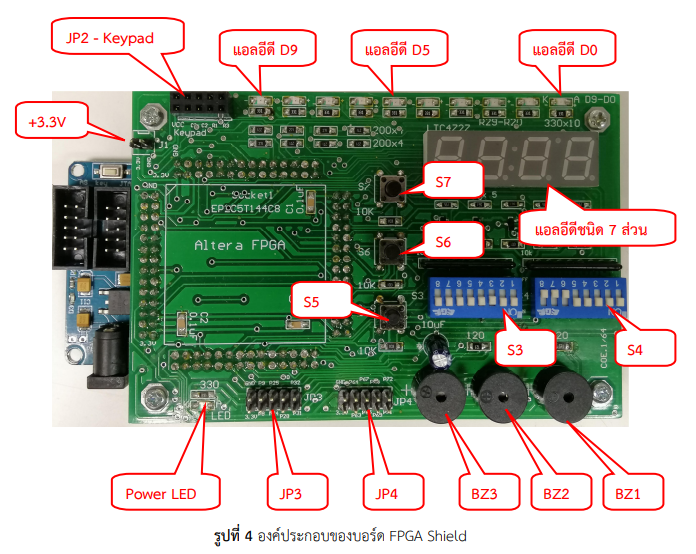
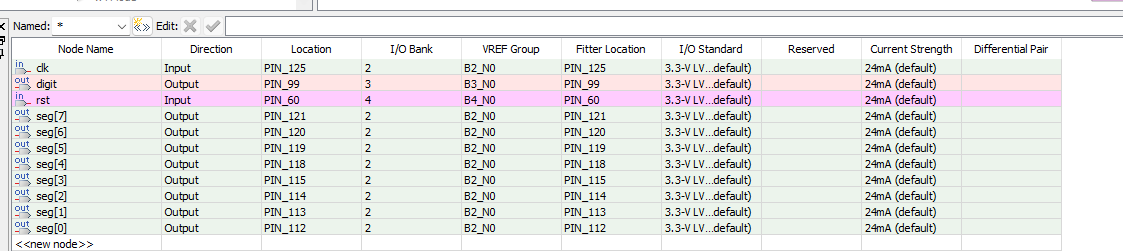
**รายงานผลการทดลอง : 3HA01: Introduction to FPGA design**

****

**c**

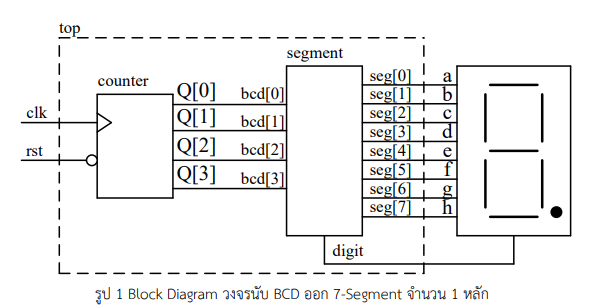
รูปที่1 Layout อุปกรณ์ย่อยของบอร์ด

**ผลการทดลอง Check Point #1**

|  |  |
| --- | --- |
| PIN | คือ |
| 125 | Switch5 [Clock] input |
| 99 | Output ที่ Digit #4 |
| 60 | S3 ซ้ายสุด |
| Seg[7:0] | แสดงผล Segmentย่อย |

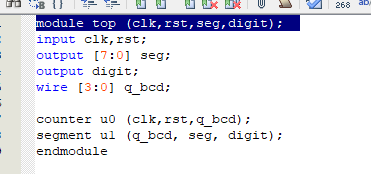
รูปที่2 Pin Planner

\*ดูแต่ละPinคืออะไรได้ที่ Document : คู่มือการใช้งานบอร์ด Altera FPGA Shield-v03.doc



รูปที่ 3 Block Diagram

**การออกแบบวงจร**

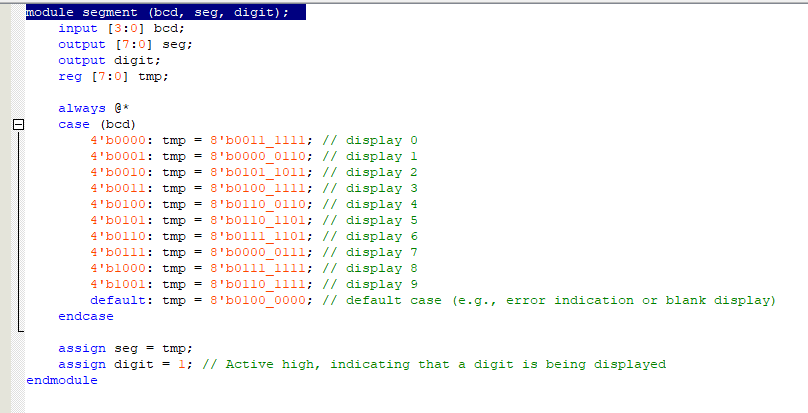
1. **Module หลัก Top.v**

รูปที่ 4 Top.v

* + counter u0 (clk,rst,q\_bcd); *// เชื่อมสัญญาณตาม Block Diagram*
  + segment u1 (q\_bcd,seg,digit); *// เชื่อมสัญญาณตาม Block Diagram*

1. **Module Counter.v :** ไม่ต้องทำการแก้ไข
2. **Module Segment.v :** Segment เป็น Segment **แ**บบ common Cathode

จ่าย 1 เพื่อให้ Segmentติด

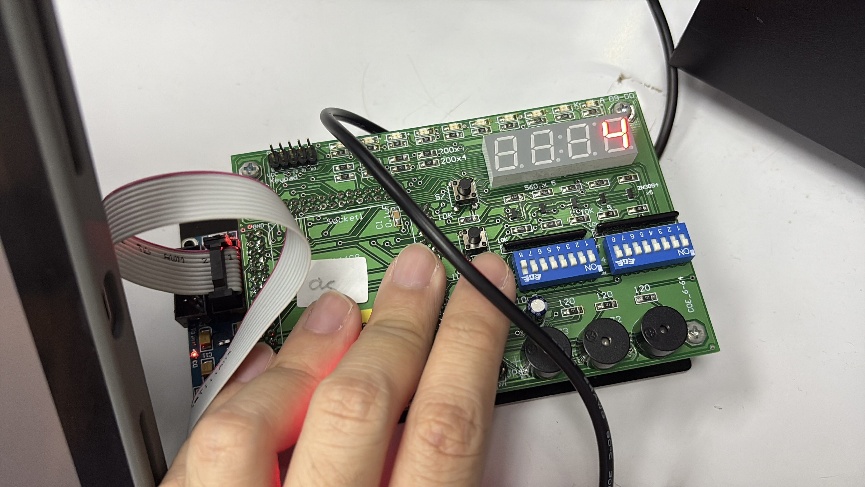


รูปที่ 5 Segment.V

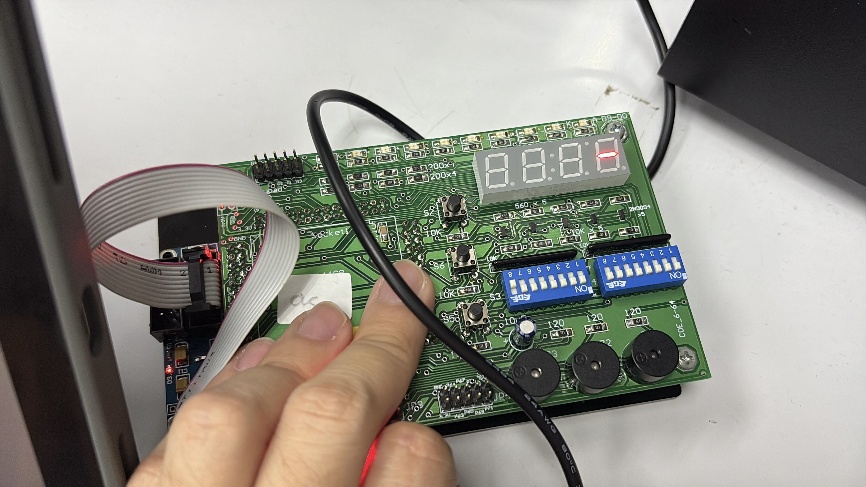
**Counter และ 7-Segment Display บน FPGA Board**

1. **การทดสอบ Counter**

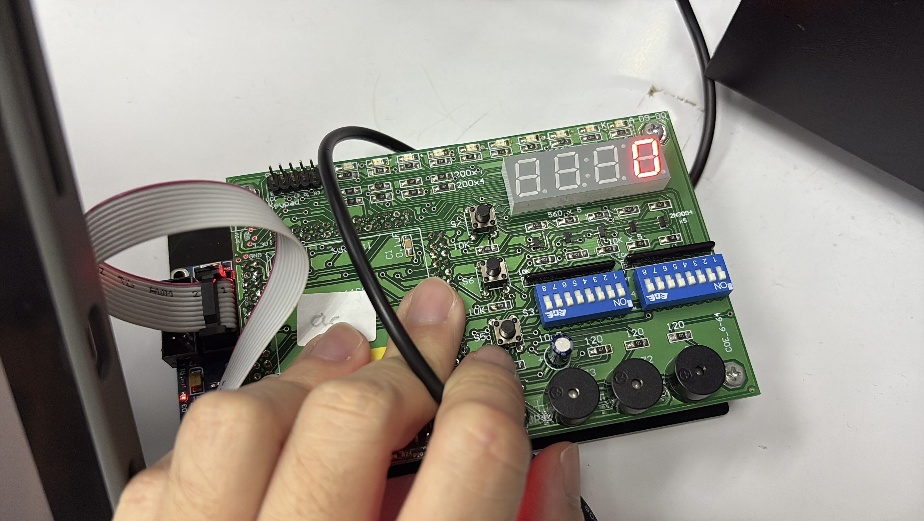
* **ขั้นตอนการทดลอง**: กดปุ่ม Switch S5 เพื่อให้สัญญาณ clock เข้าสู่วงจร counter
* **ผลที่ได้**: 7-segment display แสดงตัวเลขเพิ่มขึ้นตามลำดับ ดังนี้
  + เริ่มต้นแสดง "0"
  + กดปุ่ม S5 ครั้งที่ 1 แสดง "1"
  + กดปุ่ม S5 ครั้งที่ 2 แสดง "2"
  + กดปุ่ม S5 ครั้งที่ 3 แสดง "3"
  + ดำเนินการต่อไปจนถึง "9"
  + เมื่อกดต่อไปจะแสดงเฉพาะ segment g ("-") เนื่องจากเกิดการนับเกินค่า 9

****

รูปที่ 6 แสดงผล เลข4 เมื่อ กด SW5 เป็นครั้งที่ 5 เพื่อเพื่อ Counter



รูปที่ 7 แสดงผล “-” เมื่อ กด SW5 เป็นครั้งที่ 11 เพื่อเพื่อ Counter



รูปที่ 8 แสดงผล “-” เมื่อ กด SW5 เป็นครั้งที่ 5 เพื่อเพื่อ Counter

**2. การทดสอบ Reset Function**

* **ขั้นตอนการทดลอง**:
  1. ให้ counter นับไปยังค่าใดค่าหนึ่ง (เช่น "5" หรือ "8")
  2. โยก DipSwitch ที่เชื่อมต่อกับขา rst
* **ผลที่ได้**:
  1. 7-segment display แสดง "0" ทันที
  2. วงจรพร้อมสำหรับการเริ่มนับใหม่
  3. เมื่อโยก DipSwitch กลับไปที่ตำแหน่งเดิม สามารถกดปุ่ม S5 เพื่อนับต่อได้ปกติ

**3. การทดสอบ 7-Segment Display Pattern**

จากโค้ดในส่วน segment module พบว่า:

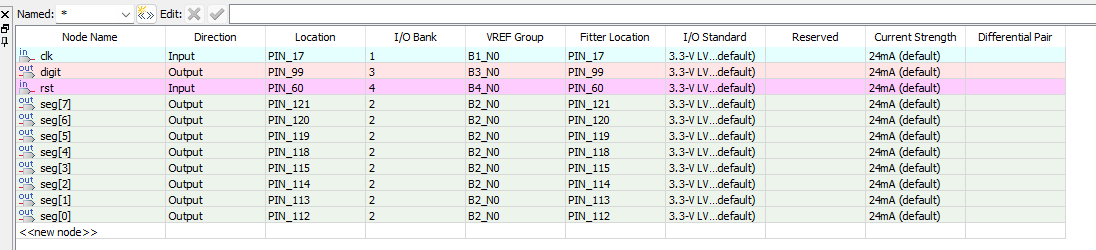
* แต่ละตัวเลข 0-9 จะแสดงผลตาม pattern ที่กำหนด
* ตัวอย่างการแสดงผล:
  + "0": แสดง segments a,b,c,d,e,f (ไม่แสดง g)
  + "1": แสดง segments b,c
  + "2": แสดง segments a,b,g,e,d
  + "9": แสดง segments a,b,c,d,f,g

**สรุปผลการทดลอง**

1. **Counter Function**: ทำงานได้อย่างถูกต้อง นับเพิ่มทีละ 1 ทุกครั้งที่กดปุ่ม S5
2. **7-Segment Display**: แสดงผลตัวเลข 0-9 ได้อย่างชัดเจน และแสดง "-" เมื่อนับเกิน
3. **Reset Function**: ทำงานได้อย่างสมบูรณ์ สามารถรีเซ็ตค่าเป็น "0" และเริ่มนับใหม่ได้

**Check Point #2: Automatic Counter with 1Hz Clock**

|  |  |
| --- | --- |
| PIN | คือ |
| 17 | 50MHz Clock จากบอร์ด |
| 99 | Output ที่ Digit #4 |
| 60 | S3 ซ้ายสุด Reset |

****

รูปที่9 Pin Planner

**การทำงานของ Clock Divider**

* รับสัญญาณนาฬิกา 50 MHz จาก PIN\_17
* หารความถี่ลงเหลือประมาณ 1 Hz โดยใช้ parameter max\_count2 = 50000000/2
* ใช้ 22-bit counter สำหรับนับ
* สร้าง toggle output เมื่อ counter ถึง 0

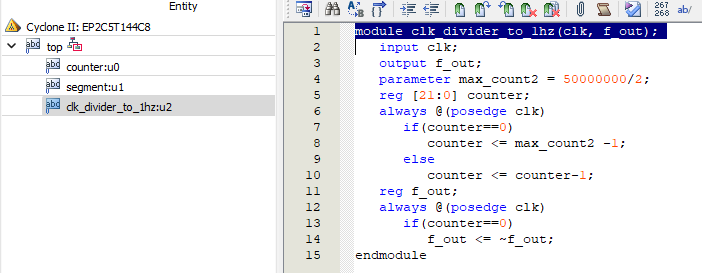
**ผลการทดลอง Check Point #2**

**การแสดงผลที่ปรับปรุงแล้ว**

* การนับอัตโนมัติ: 7-Segment แสดงตัวเลขเปลี่ยนอัตโนมัติทุกๆ 1 วินาที
* ช่วงการนับ: นับเฉพาะ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 เท่านั้น
* การวนรอบ: เมื่อนับถึง 9 แล้วจะกลับไปนับ 0 ทันที (ไม่แสดงเครื่องหมาย "-")
* ไม่ต้อว Switch Bouncing: เนื่องจากใช้สัญญาณนาฬิกาที่เสถียร นับครบทุกเลข

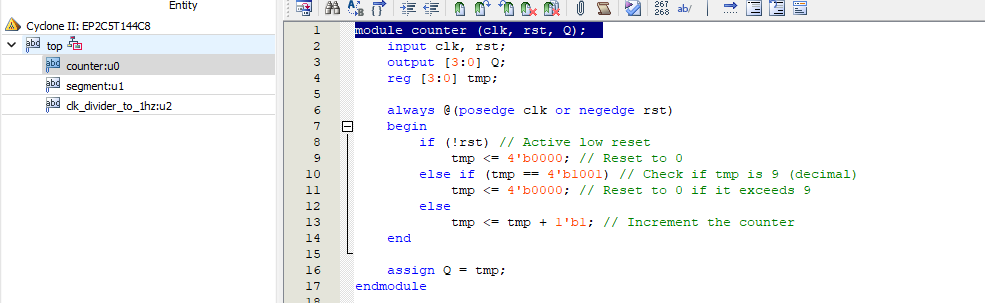
**การทำงานของ Reset Function**

* ฟังก์ชัน Reset ยังคงทำงานได้: เมื่อโยก DIP Switch ที่ต่อกับขา rst
* Reset เป็น 0: 7-Segment จะแสดงเลข 0 ทันที
* กลับมานับต่อได้: เมื่อปล่อย reset วงจรจะเริ่มนับใหม่จาก 0

1. **เพิ่ม Clock Divider Module**

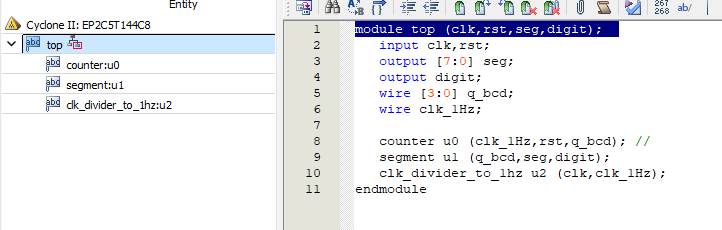
รูปที่10 Code Clock\_divider\_1Hz.v

1. **ปรับปรุง Counter Module :** เพิ่มการตรวจ เมื่อถึง 9 ให้ชี้ไปที่ 0

****

รูปที่11 Code Counter.v

1. ปรับปรุง Top Module เพิ่ม
   * counter u0 (clk\_1Hz,rst,q\_bcd); *// ใช้ clk\_1Hz แทน clk*
   * clk\_divider\_to\_1hz u2 (clk,clk\_1Hz);

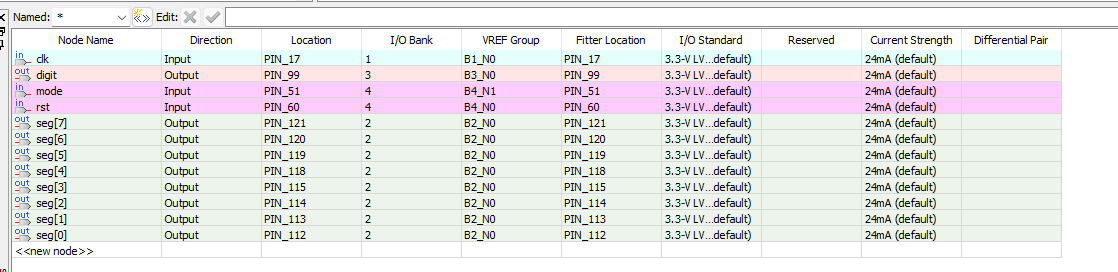


รูปที่12 Code Top.v

* + Vdo Link ผลการทดลอง Checkpoint #2 : <https://youtu.be/cdxIHFyhgvs>

**Check Point #3: Up/Down Counter with Mode Control**

|  |  |
| --- | --- |
| PIN | คือ |
| 17 | 50MHz Clock จากบอร์ด |
| 99 | Output ที่ Digit #4 |
| 51 | S3 ขวาสุด Mode Change |
| 60 | S3 ซ้ายสุด Reset |

****

รูปที่13 Pin Planner

**ผลการทดลอง Check Point #3**

**การทดสอบ Count Up Mode**

* ตั้ง DIP switch ที่ต่อกับ PIN\_51 เป็น logic high (1)
* ผลลัพธ์: 7-Segment นับขึ้น 0→1→2→3→4→5→6→7→8→9→0→...
* ความเร็ว: เปลี่ยนทุกๆ 1 วินาที (ตาม clock divider)

**การทดสอบ Count Down Mode**

* ตั้ง DIP switch ที่ต่อกับ PIN\_51 เป็น logic low (0)
* ผลลัพธ์: 7-Segment นับลง 9→8→7→6→5→4→3→2→1→0→9→...
* ความเร็ว: เปลี่ยนทุกๆ 1 วินาที (เหมือนกับ count up)

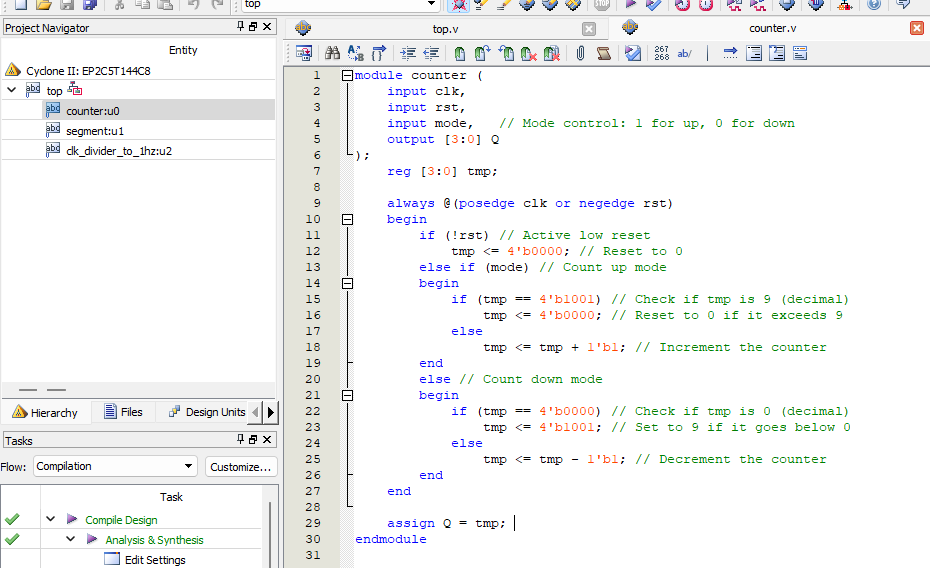
**การทดสอบ Reset Function**

* Reset ใน Count Up Mode: รีเซ็ตไป 0 แล้วนับขึ้นต่อ
* Reset ใน Count Down Mode: รีเซ็ตไป 0 แล้วนับลงไป 9, 8, 7...

**การทดสอบการเปลี่ยน Mode ระหว่างการทำงาน**

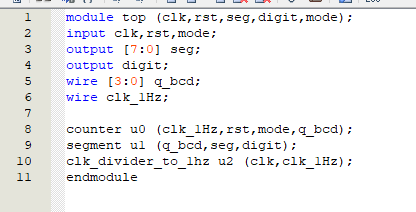
* เปลี่ยนจาก Up เป็น Down: เปลี่ยนทิศทางการนับทันทีจากตัวเลขปัจจุบัน
* เปลี่ยนจาก Down เป็น Up: เปลี่ยนทิศทางการนับทันทีจากตัวเลขปัจจุบัน

**1. ปรับปรุง Counter Module :** เพิ่ม mode เพื่อกำหนด นับขึ้น / นั่งลง



รูปที่14 Code counter

1. **ปรับปรุง Top Module :** เพิ่ม Mode control input + mode input



รูปที่15 Code counter

1. **Segment Module :** ไม่ต้องทำการแก้ไข
2. **Clock Divider Module :** ไม่ต้องทำการแก้ไข

Vdo Link ผลการทดลอง Checkpoint #2 : <https://youtu.be/5XNxqK7RQKA>