

FPGA Surveyor-III XC3S200F

FPGA Surveyor-III XC3S200F4

BoardManual

บริษัท เอเพก อินสตรูเมนต์ จำกัด

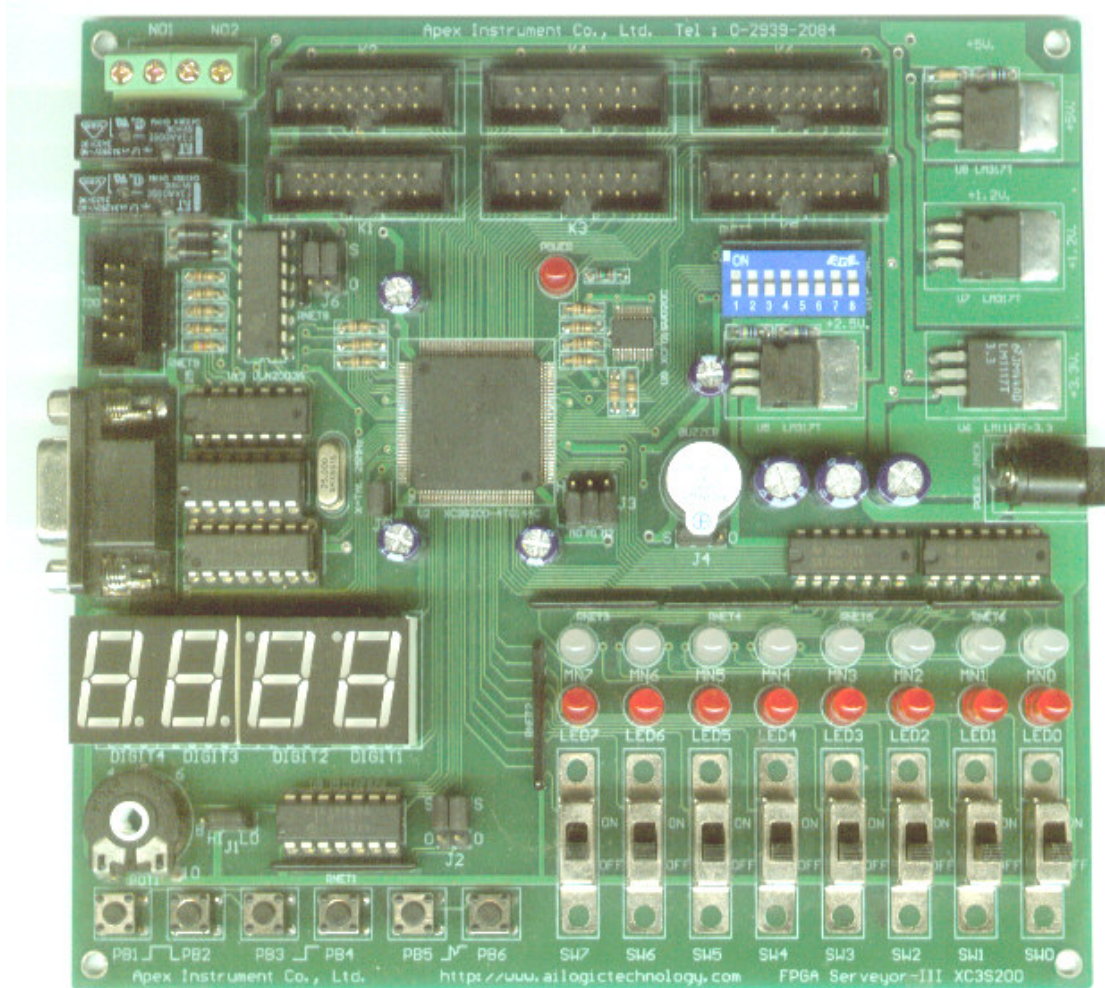
APEX INSTRUMENT CO., LTD.

77/9 ซอยลาดพร้าว1 ถนนลาดพร้าว แขวงจอมพล เขตจตุจักร กรุงเทพมหานคร 10900 โทร.:0-2939-2084 แฟกซ์: 0-2939-2084

77/9 SOI LADPRAO 1, LADPRAO ROAD, JOMPOL, JATUJAK DISTRICT, BANGKOK THAILAND 10900 TEL. 0-2939-2084 FAX.0-2939-2084

FPGA Surveyor-III XC3S200

บอร์ดทดลองรุ่น FPGA Surveyor-III XC3S200F และ FPGA Surveyor-III XC3S200F4 เป็นบอร์ดทดลอง FPGA และ Logic Trainer มีรายละเอียดแสดงดังรูปที่ 1 โดยที่บอร์ดนี้จะเป็นได้ทั้งบอร์ดทดลองและบอร์ดพัฒนา FPGA ที่มีความจุรวมมากถึง 200,000 และ 400,000 เกตตามลำดับ และใช้ Platform Flash PROM สำหรับเก็บข้อมูลวงจร ซึ่งสามารถโปรแกรมวงจรลง Platform Flash PROM ผ่านทางสายควาน์โหลดแบบ JTAG ได้โดยตรงและสามารถโปรแกรมซ้ำได้ถึง 20,000 ครั้ง บอร์ดออกแบบครั้งนี้มีอุปกรณ์อำนวยความสะดวก ที่เพียงพอพร้อมด้วยอุปกรณ์อินพุตเอาต์พุตอย่างครบครัน เพื่อให้ผู้ทดลองได้เรียนรู้การออกแบบวงจรดิจิทัลตั้งแต่วงจรขั้นพื้นฐานจนถึงขั้นนำไปพัฒนาออกแบบสร้างวงจรขนาดใหญ่ได้ด้วยตัวเอง



รูปที่ 1 บอร์ดรุ่น FPGA Surveyor-III XC3S200

คุณสมบัติทั่วไปของบอร์ดอเนกประสงค์

คุณสมบัติทั่วไปของบอร์ดทดลองอเนกประสงค์รุ่น FPGA Surveyor-III XC3S200 ประกอบด้วย

- FPGA Spartan-3 เบอร์ XC3S200-4TQ144C (XC3S400-4TQ144C สำหรับรุ่นลงท้ายด้วย F4)
- Platform Flash PROM เบอร์ XCF01SVO20C (XCF02VO20C สำหรับรุ่นลงท้ายด้วย F4)
- 7 Segment 4 หลัก
- DIP Switch 8 ดวง
- LED 3 สถานะ 8 ดวง

- LED 2 สถานะ 8 ดวง
- Logic Switch (Slide Switch) 8 ตัว
- Switch แบบ One-Shot 2 ตัว
- Switch แบบ Bounce 2 ตัว
- Switch แบบ Bounceless 2 ตัว
- 6 Expansion ports (48 Bits I/O)
- Relay 220 V/3A 2 ตัว
- Clock generator : 1–100Hz และ 10–1KHz
- Buzzer 1 ตัว
- 25 MHz Oscillator

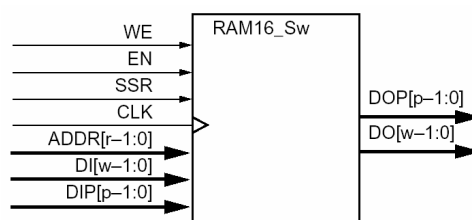
คุณสมบัติที่สำคัญของ FPGA ตระกูล Spartan-3 เบอร์ XC3S200

ชิพ FPGA ตระกูล Spartan-3 เบอร์นี้มีความโดดเด่น คือ มีหน่วยความจำ RAM รวม 216Kbits และมีตัวคูณที่เป็นฮาร์ดแวร์ 12 ชุด รวมทั้ง DCM อีก 4 ชุด ทำให้การออกแบบวงจรดิจิทัลที่มีไมโครคอนโทรลเลอร์แบบฝังตัวอยู่ด้วยทำได้ง่าย ช่วยให้การออกแบบวงจรทำได้สะดวกและอย่างมีประสิทธิภาพมาก โดยอุปกรณ์ที่อยู่ภายในชิพมีคุณสมบัติดังนี้

- ความจุวงจร 200,000 เกต
- 18Kb block RAMs จำนวน 12 ชุด รวม 216K bits(จำนวน 18 ชุด สำหรับรันลงท้ายด้วย F4)
- 18x18 hardware multiplier จำนวน 12 ชุด(จำนวน 18 ชุด สำหรับรันลงท้ายด้วย F4)
- Digital Clock Manager (DCM) จำนวน 4 ชุด
- Digitally Controlled Impedance (DCI)

1.) หน่วยความจำ 18Kb block RAM

18Kb block RAM เป็นหน่วยความจำความเร็วสูงประมาณ 200 Mhz จำนวน 12 ชุด โดย Block RAM แต่ละชุดสามารถทำเป็น RAM หรือ ROM ได้หลายแบบ ดังตัวอย่างในรูปที่ 2 และตารางในรูปที่ 3



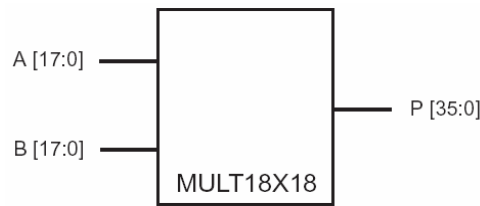
รูปที่ 2 แสดงขนาด RAM แบบ Single Port

Organization	Memory Depth	Data Width	Parity Width
512x36	512	32	4
1Kx18	1024	16	2
2Kx9	2048	8	1
4Kx4	4096	4	-
8Kx2	8192	2	-
16Kx1	16384	1	-

รูปที่ 3 RAM แบบ Single Port ขนาดต่างๆ ที่สร้างจาก Block RAM แต่ละชุด

2.) 18x18 Hardware multiplier

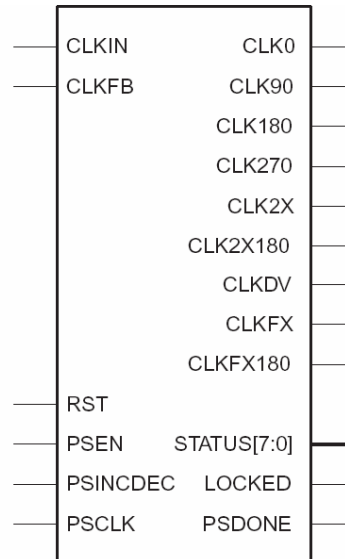
18x18 hardware multiplier เป็นฮาร์ดแวร์วงจรคูณสำเร็จรูปขนาด 18x18 บิต มีสัญลักษณ์แสดงดังรูปที่ 4 มีจำนวน 12 ชุด



รูปที่ 4 18x18 hardware multiplier

3.) Digital Clock Manager

Digital Clock Manager (DCM) เป็นวงจรที่มีความสำคัญมากที่ช่วยจัดการเกี่ยวกับสัญญาณนาฬิกา ซึ่งมีอยู่ในชิพจำนวน 4 ชุด และถือได้ว่า DCM ช่วยทำให้การออกแบบวงจรง่ายขึ้นอย่างมากเนื่องจากสามารถสร้างความถี่ต่างๆ ได้อย่างมากมายจากออสซิลเลเตอร์จากภายนอกเพียงชุดเดียว จึงไม่มีความจำเป็นต้องหาสัญญาณนาฬิกาจากภายนอกหลายๆ แหล่งอีกต่อไป และไม่เพียงเท่านั้นสัญญาณพิกัดกล่าวข้างังค์โครโนซ์กับสัญญาณนาฬิกาจากออสซิลเลเตอร์เดิมอีกด้วย สามารถนำไปใช้เป็นตัวกำเนิดสัญญาณนาฬิกาความถี่ต่างๆ ได้โดยไม่ต้องใช้ Variable Clock จากภายนอกแต่อย่างใด DCM มีสัญลักษณ์แสดงดังรูปที่ 5



รูปที่ 5 สัญลักษณ์ของวงจร DCM

DCM จะทำงานในหน้าที่ดังต่อไปนี้

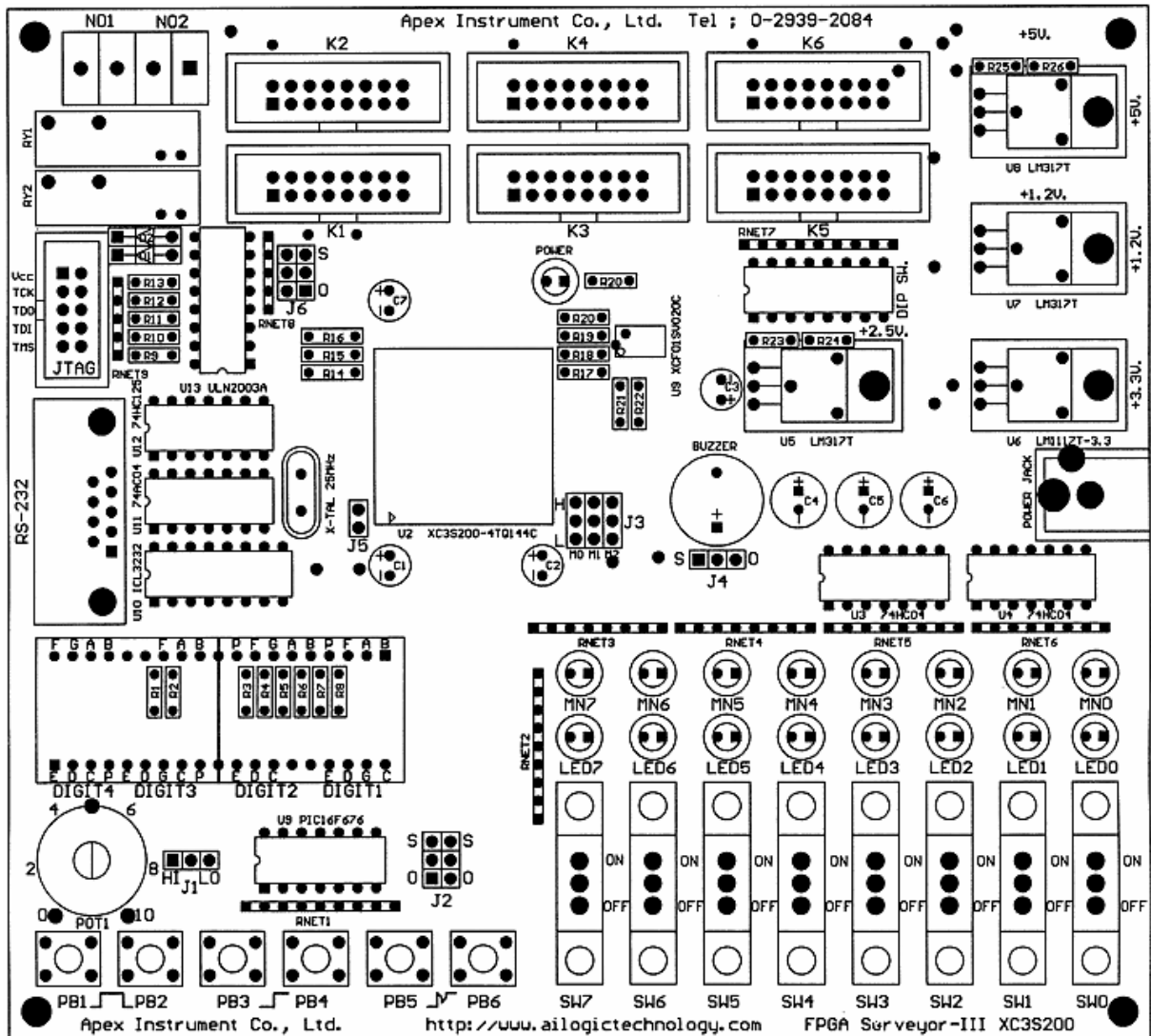
- หาคความถี่ (Clock Divider) เป็นวงจรซึ่งจะให้ความถี่เอาต์พุตเท่ากับความถี่อินพุตหารด้วยตัวเลข ดังต่อไปนี้ คือ 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5, 5.5, 6, 6.5, 7, 7.5, 8, 9, 10, 11, 12, 13, 14, 15, หรือ 16 ตามลำดับ
- สร้างความถี่สองเท่า (Clock Doubler) เป็นวงจรซึ่งจะให้ความถี่ที่เอาต์พุตจะเป็น 2 เท่าของความถี่อินพุต
- Digital Frequency Synthesizer (DFS) เป็นวงจรซึ่งสามารถกำหนดให้ความถี่เอาต์พุตเท่ากับผลคูณของความถี่อินพุตกับอัตราส่วนของ M/D โดยที่ M = 2 ถึง 32 และ D = 1 ถึง 32 ตัวอย่าง Fin = 25 MHz ถ้าเราต้องการ Fout = 200 MHz ทำได้โดยเลือก M=8, D=1 วงจร DFS นี้นำไปใช้งาน เช่น สร้างวงจรเปลี่ยนจากการส่งข้อมูลแบบขนานเป็นอนุกรม ซึ่งต้องสร้างสัญญาณนาฬิกาสูงกว่าของเดิม เช่น 10 – 11 เท่า เป็นต้น หรืองานอื่นๆ ที่ต้องใช้วงจรฟรีควเอนซีซินธิไซเซอร์
- Delay-Locked Loop (DLL) เป็นวงจรใช้แก้ปัญหาการเลื่อนเฟสในวงจรให้กลับมตรงตามเฟสที่ต้องการ
- Quadrant Phase Shift เป็นวงจรเลื่อนเฟส 90 , 180 และ 270 องศา ตามลำดับ
- Fine Phase Shift เป็นวงจรใช้ในการเลื่อนเฟสอย่างละเอียด มีความละเอียดอยู่ที่ 1/ 256 เท่าของคาบความถี่

4.) Digitally Controlled Impedance (DCI)

Digitally Controlled Impedance (DCI) ใช้ป้องกันสัญญาณสะท้อนใน PCB โดยการควบคุมเอาต์พุตอิมพีแดนซ์ที่เหมาะสม

หลักการทำงานของบอร์ดเนกประสงค์

บอร์ดรุ่น FPGA Surveyor-III XC3S200 ดังรูปที่ 6 เป็นดังนี้



รูปที่ 6 บอร์ดรุ่น FPGA Surveyor-III XC3S200

1. รายละเอียดด้านเอาต์พุต

1.1) 7-Segment (DIGIT 1 – DIGIT4)

ตัวแสดงผลเซเวนเซกเมนต์ (7-Segment) ทั้ง 4 หลัก คือ DIGIT1 – DIGIT4 จะใช้เทคนิคการสแกน (Scan) ในการแสดงตัวเลขบนตัวแสดงผลทั้ง 4 หลักพร้อมๆ กันเพื่อประหยัดสายสัญญาณ O/P จาก FPGA โดยที่ 8 เส้นใช้สำหรับส่ง Data เพื่อแสดงผล และอีก 4 เส้นใช้สำหรับแคโทดร่วม (Common Cathode) ของแต่ละหลักดังรายละเอียดในตารางที่ 1 โดยหลักการทำงานจะเป็นดังนี้คือจะทำการส่งตัวเลขไปที่หลักแรกแล้วทำการเลือกให้หลักแรกแสดงผล จากนั้นจึงเปลี่ยนไปส่งตัวเลขของหลักที่สองแล้วเลือกหลักที่สองให้แสดงผล และทำหลักถัดไปจนครบทั้ง 4 หลัก แล้วจึงวนกลับมาเริ่มที่หลักแรกใหม่และวนไปเรื่อยๆ จะทำให้รูปที่ 7(a) รายละเอียดไดอะแกรมผังวงจรบอร์ดทดลองเนกประสงค์รุ่น FPGA Discovery-III XC3S200

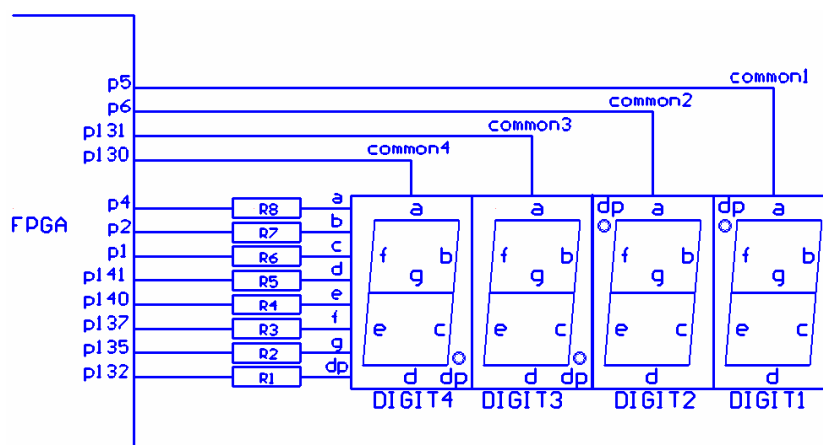
FPGA Surveyor-III XC3S200 Board Manual (Rev2.07/03/07)

ผู้ใช้งานเห็นตัวเลขติดพร้อมกันโดยมีข้อแม้ว่าการสับเปลี่ยนหลักในการแสดงผลต้องทำให้เร็วกว่าที่ตาคนเราจะมองได้ทัน ซึ่งโดยทั่วไปตาคนเราจะมองแยกได้ประมาณ 25 – 30 ครั้งต่อวินาที ดังนั้นอย่างน้อยควรต้องสแกนโดยใช้ความเร็วไม่น้อยกว่า 120 ครั้งต่อวินาที (30 ครั้ง X 4 หลัก = 120 ครั้งต่อวินาที) จึงจะมองไม่เห็นการกระพริบของตัวแสดงผล

ในการแสดงผลนั้นหากต้องการให้เซกเมนต์ใดของตัวแสดงผลติดได้นั้นต้องส่งสัญญาณลอจิก “1” โดยที่ด้านแอนโอดทุกขาจะมีตัวต้านทาน R1-R8 ขนาด 100 โอห์มต่ออนุกรมกับ I/O ของ FPGA เพื่อจำกัดกระแสแต่ละเซกเมนต์ ในส่วนที่ขาแคโอดร่วมในแต่ละหลักนั้นถ้าต้องการให้หลักใดติดก็ต่อให้ขาแคโอดร่วมของหลักนั้นเป็นลอจิก “0” และขอให้สังเกตว่าตัวแสดงผลเลขวนเซกเมนต์ DIGIT2 และ DIGIT1 หรือสองตัวสุดท้ายทางขวามือถูกออกแบบมาให้มีการกลับตัวแสดงผลโดยการหมุน 180 องศา เพื่อให้จุด (.) ของ DIGIT2 และ DIGIT1 ขึ้นไปอยู่ด้านบนเพื่อประโยชน์ในการแสดงเครื่องหมาย “ : ” (Colon) ในการทำนาฬิกาหรือการแสดงเป็นองศาในงานที่เกี่ยวข้องกับอุณหภูมิ เช่น 11:39 หรือ 20° C แต่การแสดงผลตัวเลขต่างๆ ยังคงใช้สายสัญญาณเดียวกับสองตัวแรก รายละเอียดการต่อขาอุปกรณ์เข้ากับขาของ FPGA เป็นดังตารางรูปที่ 1 โดยมีวงจรแสดงดังรูปที่ 7

ตารางรูปที่ 1 รายละเอียดการต่อขาอุปกรณ์เข้ากับขาของ FPGA

7-Segment	FPGA Pin
a	p4
b	p2
c	p1
d	p141
e	p140
f	p137
g	p135
dp (จุด)	p132
DIGIT1 (Common1)	p5
DIGIT2 (Common2)	p6
DIGIT3 (Common3)	p131
DIGIT4 (Common4)	p130



รูปที่ 7 วงจรแสดงผลด้วย 7-Segment

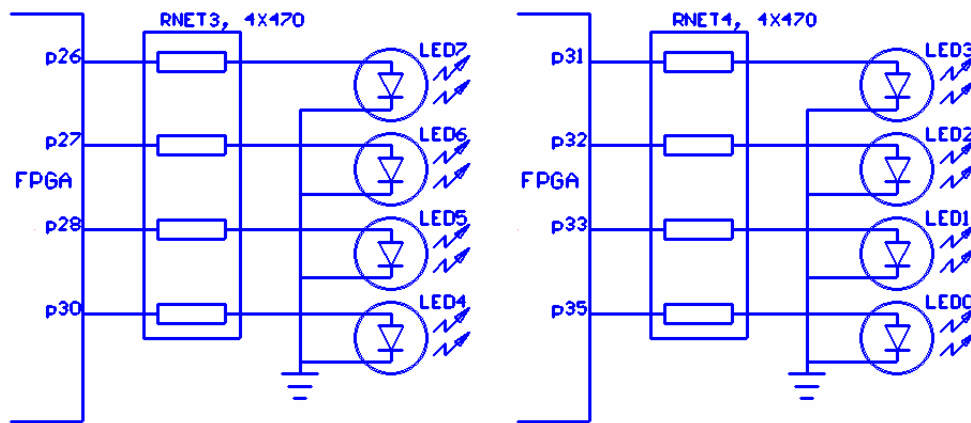
1.2 LED แสดงผล

1.) LED monitor ZLED0-LED7)

บอร์ดทดลองนี้มี LED 8 ดวง คือ LED1 – LED8 โดยต่อขาคาโทด (Cathode) ลงกราวนด์ และต่อขาแอนโนดเข้ากับขาอินพุตเอาต์พุต (I/O) ของ FPGA โดยมีตัวต้านทานแบบเนตเวิร์ค (ภายในประกอบด้วย R=470 โอห์ม 4 ตัวแยกอิสระ) คือ RNET3 และ RNET4 ต่ออนุกรมอยู่เพื่อจำกัดกระแส โดยในขณะที่ขาสัญญาณในแต่ละบิตเป็น High “1” LED เปล่งแสงสีแดง และในขณะที่แต่ละบิตเป็น Low “0” LED จะดับ หากสัญญาณที่เข้ามาเป็นพัลส์ LED จะเปล่งแสงสีแดงแต่มีความสว่างลดลงขึ้นอยู่กับค่า Duty Cycle โดยที่ขาสัญญาณของ LED จะต่ออยู่กับขาอินพุตเอาต์พุต (I/O) ของชิพ FPGA ต่างๆ ดังตารางที่ 2 โดยมีวงจรแสดงดังรูปที่ 8

ตารางที่ 2 แสดงขาสัญญาณของ LED ที่ต่ออยู่กับขาอินพุตเอาต์พุต (I/O) ของชิพ FPGA

LED	FPGA Pin
LED0	P35
LED1	P33
LED2	P32
LED3	P31
LED4	P30
LED5	P28
LED6	P27
LED7	P26



รูปที่ 8 วงจรแสดงผลด้วย LED monitor

2.) Logic monitor (MN0 – MN7)

เป็น LED แบบสองสีที่สามารถตรวจสอบสัญญาณได้สามสถานะคือ High “1”, Low “0” และ High impedance “Z” โดยแต่ละดวงของ LED จะควบคุมโดยใช้ขาสัญญาณสองขา เช่น MN0 จะมี MN0 Pin 1 และ MN0 Pin 2

โดยถ้า MN0 Pin 1 มีระดับลอจิก “1” และ MN0 Pin 2 มีระดับลอจิก “0” จะทำให้ไดโอดเปล่งแสง MN0 สว่างเป็นสีแดง และในทางกลับกัน จะได้สี เขียว หรือหากมีระดับลอจิกเดียวกัน จะดับไม่สว่าง

ใช้แทนสัญญาณเพื่อทำให้ไดโอดเปล่งแสงสามารถทำหน้าที่เป็น Logic Monitor ได้ เช่นในขณะที่ขาสัญญาณในแต่ละบิตเป็น High “1” ไดโอดจะเปล่งแสงสีแดง ในขณะที่ขาสัญญาณในแต่ละบิตเป็น Low “0” ไดโอดจะเปล่งแสงสีเขียว และในขณะที่สัญญาณเป็น High impedance “Z” ไดโอดเปล่งแสงจะดับ ทั้งนี้ทั้งนั้นหากสัญญาณที่เข้ามาเป็นสัญญาณพัลส์ไดโอดจะเปล่งแสงสี

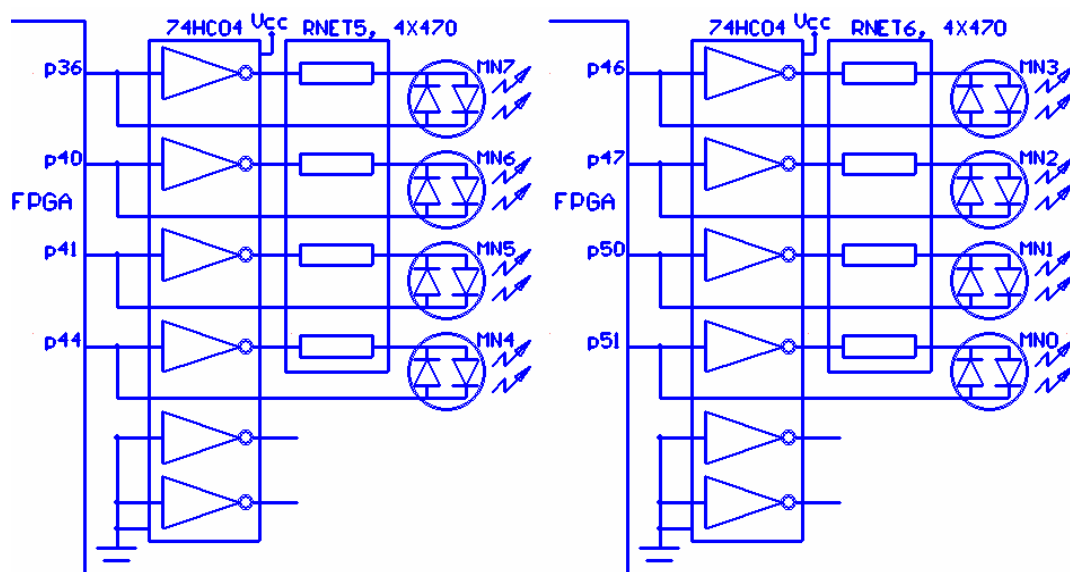
แดงและสีเขียวสลับกันทำให้เห็นเป็นแสงสีส้ม โดยขาสัญญาณของไดโอดเปล่งแสงจะต่ออยู่กับ FPGA ขาต่างๆ ดังตารางที่ 4
วงจรของ Logic monitor แสดงดังรูปที่ 9

ตารางที่ 3 แสดงสี LED ที่สถานะต่างๆ

Logic	Color
High "1"	Red
Low "0"	Green
High impedance "Z"	Off
Pulse	Orange

ตารางที่ 4 แสดงขาสัญญาณของ LED ที่ต่ออยู่กับขาอินพุตเอาต์พุต (I/O) ของชิพ FPGA

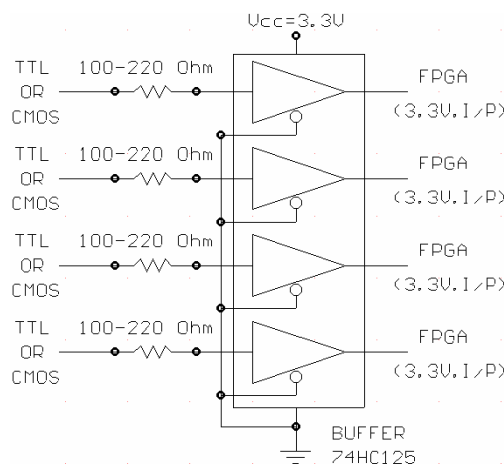
Bi-color LED	FPGA pin
MN0	P51
MN1	P50
MN2	P47
MN3	P46
MN4	P44
MN5	P41
MN6	P40
MN7	P36



รูปที่ 9 วงจรของ Logic monitor

1.4 คอนเนคเตอร์ K1- K6

พอร์ตต่อสายสัญญาณไปยังวงจรมานอกจะต่ออยู่ที่คอนเนคเตอร์ K1-K6 เพื่อเชื่อมต่อสัญญาณไปยังอุปกรณ์ภายนอกหรือรับสัญญาณเข้ามามีรายละเอียดดังตารางที่ 1 โดยที่ที่คอนเนคเตอร์ K1- K6 จะถูกออกแบบเป็นสายสัญญาณและกราวด์ประมาณอย่างละครึ่งเพื่อแก้ปัญหาการรบกวนข้ามช่องสัญญาณ (Cross talk) ดังนั้นเมื่อต่อสายแพร์หรือสายริบบอนเข้ากับคอนเนคเตอร์แล้วจะทำให้สายสัญญาณและกราวด์อยู่ในลักษณะวางเรียงสลับกันไปทุกเส้นทำให้การรบกวนข้ามช่องสัญญาณเกิดได้ยาก และเนื่อง I/O ทั้งหมดจะเป็นระบบ 3.3 โวลต์ ดังนั้นในขณะที่ I/O ทำหน้าที่เป็นเอาต์พุตก็จะสามารถขับอินพุตทั้งลอจิกที่ใช้กับระบบ 3.3 และ 5 โวลต์ได้โดยตรง แต่เมื่อ I/O เป็นอินพุตจะใช้ได้เฉพาะระบบ 3.3 โวลต์เท่านั้น หากต้องการเชื่อมต่อกับเอาต์พุตจากภายนอกที่เป็นระบบ 5 โวลต์จำเป็นต้องมีไอซีระบบ 3.3 โวลต์มาเป็นกันชนหรือทำหน้าที่เป็นบัฟเฟอร์ เช่น ไอซีตระกูล 74HCxx หรือ 74ACxx ซึ่งหลายคนอาจจะเผลอลืมไปแล้วว่าไอซีเหล่านี้ใช้ไฟเลี้ยงได้ตั้งแต่ 2 – 6 โวลต์ เช่น เบอร์ 74HC125 เป็นต้น ในกรณีนี้บัฟเฟอร์กินไฟเลี้ยง 3.3 โวลต์ ดังนั้นอินพุตของบัฟเฟอร์ที่ต่ออยู่กับเอาต์พุตระบบ 5 โวลต์จึงจำเป็นต้องมีตัวต้านประมาณ 220 โอห์มมาต่ออนุกรมที่อินพุตเพื่อจำกัดกระแสไหลเข้าอินพุตไม่ให้เกินกว่า 10 mA เพื่อป้องกันไม่ให้อินพุตเสียหาย แต่ถ้าเป็นเอาต์พุตระบบ 5 โวลต์จากลอจิกตระกูล TTL อาจลดค่าความต้านทานอนุกรมนี้ลงเหลือประมาณ 100 โอห์มได้ เนื่องจากลอจิก “ 1 ” จะให้ระดับแรงดันเอาต์พุตต่ำกว่าตระกูล CMOS การใช้ค่าความต้านทานน้อยลงจะทำให้วงจรทำงานที่ความถี่ที่สูงกว่า ตัวอย่างการเชื่อมต่อเอาต์พุตจากไอซีตระกูล TTL หรือ CMOS กับอินพุตของ FPGA แสดงดังรูปที่ 10 และในกรณีที่มี Input ระบบ 5 โวลต์เยอะๆ อาจใช้ CPLD ตระกูล XC9500XL เช่น เบอร์ XC9572XL เป็นบัฟเฟอร์ได้เพราะ I/O ของ CPLD แม้จะเป็นระบบ 3.3 โวลต์แต่สามารถทนรับสัญญาณเอาต์พุตระบบ 5 โวลต์ที่เข้ามาได้โดยตรงโดยไม่ต้องต่อตัวต้านทานอนุกรมกับขา I/O แต่อย่างใด



รูปที่ 10 ตัวอย่างการเชื่อมต่อเอาต์พุตจากไอซีตระกูล TTL หรือ CMOS กับอินพุตของ FPGA

การต่อตัวต้านทานพูลอัพ (Pulled up) ภายในชิพพร้อมกันทุกตัวทำได้โดยการต่อ Jumper J5 เพียงตัวเดียวเท่านั้น แต่ถ้าไม่ต้องการพูลอัพก็ให้ถอด J5 ออก การต่อสายสัญญาณ I/O ออกจากคอนเนคเตอร์ K1 – K6 นั้นถ้าต้องการใช้งานที่ความถี่สูงๆ หรือไม่ต้องการป้องกันการรบกวนข้ามช่องและป้องกันไม่ให้เกิดคลื่นของสัญญาณสะท้อนภายในเส้นสายทองแดงของ PCB ก็ควรใช้สายแพร์ (Flat Cable) สำหรับบอร์ดทดลองนี้แนะนำให้ใช้สายแพร์ขนาด 40 เส้นที่ความยาวสัญญาณไม่ควรเกิน 10 – 15 เซนติเมตร (เนื่องจากหาซื้อได้ง่ายตามร้านคอมพิวเตอร์ทั่วไป เพราะสายแพร์ดังกล่าวใช้เป็นสายเชื่อมต่อสัญญาณระหว่างฮาร์ดดิสก์กับเมนบอร์ดของคอมพิวเตอร์) และต้องโปรแกรมให้ I/O ของ FPGA เป็นแบบ Slow Slew Rate เพื่อลดสัญญาณรบกวนข้ามช่องและลดการสะท้อนของสัญญาณในสายแพร์ เกณฑ์ที่ใช้ในการพิจารณาว่าจะไม่มีผลกระทบเนื่องจากคุณสมบัติสายส่งหรือผลของ

FPGA Surveyor-III XC3S200 Board Manual (Rev2.07/03/07)

Transmission Line effect ที่เกิดขึ้นใน PCB และรวมถึงสายสัญญาณ (สายแพร์) ด้วยนั้น คือ ความยาวสูงสุดของเส้นลายทองแดงของ PCB รวมกับสายสัญญาณโดยประมาณไม่ควรเกิน $(2''/\text{nS}) \times \text{ช่วงเวลาขึ้น (Rise time)}$ ซึ่งในกรณีที่ O/P ของ FPGA เป็น Fast slew rate จะมีช่วงเวลาดำเนินโดยประมาณน้อยกว่า 1 nS ส่วนในกรณีที่ เป็น Slow slew rate จะมีช่วงเวลาดำเนินโดยประมาณ 3 nS ดังนั้น $(2''/\text{nS}) \times 3 \text{ nS} = 6'' = 15 \text{ เซนติเมตร}$ ในกรณีที่สายแพร์ยาวกว่านี้และมีผลกระทบเนื่องจากเกิดการสะท้อนของสัญญาณจนทำให้วงจรทำงานผิดพลาดนั้นการแก้ไขได้โดยเลือก Terminate ด้วยวิธีที่เหมาะสม สำหรับชิพตระกูล Spartan-3 นั้นจะมี Digitally Controlled Impedance (DCI) ในการช่วยแก้ไขปัญหการสะท้อน

ในการต่อสัญญาณออกภายนอกบอร์ดทดลองนั้นถ้าสายแพร์มี Vcc รวมอยู่ด้วยจะมีการต่อตัวเก็บประจุขนาด 0.1 uF และ 10nF แบบมัลติเลเยอร์หรือตัวเก็บประจุแบบชิพ (Chip capacitor) ใกล้เคียง ขั้ว Vcc และกราวด์ทุกจุด การต่อนี้ต้องทำทั้งที่บอร์ดทดลอง (ซึ่งต่อไว้แล้ว) และที่บอร์ดอื่นที่อยู่ภายนอกด้วยเพื่อให้สาย Vcc มีคุณสมบัติทางไฟฟ้าแบบ AC เสมือนว่าเป็นกราวด์

รายละเอียดการต่อขาของคอนเนคเตอร์ K1- K6 กับขาของ FPGA แสดงดังตารางที่ 5

ตารางที่ 5รายละเอียดการต่อขาของคอนเนคเตอร์ K1- K6 กับขาของ FPGA

K1 Pin	FPGA Pin	Descriptions	K2 Pin	FPGA Pin	Descriptions
1	P125	I/O, RELAY No.1	1	P112	I/O
2		GND	2		+3.3V <u>Vcc</u>
3	P124	I/O, RELAY No.2	3	P108	I/O
4		GND	4		GND
5	P123	I/O	5	P107	I/O
6		GND	6		GND
7	P122	I/O	7	P105	I/O
8		GND	8		GND
9	P119	I/O	9	P104	I/O
10		GND	10		GND
11	P118	I/O	11	P103	I/O
12		GND	12		GND
13	P116	I/O	13	P102	I/O
14		GND	14		GND
15	P113	I/O	15	P100	I/O
16		GND	16		GND

ตารางที่ 5 รายละเอียดการต่อขาของคอนเนคเตอร์ K1- K6 กับขาของ FPGA (ต่อ)

K3 Pin	FPGA Pin	Descriptions	K4 Pin	FPGA Pin	Descriptions
1	P89	I/O	1	P99	I/O
2		GND	2		+3.3V V_{cc}
3	P87	I/O	3	P98	I/O
4		GND	4		GND
5	P86	I/O	5	P97	I/O
6		GND	6		GND
7	P85	I/O	7	P96	I/O
8		GND	8		GND
9	P84	I/O	9	P95	I/O
10		GND	10		GND
11	P83	I/O	11	P93	I/O
12		GND	12		GND
13	P82	I/O	13	P92	I/O
14		GND	14		GND
15	P80	I/O	15	P90	I/O
16		GND	16		GND

ตารางที่ 5 รายละเอียดการต่อขาของคอนเนคเตอร์ K1- K6 กับขาของ FPGA (ต่อ)

K5 Pin	FPGA Pin	Descriptions	K6 Pin	FPGA Pin	Descriptions
1	P68	I/O , DIP SW1	1	P79	I/O
2		GND	2		+3.3V V_{cc}
3	P63	I/O , DIP SW2	3	P78	I/O
4		GND	4		GND
5	P60	I/O , DIP SW3	5	P77	I/O
6		GND	6		GND
7	P59	I/O , DIP SW4	7	P76	I/O
8		GND	8		GND
9	P56	I/O , DIP SW5	9	P74	I/O
10		GND	10		GND
11	P55	I/O , DIP SW6	11	P73	I/O
12		GND	12		GND
13	P53	I/O , DIP SW7	13	P70	I/O
14		GND	14		GND
15	P52	I/O , DIP SW8, BUZZER	15	P69	I/O
16		GND	16		GND

2. รายละเอียดด้านอินพุต

2.1 ดิพสวิตช์

มีดิพสวิตช์ (Dip Switch) อีก 8 สวิตช์ โดยปกติหากเลื่อนสวิตช์ลงไปที่ OFF จะทำให้ได้ลอจิก “1” (Active Low) และหากเลื่อนสวิตช์ขึ้นไปที่ ON จะทำให้ได้ลอจิก “0”

ตารางที่ 6 รายละเอียดการต่อขาของ ดิพสวิตช์ กับขาของ FPGA (ต่อ)

DIP SWITCH	FPGA Pin
DIP1	P68
DIP 2	P63
DIP 3	P60
DIP 4	P59
DIP 5	P56
DIP 6	P55
DIP 7	P53
DIP 8	P52

2.2 สวิตช์กดติดปล่อยดับและ Variable clock generator

ทางด้านอินพุตบอร์ดทดลองนี้มีสวิตช์กดติดปล่อยดับ (Push button Switch) อยู่ 6 ตัวคือ PB1 – PB6 และ Variable clock generator (VRCLK) โดยใช้ไอซี PIC16F676 เพียงตัวเดียว รายละเอียดวงจรแสดงดังรูปที่ 11 โดยที่ PB1 และ PB2 จะเป็น One-shot push button switch หรือเป็นสวิตช์กดติดปล่อยดับที่ให้สัญญาณเอาต์พุตเป็นสัญญาณพัลส์ 1 ลูก ที่มีค่าความกว้างของพัลส์ช่วง High “1” คงที่ไม่ว่าจะกดสวิตช์นานเท่าไรก็ตาม ส่วนพัลส์ที่ถูกกดไปจะเกิดเมื่อปล่อยสวิตช์และกดอีกครั้งเท่านั้น

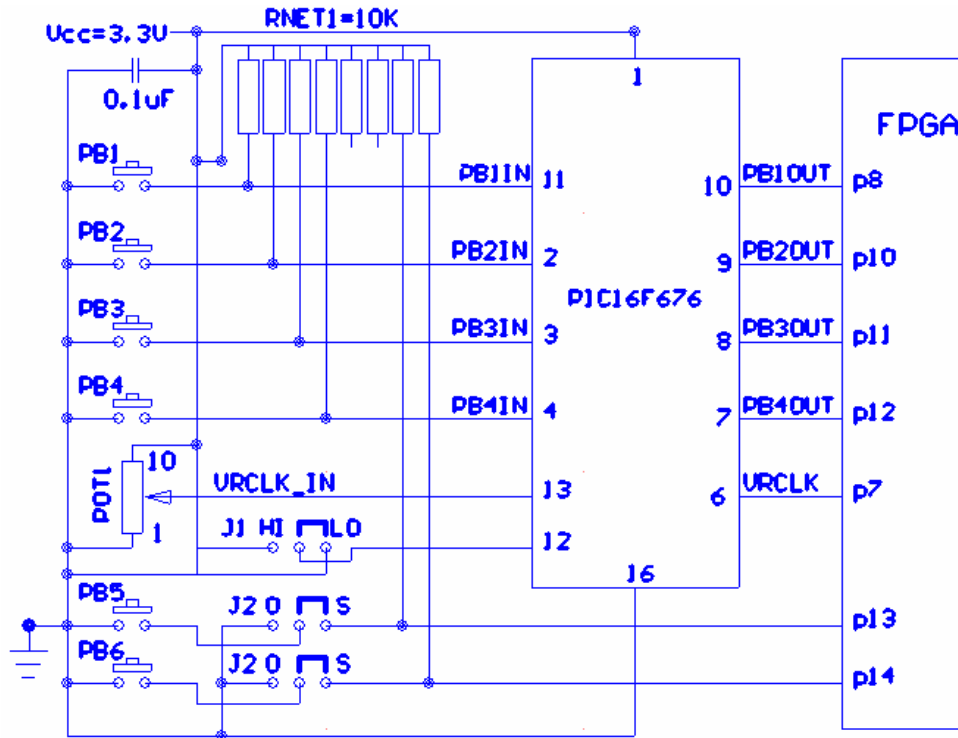
PB3 และ PB4 จะเป็น Bounce-less push button switch หรือเป็นสวิตช์กดติดปล่อยดับที่ให้เอาต์พุตเป็นสัญญาณพัลส์ 1 ลูกที่มีค่าความกว้างของพัลส์ช่วง High “1” เท่ากับระยะเวลาที่ยังคงกดสวิตช์นั้นอยู่ และจะกลับมาเป็น Low “0” เมื่อปล่อยสวิตช์นั้นซึ่งสัญญาณพัลส์ที่ได้จะเป็นสัญญาณสี่เหลี่ยม (Square Wave) อย่างแท้จริงโดยไม่มีสัญญาณเบาวซ์ (Bounce-less)

PB5และPB6 เป็น Bounce push button switch หรือสวิตช์กดติดปล่อยดับธรรมดา โดยที่ PB1-PB6 จะต่ออยู่กับขาของ FPGA โดยปกติหากไม่มีการกดจะเป็นลอจิก “1” และหากกดจะเป็นลอจิก “0” เนื่องจากมีตัวต้านทาน RNET1 ต่อพูลอัพอยู่

Range selector of on board variable clock generator (J1) เป็นจัมป์เปอร์ที่ใช้ตั้งช่วงความถี่ของสัญญาณนาฬิกา lock generator ที่สามารถปรับเปลี่ยนได้ 2 ช่วงคือ High Frequency ☐ (เชื่อม 1 และ 2) และ Low Frequency ☐ (เชื่อม 2 และ 3) ส่วนการปรับความถี่จะปรับที่ POT1

ตารางที่ 7 รายละเอียดการต่อขาเข้ากับขาของ FPGA

Pushbutton และ CLK	FPGA Pin
PB1 OUT	P8
PB2 OUT	P10
PB3 OUT	P11
PB4 OUT	P12
PB5	P13
PB6	P14
VRCLK	P7



รูปที่ 12 รายละเอียดวงจรสร้างพัลส์และClock

2.3 ออสซิลเลเตอร์

บอร์ดทดลองนี้ยังมีซ็อกเก็ตสำหรับติดตั้งออสซิลเลเตอร์ (Oscillator socket) ให้อีก 1 ตัว ซึ่งปกติจะใช้ไอซีเบอร์ 74AC04 คริสตัล 25 Mhz ตัวต้านทาน 330 โอห์มและ 1 เมกาโอห์ม และ ตัวเก็บประจุ 15 pF จำนวน 2 ตัว มาสร้างเป็นออสซิลเลเตอร์ โดยป้อนเข้าที่ขา GCLK6 หรือ p127 ของ FPGA ถ้าต้องการเปลี่ยนเป็นออสซิลเลเตอร์ก็ให้ถอดไอซี 74AC04 เพียงตัวเดียวเท่านั้นก็สามารถใส่ออสซิลเลเตอร์แบบ 3.3 โวลต์ที่ต้องการได้ทันทีตั้งแต่ความถี่ 1 Mhz ถึง 50 MHz แต่ต้องไม่ลืมว่าใน FPGA ตระกูล Spartan-3 มี Digital Clock Manager (DCM) เป็นวงจรที่มีความสำคัญมากที่ช่วยจัดการเกี่ยวกับสัญญาณนาฬิกาอยู่ในชิพจำนวน 4 ชุด ดังนั้นในหลายกรณีก็สามารถใช้ DCM ช่วยสร้างความถี่ที่ต้องการได้ อาจจะไม่มีความจำเป็นต้องใช้ออสซิลเลเตอร์จากภายนอกเพิ่มเติมแต่อย่างใด

2.4 Relay connector (NO1, NO2)

เป็นขั้วต่อหน้าคอนแทคของรีเลย์ 1 และ 2 (RLY1, RLY2) ขนาด 3A 250V จำนวน 2 ชุด ตามลำดับ โดยเป็นคอนแทคในแบบปกติเปิด (NO) วงจรชุดขับของรีเลย์ 1 และ 2 จะต่ออยู่กับขา P125 และ P124 ของ FPGA ตามลำดับ ซึ่งใช้ I/O ร่วมกับขา 1 และ 3 ของ K1 (Jumper มีเสียบที่ตำแหน่ง S) ในกรณีที่ไมต้องการใช้รีเลย์ 1 และ 2 ให้ถอด Jumper มาเสียบที่ตำแหน่ง O

2.5 JTAG connector

เป็นหัวต่อที่ใช้ต่อกับสายคาวาน์โฮลด์ JTAG Cable) ที่ใช้โปรแกรมข้อมูลลงตัว FPGA และ Flash PROM

2.6 Slide switch (SW0 – SW7)



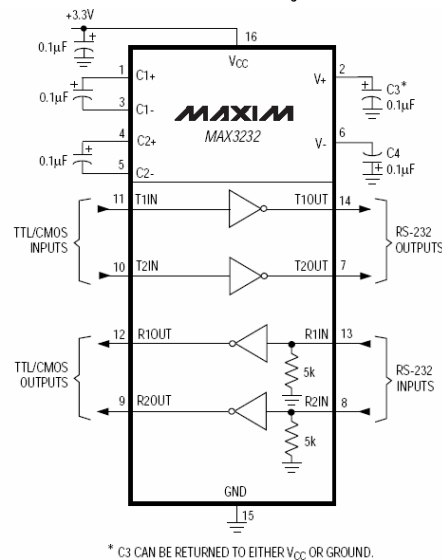
เป็นสวิตช์เลื่อนที่ใช้ป้อนข้อมูลเข้าสู่ CPLD โดยถ้าเลื่อนลงจะเป็น “1” ถ้าเลื่อนขึ้นจะเป็น “0” โดยเชื่อมต่อกับขาของ FPGA โดยทำงานแบบ Active Low เช่นกัน เนื่องจากมี R พูลอัพต่อทุกขาของ Slide switch เข้ากับ Vcc

ตารางที่ 8 รายละเอียดการต่อขา Slide switch เข้ากับขาของ FPGA

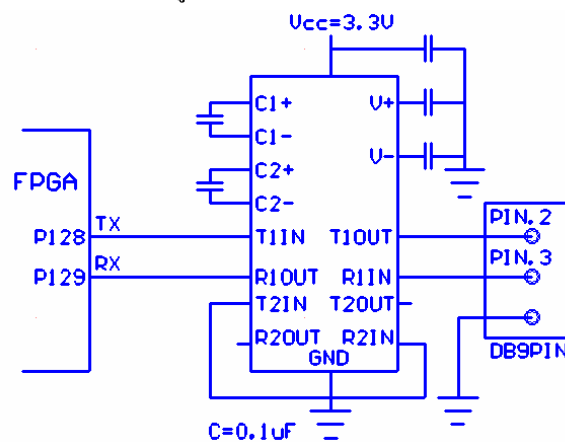
Slide switch	FPGA Pin
SW7	P15
SW6	P17
SW5	P18
SW4	P20
SW3	P21
SW2	P23
SW1	P24
SW0	P25

3 พอร์ต RS-232C

พอร์ต RS-232C จะใช้ไอซีเบอร์ ICL3232 หรือ MAX3232 ดังรูปที่ 13 โดยการต่อเข้าขา p128 และ p129 แสดงดังรูปที่ 14



รูปที่ 13 พอร์ต RS-232C



รูปที่ 14 แสดงการต่อพอร์ต RS-232C กับ FPGA

4. Misc

4.1 DC Adaptor input jack เป็นหัวต่อไฟเลี้ยงเพื่อป้อนให้แก่บอร์ดในการทำงาน ต่ออยู่กับอะแดปเตอร์ที่มีไฟออกมาเป็น 7V – 9V โดยมีขั้วด้านในเป็น บวก “+” ด้านนอกเป็น ลบ “-”

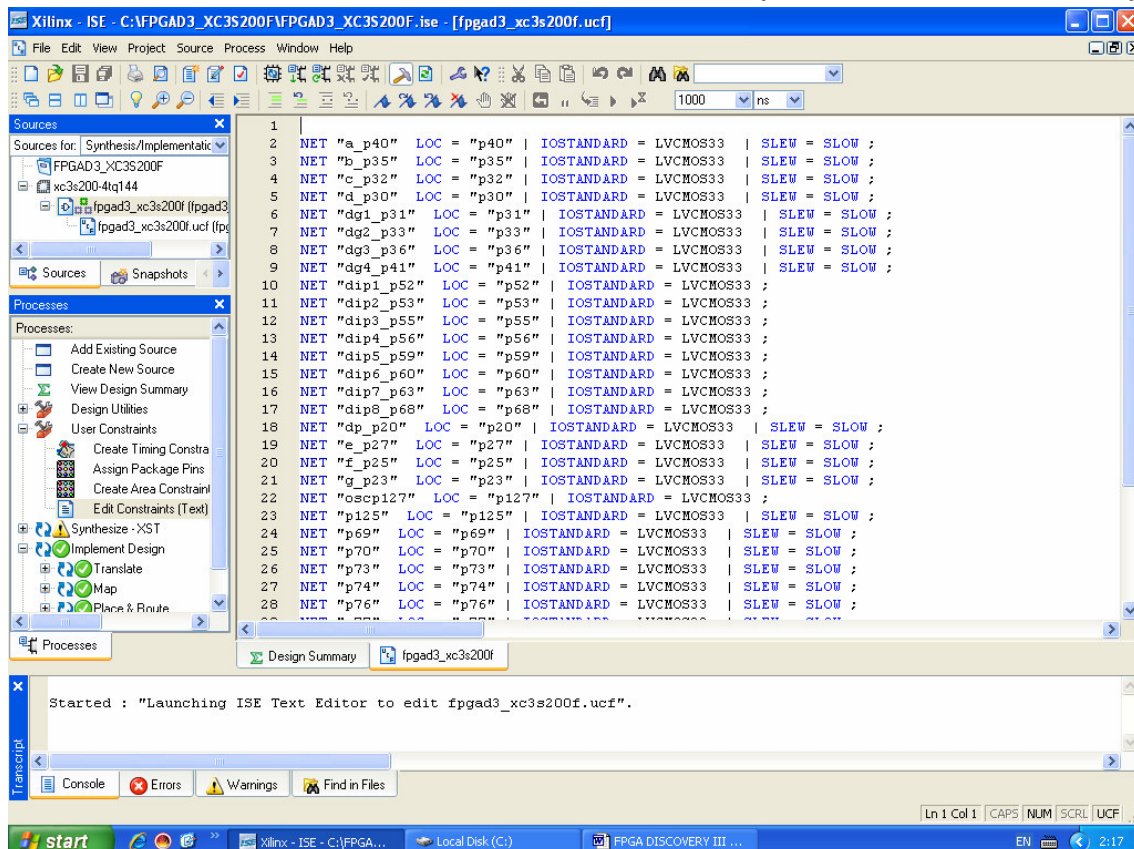
4.2 Power switch เป็นสวิตช์ที่ใช้ปิดเปิดไฟเลี้ยงให้กับบอร์ด

4.3 Power LED เป็นไดโอดเปล่งแสงว่าในขณะนั้นๆ มีไฟเลี้ยงบอร์ดอยู่หรือไม่

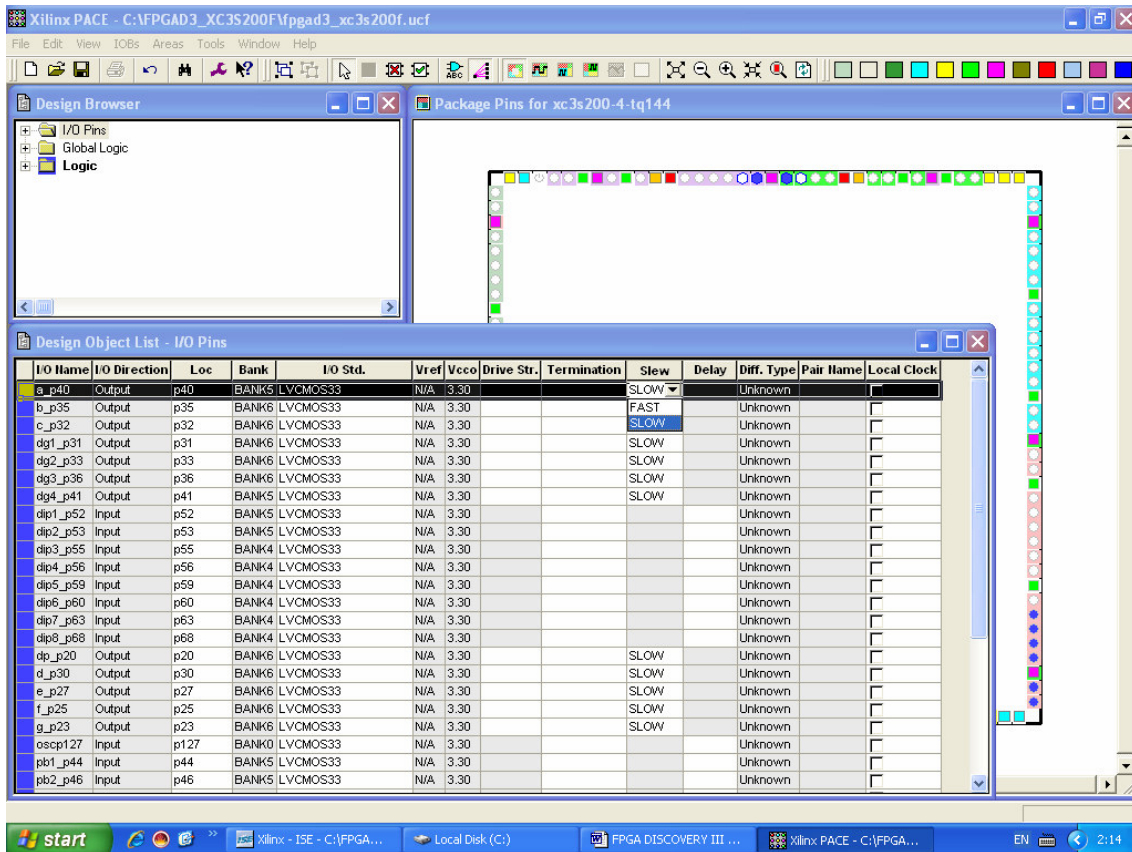
4.4 JUMPER J2 เป็น Jumper ที่ต่อกับPB5 และ PB6 เข้ากับ FPGA (เมื่ออยู่ที่ตำแหน่ง S) ไว้สำหรับทดสอบ การปล่อยขาอินพุตลอยไว้ ซึ่งจะให้ลอจิกไม่แน่นอนเช่นเดียวกับไอซี CMOSทั่วไป ซึ่งจะไม่เหมือนกับ TTL ที่ปล่อยขาลอยจะเป็นลอจิก “1” ดังนั้น จะต้องพูลอัพหรือพูลดาวน์ขาอินพุตนั้นไว้ (ด้วย R=4.7-10k) เพื่อบังคับให้เป็นลอจิก “0” หรือ “1” ยกเว้นกรณีไม่กำหนดค่าเพื่อใช้งานจึงจะปล่อยขาลอยได้ซึ่งจะแตกต่างจากไอซี CMOS ที่ต้องต่อขาอินพุตเข้ากับ Vcc หรือ Ground หรือต้องพูลอัพหรือพูลดาวน์

4.5 ตัวอย่างการ โปรแกรมเอาต์พุตของ FPGA

ไฟเลี้ยง Vcco ของบอร์ดนี้เป็น 3.3V ดังนั้น I/O ทั้งหมดจะเป็นระบบ 3.3Vเท่านั้น ซึ่งสามารถเลือก I/O ได้เพียง 2 ชนิด คือ LVC MOS33 หรือ LV TTL ดังตารางที่ 2 ตัวอย่างการ โปรแกรมเอาต์พุตของ FPGA เป็นแบบ LV TTL (IOSTANDARD = LV TTL) และเป็นแบบ Slow slew Rate (SLEW = SLOW) ใน Edit Constraints(Text)แสดงดังรูปที่15 หรือในหน้าต่าง PACE แสดงรูปที่ 16



รูปที่15 การ โปรแกรมเอาต์พุตของ FPGA เป็นแบบ LVCMOS และเป็นแบบ Slow slew Rate ใน Edit Constraints(Text)



รูปที่ 16 การ โปรแกรมเฮดฟุตของ FPGA เป็นแบบ LVCMOS และเป็นแบบ Slow slew Rate ในหน้าต่าง PACE

Table 2 Single-Ended I/O Standards (Values in Volts)

Signal Standard	Vcco		V _{REF} for Inputs ⁽¹⁾	Board Termination Voltage (V _{TT})
	For Outputs	For Inputs		
GTL	Note 2	Note 2	0.8	1.2
GTLP	Note 2	Note 2	1	1.5
HSTL_I	1.5	-	0.75	0.75
HSTL_III	1.5	-	0.9	1.5
HSTL_I_18	1.8	-	0.9	0.9
HSTL_II_18	1.8	-	0.9	0.9
HSTL_III_18	1.8	-	1.1	1.8
LVCMOS12	1.2	1.2	-	-
LVCMOS15	1.5	1.5	-	-
LVCMOS18	1.8	1.8	-	-
LVCMOS25	2.5	2.5	-	-
LVCMOS33	3.3	3.3	-	-
LVTTTL	3.3	3.3	-	-

4.6 คอนเนคเตอร์ JTAG และจัมเปอร์ J1

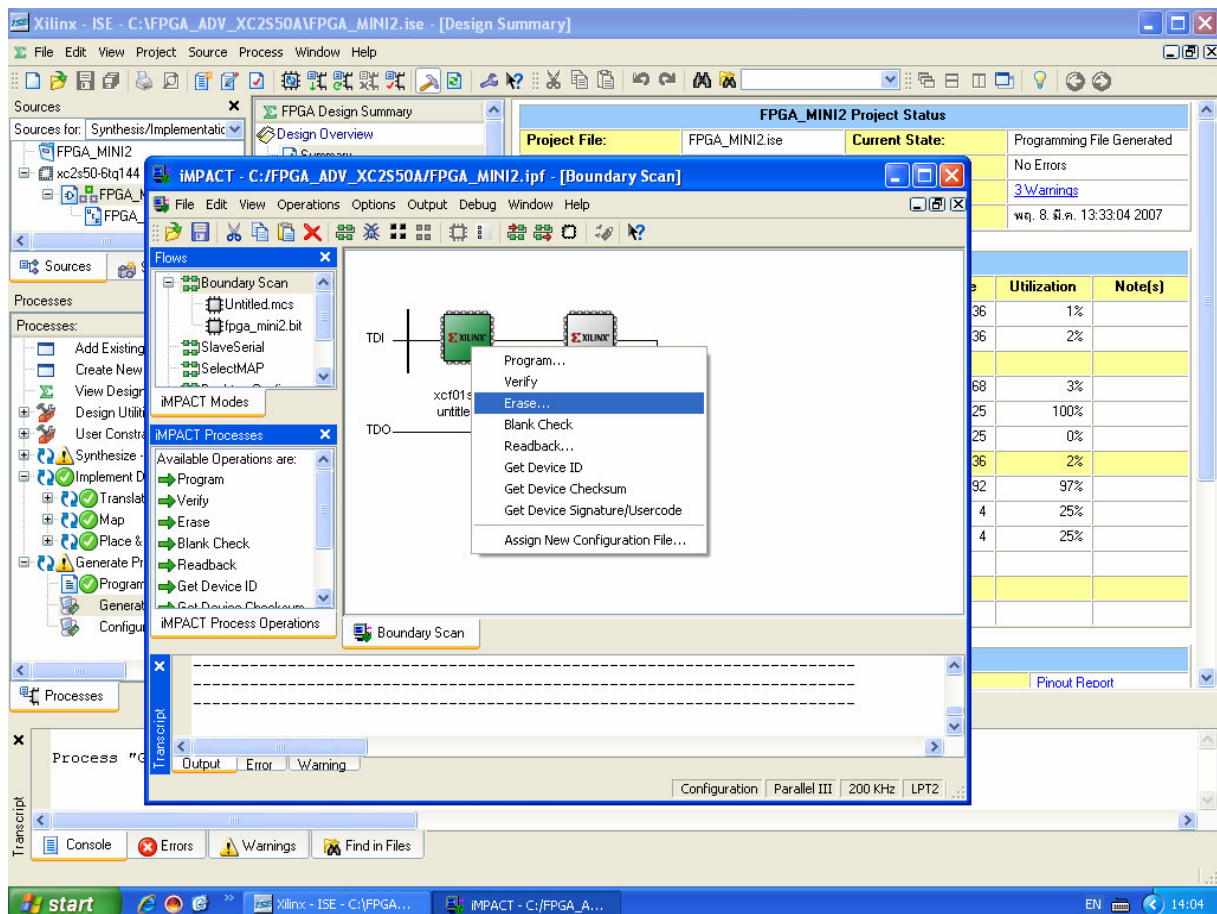
คอนเนคเตอร์ JTAG ใช้สำหรับต่อสายคาวาน์โหนด (JTAG Cable) เข้ากับพอร์ตขนาน(Printer Port)ของคอมพิวเตอร์ เพื่อโปรแกรมข้อมูลวงจร (Configuration data) ลง FPGA บอร์ดทดลองนี้ถูกออกแบบให้โปรแกรม Serial Flash PROM และ FPGA ใน JTAG Mode (Boundary Scan Mode) ได้โดยตรงโดยใช้สายคาวาน์โหนด และใช้ Master Serial Mode เพื่อให้ Serial Flash PROM สามารถโปรแกรม FPGA อย่างอัตโนมัติทุกครั้งที่จ่ายไฟเลี้ยง ใน Master Serial Mode นั้นต้องเซต Jumper J1 ให้ M0, M1, M2 = 000 ตามที่สรุปไว้ในตารางที่ 3 สายคาวาน์โหนดสามารถใช้ได้ทั้งของ Xilinx หรือของบริษัทเอเพกซ์เองก็ได้ (ให้มาพร้อมกับบอร์ดทดลองอยู่แล้ว) และไม่ว่าจะตั้งค่า M0, M1 และ M2 อยู่ในโหมดใดก็ยังสามารถโปรแกรมโดยใช้สาย JTAG ได้

ตารางที่ 3 รายการเซตโหมดในการโปรแกรม FPGA

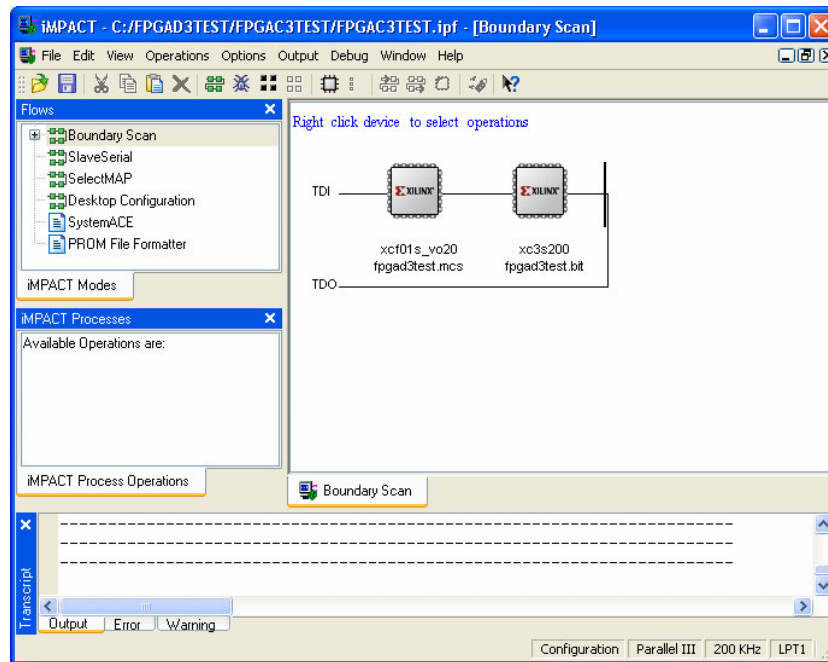
Configuration Mode	M0	M1	M2
Master Serial	0	0	0
Slave Serial	1	1	1
Master Parallel	1	1	0
Slave Parallel	0	1	1
JTAG	1	0	1

การโปรแกรม FPGA โดยตรงผ่านทางสาย JTAG นั้นผู้ใช้จะต้องลบข้อมูลใน Flash PROM ทิ้งก่อนเสมอ ตัวอย่างแสดง
 ดังรูปที่ 17 การไม่ลบข้อมูลจากรอออกจาก Flash PROM ก่อนอาจทำให้ข้อมูลที่โปรแกรมลง FPGA ไม่สมบูรณ์ (โดยไม่มีการ
 เตือนว่าเกิดข้อผิดพลาด) เพราะ FPGA จะถูกเซตอยู่ใน Master Serial Mode จึงถูกโปรแกรมจาก Flash PROM อย่างอัตโนมัติ
 เรียบร้อยแล้วทันทีที่เริ่มต้นจ่ายไฟเลี้ยง

การโปรแกรมวงจรลง FPGA นั้นเราจะต้องสร้างไฟล์พร้อมที่จะโปรแกรมลง Platform Flash PROM และ FPGA ก่อน
 จากนั้นแล้วทำการต่อสาย JTAG และต่อไฟเลี้ยงเข้าบอร์ดแล้วทำการดาวน์โหลดวงจรที่ต้องการลงสู่ Flash PROM และชิพ FPGA
 ตามลำดับ ซึ่งขั้นตอนดาวน์โหลดนั้นที่จอกคอมพิวเตอร์จะปรากฏขึ้นทั้ง 2 ตัวพร้อมกันดังรูปที่ 18 เพราะมีการออกแบบให้ในโหมด
 JTAG ต่อถึงกันแบบลูกโซ่เพื่อสะดวกเมื่อโปรแกรม เราจึงสามารถเลือกที่จะดาวน์โหลดข้อมูลลงชิพทั้งสองตัวหรือตัวใดตัว
 หนึ่งก็ได้ และเนื่องจาก Platform Flash PROM และ FPGA ต่อกันในโหมด Master serial (M0 , M1 , M2 = 0) อีกด้วย ดังนั้นทุก
 ครั้งที่เริ่มจ่ายไฟให้บอร์ดทดลอง FPGA จะดาวน์โหลดข้อมูลจาก Platform Flash PROM มาที่ FPGA อย่างอัตโนมัติ



รูปที่ 17 แสดงการลบข้อมูลใน Flash PROM ทิ้งก่อนการโปรแกรม FPGA โดยตรงผ่านทางสาย JTAG



รูปที่ 18 ขั้นตอนดาวน์โหลดที่จอกอมพิวเตอร์จะปรากฏชิพ Flash PROM และ FPGA พร้อมกัน