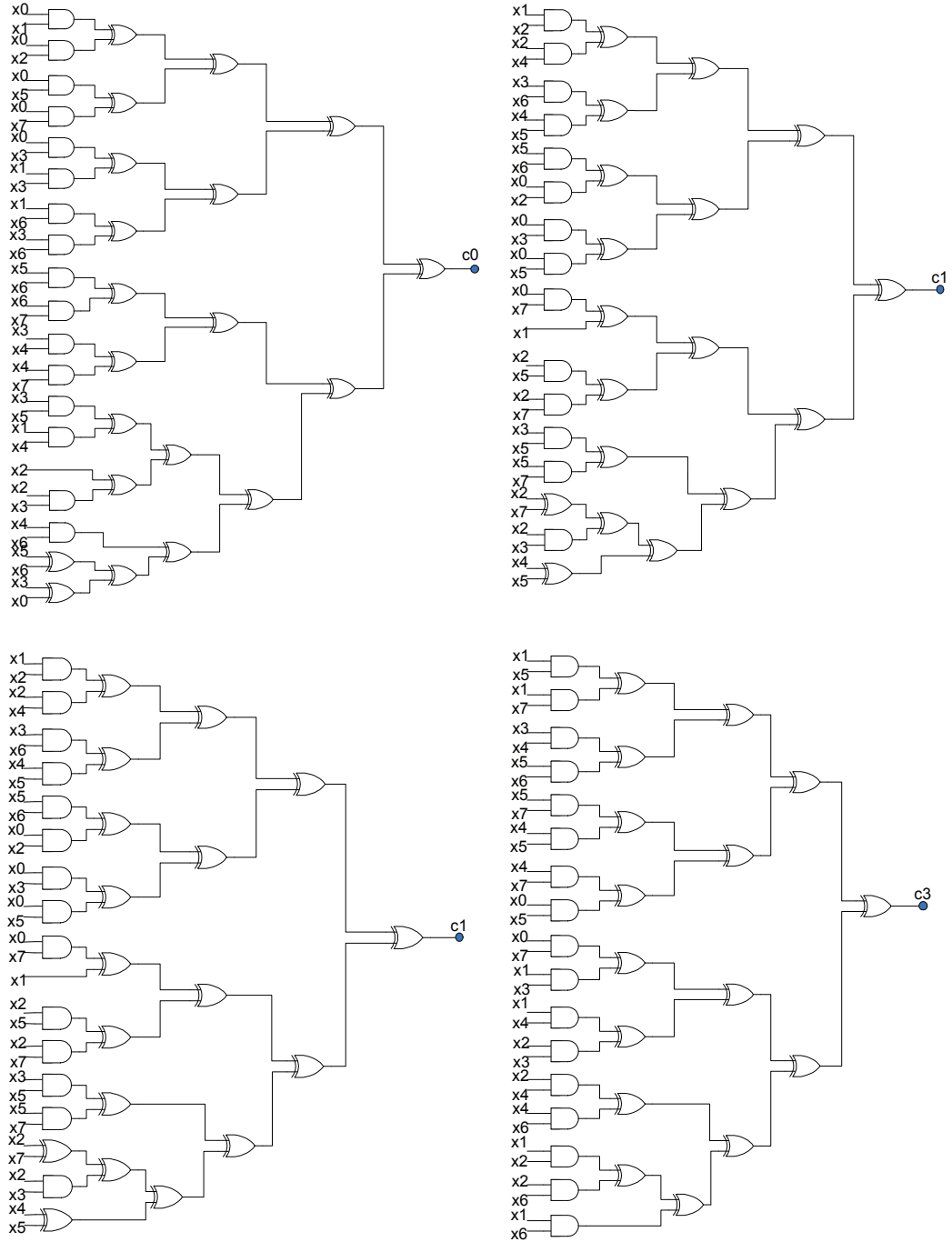


Figure B.2: Optimized combinational logic of internal wires of a_0 – a_3 and b_0 – b_3 using logic sharing technique in this work.

Figure B.3: Combinational logic of internal wires of c_0 – c_3 in appendix of [102].

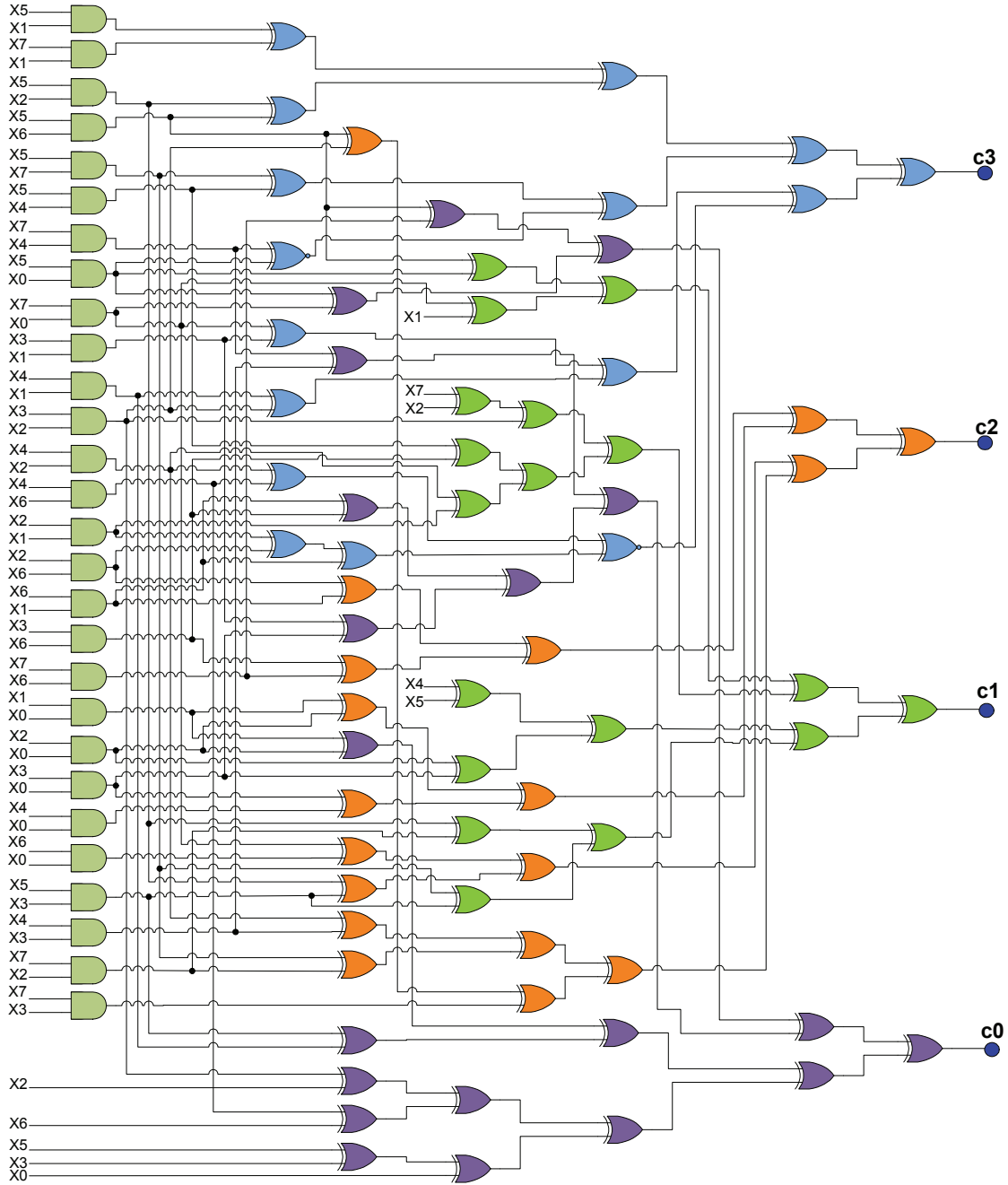


Figure B.4: Optimized combinational logic of internal wires of c_0 – c_3 using logic sharing technique in this work.

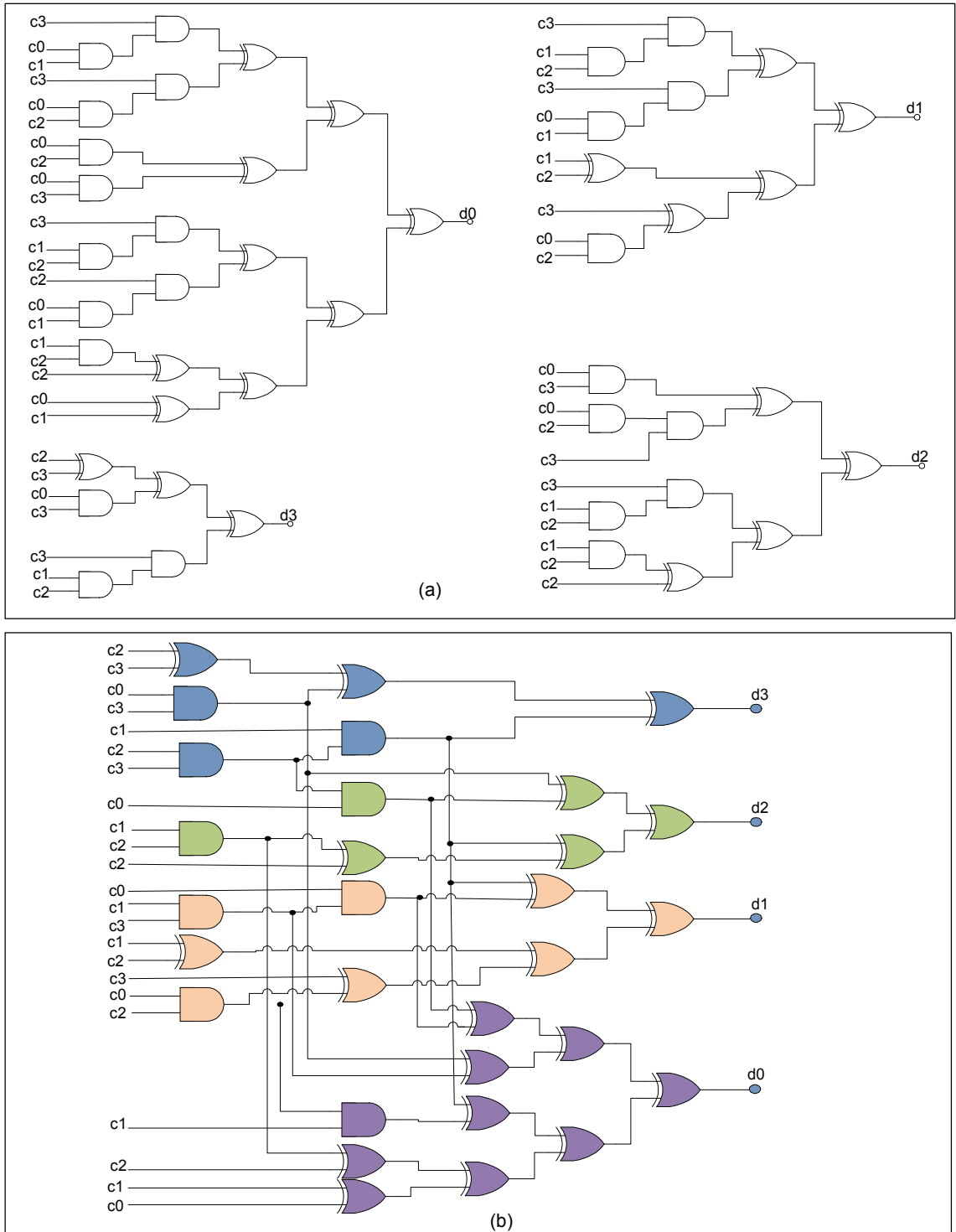
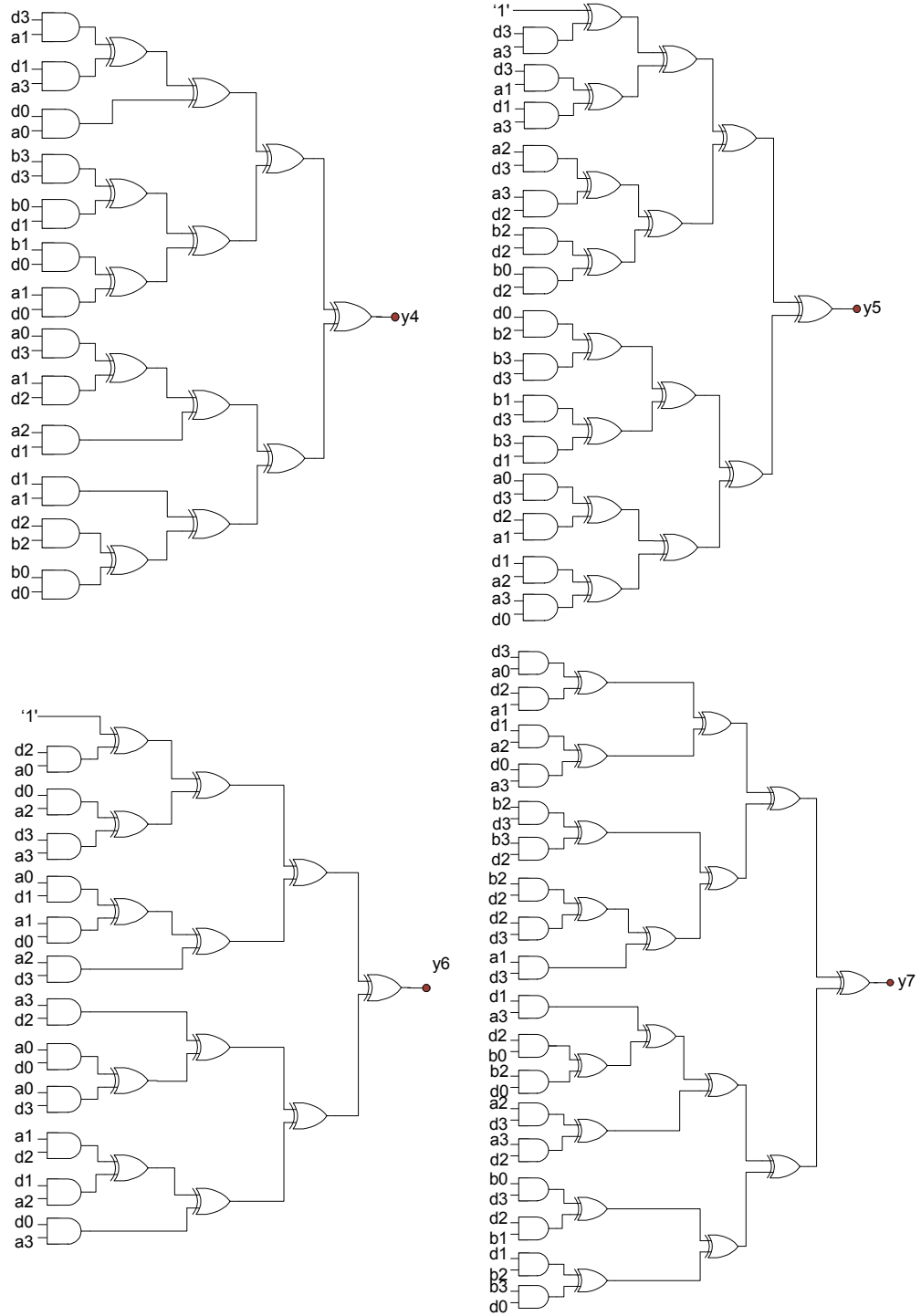


Figure B.5: (a) Internal wires of d_0-d_3 in [102] and (b) Optimized combinational logic of internal wires of d_0-d_3 using logic sharing technique.



Figure B.7: Combinational logic of output $y_4 - y_7$.

Appendix C

Trapezoidal Power Clock Generator

In this chapter, the author analyzes and re-simulates the three-phase trapezoidal power clock generator that was proposed in [113]. The circuit schematic diagram is shown in Fig. C.1. The circuit was re-simulated using transistor size of $W/L = 0.6 \mu\text{m}/0.18 \mu\text{m}$ in LTspice simulator. As shown in this circuit diagram that control signal for rise and fall time is produced by LC resonant oscillator, in which by given a desired frequency f , the required inductance L can be obtained using equation as follows:

$$3f = \frac{1}{2\pi\sqrt{LC_{eq}}}, \quad (\text{C.0.1})$$

where C_{eq} is the equivalent capacitance at the clock node. The reference frequency is three time than the generated power clock frequency. The parameters of the simulation are tabulated in Table C.1.

Table C.1: Parameters for elements in trapezoidal power clock generator circuit.

Parameters' values	
L	45 μH
Capacitance C_{E1}, C_{E2}, C_{E3}	0.8 pF
PMOSs' W/L	0.6 $\mu\text{m}/0.18 \mu\text{m}$
MMOSs' W/L	0.6 $\mu\text{m}/0.18 \mu\text{m}$
Vdd	1.8 V
$f_{reference}$	25 MHz
Generated clock Freq.	8.33 MHz

Input control signals and the generated power clock signals are depicted in Fig. C.2. Although the three-phase power clock signals are correctly generated,

the author still needs to do a few modification for phase delay and the phase ratio setting, so that it can be supplied for CSSAL S-box circuit which requires triple power clock signals.

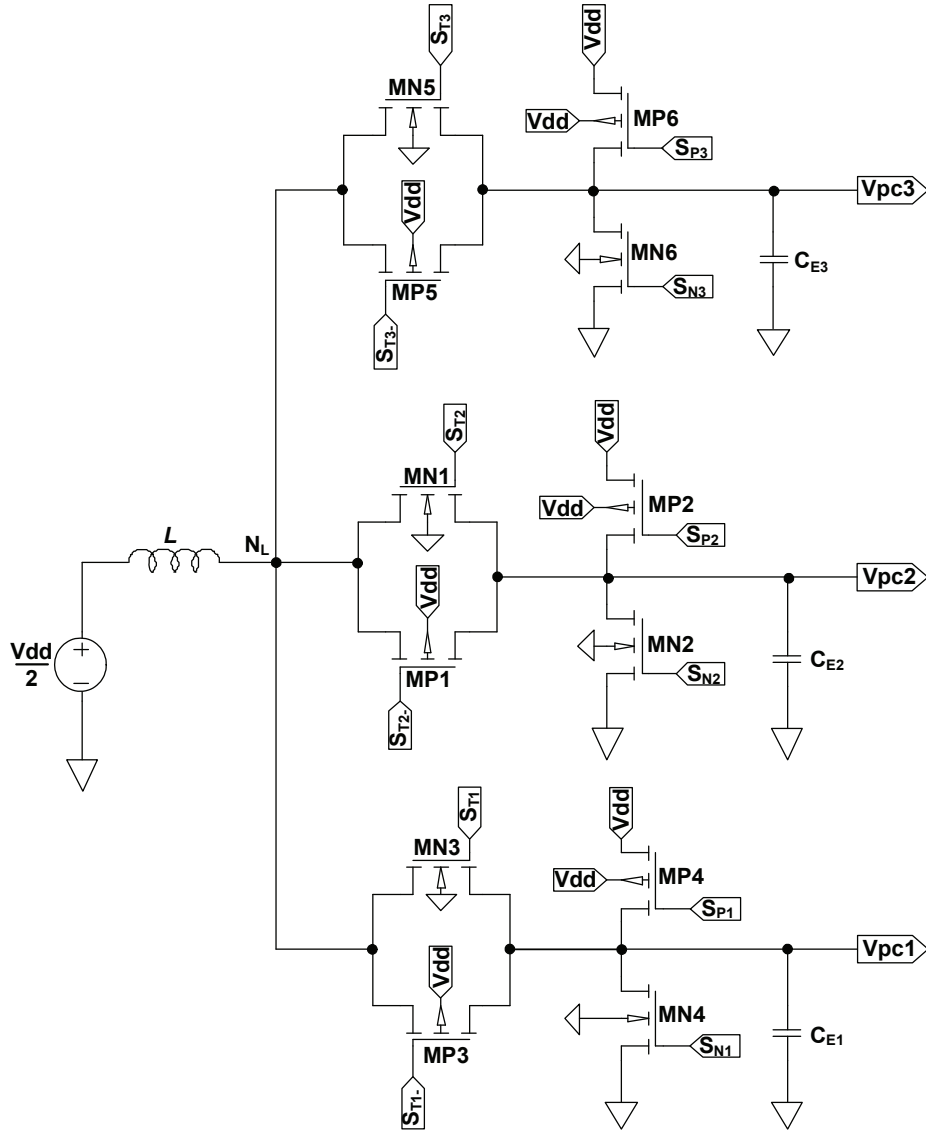


Figure C.1: Schematic diagram of the three-phase trapezoidal power clock generator circuit.

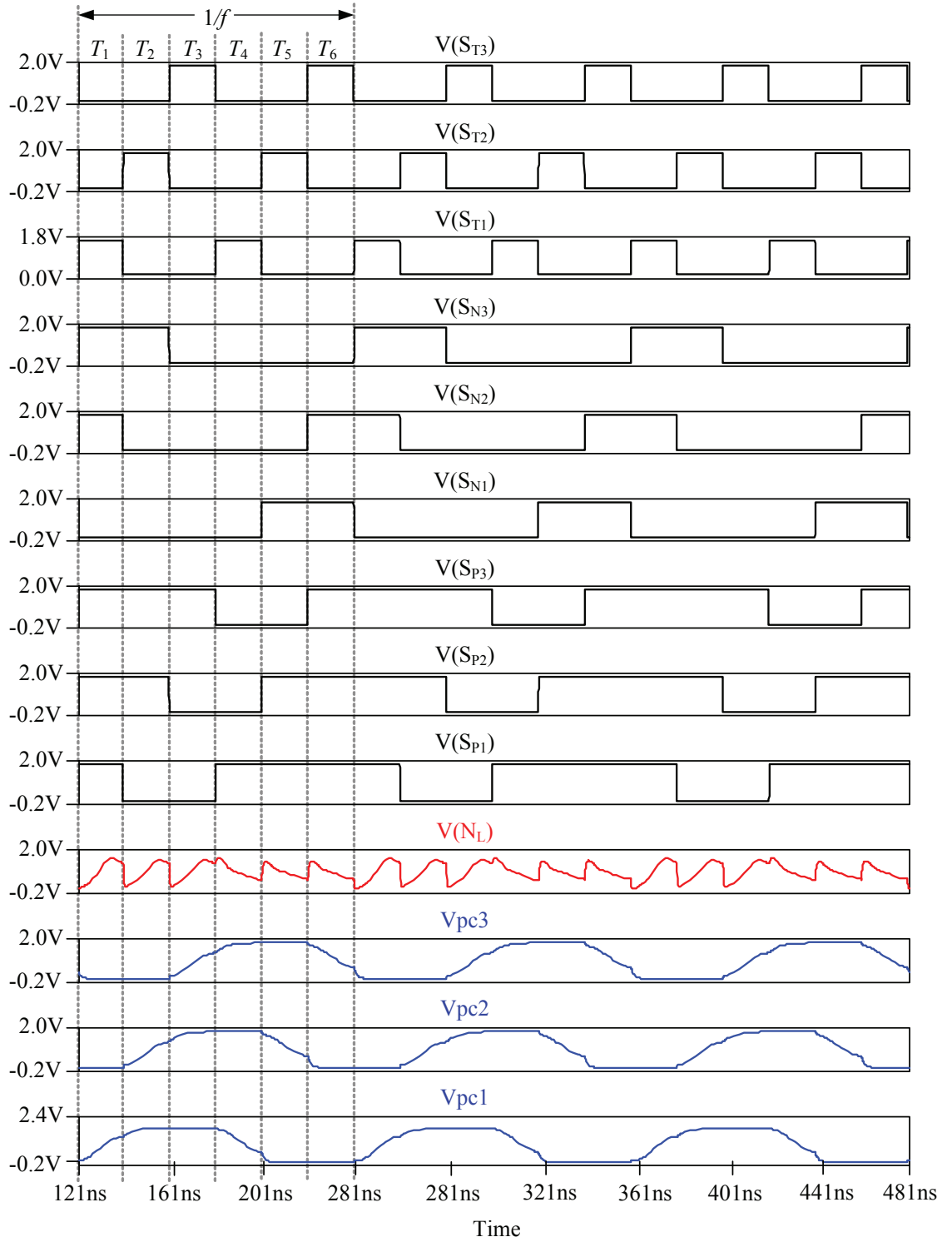


Figure C.2: Input-output signals of the three-phase trapezoidal power clock generator.

Index

- Δf , 33, 35
- Δt , 32
- 2N-2N2P AND/NAND, 74, 77
- 2N-2N2P inverter, 74
- S-box LSI photomicrograph, 183
- adiabatic, 23, 69
- ASIC, 5
- asymmetric algorithm, 6
- average energy \overline{E} , 31
- average energy \overline{E} , 106, 130, 143, 177
- back-annotation, 29, 164
- Bit Parallel Cellular Multiplier, 108
- bit parallel multiplier, 107, 110, 114
- ciphertext, 6
- CMOS parameters, 221
- CMOS power, 16, 19, 143
- CPA, 42
- cryptanalysis, 1, 6, 8
- cryptography, 1
- CSSAL inverter, 83
- CSSAL NAND/AND, 89, 97
- CSSAL OR/NOR, 96
- CSSAL XOR/XNOR, 94
- current transition, 136
- current transition of inverter, 89
- DCVSL, 49
- DCVSL logic, 50
- DEMA, 11
- DEMA, 9, 44
- DPA, 11
- DPA, 9, 39, 41
- dual-input current transitions, 101, 102
- dynamic Logic, 54
- dynamic power, 16, 19
- ECRL AND/NAND, 71
- ECRL inverter, 70
- energy variance σ_E , 130, 143
- Enigma, 2
- energy variance σ_E , 31
- energy variance σ_E , 106, 177
- FFT, 32, 130
- Fault attacks, 9
- FPGA, 5
- full-custom, 165
- f_0 , 130, 162
- glitches, 22, 97
- Hamming-distance, 11, 26, 49
- Hamming-weight, 27, 43
- hiding countermeasure, 48
- hieroglyphics, 2
- inner cell, 113
- input transition, 50
- interface, 5
- Invasive attacks, 9
- Kerckhoffs Principle, 10

- layout, 164, 165, 170
- LPA, 43
- LSI photomicrograph, 183
- LSI summary, 202

- majority gate, 66
- masking countermeasure, 65
- MDPL, 66
- measurement, 183, 188, 194
- memory, 5
- multiplier LSI, 183

- Non-invasive attacks, 10
- NED, 30, 106, 130, 143, 177
- NSD, 31, 106, 130, 136, 143, 177, 221

- plaintext, 6
- PLCC, 4
- post-layout, 164, 171, 177
- power clock generator, 239
- pre-charged logic, 56
- Probing Attacks, 10
- process variation, 137, 143, 221

- RC model of CSSAL inverter, 89
- RFID, 4

- S-box LSI, 183
- SABL, 56
- SAL AND/NAND, 78
- SAL inveter, 77
- SCA, 8, 37
- scCMOS, 48
- scCMOS logic, 50
- Semi-invasive attacks, 9
- shift cipher, 2
- short-circuit power, 19
- smart card, 3
- SPA, 40
- standard deviation σ_E , 31
- static power, 20
- SyAL AND/NAND, 80
- SyAL inverter, 80
- symmetric algorithm, 6

- TDPL, 61
- timing attack, 9, 38

- USB token, 3