รายการอ้างอิง

[1] Davis, A. ,and Nowick, S.M. An Introduction to Asynchronous Circuit Design. Department of Computer Science University of Utah Technical report, September 1997.

[2] Sparsø, J. ,and Furber, S. Principles of asynchronous circuit design – A systems perspective. pp.9-14,87. Kluwer Academic Publishers, 2001.

[3] Verhoeff, T. Delay-insensitive Codes-an Overview. Department of Mathematics and Computing Science Eindhoven University of Technology, Springer Berlin / Heidelberg, 1987.

[4] Sufian, S. A Design of System Bus for Asynchronous Circuits. Master's thesis, Department of Computer Engineering Faculty of Engineering Chulalongkorn University, 2006.

[5] Hauck, S. Asynchronous Design Methodologies: An Overview. Proceedings of the IEEE, pp. 69-93 Vol. 83 No. 1. January 1995.

[6] Ginosar, R. Asynchronous Design and Synchronization. [Online]. 2009. Available from: http://webee.technion.ac.il/courses/048878.htm [2011,March 8]

[7] Bainbridge, W.J. ,and Furber, S.B. Delay Insensitive System-on-Chip Interconnect using 1-of-4 Data Encoding. Proceedings Async 2001 IEEE Computer Society Press , March 2001.

[8] Takamura, A., Kuwako, M., Ueno, Y., Kagotani, H. ,and Nanya, T. TITAC : Design of a Quasi-Delay-Insensitive Microprocessor. IEEE Design & Test of Computers, pp. 58-59,61. Summer 1994.

[9] Miller, R.E. Combinational Circuit. IEEE Transactions on Computers, Volume 1 of Switching Theory. 1965.

[10] Godse, A. P. ,and Godse, D.A. Microprocessor – I. First Edition chapter 8 pp.2,10. Technical Publications, 2008.

[11] Stallings, W. Computer organization and architecture: designing for performance. Seventh edition pp.223. Prentice Hall, 2006.

[12] Lewis, M.G. Lower Power Asynchronous Digital Signal Processing. Doctoral dissertation, Department of Computer Science Faculty of Engineering and Physical Sciences University of Manchester, 2000.

[13] Xilinx, Inc. Application Note: Spartan-3 FPGA Family. Using Block RAM in Spartan-3 Generation FPGAs. [Online]. 2005. Available from: http:// www.xilinx.com [2011, March 8]

[14] Bainbridge, W.J. ,and Furber, S.B. Asynchronous Macrocell Interconnect using MARBLE. Proc. Async'98 San Diego, pp. 122-132. April 1998.

[15] Plana, L.A., Riocreux, P.A., Bainbridge, W.J., Bardsley, A., Garside, J.D. ,and Temple, S. SPA - A Synthesisable Amulet Core for Smartcard Applications. Proceedings of Async'2002 Manchester, pp. 201-210. April 2002.

[16] Takamura, A. and others. TITAC–2 : An asynchronous 32-bit microprocessor based on Scalable-Delay-Insensitive model. In Proc. International Conf. Computer Design (ICCD'97), pp. 288–294. MIT Press, October 1997.

[17] Ruangsinsup, P. Design of 8-bit scalable-delay-Insensitive microprocessor using FPGA. Master's thesis, Department of Computer Engineering Faculty of Engineering Chulalongkorn University, 2001.

[18] Cortadella, J. The Department of Computer Architecture (DAC) Universitat Politècnica de Catalunya Spain. Petrify : a tutorial for the designer of asynchronous circuits. [Online]. (no year given). Available from: http://www.cs.unc.edu/~montek/teaching/spring-04.htm [2011, March 8]

[19] Mangino, J. Texas Instruments Incorporated. Using DMA with High Performance Peripherals to Maximize System Performance. [Online]. 2007. Available from: http://focus.ti.com/lit/wp/spna105/spna105.pdf [2011, March 8]

[20] Saxena, N.R. ,and Robinson, J.P. Syndrome and Transition Count are Uncorrelated. IEEE Transactions on Information Theory archive, IEEE Press Piscataway NJ USA, pp. 64. 1988.

[21] Xilinx, Inc. ISE Web pack 11.1. Xilinx Inc. : Xilinx Inc. 2009.

[22] Xilinx, Inc. ModelSim XE 6.4b. Xilinx Inc. : Xilinx Inc. 2009.

[23] Xilinx, Inc. Spartan-3 Generation FPGA User Guide. Package Marking. [Online]. 2009. Available from: http:// www.xilinx.com [2011, March 8]

[24] Xilinx, Inc. XA Spartan-3E Automotive FPGA Family Data Sheet. Architectural Overview. [Online]. 2009. Available from: http:// www.xilinx.com [2011, March 8]

[25] Petrov, P. Project II: Implementation of a Booth Multiplier. ENEE 359V: Advanced Digital Design with HDLs, Department of Computer Science University of Marryland, pp.1-3, Fall 2009.

[26] Hennessy, J.L. ,and Patterson, D.A. Computer Architecture : A Quantitative approach, pp.700-701. Morgan Kaifmann Publishers is an Imprint of Elsevier, 2007.

ภาคผนวก

ภาคผนวก ก

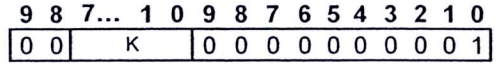
ชุดคำสั่งและรหัสดำเนินการ

Description	Mnemonic code

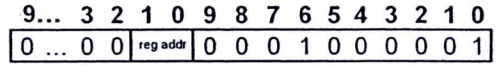
DATA TRANSFER

1 **LD** = Load

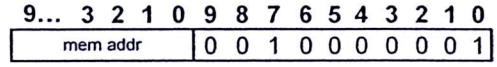
A, #K
(A <- K)



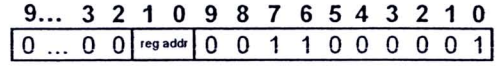
A, Reg
(A <- Reg)



A, @K
(A <- Mem[K])

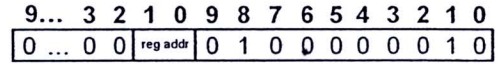


A, @Reg
(A <- Mem[Reg])

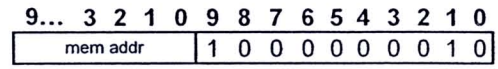


2 **ST** = Store

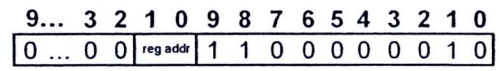
A, Reg
(Reg <- A)



A, @K
(Mem[K] <- A)

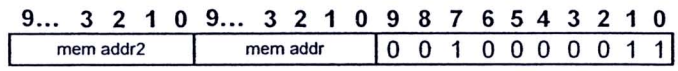


A, @Reg
(Mem[Reg] <- A)

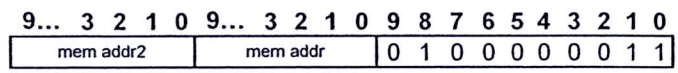


3 **IN** = Input from

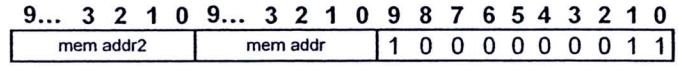
@K2, @K
(Mem2[K2] <- Mem[K])



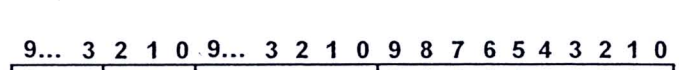
@K2, @K
(Mem2[K2] <- Mem[K];
Mem2[K2+1] <- Mem[K+1])



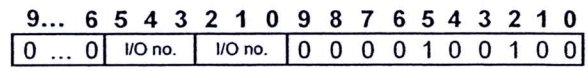
@K2, @K
(Mem2[K2] <- Mem[K];
Mem2[K2+1] <- Mem[K+1];
Mem2[K2+2] <- Mem[K+2])



I/O,@K
(I/O <- Mem[K])



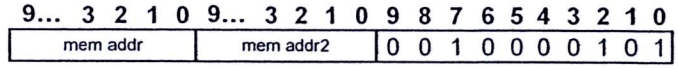
I/O,I/O
(I/O <- I/O)



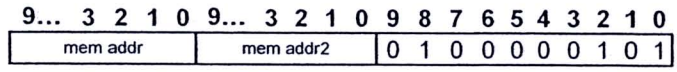
Description	Mnemonic code

4 **OUT** = Output from

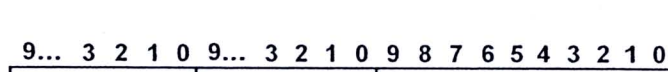
@K, @K2
(Mem[K] <- Mem2[K2])



@K, @K2
(Mem[K] <- Mem2[K2];
Mem[K+1] <- Mem2[K2+1])



@K, @K2
(Mem[K] <- Mem2[K2];
Mem[K+1] <- Mem2[K2+1];
Mem[K+2] <- Mem2[K2+2])

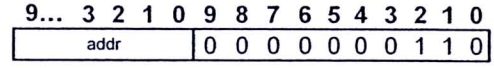


I/O,@K
(Mem[K] <- I/O)

PROGRAM AND MACHINE CONTROL

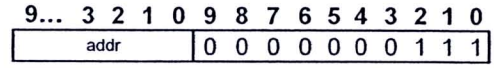
5 **JMP** = Unconditional Jump

Address
(PC <- Address)



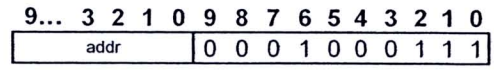
6 **JZ** = Jump on Zero

Address
(PC <- Address if Z = 1)



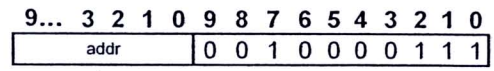
7 **JNZ** = Jump on Not Zero

Address
(PC <- Address S if Z = 0)



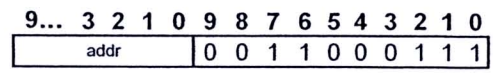
8 **JC** = Jump on Carry

Address
(PC <- Address if C = 1)



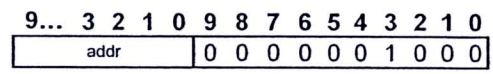
9 **JNC** = Jump on Not Carry

Address
(PC <- Address if C = 0)



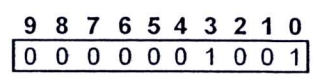
10 **CALL** = Call

Address
(SP <- PC; PC <- Address)



11 **RET** = Return

(PC <- SP)



Description	Mnemonic code

12 **RETI** = Return from Interrupt

9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	1	0	1	1

(PC <- SP)

ARITHMETIC

13 **ADD** = Add

A, #K
(A <- A + K)

9	8	7...	1	0	9	8	7	6	5	4	3	2	1	0
0	0	K			0	0	0	0	0	0	1	1	0	0

A, Reg
(A <- A + Reg)

9...	3	2	1	0	9	8	7	6	5	4	3	2	1	0
0 ...	0	0	reg addr		0	0	0	1	0	0	1	1	0	0

A, @K
(A <- A + Mem[K])

9...	3	2	1	0	9	8	7	6	5	4	3	2	1	0
mem addr					0	0	1	0	0	0	1	1	0	0

A, @Reg
(A <- A + Mem[Reg])

9...	3	2	1	0	9	8	7	6	5	4	3	2	1	0
0 ...	0	0	reg addr		0	0	1	1	0	0	1	1	0	0

14 **SUB** = Substract

A, #K
(A <- A – K)

9	8	7...	1	0	9	8	7	6	5	4	3	2	1	0
0	0	K			0	0	0	0	0	0	1	1	0	1

A, Reg
(A <- A – Reg)

9...	3	2	1	0	9	8	7	6	5	4	3	2	1	0
0 ...	0	0	reg addr		0	0	0	1	0	0	1	1	0	1

A, @K
(A <- A - Mem[K])

9...	3	2	1	0	9	8	7	6	5	4	3	2	1	0
mem addr					0	0	1	0	0	0	1	1	0	1

A, @Reg
(A <- A - Mem[Reg])

9...	3	2	1	0	9	8	7	6	5	4	3	2	1	0
0 ...	0	0	reg addr		0	0	1	1	0	0	1	1	0	1

15 **INC** = Increment

A, #K
(A <- K +1)

9	8	7...	1	0	9	8	7	6	5	4	3	2	1	0
0	0	K			0	0	0	0	0	0	1	1	1	0

A, Reg
(A <- Reg + 1)

9...	3	2	1	0	9	8	7	6	5	4	3	2	1	0
0 ...	0	0	reg addr		0	0	0	1	0	0	1	1	1	0

A, @K
(A <- Mem[K] + 1)

9...	3	2	1	0	9	8	7	6	5	4	3	2	1	0
mem addr					0	0	1	0	0	0	1	1	1	0

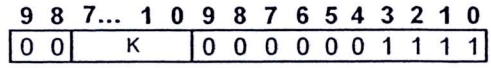
A, @Reg
(A <- Mem[Reg] + 1)

9...	3	2	1	0	9	8	7	6	5	4	3	2	1	0
0 ...	0	0	reg addr		0	0	1	1	0	0	1	1	1	0

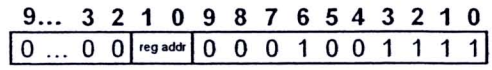
Description	Mnemonic code

16 **DEC** = Decrement

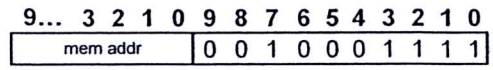
A, #K
(A <- K -1)



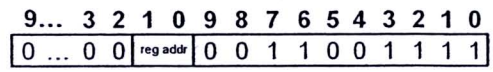
A, Reg
(A <- Reg - 1)



A, @K
(A <- Mem[K] - 1)

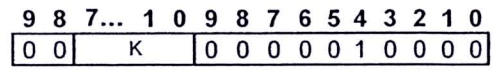


A, @Reg
(A <- Mem[Reg] - 1)

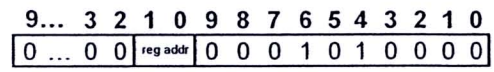


17 **MUL** = Multiplication

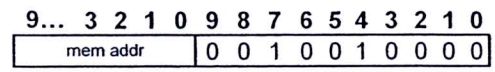
A, #K
(A <- A*K)



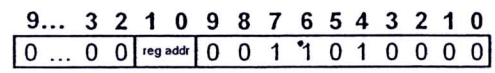
A, Reg
(A <- A*Reg)



A, @K
(A <- A*Mem[K])



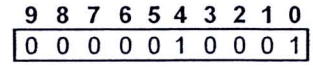
A, @Reg
(A <- A*Mem[Reg])



LOGIC

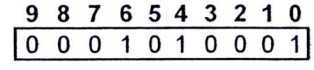
18 **SLL** = Shift Logical Left

A
(A <- A[MSB-1..0] & '0')



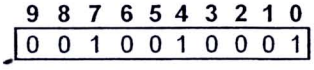
19 **SLR** = Shift Logical Right

A
(A <- '0' & A[MSB..1])



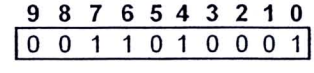
20 **SAL** = Shift Arithmetic Left

A
(A <- A[MSB-1..0] & 'A[LSB]')



21 **SAR** = Shift Arithmetic Right

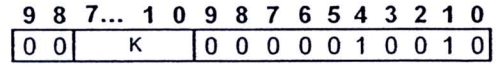
A
(A <- 'A[MSB]' & A[MSB..1])



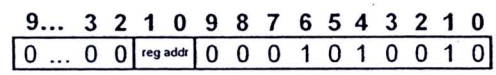
Description	Mnemonic code

22 **AND** = Logical And

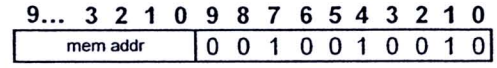
A, #K
(A <- A AND K)



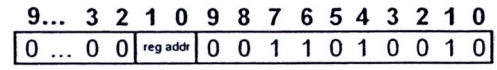
A, Reg
(A <- A AND Reg)



A, @K
(A <- A AND Mem[K])

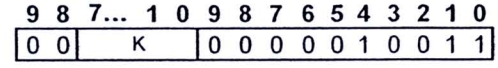


A, @Reg
(A <- A AND Mem[Reg])

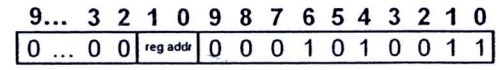


23 **OR** = Logical Or

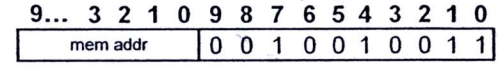
A, #K
(A <- A OR K)



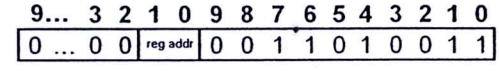
A, Reg
(A <- A OR Reg)



A, @K
(A <- A OR Mem[K])

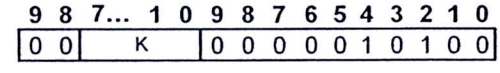


A, @Reg
(A <- A OR Mem[Reg])

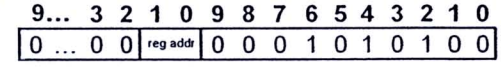


24 **XOR** = Logical Exclusive or

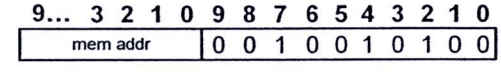
A, #K
(A <- A XOR K)



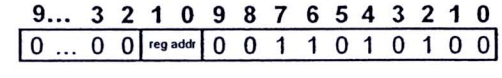
A, Reg
(A <- A XOR Reg)



A, @K
(A <- A XOR Mem[K])

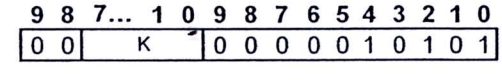


A, @Reg
(A <- A XOR Mem[Reg])

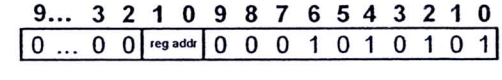


25 **ANDB** = Bitwise And

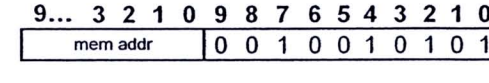
A, #K
(A <- A AND K)



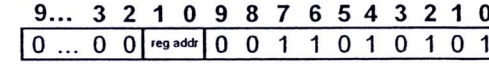
A, Reg
(A <- A AND Reg)

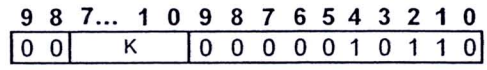
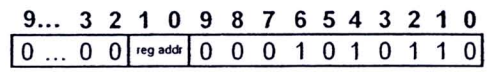
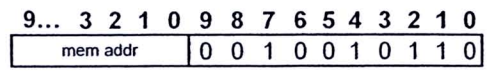
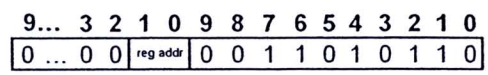
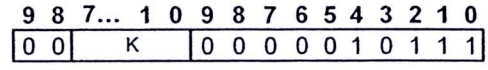
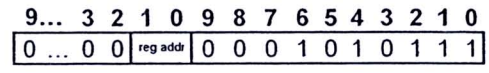
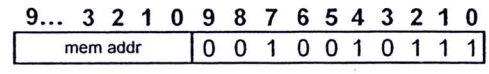
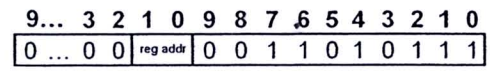


A, @K
(A <- A AND Mem[K])



A, @Reg
(A <- A AND Mem[Reg])



Description	Mnemonic code
26 **ORB** = Bitwise Or	
A, #K (A <- A OR K)	
A, Reg (A <- A OR Reg)	
A, @K (A <- A OR Mem[K])	
A, @Reg (A <- A OR Mem[Reg])	
27 **XORB** = Bitwise Exclusive or	
A, #K (A <- A XOR K)	
A, Reg (A <- A XOR Reg)	
A, @K (A <- A XOR Mem[K])	
A, @Reg (A <- A XOR Mem[Reg])	

หมายเหตุ: ความหมายของสัญลักษณ์มีดังนี้

A หมายถึง รีจิสเตอร์ตัวสะสม

K หมายถึง ค่าคงที่

Reg หมายถึง รีจิสเตอร์ทั่วไป ('01' คือรีจิสเตอร์ R0, '10' คือรีจิสเตอร์ R1)

Address หมายถึง ตำแหน่งอ้างอิงในหน่วยความจำสำหรับโปรแกรม

หมายถึง การอ้างอิงค่าคงที่

@หมายถึง การอ้างอิงค่าในหน่วยความจำสำหรับข้อมูล

ภาคผนวก ข

คำศัพท์ที่ใช้ในวิทยานิพนธ์

	คำศัพท์ภาษาไทย	คำศัพท์ภาษาอังกฤษ
ก	กราฟบรรยายการเปลี่ยนสัญญาณ	Signal Transition Graph (STG)
	กราฟแสดงสถานะ	State Graph
	กลุ่มคำสั่งควบคุมทำงาน	Program and Machine Control Operation
	กลุ่มคำสั่งเคลื่อนย้ายข้อมูล	Data Transfer Operation
	กลุ่มคำสั่งดำเนินการทางคณิตศาสตร์ และตรรกะ	Arithmetic and Logic Operation
	เกตผกผัน	Weak Inverter
ข	ข้อมูลรวมชุด	Bundle Data
	เขียนข้อมูลลงหน่วยความจำ	Memory Write
	เขียนอินพุท/เอาท์พุท	I/O Write
ค	ความคลาดเคลื่อนของสัญญาณนาฬิกา	Clock Skew
	ค่าเวลาล่าช้า	Delay Time
	คำสั่งกระโดดแบบไร้เงื่อนไข	Jump (JMP)
	คำสั่งกระโดดเมื่อค่าในรีจิสเตอร์ตัวสะสมเท่ากับศูนย์	Jump Zero (JZ)
	คำสั่งกระโดดเมื่อค่าในรีจิสเตอร์ตัวสะสมมีตัวทด	Jump Carry (JC)
	คำสั่งกระโดดเมื่อค่าในรีจิสเตอร์ตัวสะสมไม่เท่ากับศูนย์	Jump not Zero (JNZ)
	คำสั่งกระโดดเมื่อค่าในรีจิสเตอร์ตัวสะสมไม่มีตัวทด	Jump not Carry (JNC)
	คำสั่งกระทำบิตต่อบิตออร์	Bitwise or (ORB)

คำศัพท์ภาษาไทย	คำศัพท์ภาษาอังกฤษ
คำสั่งกระทำบิตต่อบิตเอ็กคลูซีฟออร์	Bitwise exclusive-or (XORB)
คำสั่งกระทำบิตต่อบิตแอนด์	Bitwise and (ANDB)
คำสั่งกระทำลอจิกออร์	Logical or (OR)
คำสั่งกระทำลอจิกเอ็กคลูซีฟออร์	Logical exclusive-or (XOR)
คำสั่งกระทำลอจิกแอนด์	Logical and (AND)
คำสั่งกลับจากงานบริการอินเตอร์รัพท์	Return from Interrupt (RET)
คำสั่งกลับไปใช้งานโปรแกรมหลัก	Return (RET)
คำสั่งคูณ	Multiplicand (MUL)
คำสั่งบวก	Addition (ADD)
คำสั่งเพิ่มค่าหนึ่งค่า	Increment (INC)
คำสั่งเรียกใช้งานโปรแกรมย่อย	CALL
คำสั่งลดค่าหนึ่งค่า	Decrement (DEC)
คำสั่งลบ	Subtract (SUB)
คำสั่งเลื่อนทุกบิตไปทางขวา แบบคิดเครื่องหมาย	Shift Arithmetic Right (SAR)
คำสั่งเลื่อนทุกบิตไปทางขวา แบบไม่คิดเครื่องหมาย	Shift Logical Right (SLR)
คำสั่งเลื่อนทุกบิตไปทางซ้าย แบบคิดเครื่องหมาย	Shift Arithmetic Left (SAL)
คำสั่งเลื่อนทุกบิตไปทางซ้าย แบบไม่คิดเครื่องหมาย	Shift Logical Left (SLL)
คำสั่งสโตร์	Store
คำสั่งโหลด	Load
คำสั่งอิน	IN
คำสั่งเอาท์	OUT

	คำศัพท์ภาษาไทย	คำศัพท์ภาษาอังกฤษ
	แคชเก็บคำสั่ง	Instruction Cache
ง	งานของอินเตอร์รัพท์	Interrupt Service Routine (ISR)
จ	จำลองการทำงานแบบอิงเวลา	Simulate
ช	ชิปของสมาร์ตคาร์ด	Smartcard Chip
	ชุดคำสั่ง	Instruction Set
	ใช้พลังงาน	Power Consumption
ด	ดีเอ็มเอ	Direct Memory Access (DMA)
ต	ตอบรับอินเตอร์รัพท์	Interrupt Acknowledge
	ตัวกันผลการเด้ง	Debouncer
	ตัวขับบัส	Bus Driver
	ตัวควบคุมดีเอ็มเอ	DMA Controller
	ตัวควบคุมบัส	Bus Controller
	ตัวคูณ	Multiplier
	ตัวตั้ง	Multiplicand
	ตัวตัดสินใจ	Bus Arbiter
	ตัวต้านทานพูลดาวน์	Pull-down Resistor
	ตัวต้านทานพูลอัพ	Pull-up Resistor
	ตัวรับบัส	Bus Receiver
	ตัวเลือกแยกสัญญาณ	Demultiplexer
	ตัวเลือกรวมสัญญาณ	Multiplexer
	ตารางค้นหาแบบสี่อินพุท	4 Input LUTs
	ตารางตำแหน่งอินเตอร์รัพท์	Interrupt Vector Table (IVT)
	ตำแหน่งของคำสั่ง	Instruction Pointer (IP)
น	นับจำนวนการเปลี่ยนสถานะ	
	ของสัญญาณ	Transition Count

	คำศัพท์ภาษาไทย	คำศัพท์ภาษาอังกฤษ
บ	บัฟเฟอร์ค่าควบคุม	Control Buffer
	บัส	Bus
	บัสข้อมูล	Data Bus
	บัสควบคุม	Control Bus
	บัสแบบเดดิเคต	Dedicated Bus
	บัสแบบมัลติเพล็กซ์	Multiplex Bus
	บัสระบบแบบอสมวาร	Asynchronous System Bus
	บัสเลขที่อยู่	Address Bus
	บูทอัลกอลิทึม	Booth Agorithm
	แบบจำลองความหน่วง	Delay Model
	แบบจำลองความหน่วงที่ไม่ไว	
	ต่อความหน่วง	Delay Insensitive (DI)
ป	ประมาณค่าการใช้อุปกรณ์บนเอฟพีจีเอ	Device Utilization Summary
ผ	ผลลัพธ์	Product
	แผนที่คาร์นอฟ	Karnaugh Map
	แผนภาพตัดสินใจแบบทวิภาค	Binary Decision Diagram
	ชนิดมีการลดทอนอันดับ	Reduced-Ordered-Binary Decision Diagram (ROBDD)
พ	พลังงานที่ใช้เปลี่ยนสถานะของสัญญาณ	Switching Power
	พลังงานไดนามิก	Dynamic Power
	พลังงานไฟฟ้ารั่วไหล	Leakage Power
	พลังงานสแตติก	Static Power
	พอร์ทอินพุทเอาท์พุท	Bonded IOBs
	พีชคณิตบูลีน	Boolean Algebra
ฟ	ไฟโฟ	First-in-first-out (FIFO)

	คำศัพท์ภาษาไทย	คำศัพท์ภาษาอังกฤษ
ม	ไมโครโพรเซสเซอร์แบบอสมวาร	Asynchronous Processor
ร	รหัสฐานสอง	Binary Code
	รหัสฐานสองจำนวน n บิต	n'b
	รหัสรางคู่	Dual-rail Code
	รหัสหนึ่งในสิบหก	1-of-16 Code
	รหัสหนึ่งในสี่	1-of-4 Code
	รหัสหนึ่งในสี่จำนวน n บิต	n'1of4
	ร้องขออินเตอร์รัพท์	Interrupt Request
	ระดับเกต	Gate Level
	ระดับทรานซิสเตอร์	Transistor Level
	ระบบตอบรับ	Acknowledgement Circuits
	รีจิสเตอร์	Register
	รีจิสเตอร์ฐานตัวนับ	Base Word Count Register
	รีจิสเตอร์ฐานเลขที่อยู่ต้นทาง	Base Source Address Register
	รีจิสเตอร์ฐานเลขที่อยู่ปลายทาง	Base Destination Address Register
	รีจิสเตอร์ตัวนับปัจจุบัน	Current Word Count Register
	รีจิสเตอร์ตัวสะสม	Accumulator Register (ACC, A)
	รีจิสเตอร์พักข้อมูล	Temp Register
	รีจิสเตอร์เลขที่อยู่ต้นทางปัจจุบัน	Current Source Address Register
	รีจิสเตอร์เลขที่อยู่ปลายทางปัจจุบัน	Current Destination Address Register
	รีเซ็ต	Reset
ล	ลอจิกสามสถานะ	Tri-state Buffer
	แลตช์	Latch
ว	วงจรควบคุมที่ไม่ขึ้นต่ออัตราเร็ว	Speed-independent Control Circuits

คำศัพท์ภาษาไทย	คำศัพท์ภาษาอังกฤษ
วงจรคูณครั้งละ 2 หลักเข้ารหัสหนึ่งในสี่ ใช้ฟังก์ชันเข้ารหัสรางคู่	1-of-4 Radix-4 Booth Multiplier with Dual-rail Function Unit
วงจรคูณครั้งละ 2 หลักเข้ารหัสหนึ่งในสี่ ใช้ฟังก์ชันเข้ารหัสหนึ่งในสี่	1-of-4 Radix-4 Booth Multiplier with 1-of-4 Function Unit
วงจรคูณครั้งละ 1 หลักเข้ารหัสหนึ่งในสี่ ใช้ฟังก์ชันเข้ารหัสรางคู่	1-of-4 Radix-2 Booth Multiplier with Dual-rail Function Unit
วงจรคูณครั้งละ 1 หลักเข้ารหัสหนึ่งในสี่ ใช้ฟังก์ชันเข้ารหัสหนึ่งในสี่	1-of-4 Radix-2 Booth Multiplier with 1-of-4 Function Unit
วงจรถอดรหัสคำสั่งดีเอ็มเอ	DMA Decoder
วงจรเพิ่มค่า	Incrementor
วงจรลดค่า	Decrementor
วงจรเลือกหลักสำหรับวงจรเพิ่มค่าเข้ารหัสหนึ่งในสี่	1-of-4 Selector
วงจรสมวาร	Synchronous Circuit
วงจรสร้างสัญญาณควบคุมหน่วยความจำ	Memory Controller
วงจรอสมวาร	Asynchronous Circuits
วิถีวิกฤต	Critical Path
เวลาของสัญญาณนาฬิกาหนึ่งลูก	Clock Cycle Time
เวลาในเข้าถึงหน่วยความจำ	Memory Access Time

	คำศัพท์ภาษาไทย	คำศัพท์ภาษาอังกฤษ
ส	สถานะ	Status
	สถานะของสัญญาณ	State Transition
	สไลด์	Slice
	ส่วน	Path
	ส่วนเขียนผลลัพธ์	Write Result Unit
	ส่วนควบคุม	Control Unit
	ส่วนติดต่อกับแอลซีดี	LCD Interface
	ส่วนติดต่อกับอุปกรณ์ต่อพ่วง	I/O Interface
	ส่วนติดต่อของบัส	Bus Interface
	ส่วนติดต่อของหน่วยความจำ	Memory Interface
	ส่วนบริการอินเตอร์รัพท์	Interrupt Unit
	ส่วนประมวลผล	Execute Unit
	ส่วนแปลความหมายของคำสั่ง	Decode Unit
	ส่วนพักข้อมูล	Latch
	ส่วนพักข้อมูลต้นทาง	Source Latch
	ส่วนพักข้อมูลปลายทาง	Destination Latch
	ส่วนฟังก์ชัน	Function Unit
	ส่วนรับส่งข้อมูล	Data Path
	ส่วนอ่านคำสั่ง	Fetch Unit
	สังเคราะห์	Synthesis
	สัญญาณตอบรับ	Acknowledge Signal
	สัญญาณนาฬิกา	Clock
	สัญญาณร้องขอ	Request Signal
	สัญญาณอาณัติแบบ 4 ขั้น	4-Cycle Protocol, 4 Phase Protocol
	สายท่อแบบอสมวาร	Asynchronous Pipelines

	คำศัพท์ภาษาไทย	คำศัพท์ภาษาอังกฤษ
	สายสัญญาณบัส	Bus Line
ห	หน่วยคำนวณทางคณิตศาสตร์และตรรกะ	Arithmetic and Logic Units (ALU)
อ	อนุญาตให้ใช้บัส	Bus Grant
	อ่านข้อมูลจากหน่วยความจำ	Memory Read
	อ่านอินพุท/เอาท์พุท	I/O Read
	อิมพลิเมนท์	Implement
	อุปกรณ์ต่อพ่วง	I/O
	อุปกรณ์ไปป์ไลน์	Asynchronous Pipeline Module
	อุปกรณ์สหสัญญาณ	Multiplexer
	อุปกรณ์ออโต้สวีป	Autosweeping Module (ASM)
	เอฟพีจีเอแบบสมวาร	Synchronous FPGA
	เอฟพีจีเอแบบอสมวาร	Asynchronous FPGA
	แอนด์เกต	And-gate
ฮ	ไฮ-อิมพีแดนซ์	High Impedance (Hi-Z)

ประวัติผู้เขียนวิทยานิพนธ์

นางสาว กิตติมา ฐานพีรภัทร์ เกิดเมื่อวันที่ 17 เมษายน พ.ศ. 2528 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาปริญญาวิศวกรรมศาสตร์บัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ จากภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสยาม ในปีการศึกษา 2549 และเข้าศึกษาต่อในหลักสูตรวิศวกรรมศาสตร์มหาบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ ที่ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ปีการศึกษา 2550