และเมื่อเป็นแบบ Logic High Disable คือ

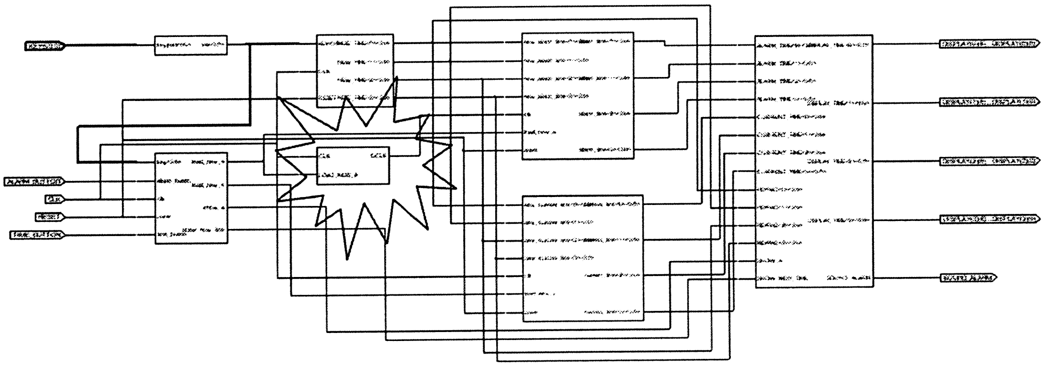
```
GClk_A <= not (Clk and GSignal0);

control0 : process (Clk,Reset)
begin
      if Reset = '1' then
            GSignal0 <= '0';
      elsif falling_edge(Clk) then
            GSignal0 <= Load_Alarm_Time;
      end if;
end process control0;
```

ผลการทดลองแสดงให้เห็นว่าสัญญาณ Gated_Clock (GClk_A) ซึ่งเป็นสัญญาณนาฬิกาใหม่ที่เข้าไปกระตุ้นการทำงานของวงจร Alarm Register มีลักษณะแตกต่างจากสัญญาณนาฬิกาปกติก่อนที่จะทำ Clock Gating อย่างชัดเจน

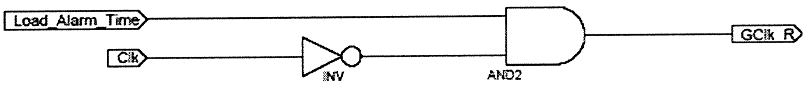
รายละเอียดของผลการทดลองของแต่ละชุดการทดลองเป็นดังนี้

การทดลอง ALR-01 (Logic Low Disable + Latch Free + Active High)

เมื่อทำการสังเคราะห์วงจรแล้วจะได้ผลดังแสดงในภาพที่ 23 วงจรที่ถูกล้อมด้วยกรอบคือวงจรควบคุมที่เพิ่มเข้ามาซึ่งมีรายละเอียดภายในดังแสดงในภาพที่ 24



ภาพที่ 23 วงจรที่เพิ่มเข้ามาใน Alarm Clock ตามการทดลอง ALR-01



ภาพที่ 24 รายละเอียดวงจรที่เพิ่มเข้ามาใน Alarm Clock ตามการทดลอง ALR-01

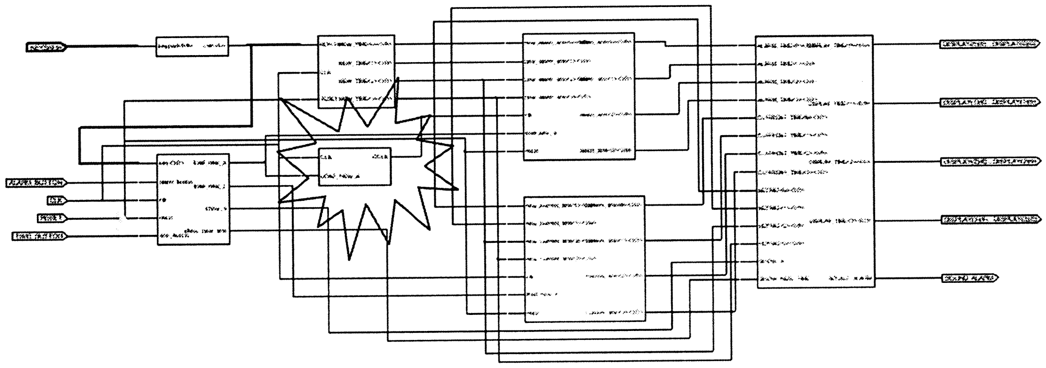
จากผลการทดลอง นอกจากที่สัญญาณนาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิม ผลการทำงานนั้นยังทำให้ Alarm Register เก็บค่าเวลาปลุกเร็วกว่าเดิมถึง 1 รอบสัญญาณนาฬิกา แต่อย่างไรก็ตามหากพิจารณาการทำงานโดยรวมแล้วจะเห็นว่า Alarm Register ทำการจำค่าเวลาปลุกใหม่เร็วขึ้นเพียงเล็กน้อยเท่านั้น ไม่ได้ทำให้เวลาปลุกเปลี่ยนค่าไป จึงอาจกล่าวได้ว่าการทำงานที่เร็วขึ้นเล็กน้อยของ Alarm Register นี้ไม่ได้มีผลให้เกิดการทำงานที่ผิดพลาดใด ๆ ขึ้นมา

<u>การทดลอง ALR-02 (Logic Low Disable + Latch Free + Active Low)</u>

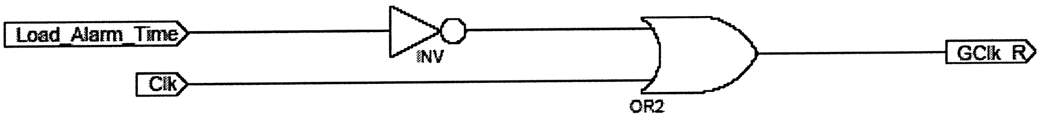
เมื่อทำการสังเคราะห์วงจรแล้วจะได้วงจรควบคุมที่เพิ่มเข้ามาซึ่งมีรายละเอียดเหมือนกับในการทดลอง ALR-01 ดังแสดงในภาพที่ 23 และ 24 จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมอย่างชัดเจนและไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้ Alarm Register ทำงานได้เหมือนเดิมทุกประการ

<u>การทดลอง ALR-03 (Logic High Disable + Latch Free + Active High)</u>

เมื่อทำการสังเคราะห์วงจรแล้วจะได้ผลดังแสดงในภาพที่ 25 วงจรที่ถูกล้อมด้วยกรอบคือวงจรควบคุมที่เพิ่มเข้ามาซึ่งมีรายละเอียดภายในดังแสดงในภาพที่ 26



ภาพที่ 25 วงจรที่เพิ่มเข้ามาใน Alarm Clock ตามการทดลอง ALR-03



ภาพที่ 26 รายละเอียดวงจรที่เพิ่มเข้ามาใน Alarm Clock ตามการทดลอง ALR-03

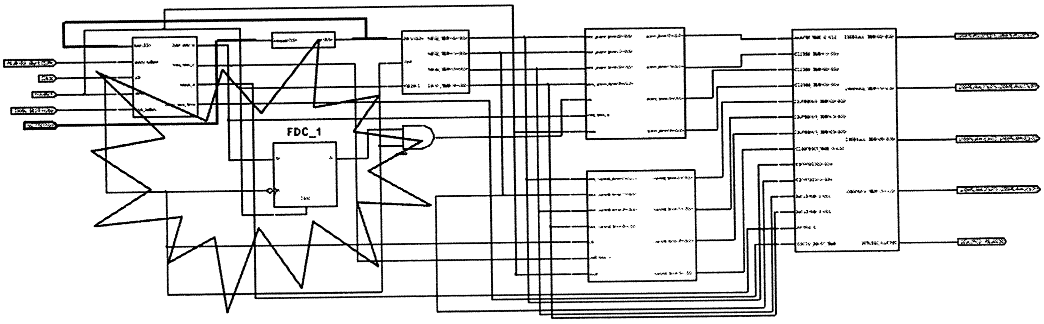
จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมอย่างชัดเจนและไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้ Alarm Register ทำงานได้เหมือนเดิมทุกประการ

<u>การทดลอง ALR-04 (Logic High Disable + Latch Free + Active Low)</u>

เมื่อทำการสังเคราะห์วงจรแล้วจะได้วงจรควบคุมที่เพิ่มเข้ามาซึ่งมีรายละเอียดเหมือนกับในการทดลอง ALR-03 ดังแสดงในภาพที่ 25 และ 26 จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมอย่างชัดเจนและไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้ Alarm Register เก็บค่าเวลาปลุกเร็วกว่าเดิมครึ่งรอบสัญญาณนาฬิกา

<u>การทดลอง ALR-05 (Logic Low Disable + Latch Base + Active High)</u>

เมื่อทำการสังเคราะห์วงจรแล้วจะได้ผลดังแสดงในภาพที่ 27 วงจรที่ถูกล้อมด้วยกรอบคือวงจรควบคุมที่เพิ่มเข้ามา



ภาพที่ 27 วงจรที่เพิ่มเข้ามาใน Alarm Clock ตามการทดลอง ALR-05

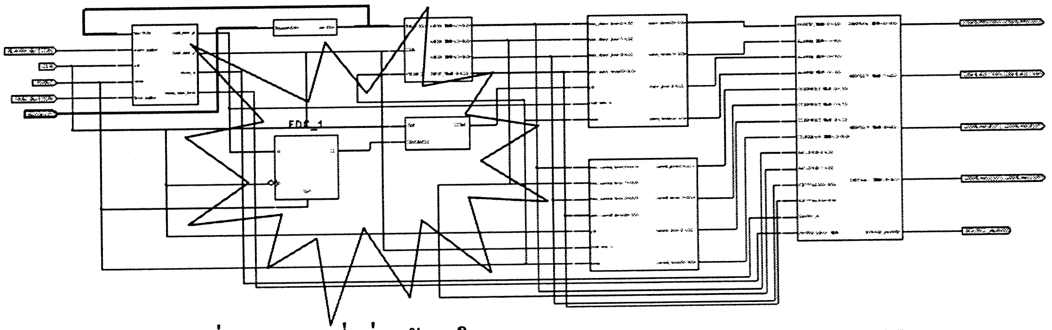
จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมอย่างชัดเจนและไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้ Alarm Register ทำงานได้เหมือนเดิมทุกประการ

<u>การทดลอง ALR-06 (Logic Low Disable + Latch Base + Active Low)</u>

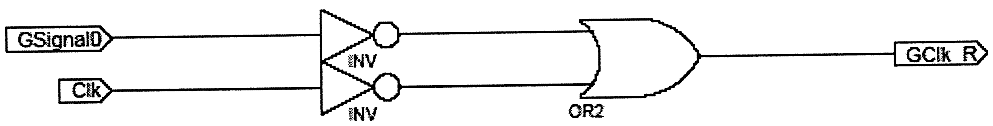
เมื่อทำการสังเคราะห์วงจรแล้วจะได้วงจรควบคุมที่เพิ่มเข้ามาซึ่งมีรายละเอียดเหมือนกับในการทดลอง ALR-05 ดังแสดงในภาพที่ 27 จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมอย่างชัดเจนและไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้ Alarm Register เกิดการทำงานที่ผิดพลาดดังแสดงในภาพที่ 30 จะเห็นว่า Alarm Register ไม่ได้เก็บค่านาฬิกาปลุกใหม่ ทั้งนี้เพราะช่วงที่วงจร Alarm Register ทำงานนั้น (สัญญาณนาฬิกาขาลง) สัญญาณ Load_Alarm_Time ก็หยุดการทำงานไปก่อนแล้ว

<u>การทดลอง ALR-07 (Logic High Disable + Latch Base + Active High)</u>

เมื่อทำการสังเคราะห์วงจรแล้วจะได้ผลดังแสดงในภาพที่ 28 วงจรที่ถูกล้อมด้วยกรอบคือวงจรควบคุมที่เพิ่มเข้ามาซึ่งมีรายละเอียดภายในดังแสดงในภาพที่ 29



ภาพที่ 28 วงจรที่เพิ่มเข้ามาใน Alarm Clock ตามการทดลอง ALR-07



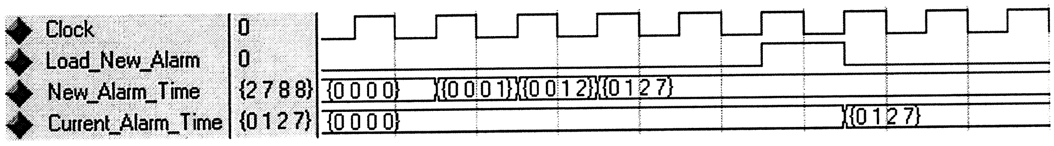
ภาพที่ 29 รายละเอียดวงจรที่เพิ่มเข้ามาใน Alarm Clock ตามการทดลอง ALR-07

จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมอย่างชัดเจนและไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้ Alarm Register เกิดการทำงานที่ผิดพลาดดังแสดงในภาพที่ 30 จะเห็นว่า Alarm Register ไม่ได้เก็บค่านาฬิกาปลุกใหม่ ทั้งนี้เพราะช่วงที่วงจร Alarm Register ทำงานนั้น (สัญญาณนาฬิกาขาขึ้น) สัญญาณ Load_Alarm_Time ก็หยุดการทำงานไปก่อนแล้ว

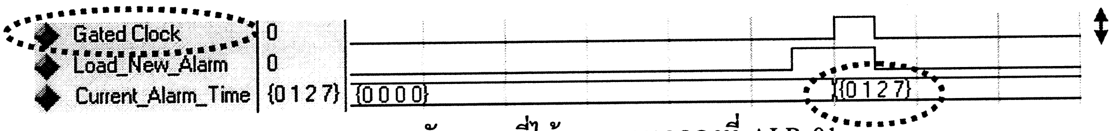
<u>การทดลอง ALR-08 (Logic High Disable + Latch Base + Active Low)</u>

เมื่อทำการสังเคราะห์วงจรแล้วจะได้วงจรควบคุมที่เพิ่มเข้ามาซึ่งมีรายละเอียดเหมือนกับในการทดลอง ALR-07 ดังแสดงในภาพที่ 28 และ 29 จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมอย่างชัดเจนและไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้ Alarm Register ทำงานได้เหมือนเดิมทุกประการ

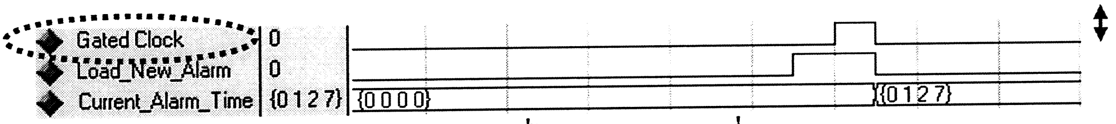
ผลการทดลองที่ได้จากการจำลองการทำงานซึ่งจะแสดงเฉพาะส่วนที่มีความเกี่ยวข้องกับการทดลอง โดยจะแสดงเฉพาะผลการทดลองการทำงานในโหมดการทำงานของวงจรในช่วงตั้งเวลาปลุก ซึ่งมีรายละเอียดดังต่อไปนี้



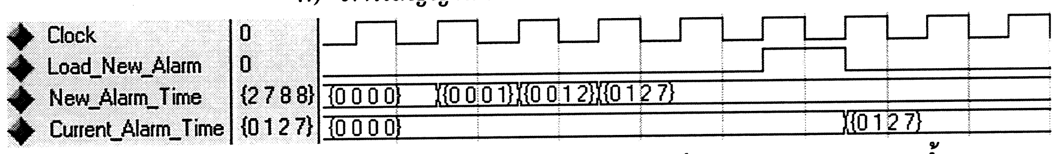
ก) ภาพสัญญาณการทำงานของ Alarm Register ก่อนทำ Clock Gating



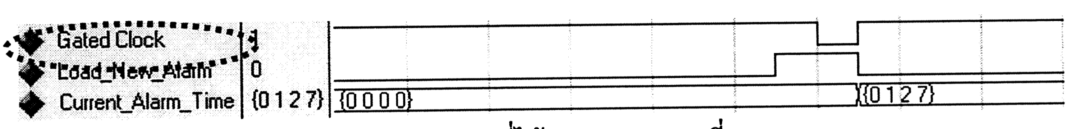
ข) ภาพสัญญาณที่ได้จากการทดลองที่ ALR-01



ค) ภาพสัญญาณที่ได้จากการทดลองที่ ALR-02

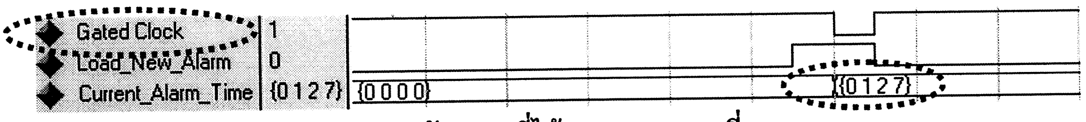


ง) ภาพสัญญาณการทำงานของ Alarm Register ก่อนทำ Clock Gating (ซ้ำ)

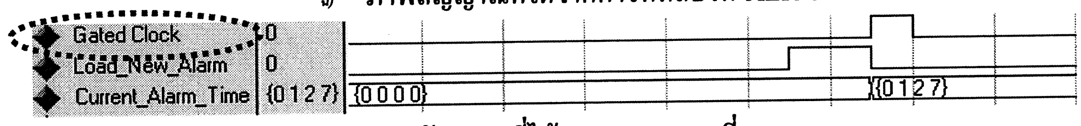


จ) ภาพสัญญาณที่ได้จากการทดลองที่ ALR-03

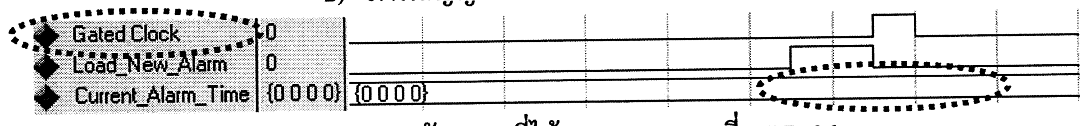
ภาพที่ 30 ภาพสัญญาณการทำงานของ Alarm Register ในช่วงตั้งเวลาปลุก



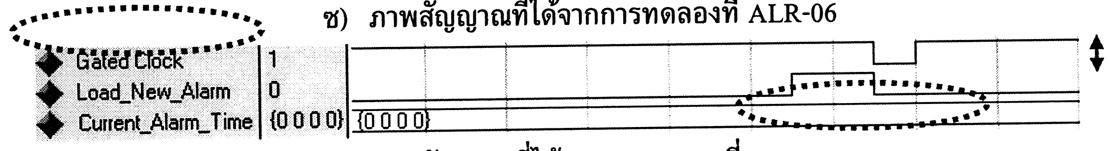
ฉ) ภาพสัญญาณที่ได้จากการทดลองที่ ALR-04



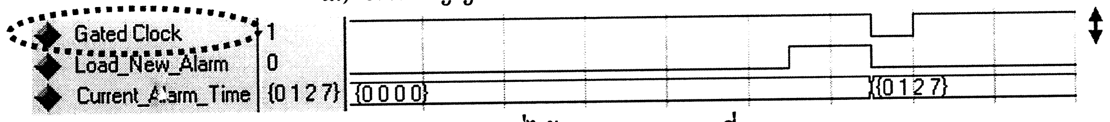
ช) ภาพสัญญาณที่ได้จากการทดลองที่ ALR-05



ซ) ภาพสัญญาณที่ได้จากการทดลองที่ ALR-06



ฌ) ภาพสัญญาณที่ได้จากการทดลองที่ ALR-07



ญ) ภาพสัญญาณที่ได้จากการทดลองที่ ALR-08

ภาพที่ 30 ภาพสัญญาณการทำงานของ Alarm Register ในช่วงตั้งเวลาปลุก (ต่อ)

ผลจากการวิเคราะห์การใช้พลังงานโดยใช้ XPower พบว่าการใช้พลังงานเป็นไปตามที่ได้ตั้งสมมุติฐานไว้โดยมีรายละเอียดดังตารางที่ 3

ตารางที่ 3 ผลการวิเคราะห์การใช้พลังงานโดยใช้ XPower ของ Alarm Register

การทดลอง / รายละเอียด	ต้นแบบ	ALR01	ALR02	ALR03	ALR04	ALR05	ALR06	ALR07	ALR08
Clock Power (mW)	1.36	1.26	1.28	1.22	1.28	1.26	1.31	1.26	1.23
Inputs Power (mW)	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15
Logic Power (mW)	0.00	0.07	0.07	0.07	0.07	0.07	0.07	0.07	0.07
Outputs Power (mW)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Signals Power (mW)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Summary (mW)	1.52	1.49	1.51	1.44	1.51	1.49	1.53	1.48	1.45
Glitch	✘	✘	✘	✘	✘	✘	✘	✘	✘
การทำงาน	☺	😐 A	☺	☺	😐 A	☺	☹	☹	☺

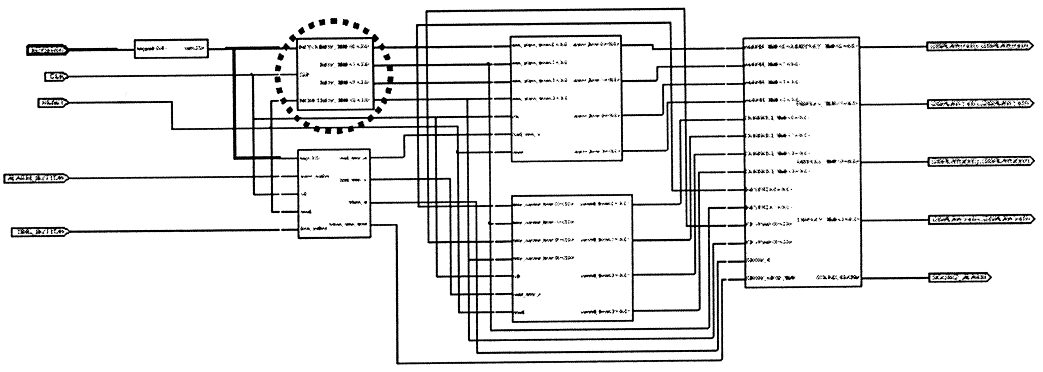
****หมายเหตุ :** ☺ = ปกติ 😐A = เร็วกว่าปกติ ☹ = ผิดปกติ

1.3 การทดลองประยุกต์ใช้เทคนิค Clock Gating ในส่วน Keyboard Buffer

เมื่อพิจารณาพฤติกรรมการทำงานจากโครงสร้างเดิมของวงจร Alarm Clock ซึ่งได้อธิบายไว้แล้วในบทที่ 2 จะสังเกตเห็นว่าในส่วน Keyboard Buffer ซึ่งเป็นส่วนที่ใช้พักค่าเวลาที่มีการป้อนเข้ามาเพื่อใช้ตั้งเวลาปลุกหรือเพื่อตั้งเวลาปัจจุบันจะมีการทำงานเฉพาะในช่วงเวลาที่มีการกดปุ่มตัวเลขเท่านั้นดังแสดงในภาพที่ 30ก) จุดนี้จึงเป็นโอกาสที่จะนำเทคนิค Clock Gating เข้ามาใช้ตัดสัญญาณนาฬิกาเพื่อลดการใช้พลังงานในส่วนนี้

แต่เนื่องจากสัญญาณควบคุมนี้เป็นสัญญาณที่ได้จากการกดปุ่มซึ่งไม่มีรูปแบบการทำงานที่ตายตัวซึ่งจะส่งผลเสียต่อการทำงานโดยรวมเมื่อมีการทำ Clock Gating ดังนั้นจึงต้องใช้สัญญาณ Show_New_Time ซึ่งเป็นสัญญาณภายในที่ทำงานสอดคล้องกันในช่วงที่มีการกดปุ่มตัวเลขเป็นสัญญาณควบคุมร่วมด้วย

ภาพที่ 31 แสดงถึงโครงสร้างวงจร Alarm Clock เดิมก่อนที่จะมีการทดลองซึ่งถูกสังเคราะห์ขึ้นมาจากโปรแกรม ISE 6.3i โดยส่วนที่เป็น Keyboard Buffer คือส่วนที่ถูกเน้นในกรอบ



ภาพที่ 31 วงจร Alarm Clock เดิม (Keyboard Buffer)

ในการทดลองแต่ละชุดการทดลองนั้น มีการประยุกต์ใช้ Clock Gating (GClk_K) แทนสัญญาณนาฬิกา (Clk) เดิม โดยปรับ Code ใน Alarm_Clock.vhd ของแต่ละชุดการทดลองดังนี้

```
U_KEYPAD_BUFFER :KEYPAD_BUFFER
port map (  Clk       =>    GClk_K,      -- replace Clk with GClk_K
            Key       =>    Key,
            Reset     =>    Reset,
            New_Time  =>    Show_New_Time
          );
```

เนื่องจากชุดการทดลองแต่ละชุด (ALK-01 ถึง ALK-08) มีการประยุกต์ใช้ Clock Gating ที่แตกต่างกันไป ทำให้ Code ของการได้มาซึ่งสัญญาณ GClk_K ใน Alarm_Clock.vhd ของแต่ละชุดการทดลองจึงมีการปรับที่แตกต่างกันไปดังนี้

ในกรณีของ Active High (กรณีที่เป็นเลขคี่) สัญญาณนาฬิกาจะทำงานที่ขอบขาขึ้น (Rising Edge) แต่สำหรับกรณี Active Low (กรณีที่เป็นเลขคู่) จะต้องมีการปรับจากสัญญาณนาฬิกาที่เคยทำงานที่ขอบขาขึ้น ให้ทำงานที่ขอบขาลง (Falling Edge) ดังนี้

```
        elsif (Clk'event and Clk='0') then
```

จากนั้น Code ที่เหลือของแต่ละคู่ชุดทดลอง (คู่ 01 และ 02 คู่ 03 และ 04 คู่ 05 และ 06 และคู่ 07 และ 08) จะมีลักษณะเหมือนกัน

ส่วนข้อแตกต่างของ Code ที่มีลักษณะ Logic Low Disable และ Logic High Disable คือ ในช่วงที่สัญญาณ Clock Gating ไม่ทำงานจะมีค่าเป็น 0 และ 1 ตามลำดับ ซึ่งหมายถึงการเป็นนิเสธซึ่งกันและกันนั่นเอง และสุดท้ายคือข้อแตกต่างของ Code ที่มีลักษณะ Latch Free และ Latch Base คือ การปรับ Code ในส่วนของ GClk_K ให้มี Latch หรือ ไม่มี Latch นั่นเอง Code ที่ถูกปรับแล้วจะมีลักษณะที่แตกต่างกันดังนี้

สำหรับแบบ Latch Free เมื่อเป็นแบบ Logic Low Disable คือ

```
GClk_K <= not Clk when Key /= 10 or Show_New_Time = '1'
          else '0';
```

และเมื่อเป็นแบบ Logic High Disable คือ

```
GClk_K <= Clk when Key /= 10 or Show_New_Time = '1'
          else '1';
```

และสำหรับแบบ Latch Base เมื่อเป็นแบบ Logic Low Disable คือ

```
GClk_K <= Clk and GSignal4;
GSignal3 <= '1' when Key /= 10 or Show_New_Time = '1' else '0';
control3 : process (Reset,Clk)
begin
      if Reset = '1' then
            GSignal4 <= '0';
      elsif falling_edge(Clk) then
            GSignal4 <= GSignal3;
      end if;
end process control3;
```

และเมื่อเป็นแบบ Logic High Disable คือ

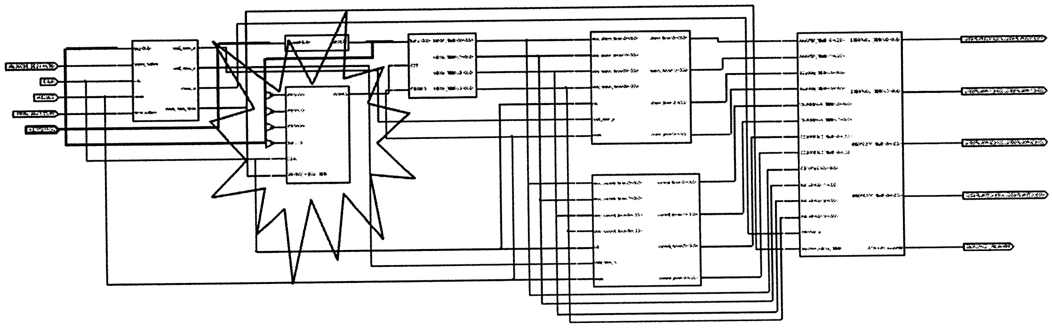
```
GClk_K <= not Clk or GSignal4;
GSignal3 <= '0' when Key /= 10 or Show_New_Time = '1' else '1';
control3 : process (Reset,Clk)
begin
      if Reset = '1' then
            GSignal4 <= '0';
      elsif falling_edge(Clk) then
            GSignal4 <= GSignal3;
      end if;
end process control3;
```

ผลการทดลองแสดงให้เห็นว่าสัญญาณ Gated_Clock (GClk_K) ซึ่งเป็นสัญญาณนาฬิกาใหม่ที่เข้าไปกระตุ้นการทำงานของวงจร Keyboard Buffer มีลักษณะแตกต่างจากสัญญาณนาฬิกาปกติก่อนที่จะทำ Clock Gating อย่างชัดเจน

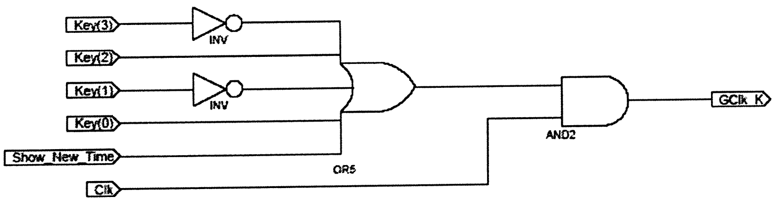
รายละเอียดของผลการทดลองของแต่ละชุดการทดลองเป็นดังนี้

<u>การทดลอง ALK-01 (Logic Low Disable + Latch Free + Active High)</u>

เมื่อทำการสังเคราะห์วงจรแล้วจะได้ผลดังแสดงภาพที่ 32 วงจรที่ถูกล้อมด้วยกรอบคือวงจรควบคุมที่เพิ่มเข้ามาซึ่งมีรายละเอียดภายในดังแสดงในภาพที่ 33



ภาพที่ 32 วงจรที่เพิ่มเข้ามาใน Alarm Clock ตามการทดลอง ALK-01



ภาพที่ 33 รายละเอียดวงจรที่เพิ่มเข้ามาใน Alarm Clock ตามการทดลอง ALK-01

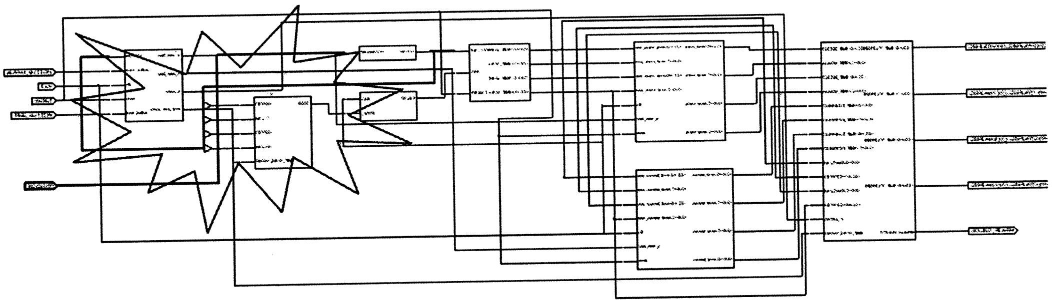
จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมแต่ไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้สัญญาณ New_Time ใน Keyboard Buffer เก็บค่าที่มีการกดปุ่มเร็วขึ้นเมื่อการกดปุ่มตัวเลขอยู่ในช่วงสัญญาณนาฬิกาขาขึ้น และเก็บค่าได้เป็นปกติเมื่อมีการกดปุ่มในสัญญาณนาฬิกาขาลงซึ่งจะมีผลทำให้เกิดการทำงานที่ผิดพลาดได้

<u>การทดลอง ALK-02 (Logic Low Disable + Latch Free + Active Low)</u>

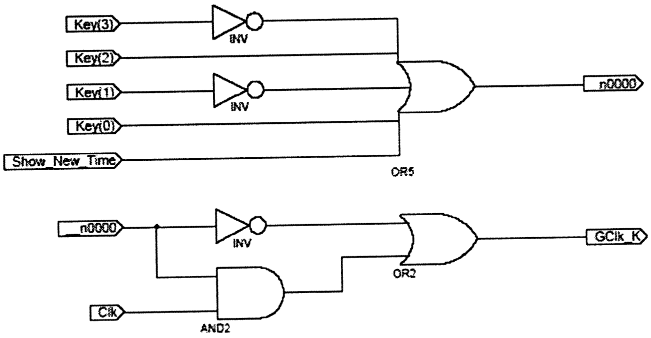
เมื่อทำการสังเคราะห์วงจรแล้วจะได้วงจรควบคุมที่เพิ่มเข้ามาซึ่งมีรายละเอียดเหมือนกับในการทดลอง ALK-01 ดังแสดงในภาพที่ 32 และ 33 จากผลการทดลองสัญญาณนาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมแต่ไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้ Keyboard Buffer ทำงานได้เหมือนเดิมทุกประการ

<u>การทดลอง ALK-03 (Logic High Disable + Latch Free + Active High)</u>

เมื่อทำการสังเคราะห์วงจรแล้วจะได้ผลดังแสดงในภาพที่ 34 วงจรที่ถูกล้อมด้วยกรอบคือวงจรควบคุมที่เพิ่มเข้ามาซึ่งมีรายละเอียดภายในดังแสดงในภาพที่ 35



ภาพที่ 34 วงจรที่เพิ่มเข้ามาใน Alarm Clock ตามการทดลอง ALK-03



ภาพที่ 35 รายละเอียดวงจรที่เพิ่มเข้ามาใน Alarm Clock ตามการทดลอง ALK-03

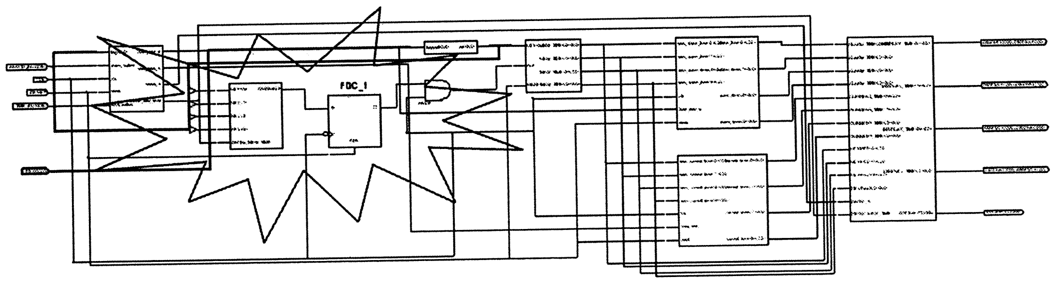
จากผลการทดลองสัญญาณาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมแต่ไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้ Keyboard Buffer ทำงานได้เหมือนเดิมทุกประการ

<u>การทดลอง ALK-04 (Logic High Disable + Latch Free + Active Low)</u>

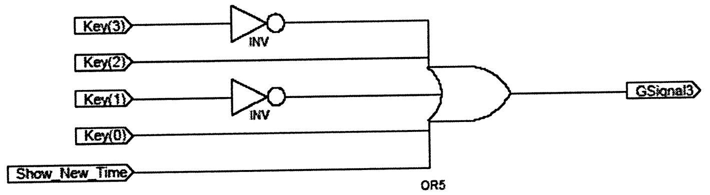
เมื่อทำการสังเคราะห์วงจรแล้วจะได้วงจรควบคุมที่เพิ่มเข้ามาซึ่งมีรายละเอียดเหมือนกับในการทดลอง ALK-03 ดังแสดงในภาพที่ 34 และ 35 จากผลการทดลองสัญญาณาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมแต่ไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้สัญญาณ New_Time ใน Keyboard Buffer เก็บค่าที่มีการกดปุ่มเร็วขึ้นเมื่อการกดปุ่มตัวเลขอยู่ในช่วงสัญญาณนาฬิกาขาขึ้น และเก็บค่าได้เป็นปกติเมื่อมีการกดปุ่มในสัญญาณนาฬิกาขาลงซึ่งจะมีผลทำให้เกิดการทำงานที่ผิดพลาดได้

<u>การทดลอง ALK-05 (Logic Low Disable + Latch Base + Active High)</u>

เมื่อทำการสังเคราะห์วงจรแล้วจะได้ผลดังแสดงในรูปที่ ภาพที่ 36 วงจรที่ถูกล้อมด้วยกรอบคือวงจรควบคุมที่เพิ่มเข้ามาซึ่งมีรายละเอียดภายในดังแสดงในภาพที่ 37



ภาพที่ 36 วงจรที่เพิ่มเข้ามาใน Alarm Clock ตามการทดลอง ALK-05



ภาพที่ 37 รายละเอียดวงจรที่เพิ่มเข้ามาใน Alarm Clock ตามการทดลอง ALK-05

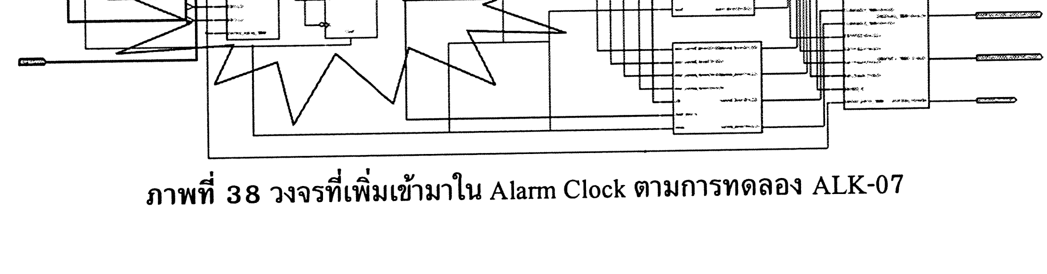
จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมแต่ไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้สัญญาณ New_Time ใน Keyboard Buffer เก็บค่าที่มีการกดปุ่มช้าลงเมื่อการกดปุ่มตัวเลขอยู่ในช่วงสัญญาณนาฬิกาขาลง และเก็บค่าได้เป็นปกติเมื่อมีการกดปุ่มในสัญญาณนาฬิกาขาขึ้นซึ่งจะมีผลทำให้เกิดการทำงานที่ผิดพลาดได้

<u>การทดลอง ALK-06 (Logic Low Disable + Latch Base + Active Low)</u>

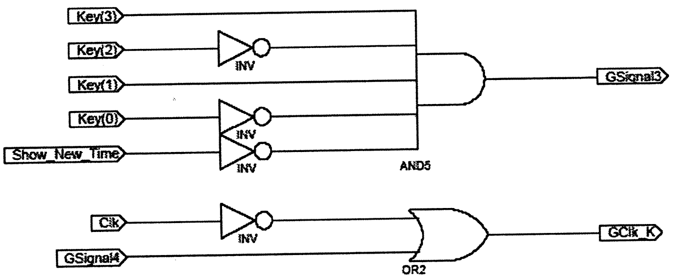
จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมแต่ไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้สัญญาณ New_Time ใน Keyboard Buffer เก็บค่าที่มีการกดปุ่มช้าลงในทุกกรณีซึ่งจะมีผลทำให้เกิดการทำงานที่ผิดพลาดได้ เมื่อทำการสังเคราะห์วงจรแล้วจะได้วงจรควบคุมที่เพิ่มเข้ามาซึ่งมีรายละเอียดเหมือนกับในการทดลอง ALK-05 ดังแสดงในภาพที่ 36 และ 37

<u>การทดลอง ALK-07 (Logic High Disable + Latch Base + Active High)</u>

เมื่อทำการสังเคราะห์วงจรแล้วจะได้ผลดังแสดงในภาพที่ 38 วงจรที่ถูกล้อมด้วยกรอบคือวงจรควบคุมที่เพิ่มเข้ามาซึ่งมีรายละเอียดภายในดังแสดงในภาพที่ 39



ภาพที่ 38 วงจรที่เพิ่มเข้ามาใน Alarm Clock ตามการทดลอง ALK-07



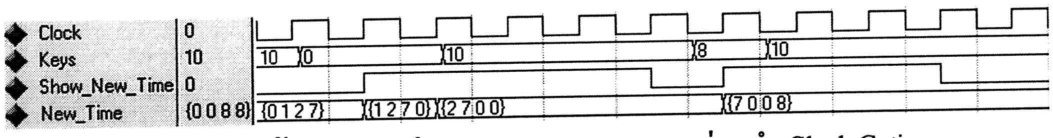
ภาพที่ 39 รายละเอียดวงจรที่เพิ่มเข้ามาใน Alarm Clock ตามการทดลอง ALK-07

จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมแต่ไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้สัญญาณ New_Time ใน Keyboard Buffer เก็บค่าที่มีการกดปุ่มช้าลงในทุกกรณีซึ่งจะมีผลทำให้เกิดการทำงานที่ผิดพลาดได้

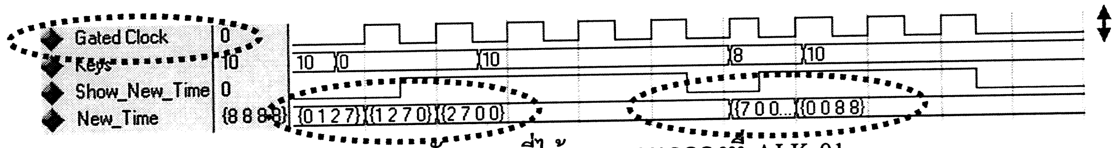
<u>การทดลอง ALK-08 (Logic High Disable + Latch Base + Active Low)</u>

จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมแต่ไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้สัญญาณ New_Time ใน Keyboard Buffer เก็บค่าที่มีการกดปุ่มช้าลงเมื่อการกดปุ่มตัวเลขอยู่ในช่วงสัญญาณนาฬิกาขาลง และเก็บค่าได้เป็นปกติเมื่อมีการกดปุ่มในสัญญาณนาฬิกาขาขึ้นซึ่งจะมีผลทำให้เกิดการทำงานที่ผิดพลาดได้ เมื่อทำการสังเคราะห์วงจรแล้วจะได้วงจรควบคุมที่เพิ่มเข้ามาซึ่งมีรายละเอียดเหมือนกับในการทดลอง ALK-07 ดังแสดงในภาพที่ 38 และ 39

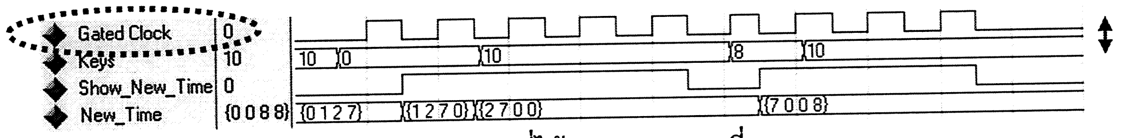
ผลการทดลองที่ได้จากการจำลองการทำงานซึ่งจะแสดงเฉพาะส่วนที่มีความเกี่ยวข้องกับการทดลอง โดยจะแสดงเฉพาะผลการทดลองในโหมดการทำงานของวงจรในช่วงที่มีการกดปุ่มตัวเลข ซึ่งมีรายละเอียดดังต่อไปนี้



ก) ภาพสัญญาณการทำงานของ Alarm Register ก่อนทำ Clock Gating

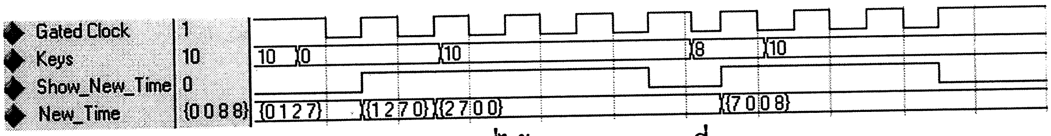


ข) ภาพสัญญาณที่ได้จากการทดลองที่ ALK-01

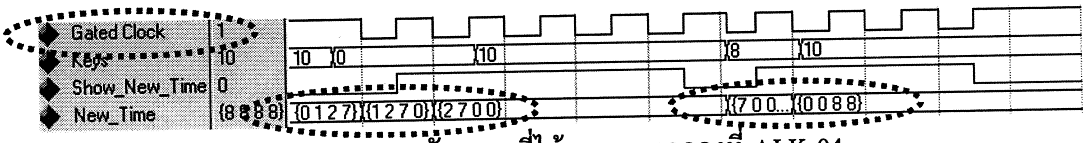


ค) ภาพสัญญาณที่ได้จากการทดลองที่ ALK-02

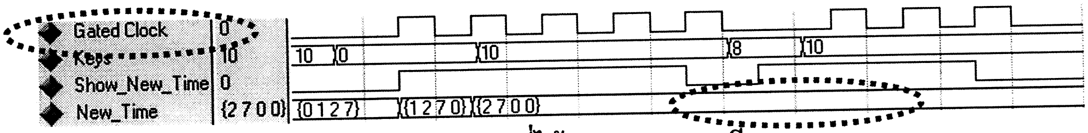
ภาพที่ 40 ภาพสัญญาณการทำงานของ Alarm Register ในช่วงเวลาที่มีการกดปุ่มตัวเลข



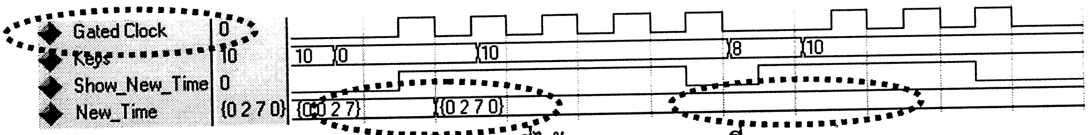
ง) ภาพสัญญาณที่ได้จากการทดลองที่ ALK-03



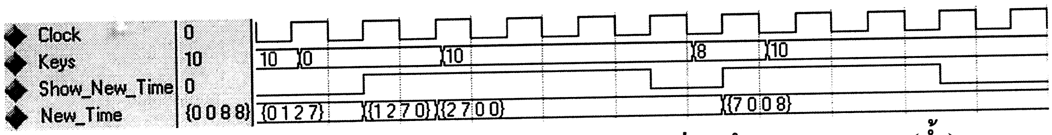
จ) ภาพสัญญาณที่ได้จากการทดลองที่ ALK-04



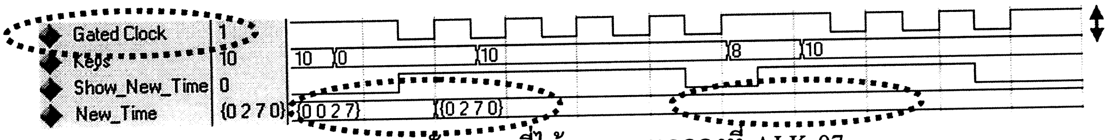
ฉ) ภาพสัญญาณที่ได้จากการทดลองที่ ALK-05



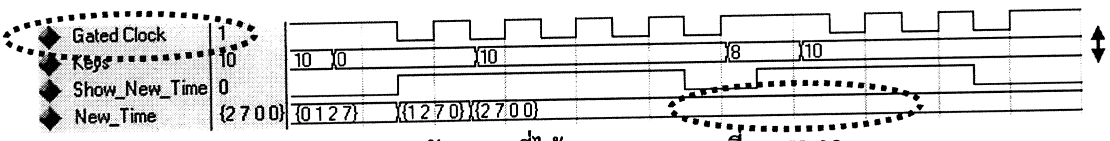
ช) ภาพสัญญาณที่ได้จากการทดลองที่ ALK-06



ซ) ภาพสัญญาณการทำงานของ Alarm Register ก่อนทำ Clock Gating (ซ้ำ)



ฌ) ภาพสัญญาณที่ได้จากการทดลองที่ ALK-07



ญ) ภาพสัญญาณที่ได้จากการทดลองที่ ALK-08

ภาพที่ 40 ภาพสัญญาณการทำงานของ Alarm Register ในช่วงเวลาที่มีการกดปุ่มตัวเลข (ต่อ)

ผลจากการวิเคราะห์การใช้พลังโดยใช้ XPower พบว่าการใช้พลังงานเป็นไปตามที่ได้ตั้งสมมุติฐานไว้โดยมีรายละเอียดดังตารางที่ 4

ตารางที่ 4 ผลการวิเคราะห์การใช้พลังงานโดยใช้ XPower ของ Keyboard Buffer

รายละเอียด \ การทดลอง	ต้นแบบ	ALK01	ALK02	ALK03	ALK04	ALK05	ALK06	ALK07	ALK08
Clock Power (mW)	1.36	1.42	1.32	1.20	1.28	1.34	1.26	1.37	1.37
Inputs Power (mW)	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15
Logic Power (mW)	0.00	0.07	0.07	0.07	0.07	0.07	0.07	0.07	0.07
Outputs Power (mW)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Signals Power (mW)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Summary (mW)	1.52	1.64	1.54	1.43	1.51	1.56	1.48	1.59	1.59
Glitch	✘	✘	✘	✘	✘	✘	✘	✘	✘
การทำงาน	☺	☹	☺	☺	☹	☹	☹	☹	☹

****หมายเหตุ :** ☺ = ปกติ ☹ = ผิดปกติ

1.4 การทดลองประยุกต์ใช้เทคนิค Clock Gating รวมทุกส่วนใน Alarm Clock

จากผลการทดลองประยุกต์ใช้เทคนิค Clock Gating ในวงจร Alarm Clock โดยแยกพิจารณาส่วนประกอบทั้งสามส่วนที่ผ่านมา จะเห็นว่าเทคนิค Clock Gating สามารถลดอัตราการใช้พลังงานได้จริงโดยได้ให้ผลการทำงานและผลทางพลังงานที่แตกต่างกันในแต่ละกรณีและวิธีไป

การทดลองในหัวข้อนี้จะเป็นการทดลองประยุกต์ใช้เทคนิค Clock Gating ในทั้งสามส่วนพร้อมกัน โดยจะแบ่งการทดลองเป็น 2 ชุดการทดลอง คือ

- การทดลองที่ ALL-I แยกวงจรควบคุมในแต่ละส่วน
การทดลองนี้จะเลือกวิธีที่ให้ผลดีที่สุดของแต่ละส่วนมาทดลองร่วมกัน โดยจะนำไปสู่การมีวงจรควบคุมและสร้างสัญญาณ Gated Clock (GClk) ที่แยกกันไปในแต่ละส่วนโดยผลการทดลองในเชิงของการใช้พลังงาน น่าจะมีสัดส่วนที่ลดลง มากกว่าผลที่ได้ลำพังจากแต่ละส่วน ซึ่งจากการทดลองที่ผ่านมาจะพบว่าการทดลองที่ 3 (Logic High Disable + Latch Free + Active High) ให้ผลการทดลองที่ดีที่สุดเหมือนกันทั้ง 3 ส่วน
- การทดลองที่ ALL-II รวมวงจรควบคุมในแต่ละส่วนเป็นวงจรเดียว
ในการทดลองนี้จะพิจารณาวงจรควบคุมและสร้างสัญญาณ Gated Clock (GClk) ทั้งระบบร่วมกัน ซึ่งน่าจะส่งผลให้วงจรควบคุมดังกล่าวมีขนาดเล็กลง

รายละเอียดของผลการทดลองของแต่ละชุดการทดลองเป็นดังนี้

<u>การทดลองที่ ALL-I แยกวงจรควบคุมในแต่ละส่วน</u>

การทดลองนี้มีการเพิ่ม Code 2 ส่วนเข้าไปสำหรับแต่ละส่วนประกอบทั้งสามส่วนดังที่ได้กล่าวมา ใน Alarm_Clock.vhd รายละเอียดของ Code ดังกล่าวมีดังต่อไปนี้

สำหรับส่วน Alarm Controller

```
GClk_C <= Clk WHEN Key /= 10 or Alarm_Button = '1' or
      Load_Alarm_Time = '1' or Show_Alarm_Time = '1' or
     Show_New_Time = '1' or Load_New_Time = '1' ELSE '1';
```

```
        U_ALARM_CONTROL: ALARM_CONTROLLER
        port map(    Clk   => GClk_C,         -- replace Clk wth GClk_C
                     Reset              =>    Reset,
                     Key                =>    Key,
                     Alarm_Button       =>    Alarm_Button,
                     Time_Button        =>    Time_Button,
                     Load_Alarm_Time    =>    Load_Alarm_Time,
                     Show_Alarm_Time    =>    Show_Alarm_Time,
                     Show_New_Time      =>    Show_New_Time,
                     Load_New_Time      =>    Load_New_Time
             );
```

สำหรับส่วน Alarm Register

```
GClk_A <= Clk or not Load_Alarm_Time;
```

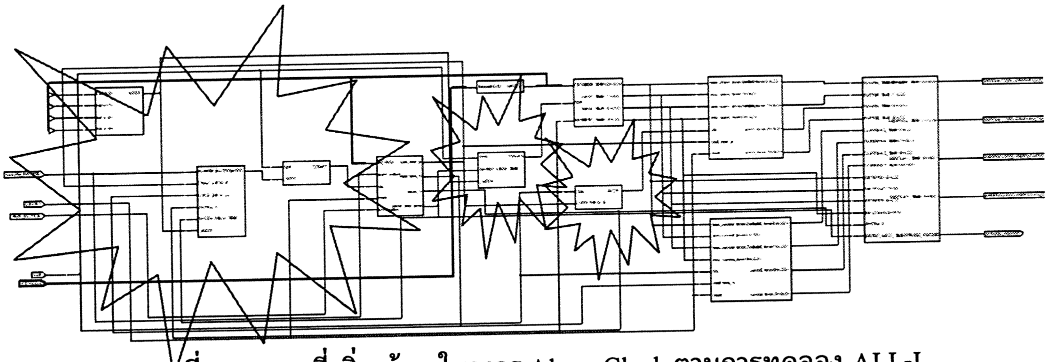
```
        U_ALARM_REG  : ALARM_REG
        port map  (  Clk    => GClk_A,          -- replace Clk wth GClk_A
                     New_Alarm_Time     => New_Time,
                     Load_Alarm_Time    => Load_Alarm_Time,
                     Reset              => Reset,
                     Alarm_Time         => Alarm_Time
             );
```

สำหรับส่วน Keyboard Buffer

```
GClk_K <= Clk when Key /= 10 or Show_New_Time = '1' else '1';
```

```
        U_KEYPAD_BUFFER  :KEYPAD_BUFFER
        port map  (  Clk    => GClk_K,          -- replace Clk wth GClk_K
                     Key          =>    Key,
                     Reset        =>    Reset,
                     New_Time     =>    Show_New_Time
             );
```

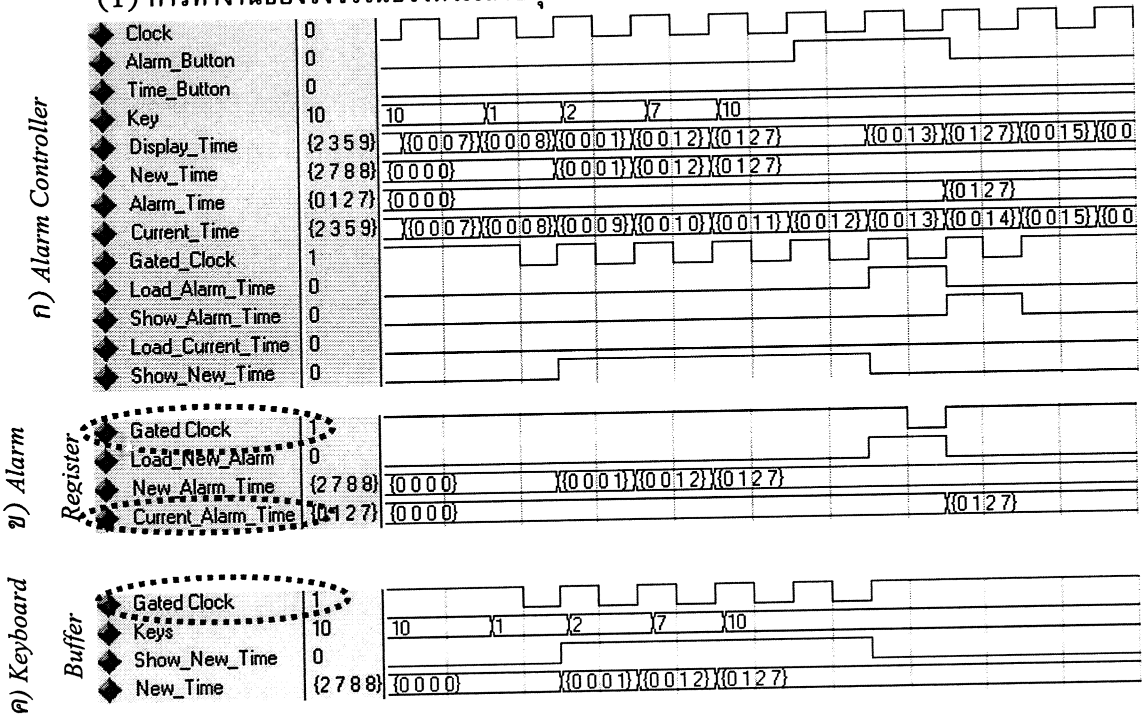
เมื่อทำการสังเคราะห์วงจรแล้วจะได้ผลดังแสดงในภาพที่ 41 วงจรที่ถูกล้อมด้วยกรอบคือวงจรควบคุมที่เพิ่มเข้ามา



ภาพที่ 41 วงจรที่เพิ่มเข้ามาในวงจร Alarm Clock ตามการทดลอง ALL-I

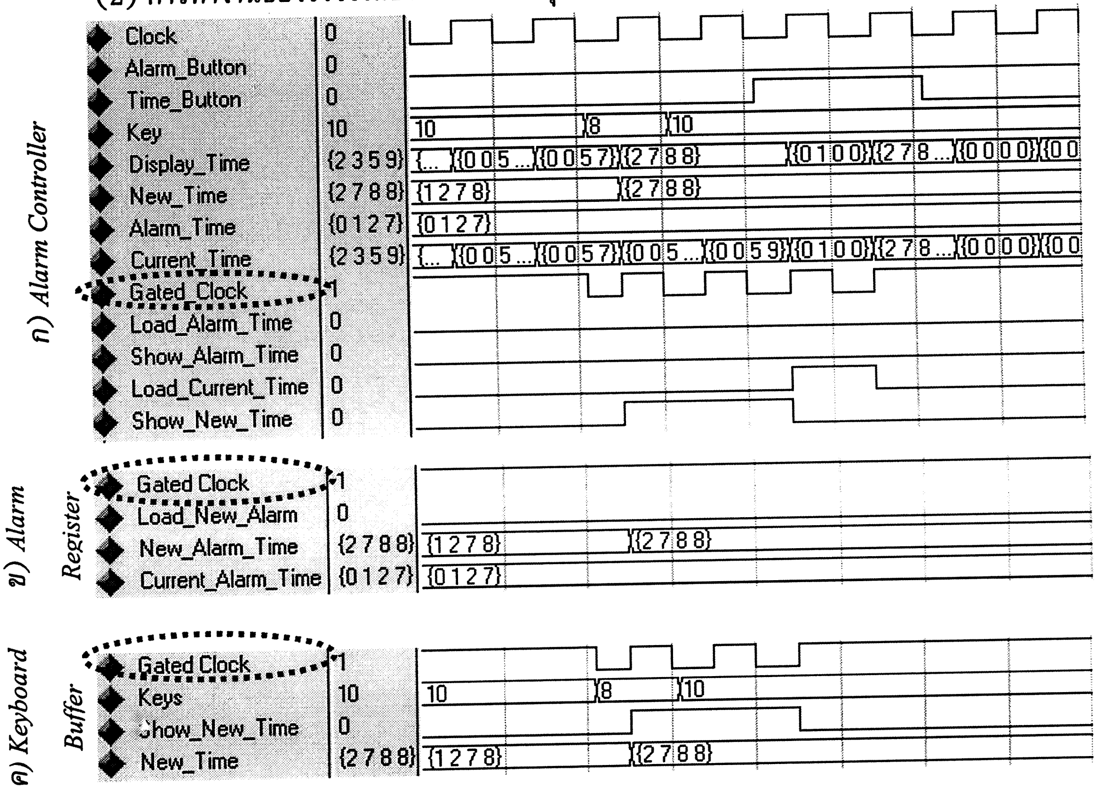
ผลการทดลองที่ได้จากการจำลองการทำงานซึ่งจะแสดงเฉพาะส่วนที่มีความเกี่ยวข้องกับการทดลอง โดยจะแยกเป็นผลจากการทดลองการทำงานในโหมดต่างๆ คือ การทำงานของวงจรในช่วงตั้งเวลาปลุก การทำงานของวงจรในช่วงตั้งเวลาปัจจุบัน การทำงานของวงจรในช่วงที่มีการกดปุ่ม Alarm_Button เพื่อดูเวลาปลุก และการทำงานของวงจรในช่วงที่มีการกดปุ่มตัวเลขทิ้งไว้ ซึ่งมีรายละเอียดดังภาพที่ 42 ถึง 44

(1) การทำงานของวงจรในช่วงตั้งเวลาปลุก



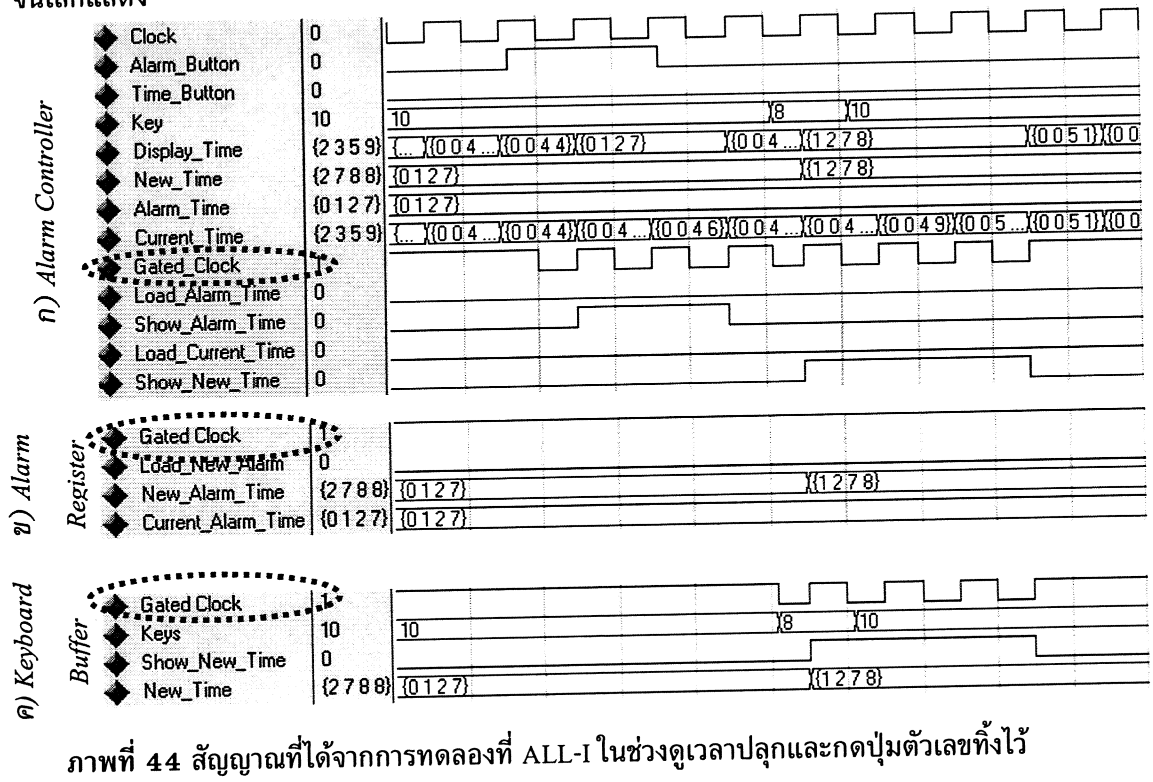
ภาพที่ 42 สัญญาณที่ได้จากการทดลองที่ ALL-I ในช่วงตั้งเวลาปลุก

(2) การทำงานของวงจรในช่วงตั้งเวลาปัจจุบัน



ภาพที่ 43 สัญญาณที่ได้จากการทดลองที่ ALL-I ในช่วงตั้งเวลาปัจจุบัน

(3) การทำงานของวงจรในช่วงกดปุ่ม Alarm เพื่อดูเวลาปลุก และช่วงที่กดปุ่มตัวเลขทิ้งไว้จนเลิกแสดง



ภาพที่ 44 สัญญาณที่ได้จากการทดลองที่ ALL-I ในช่วงดูเวลาปลุกและกดปุ่มตัวเลขทิ้งไว้

จากผลการทดลองสัญญาณนาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมอย่างชัดเจน ส่วนผลการทำงานนั้นทำให้วงจร Alarm Clock ทำงานได้เหมือนเดิมทุกประการ

<u>การทดลองที่ ALL-II รวมวงจรควบคุมในแต่ละส่วนเป็นวงจรเดียว</u>

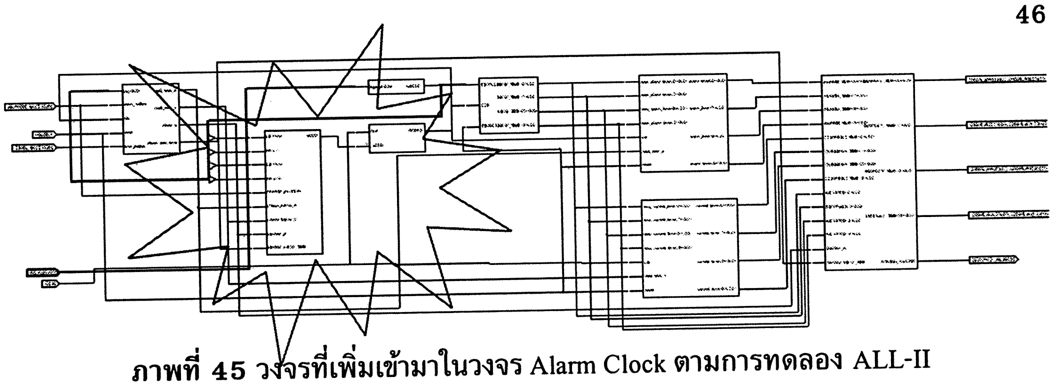
จากการทดลองที่ ALL-II จะเห็นว่าการทำงานของ Alarm Register และ Keyboard Buffer จะทำงานสอดคล้องกับการทำงานของ Alarm Controller ถึงแม้ว่าช่วงเวลาที่ใช้ในการทำงานของแต่ละส่วนจะไม่เท่ากันก็ตามแต่ก็จะสังเกตเห็นว่าทั้ง Alarm Register และ Keyboard Buffer ทำงานอยู่ในช่วงที่ Alarm Controller มีการทำงานอยู่ในทุกกรณี ดังนั้นเราจึงสามารถใช้สัญญาณ Gated Clock ของ Alarm Controller ไปใช้แทนสัญญาณนาฬิกาของ Alarm Register และ Keyboard Buffer ได้

การทดลองนี้มีการเพิ่ม Code 2 ส่วนเข้าไปใน Alarm_Clock.vhd รายละเอียดของ Code ดังกล่าวมีดังต่อไปนี้

```
GClk_C <= Clk WHEN Key /= 10 or Alarm_Button = '1' or
          Load_Alarm_Time  = '1' or Show_Alarm_Time = '1' or
          Show_New_Time = '1' or Load_New_Time = '1' ELSE '1';
```

```
U_ALARM_REG  : ALARM_REG
port map (   Clk      =>     GClk_C,        -- replace Clk wth GClk_C
             New_Alarm_Time   => New_Time,
             Load_Alarm_Time  => Load_Alarm_Time,
             Reset            => Reset,
             Alarm_Time       => Alarm_Time
         );
U_ALARM_CONTROL: ALARM_CONTROLLER
port map(    Clk     =>     GClk_C,         -- replace Clk wth GClk_C
             Reset                =>    Reset,
             Key                  =>    Key,
             Alarm_Button         =>    Alarm_Button,
             Time_Button          =>    Time_Button,
             Load_Alarm_Time      =>    Load_Alarm_Time,
             Show_Alarm_Time      =>    Show_Alarm_Time,
             Show_New_Time        =>    Show_New_Time,
             Load_New_Time        =>    Load_New_Time
      );
U_KEYPAD_BUFFER  :KEYPAD_BUFFER
port map  (  Clk     =>     GClk_C,         -- replace Clk wth GClk_C
             Key          =>     Key,
             Reset        =>     Reset,
             New_Time     =>     Show_New_Time
         );
```

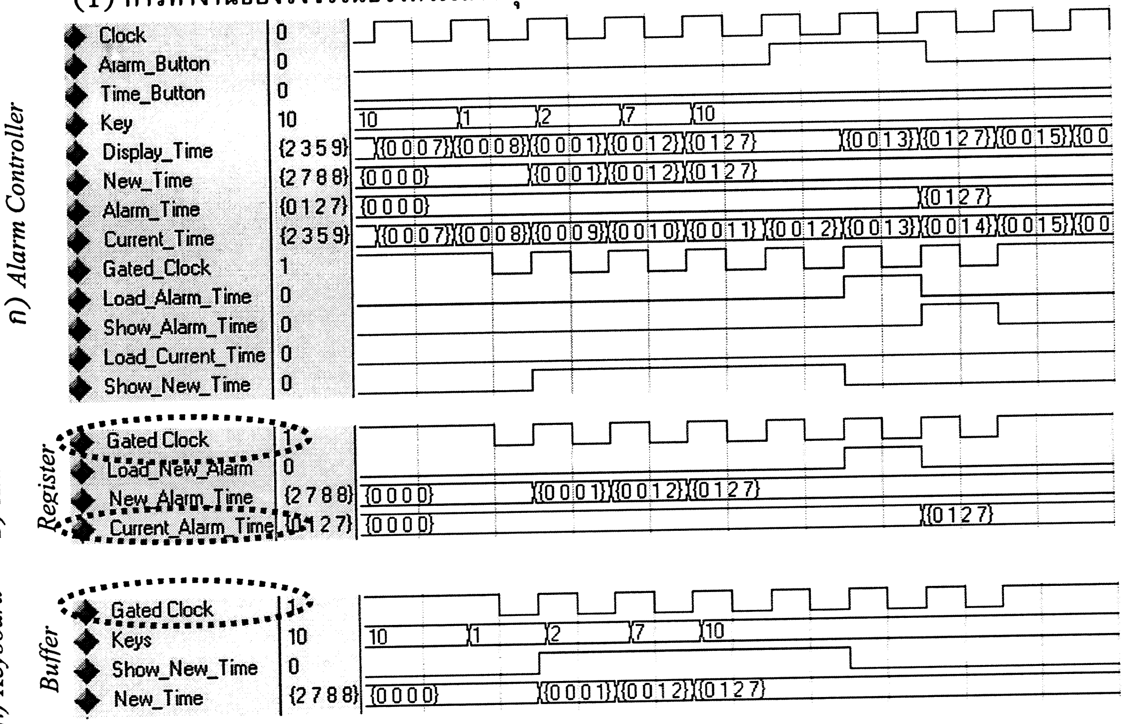
เมื่อทำการสังเคราะห์วงจรแล้วจะได้ผลดังแสดงในภาพที่ 44 วงจรที่ถูกล้อมด้วยกรอบคือวงจรควบคุมที่เพิ่มเข้ามา



ภาพที่ 45 วงจรที่เพิ่มเข้ามาในวงจร Alarm Clock ตามการทดลอง ALL-II

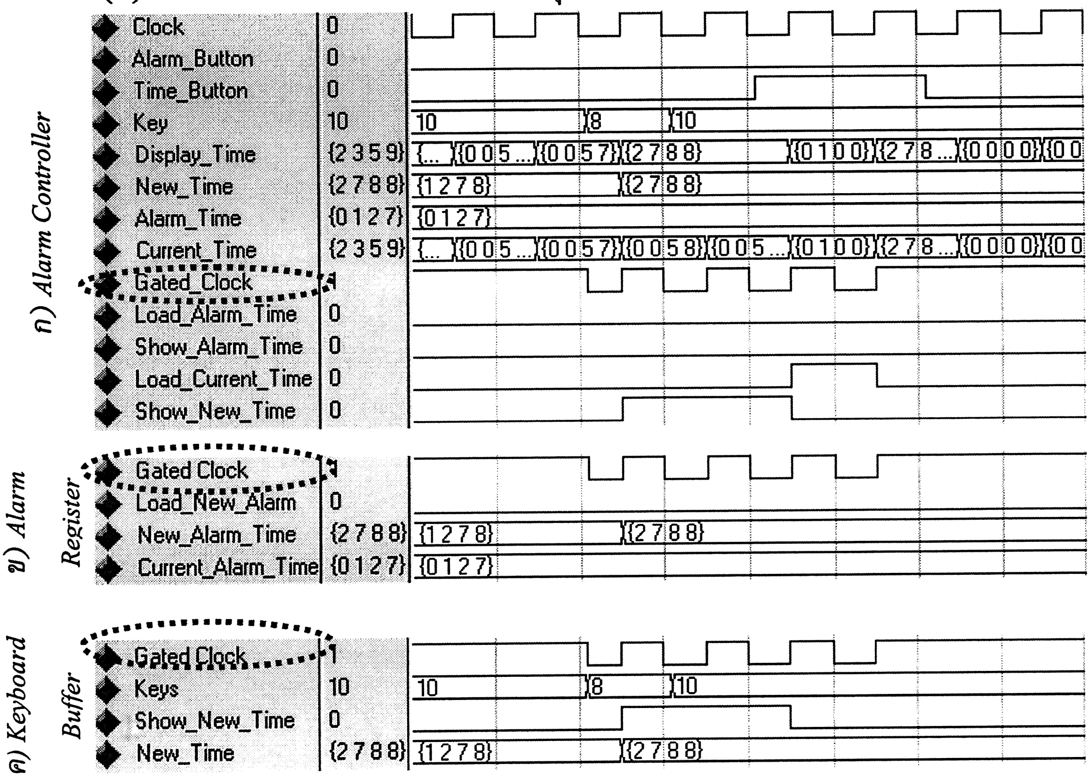
ผลการทดลองที่ได้จากการจำลองการทำงานซึ่งจะแสดงเฉพาะส่วนที่มีความเกี่ยวข้องกับการทดลอง โดยจะแยกเป็นผลจากการทดลองการทำงานในโหมดต่างๆ คือ การทำงานของวงจรในช่วงตั้งเวลาปลุก การทำงานของวงจรในช่วงตั้งเวลาปัจจุบัน การทำงานของวงจรในช่วงที่มีการกดปุ่ม Alarm_Button เพื่อดูเวลาปลุก และการทำงานของวงจรในช่วงที่มีการกดปุ่มตัวเลขทิ้งไว้ ซึ่งมีรายละเอียดดังภาพที่ 46 ถึง 48

(1) การทำงานของวงจรในช่วงตั้งเวลาปลุก



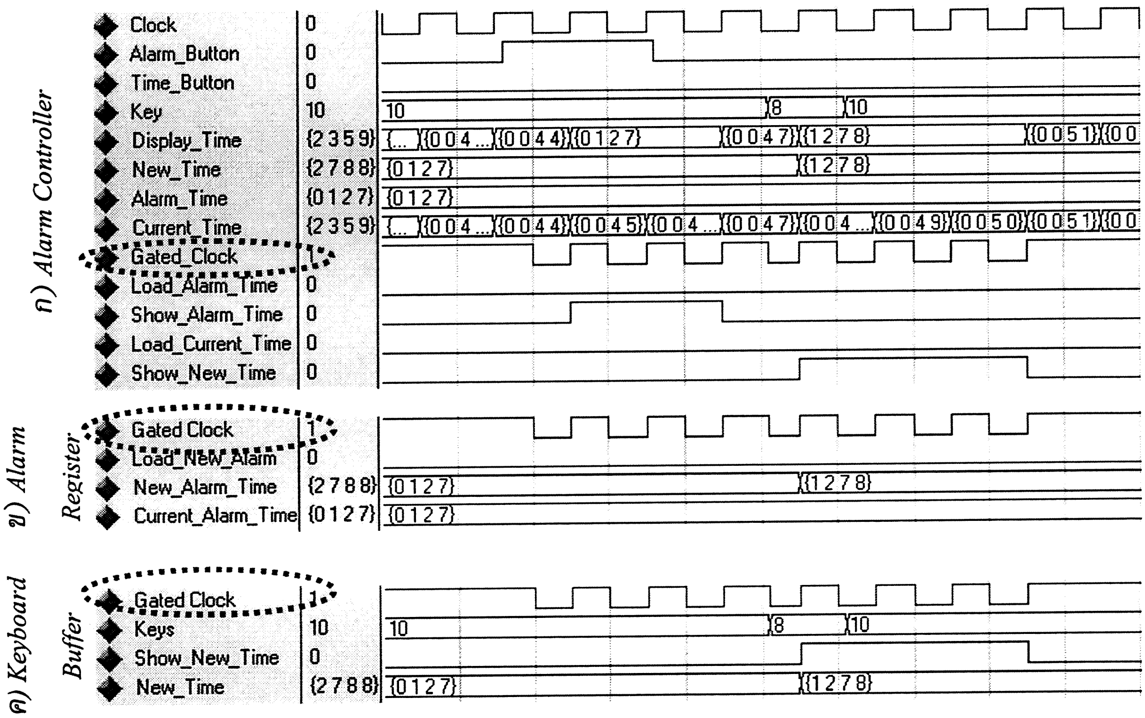
ภาพที่ 46 สัญญาณที่ได้จากการทดลองที่ ALL-II ในช่วงตั้งเวลาปลุก

(2) การทำงานของวงจรในช่วงตั้งเวลาปัจจุบัน



ภาพที่ 47 สัญญาณที่ได้จากการทดลองที่ ALL-II ในช่วงตั้งเวลาปัจจุบัน

(3) การทำงานของวงจรในช่วงกดปุ่ม Alarm เพื่อดูเวลาปลุก และช่วงที่กดปุ่มตัวเลขทิ้งไว้จนเลิกแสดง



ภาพที่ 48 สัญญาณที่ได้จากการทดลองที่ ALL-II ในช่วงดูเวลาปลุกและกดปุ่มตัวเลขทิ้งไว้

จากผลการทดลองสัญญาณนาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมอย่างชัดเจน ส่วนผลการทำงานนั้นทำให้วงจร Alarm Clock ทำงานได้เหมือนเดิมทุกประการ ดังตารางที่ 5

ตารางที่ 5 ผลการวิเคราะห์การใช้พลังงานโดยใช้ XPower รวมทุกส่วนใน Alarm Clock

รายละเอียด \ การทดลอง	ต้นแบบ	ALL-I	ALL-II
Clock Power (mW)	1.36	0.91	0.94
Inputs Power (mW)	0.15	0.15	0.15
Logic Power (mW)	0.00	0.22	0.07
Outputs Power (mW)	0.00	0.00	0.00
Signals Power (mW)	0.00	0.00	0.00
Summary (mW)	1.52	1.28	1.16
Glitch	✗	✗	✗
การทำงาน	☺	☺	☺

****หมายเหตุ :** ☺ = ปกติ

จากผลการทดลองจะเห็นว่าผลทางพลังงานที่ลดลงเมื่อทดลองประยุกต์ใช้เทคนิค Clock Gating ในวงจรทีละส่วน (ส่วน Alarm Controller ส่วน Alarm Register และส่วน Keyboard Buffer) ไม่แตกต่างกันมากเมื่อเทียบกับการทดลองที่ ALL-I ซึ่งเป็นการทำ Clock Gating จากทุกส่วนรวมในวงจร Alarm Clock เดียวกัน แต่จะลดลงมากขึ้นเมื่อทุกส่วนใช้สัญญาณ Gated Clock จากวงจรเดียวกัน (วงจร Clock Gating ของ Alarm Controller) ทั้งนี้เพราะแม้พลังงานที่ใช้ในโครงข่ายสัญญาณนาฬิกา (Clock Power) ในการทดลองที่ ALL-II จะมีค่ามากกว่าการทดลองที่ ALL-I เนื่องจากสัญญาณนาฬิกาที่ส่งไปยัง Alarm Register และ Keyboard Buffer จะมีการทำงานที่มากเกินความจำเป็นก็ตาม (ไม่ใช่สัญญาณ Gated Clock เฉพาะวงจร) แต่พลังงานในส่วนของ Logic Power ของการทดลองที่ ALL-II ก็มีค่าน้อยกว่าการทดลองที่ ALL-I อย่างมากจากการที่ใช้วงจร Gated Clock เพียงวงจรเดียวจึงทำให้อัตราการใช้พลังงานโดยรวมของการทดลองที่ ALL-II ลดลงมากกว่าการทดลองที่ ALL-I อย่างเห็นได้ชัด

นอกจากนี้วงจรที่สังเคราะห์ยังในผลการใช้ทรัพยากรในระบบเป็นดังนี้

(1) ผลการใช้ทรัพยากรของวงจรเดิม

```
Device utilization summary:
---------------------------
Selected Device : v300epq240-6

 Number of Slices:                 108  out of   3072     3%
 Number of Slice Flip Flops:        56  out of   6144     0%
 Number of 4 input LUTs:           188  out of   6144     3%
 Number of bonded IOBs:             42  out of    162    25%
 Number of GCLKs:                    1  out of      4    25%
```

(2) ผลการใช้ทรัพยากรของวงจรในการทดลองที่ ALL-I

```
Device utilization summary:
---------------------------
Selected Device : v300epq240-6

 Number of Slices:                 116  out of   3072     3%
 Number of Slice Flip Flops:        55  out of   6144     0%
 Number of 4 input LUTs:           202  out of   6144     3%
 Number of bonded IOBs:             42  out of    162    25%
 Number of GCLKs:                    1  out of      4    25%
```

(3) ผลการใช้ทรัพยากรของวงจรในการทดลองที่ ALL-II

```
Device utilization summary:
---------------------------
Selected Device : v300epq240-6

 Number of Slices:                 114  out of   3072     3%
 Number of Slice Flip Flops:        55  out of   6144     0%
 Number of 4 input LUTs:           198  out of   6144     3%
 Number of bonded IOBs:             42  out of    162    25%
 Number of GCLKs:                    1  out of      4    25%
```

จะเห็นว่าผลการใช้ทรัพยากรของวงจรที่มีการทำ Clock Gating ทั้งสองกรณีนั้นมีการใช้ทรัพยากรมากกว่าวงจรเดิมเพียงเล็กน้อยเท่านั้น นอกจากนี้จะเห็นว่าวงจรในการทดลองที่ ALL-II นั้นมีการใช้ทรัพยากรน้อยกว่าที่ใช้ในการทดลองที่ ALL-I ซึ่งก็เป็นผลมาจากจำนวนวงจรการทำ Clock Gating ที่ถูกเพิ่มเข้าไปนั่นเอง

2. ผลการทดลอง T80 Microprocessor

เมื่อได้พิจารณาโครงสร้างและการทำงานโดยรวมของวงจร T80 แล้วพบว่ามีส่วนที่น่าจะสามารถนำเทคนิค Clock Gating เข้าไปประยุกต์ใช้เพื่อลดอัตราการใช้พลังงานคือส่วน Register ซึ่งเป็นส่วนที่ใช้เก็บ General-Purpose Registers และ Index Registers ของ T80 ซึ่งการทำงานในส่วนนี้ได้อธิบายไว้แล้วในบทที่ 4 โดยจากการทำงานจะเห็นว่า T80 Register จะทำงานก็ต่อเมื่อสัญญาณ ClkEN ทำงาน (มีค่าเป็น 1) พร้อมกับสัญญาณ RegWEH หรือสัญญาณ RegWEL เท่านั้น จุดนี้จึงเป็นโอกาสในการทำเทคนิค Clock Gating มาใช้เพื่อลดอัตราการใช้พลังงานในจุดนี้

ในการจำลองการทำงานจะสมมุติให้ T80 ทำงานที่ความเร็ว 20 MHz โดย Run โปรแกรม Benchmark ซึ่งเขียนขึ้นเอง 3 โปรแกรมดังนี้

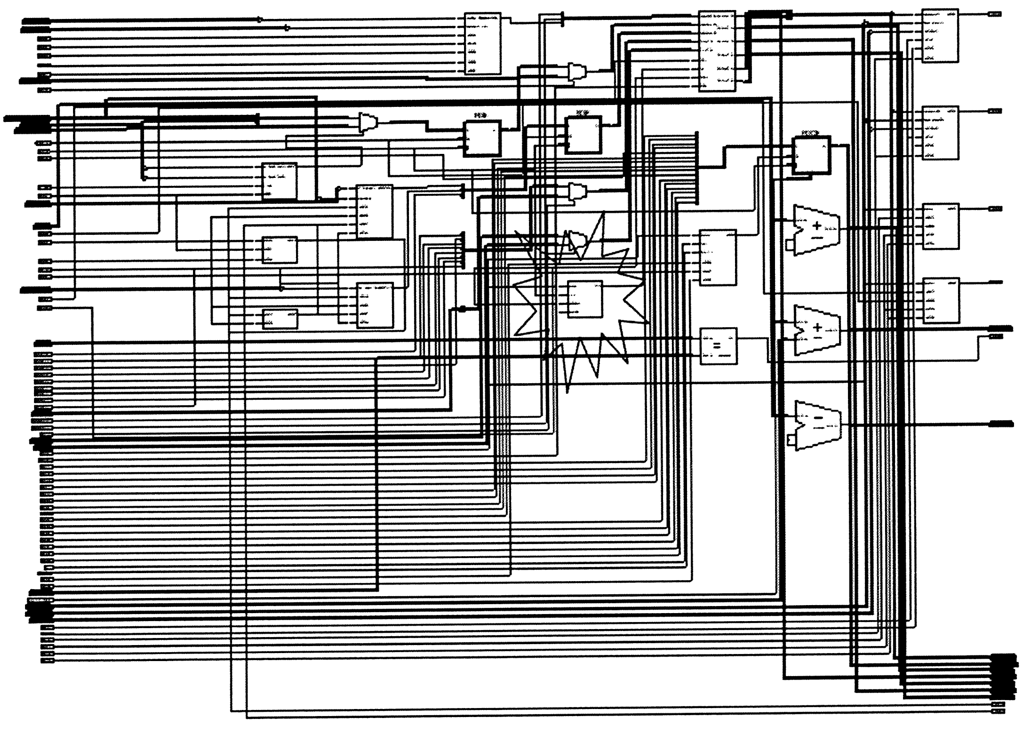
1. โปรแกรมหาค่า Factorial โดยในการทดลองกำหนดให้หาค่า 80!
2. โปรแกรมเรียงลำดับตัวเลขโดยใช้วิธีจัดเรียงค่าแบบ Bubble Sort ซึ่งในการทดลองจะเป็นการเรียงลำดับตัวเลขขนาด 1 Byte จำนวน 400 จำนวน
3. โปรแกรมหาตัวเลข Pascal's Triangle โดยในการทดลองจะเป็นการหาตัวเลข Pascal's Triangle จำนวน 50 รอบ

ซึ่งแต่ละโปรแกรมจะมีช่วงเวลาที่ไม่ได้ใช้งาน T80 Register ดังตารางที่ 6

ตารางที่ 6 ช่วงเวลาที่ไม่ได้ใช้งานรีจีสเตอร์ของแต่ละโปรแกรม

โปรแกรม	ช่วงเวลาที่ไม่ได้ใช้งาน (%)
1. Factorial	94.82
2. Buble Sort	95.52
3. Pascal's Triangle	89.15

เมื่อทำการสังเคราะห์วงจรหลังจากประยุกต์ใช้เทคนิค Clock Gating แล้วจะได้ผลดังแสดงในภาพที่ 49 วงจรที่ถูกล้อมด้วยกรอบคือวงจรควบคุมที่เพิ่มเข้ามาซึ่งมีรายละเอียดภายในที่แตกต่างกันไปแล้วแต่ละกรณี ซึ่งในบางกรณีวงจรควบคุมที่เพิ่มเข้ามานั้นจะกระจายอยู่ในวงจรเดิมไม่ได้เกาะกลุ่มกันอยู่ ทำให้ไม่สามารถแสดงออกมาอย่างชัดเจนได้



ภาพที่ 49 วงจรที่เพิ่มเข้ามาหลังประยุกต์ใช้ Clock Gating ในวงจร T80

การทดลองนี้มีการเพิ่ม Code 2 ส่วนเข้าไปใน t80.vhd ซึ่งเป็นการทำ Clock Gating ตามหลักการในบทที่ 2 รายละเอียดของ Code ดังกล่าวมีดังต่อไปนี้

ในการทดลองแต่ละชุดการทดลองนั้น มีการประยุกต์ใช้ Clock Gating (GClk) แทนสัญญาณนาฬิกา (Clk_n) เดิม โดยปรับ Code ใน t80.vhd ของแต่ละชุดการทดลองดังนี้

```
Regs : T80_Reg
      port map(
            Clk => GClk,        //*** Gated Clock
            CEN => ClkEn,
            WEH => RegWEH,
            WEL => RegWEL,
            AddrA => RegAddrA,
            AddrB => RegAddrB,
            AddrC => RegAddrC,
            DIH => RegDIH,
            DIL => RegDIL,
            DOAH => RegBusA(15 downto 8),
            DOAL => RegBusA(7 downto 0),
            DOBH => RegBusB(15 downto 8),
            DOBL => RegBusB(7 downto 0),
            DOCH => RegBusC(15 downto 8),
            DOCL => RegBusC(7 downto 0));
```

เนื่องจากชุดการทดลองแต่ละชุด (T80-01 ถึง T80-08) มีการประยุกต์ใช้ Clock Gating ที่แตกต่างกันไป ทำให้ Code ของการได้มาซึ่งสัญญาณ GClk ใน t80.vhd ของแต่ละชุดการทดลองจึงมีการปรับที่แตกต่างกันไปดังนี้

ในกรณีของ Active High (กรณีที่เป็นเลขคี่) สัญญาณนาฬิกาจะทำงานที่ขอบขาขึ้น (Rising Edge) แต่สำหรับกรณี Active Low (กรณีที่เป็นเลขคู่) จะต้องมีการปรับจากสัญญาณนาฬิกาที่เคยทำงานที่ขอบขาขึ้น ให้ทำงานที่ขอบขาลง (Falling Edge) ดังนี้

```
if Clk'event and Clk = '0' then
```

จากนั้น Code ที่เหลือของแต่ละคู่ชุดทดลอง (คู่ 01 และ 02 คู่ 03 และ 04 คู่ 05 และ 06 และคู่ 07 และ 08) จะมีลักษณเหมือนกัน

ส่วนข้อแตกต่างของ Code ที่มีลักษณะ Logic Low Disable และ Logic High Disable คือในช่วงที่สัญญาณ Clock Gating ไม่ทำงานจะมีค่าเป็น 0 และ 1 ตามลำดับ ซึ่งหมายถึงการเป็นนิเสธซึ่งกันและกันนั่นเอง และสุดท้ายคือข้อแตกต่างของ Code ที่มีลักษณะ Latch Free และ Latch Base คือ การปรับ Code ในส่วนของ GClk ให้มี Latch หรือ ไม่มี Latch นั่นเอง Code ที่ถูกปรับแล้วจะมีลักษณะที่แตกต่างกันดังนี้

สำหรับแบบ Latch Free เมื่อเป็นแบบ Logic Low Disable คือ

```
GClk <= not Clk_n and (ClkEn and (RegWEH or RegWEL));
```

และ เมื่อเป็นแบบ Logic High Disable คือ

```
GClk <= CLK_n or not (ClkEn and (RegWEH or RegWEL));
```

และสำหรับแบบ Latch Base เมื่อเป็นแบบ Logic Low Disable คือ

```
GClk <= CLK_n and GSingal;

L1 : process(CLK_n,RESET_n)
begin
      if RESET_n = '0' then
            GSingal <= '0';
      elsif falling_edge(CLK_n) then
            GSingal <= ClkEn and (RegWEH or RegWEL);
      end if;
end process L1;
```

และ เมื่อเป็นแบบ Logic High Disable คือ

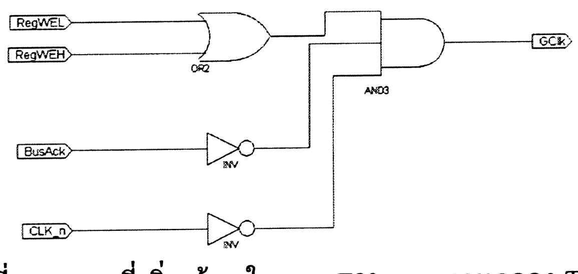
```
GClk <= not (CLK_n and GSingal);

L1 : process(CLK_n,RESET_n)
begin
      if RESET_n = '0' then
            GSingal <= '0';
      elsif falling_edge(CLK_n) then
            GSingal <= ClkEn and (RegWEH or RegWEL);
      end if;
er.d process L1;
```

ผลการทดลองแสดงให้เห็นว่าสัญญาณ Gated_Clock (GClk) ซึ่งเป็นสัญญาณนาฬิกาใหม่ที่เข้าไปกระตุ้นการทำงานของวงจร T80 มีลักษณะแตกต่างจากสัญญาณนาฬิกาปกติก่อนที่จะทำ Clock Gating อย่างชัดเจน

รายละเอียดของผลการทดลองของแต่ละชุดการทดลองเป็นดังนี้

<u>การทดลอง T80-01 (Logic Low Disable + Latch Free + Active High)</u>

เมื่อทำการสังเคราะห์วงจรแล้วจะได้วงจรควบคุมที่เพิ่มเข้ามาซึ่งมีรายละเอียดภายในดังแสดงในภาพที่ 50



ภาพที่ 50 วงจรที่เพิ่มเข้ามาในวงจร T80 ตามการทดลอง T80-01

จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมแต่ไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้ T80 Register ทำงานเร็วขึ้นครึ่งรอบสัญญาณนาฬิกาและมีผลทำให้การรันโปรแกรมหาค่า Factorial และ โปรแกรมเรียงลำดับตัวเลขโดยใช้วิธี

จัดเรียงค่าแบบ Bubble Sort มีผลลัพธ์ที่ผิดไปจากเดิม แต่อย่างไรก็ตามผลลัพธ์ที่ได้จากการรันโปรแกรมหาตัวเลข Pascal's Triangle ยังคงถูกต้องเหมือนเดิม

เมื่อตรวจสอบการทำงานโดยละเอียดของวงจรพบว่าสาเหตุที่ทำให้ได้ผลลัพธ์ที่ผิดไปจากเดิมของการรันโปรแกรมหาค่า Factorial และ โปรแกรมเรียงลำดับตัวเลขโดยใช้วิธีจัดเรียงค่าแบบ Bubble Sort มาจากการทำงานในคำสั่ง LDIR ซึ่งมีการลดค่าจำนวนรอบของการทำงาน (เก็บใน Register BC) พร้อม ๆ กับเปรียบเทียบเพื่อหยุดการทำงานคำสั่ง ดังนั้นเมื่อมีการทำ Clock Gating ใน T80 Register ที่มีผลทำให้การเก็บค่าใน T80 Register ทำงานเร็วขึ้นจึงทำให้การเปรียบเทียบและการลดค่าจำนวนรอบทำงานไม่สอดคล้องกันจึงมีผลทำให้การทำงานของ LDIR เสร็จเร็วกว่าเดิม 1 รอบคำสั่งจึงมีผลให้ผลลัพธ์การทำงานผิดไปจากเดิมในขณะที่โปรแกรมหาตัวเลข Pascal's Triangle ไม่ได้ใช้คำสั่ง LDIR ผลลัพธ์ที่ได้จึงยังคงถูกต้องดังเดิม

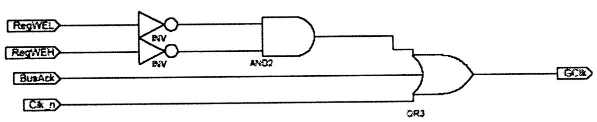
<u>การทดลอง T80-02 (Logic Low Disable + Latch Free + Active Low)</u>

เมื่อทำการสังเคราะห์วงจรแล้วจะได้วงจรควบคุมที่เพิ่มเข้ามาซึ่งมีรายละเอียดเหมือนกับในการทดลอง T80-01 ดังแสดงในภาพที่ 50

จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมแต่ไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้ T80 Register ทำงานได้เหมือนเดิมทุกประการ อีกทั้งผลลัพธ์ที่ได้จากการรันโปรแกรมทั้งก็ยังคงถูกต้องเหมือนเดิม

<u>การทดลอง T80-03 (Logic High Disable + Latch Free + Active High)</u>

เมื่อทำการสังเคราะห์วงจรแล้วจะได้วงจรควบคุมที่เพิ่มเข้ามามีรายละเอียดภายในดังแสดงในภาพที่ 51



ภาพที่ 51 วงจรที่เพิ่มเข้ามาในวงจร T80 ตามการทดลอง T80-03

จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมแต่ไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้ T80 Register ทำงานได้เหมือนเดิมทุกประการ อีกทั้งผลลัพธ์ที่ได้จากการรันโปรแกรมทั้งก็ยังคงถูกต้องเหมือนเดิม

<u>การทดลอง T80-04 (Logic High Disable + Latch Free + Active Low)</u>

เมื่อทำการสังเคราะห์วงจรแล้วจะได้วงจรควบคุมที่เพิ่มเข้ามาซึ่งมีรายละเอียดเหมือนกับในการทดลอง T80-03 ดังแสดงในภาพที่ 51

จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมแต่ไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้ T80 Register ทำงานเร็วขึ้นครึ่งรอบสัญญาณนาฬิกาและมีผลทำให้การรันโปรแกรมหาค่า Factorial และ โปรแกรมเรียงลำดับตัวเลขโดยใช้วิธี

จัดเรียงค่าแบบ Bubble Sort มีผลลัพธ์ที่ผิดไปจากเดิม แต่อย่างไรก็ตามผลลัพธ์ที่ได้จากการรันโปรแกรมหาตัวเลข Pascal's Triangle ยังคงถูกต้องเหมือนเดิม

เมื่อตรวจสอบการทำงานโดยละเอียดของวงจรพบว่าสาเหตุที่ทำให้ได้ผลลัพธ์ที่ผิดไปจากเดิมของการรันโปรแกรมหาค่า Factorial และ โปรแกรมเรียงลำดับตัวเลขโดยใช้วิธีจัดเรียงค่าแบบ Bubble Sort มาจากการทำงานในคำสั่ง LDIR ซึ่งมีการลดค่าจำนวนรอบของการทำงาน (เก็บใน Register BC) พร้อม ๆ กับเปรียบเทียบเพื่อหยุดการทำงานคำสั่ง ดังนั้นเมื่อมีการทำ Clock Gating ใน T80 Register ที่มีผลทำให้การเก็บค่าใน T80 Register ทำงานเร็วขึ้นจึงทำให้การเปรียบเทียบและการลดค่าจำนวนรอบทำงานไม่สอดคล้องกันจึงมีผลทำให้การทำงานของ LDIR เสร็จเร็วกว่าเดิม 1 รอบคำสั่งจึงมีผลให้ผลลัพธ์การทำงานผิดไปจากเดิมในขณะที่โปรแกรมหาตัวเลข Pascal's Triangle ไม่ได้ใช้คำสั่ง LDIR ผลลัพธ์ที่ได้จึงยังคงถูกต้องดังเดิม

<u>**การทดลอง T80-05 (Logic Low Disable + Latch Base + Active High)**</u>

สำหรับผลการสังเคราะห์วงจรไม่สามารถนำมาแสดงได้เนื่องจากวงจรหลังการทำ Clock Gating ที่เพิ่มเข้ามาจะกระจายอยู่ในวงจรเดิมไม่ได้เกาะกลุ่มกันอยู่เช่นการทดลองที่ผ่านมา

จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมแต่ไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้ T80 Register ทำงานได้เหมือนเดิมทุกประการอีกทั้งผลลัพธ์ที่ได้จากการรันโปรแกรมทั้งก็ยังคงถูกต้องเหมือนเดิม

<u>**การทดลอง T80-06 (Logic Low Disable + Latch Base + Active Low)**</u>

สำหรับผลการสังเคราะห์วงจรไม่สามารถนำมาแสดงได้เนื่องจากวงจรหลังการทำ Clock Gating ที่เพิ่มเข้ามาจะกระจายอยู่ในวงจรเดิมไม่ได้เกาะกลุ่มกันอยู่เช่นการทดลองที่ผ่านมา

จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมแต่ไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้ T80 Register ทำงานผิดพลาดโดยไม่มีการเก็บค่าที่ส่งเข้ามาตามที่ควรจะเป็น

<u>**การทดลอง T80-07 (Logic High Disable + Latch Base + Active High)**</u>

สำหรับผลการสังเคราะห์วงจรไม่สามารถนำมาแสดงได้เนื่องจากวงจรหลังการทำ Clock Gating ที่เพิ่มเข้ามาจะกระจายอยู่ในวงจรเดิมไม่ได้เกาะกลุ่มกันอยู่เช่นการทดลองที่ผ่านมา

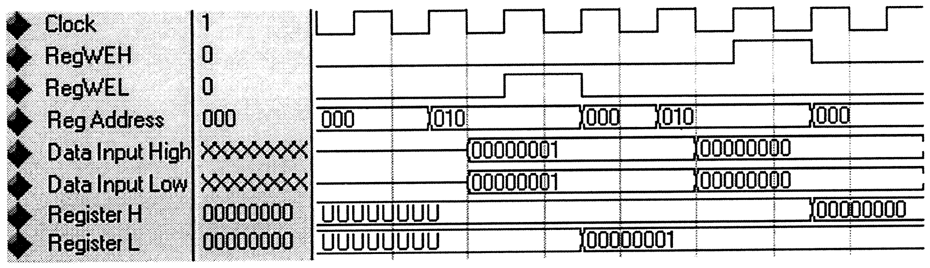
จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมแต่ไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้ T80 Register ทำงานผิดพลาดโดยไม่มีการเก็บค่าที่ส่งเข้ามาตามที่ควรจะเป็น

<u>**การทดลอง T80-08 (Logic High Disable + Latch Base + Active Low)**</u>

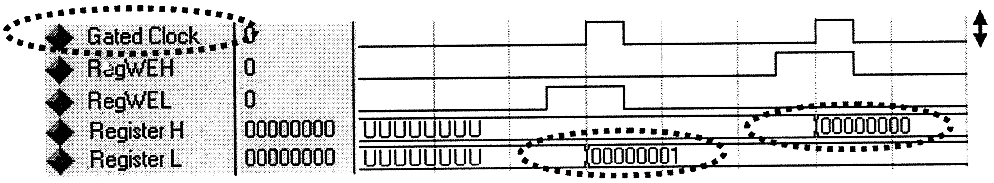
สำหรับผลการสังเคราะห์วงจรไม่สามารถนำมาแสดงได้เนื่องจากวงจรหลังการทำ Clock Gating ที่เพิ่มเข้ามาจะกระจายอยู่ในวงจรเดิมไม่ได้เกาะกลุ่มกันอยู่เช่นการทดลองที่ผ่านมา

จากผลการทดลองสัญญานาฬิกาที่ผ่านการทำ Clock Gating มีลักษณะที่เปลี่ยนไปจากเดิมแต่ไม่เกิด Glitch ส่วนผลการทำงานนั้นทำให้ T80 Register ทำงานผิดพลาดโดยไม่มีการเก็บค่าที่ส่งเข้ามาตามที่ควรจะเป็น

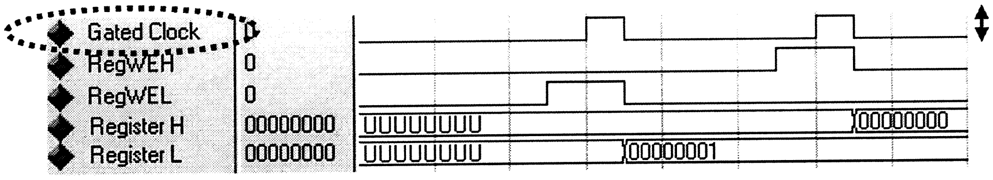
ผลการทดลองที่ได้จากการจำลองการทำงานซึ่งจะแสดงเฉพาะส่วนที่มีความเกี่ยวข้องกับการทดลอง โดยจะแสดงเฉพาะผลการทดลองภายในส่วน T80 Register ซึ่งมีรายละเอียดดังต่อไปนี้



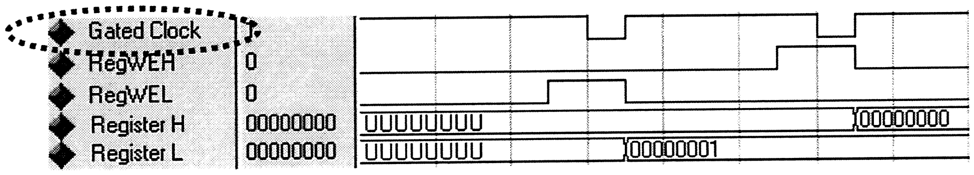
ก) ภาพสัญญาณการทำงานภายใน T80 Register



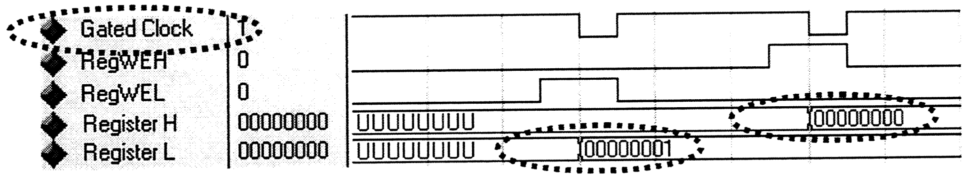
ข) ภาพสัญญาณหลังทำ Clock Gating (T80-01)



ค) ภาพสัญญาณหลังทำ Clock Gating (T80-02)

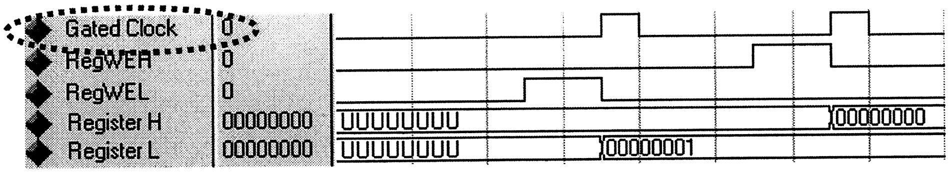


ง) ภาพสัญญาณหลังทำ Clock Gating (T80-03)

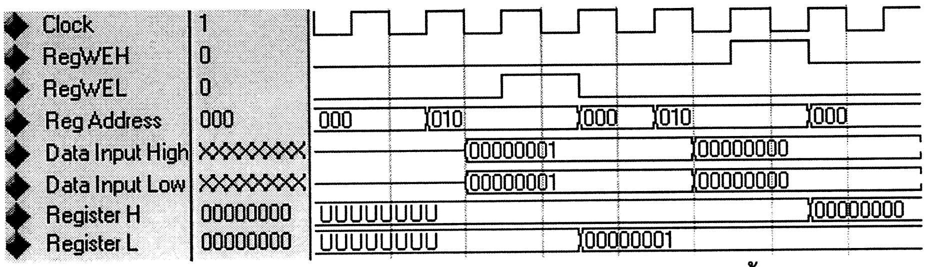


จ) ภาพสัญญาณหลังทำ Clock Gating (T80-04)

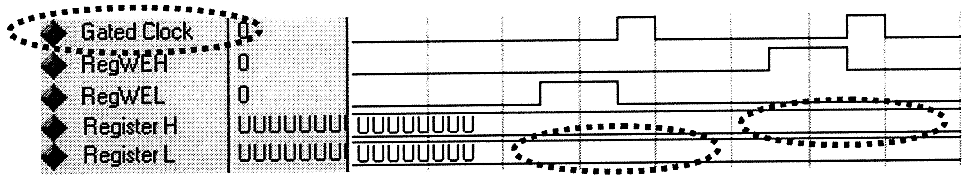
ภาพที่ 52 สัญญาณแสดงการทำงานภายใน T80 Register



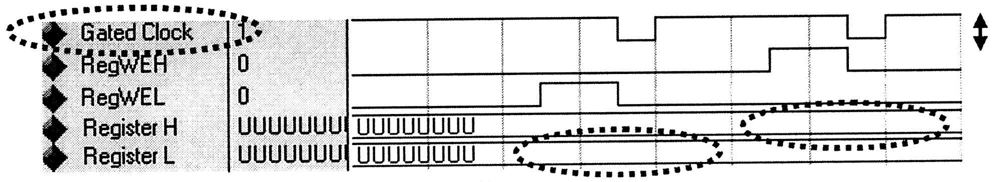
ฉ) ภาพสัญญาณหลังทำ Clock Gating (T80-05)



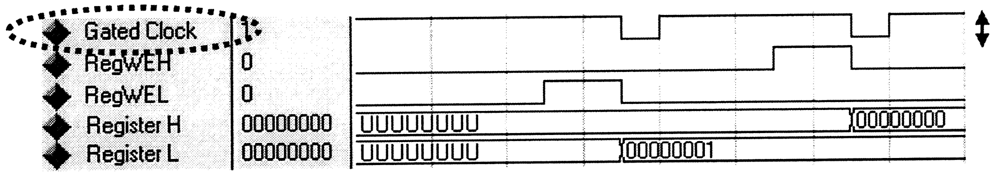
ช) ภาพสัญญาณการทำงานภายใน T80 Register (ซ้ำ)



ซ) ภาพสัญญาณหลังทำ Clock Gating (T80-06)



ฌ) ภาพสัญญาณหลังทำ Clock Gating (T80-07)



ญ) ภาพสัญญาณหลังทำ Clock Gating (T80-08)

ภาพที่ 52 สัญญาณแสดงการทำงานภายใน T80 Register (ต่อ)

ผลจากการวิเคราะห์การใช้พลังโดยใช้ XPower พบว่าการใช้พลังงานเป็นไปตามที่ได้ตั้งสมมุติฐานไว้โดยมีรายละเอียดดังตารางที่ 7 ถึง 9

ตารางที่ 7 ผลการทดลองจากการรันโปรแกรมหาค่า Factorial

รายละเอียด \ การทดลอง	ต้นแบบ	T8001	T8002	T8003	T8004	T8005	T8006	T8007	T8008
Clock Power (mW)	10.03	9.30	9.30	9.45	9.45	9.71	-	-	10.19
IOs Power (mW)	0.54	0.57	0.54	0.54	0.57	0.54	-	-	0.54
Inputs Power (mW)	0.30	0.30	0.30	0.30	0.30	0.30	-	-	0.30
Logic Power (mW)	0.70	0.86	0.85	0.85	0.86	0.85	-	-	0.85
Outputs Power(mW)	0.00	0.00	0.00	0.00	0.00	0.00	-	-	0.00
Signals Power(mW)	1.95	2.31	2.20	2.04	2.14	2.03	-	-	2.15
Summary(mW)	13.53	13.34	13.19	13.18	13.32	13.44	-	-	14.04
Glitch	✘	✘	✘	✘	✘	✘	✘	✘	✘
ผลลัพธ์	☺	☹	☺	☺	☹	☺	☹	☹	☺

**หมายเหตุ : ☺ = ปกติ ☹ = ผิดปกติ

ตารางที่ 8 ผลการทดลองการรันโปรแกรมเรียงลำดับตัวเลขโดยใช้วิธีเรียงค่าแบบ Bubble Sort

รายละเอียด \ การทดลอง	ต้นแบบ	T8001	T8002	T8003	T8004	T8005	T8006	T8007	T8008
Clock Power (mW)	9.94	10.18	10.38	9.34	9.34	9.99	-	-	9.88
IOs Power (mW)	0.55	0.53	0.55	0.55	0.53	0.55	-	-	0.55
Inputs Power (mW)	0.30	0.30	0.30	0.30	0.30	0.30	-	-	0.30
Logic Power (mW)	0.69	0.83	0.84	0.84	0.83	0.84	-	-	0.84
Outputs Power (mW)	0.00	0.00	0.00	0.00	0.00	0.00	-	-	0.00
Signals Power (mW)	2.17	2.24	2.18	2.02	1.98	2.24	-	-	2.38
Summary (mW)	13.65	14.09	14.25	13.05	12.98	13.91	-	-	13.95
Glitch	✘	✘	✘	✘	✘	✘	✘	✘	✘
ผลลัพธ์	☺	☹	☺	☺	☹	☺	☹	☹	☺

**หมายเหตุ : ☺ = ปกติ ☹ = ผิดปกติ

ตารางที่ 9 ผลการทดลองจากการรันโปรแกรมหาตัวเลข Pascal's Triangle

รายละเอียด \ การทดลอง	ต้นแบบ	T8001	T8002	T8003	T8004	T8005	T8006	T8007	T8008
Clock Power (mW)	10.40	9.86	9.38	9.80	10.03	9.39	-	-	9.61
IOs Power (mW)	0.47	0.47	0.47	0.47	0.47	0.47	-	-	0.47
Inputs Power (mW)	0.30	0.30	0.30	0.30	0.30	0.30	-	-	0.30
Logic Power (mW)	0.65	0.80	0.80	0.80	0.80	0.80	-	-	0.80
Outputs Power(mW)	0.00	0.00	0.00	0.00	0.00	0.00	-	-	0.00
Signals Power(mW)	1.91	1.94	1.94	1.91	1.96	1.90	-	-	2.03
Summary(mW)	13.73	13.37	12.88	13.28	13.56	12.86	-	-	13.21
Glitch	✘	✘	✘	✘	✘	✘	✘	✘	✘
ผลลัพธ์	☺	☺	☺	☺	☺	☺	☹	☹	☺

**หมายเหตุ : ☺ = ปกติ ☹ = ผิดปกติ

และมีผลการใช้ทรัพยากรในระบบเป็นดังนี้

(1) การใช้ทรัพยากรของระบบในวงจรเดิม

```
Device utilization summary:
---------------------------
Selected Device : v300epq240-6

 Number of Slices:                  1493  out of   3072    48%
 Number of Slice Flip Flops:         290  out of   6144     4%
 Number of 4 input LUTs:            2483  out of   6144    40%
 Number of bonded IOBs:               34  out of    162    20%
 Number of TBUFs:                     44  out of   3072     1%
 Number of GCLKs:                      1  out of      4    25%
```

(2) การใช้ทรัพยากรของระบบหลังทำ Clock Gating

```
Device utilization summary:
---------------------------
Selected Device : v300epq240-6

 Number of Slices:                  1537  out of   3072    50%
 Number of Slice Flip Flops:         293  out of   6144     4%
 Number of 4 input LUTs:            2565  out of   6144    41%
 Number of bonded IOBs:               34  out of    162    20%
 Number of TBUFs:                     44  out of   3072     1%
 Number of GCLKs:                      1  out of      4    25%
```

บทที่ 4
สรุปและวิเคราะห์ผลการทดลอง

ในบทนี้จะกล่าวถึงการวิเคราะห์ผลที่ได้จากการทดลอง จากนั้นก็เป็นการสรุปผลงานวิจัยและสุดท้ายจะเป็นข้อเสนอแนะสำหรับผู้สนใจจะนำเทคนิคนี้ไปใช้ในงานวิจัยอื่น ๆ ต่อไป

1. วิเคราะห์ผลการทดลอง

ผลการทำงานซึ่งเกิดจากการทำ Clock Gating ในการทดลองสามารถสรุปได้ดังนี้

(1) เมื่อใช้ Logic Low Disable ร่วมกับ Latch Free Clock Gating และเลือกใช้สัญญาณกระตุ้นการทำงานแบบ Active High สัญญาณนาฬิกาที่ได้จะมีการกลับเฟสกับสัญญาณเดิมและส่งผลทำให้การทำงานของวงจรทำงานเร็วขึ้นครึ่งรอบสัญญาณนาฬิกา

(2) เมื่อใช้ Logic Low Disable ร่วมกับ Latch Free Clock Gating และเลือกใช้สัญญาณกระตุ้นการทำงานแบบ Active Low สัญญาณนาฬิกาที่ได้จะมีการกลับเฟสกับสัญญาณเดิมและมีการทำงานที่ถูกต้อง

(3) เมื่อใช้ Logic High Disable ร่วมกับ Latch Free Clock Gating และเลือกใช้สัญญาณกระตุ้นการทำงานแบบ Active High สัญญาณนาฬิกาที่ได้จะมีเฟสเหมือนกับสัญญาณเดิมและมีการทำงานที่ถูกต้อง

(4) เมื่อใช้ Logic High Disable ร่วมกับ Latch Free Clock Gating และเลือกใช้สัญญาณกระตุ้นการทำงานแบบ Active Low สัญญาณนาฬิกาที่ได้จะมีเฟสเหมือนกับสัญญาณเดิมและส่งผลทำให้การทำงานของวงจรทำงานเร็วขึ้นครึ่งรอบสัญญาณนาฬิกา

(5) เมื่อใช้ Logic Low Disable ร่วมกับ Latch Base Clock Gating และเลือกใช้สัญญาณกระตุ้นการทำงานแบบ Active High สัญญาณนาฬิกาที่ได้จะมีเฟสเหมือนกับสัญญาณเดิมและมีการทำงานที่ถูกต้อง

(6) เมื่อใช้ Logic Low Disable ร่วมกับ Latch Base Clock Gating และเลือกใช้สัญญาณกระตุ้นการทำงานแบบ Active Low สัญญาณนาฬิกาที่ได้จะมีเฟสเหมือนกับสัญญาณเดิมและส่งผลทำให้การทำงานของวงจรทำงานช้าลงครึ่งรอบสัญญาณนาฬิกา

(7) เมื่อใช้ Logic High Disable ร่วมกับ Latch Base Clock Gating และเลือกใช้สัญญาณกระตุ้นการทำงานแบบ Active High สัญญาณนาฬิกาที่ได้จะมีการกลับเฟสกับสัญญาณเดิมและส่งผลทำให้การทำงานของวงจรทำงานช้าลงครึ่งรอบสัญญาณนาฬิกา

(8) เมื่อใช้ Logic High Disable ร่วมกับ Latch Base Clock Gating และเลือกใช้สัญญาณกระตุ้นการทำงานแบบ Active Low สัญญาณนาฬิกาที่ได้จะมีการกลับเฟสกับสัญญาณเดิมและมีการทำงานที่ถูกต้อง

ผลสรุปการทำงานที่กล่าวมาข้างต้นสรุปจากผลการทดลองกับวงจรที่มีการทำงานในช่วงสัญญาณนาฬิกาขาขึ้นเป็นหลัก (Active High) แต่ถ้าเป็นวงจรที่มีการทำงานในช่วงสัญญาณนาฬิกาขาลงผลการทำงานก็จะเปลี่ยนไปโดยจะสลับกันในเรื่องการกลับเฟสของสัญญาณนาฬิกา

ผลการทำงานของการทดลองที่ทำให้เกิดการเลื่อนการทำงานของวงจรในบางกรณีก็ไม่ทำให้เกิดการทำงานที่ผิดพลาดต่อการทำงานของวงจรโดยรวมเช่นการทดลองใน Alarm Register ซึ่งทำให้จำเวลาปลุกที่ต้องการตั้งเร็วขึ้นเพียงเล็กน้อย (ครึ่งคาบเวลา) จึงไม่มีผลต่อการทำงานโดยรวม หรือแม้แต่ T80 เองก็จะเห็นว่ามีเพียงคำสั่งในกลุ่ม LDIR (คำสั่ง LDIR และ LDDR) เท่านั้นที่มีการทำงานที่ผิดพลาดซึ่งถ้าหากแก้ไขได้ไม่ยากและให้ผลทางพลังงานที่ดีก็น่าจะเป็นทางเลือกที่ดีได้

ในการทดลองทั้ง 8 กรณีในการทำ Clock Gating ซึ่งได้ทำตามหลักการในบทที่ 2 จะไม่ทำให้สัญญาณนาฬิกาที่ได้เกิด Glitch ขึ้นแต่ผลการทดลองในส่วนของ Alarm Controller กลับพบว่าสองกรณีของการทำ Clock Gating แบบ Latch Free กลับทำให้เกิด Glitch ขึ้น ทั้งนี้เพราะโดยหลักการแล้วสัญญาณควบคุมเป็นสัญญาณควบคุมจากภายนอกวงจร ซึ่งไม่มีการทำงานในช่วงสัญญาณนาฬิกาเดียวกับทั้งวงจร แต่ในกรณีของ Alarm Controller นั้นสัญญาณควบคุมส่วนหนึ่งเกิดขึ้นมาจากการทำงานของ Alarm Controller เอง ดังนั้นเมื่อมีการทำ Clock Gating ในแบบที่ทำให้การเลื่อนการทำงานภายใน Alarm Controller การเกิดของสัญญาณควบคุมก็จะเปลี่ยนตามไปด้วยทำให้ไม่เป็นไปตามหลักการทำ Clock Gating ดังที่ตั้งใจไว้และมีผลทำให้เกิด Glitch ขึ้น แต่อย่างไรก็ตามผลจากการเลื่อนการทำงานไม่ได้ทำให้วงจรควบคุมแบบ Latch Base Clock Gating มีการทำงานที่เปลี่ยนไปจนเกิด Glitch ขึ้นมาด้วยทั้งนี้เพราะ Latch ที่ใช้จำค่าสัญญาณควบคุมก่อนนำไปทำ Clock Gating ยังคงทำงานอยู่ในช่วงเวลาเดิม

จากการทดลองจะเห็นว่าการทำ Clock Gating แบบ Latch Base ในกรณีที่ทำให้มีการเลื่อนการทำงาน (ช้าลงครึ่งรอบสัญญาณนาฬิกา) มีผลทำให้การทำงานผิดพลาดในทุกกรณีซึ่งต่างจากแบบ Latch Free ในกรณีที่ทำให้มีการเลื่อนการทำงาน (เร็วขึ้นครึ่งรอบสัญญาณนาฬิกา) ที่บางกรณีก็ให้ผลการทำงานโดยรวมที่เป็นปกติ ทั้งนี้เพราะการทำงานของวงจรไม่ได้อาศัยเฉพาะสัญญาณนาฬิกาเท่านั้นแต่ยังต้องอาศัยสัญญาณควบคุมในการทำงานภายในวงจรด้วยและในช่วงที่วงจรถูกกระตุ้นการทำงานโดยสัญญาณนาฬิกาสุดท้ายในกรณีของ Latch Base ที่ทำให้มีการเลื่อนการทำงานนั้นสัญญาณควบคุมก็ได้หยุดการทำงานไปแล้วจึงทำให้ไม่เกิดการทำงานใด ๆ ต่างจากการทำ Clock Gating แบบ Latch Free ที่ถึงแม้จะมีการเลื่อนช่วงเวลาการทำงานแต่จะเป็นการเลื่อนขึ้นซึ่งก็ยังอยู่ในช่วงที่สัญญาณควบคุมทำงานอยู่ แต่อย่างไรก็ตามวงจรที่มีสัญญาณควบคุมเพียงสัญญาณเดียวเช่นวงจร Alarm Register หลังจากการทำ Clock Gating แล้วยังสามารถที่จะตัดสัญญาณควบคุมออกได้เพราะไม่จำเป็นต้องใช้เลือกการทำงานในวงจร ซึ่งนอกจากจะทำให้ลดการใช้พลังงานลงได้แล้ว (ตามทฤษฎี) ยังเปิดโอกาสให้สามารถใช้การทำ Latch Base Clock Gating ในรูปแบบที่ทำเกิดการทำงานที่ช้าลงได้อีกด้วย

ตามหลักการแล้ว Latch Base Clock Gating ถือเป็นวิธีที่นิยมใช้มากที่สุดเพราะนอกจากจะให้ผลการทำงานที่เป็นปกติแล้วยังสามารถกำจัดการเกิด Glitch ได้ทุกกรณีและสามารถใช้ได้ดีกับสัญญาณควบคุมทั้งจากสัญญาณภายในตัววงจรเองหรือจากสัญญาณภายนอกวงจรในขณะที่การทำ Latch Free Clock Gating จะให้ผลการทำงานที่ไม่ดีเมื่อใช้กับสัญญาณควบจากภายนอกซึ่งมีรูปแบบการเกิดที่ไม่แน่นอน แต่จากการทดลองกับวงจร Alarm Controller และ Keyboard Buffer ซึ่งใช้สัญญาณควบคุมจากภายนอก (การกดปุ่ม) ในงานวิจัยนี้กลับพบว่าการทำ Clock

Gating แบบ Latch Base ให้ผลผิดพลาดในทุกกรณีและให้ผลการทำงานที่ดีเมื่อใช้การทำ Clock Gating แบบ Latch Free ที่เป็นเช่นนี้เพราะในการออกแบบการทดลองนั้นจะต้องออกแบบให้ค่าและช่วงเวลาส่งสัญญาณจากภายนอกตรงกันทุกการทดลองและจากคุณสมบัติของ Latch Base Clock Gating ที่อาจทำให้การทำงานเลื่อนออกไป 1-1.5 รอบสัญญาณนาฬิกาเมื่อใช้สัญญาณควบคุมจากภายนอกที่มีรูปแบบไม่แน่นอนทั้งนี้เพราะช่วงเวลาสำคัญของสัญญาณควบคุมที่มีการเปลี่ยนไป (วงจรปกติที่ทำงานสัญญาณขาขึ้นช่วงเวลาสำคัญของสัญญาณควบคุมคือช่วงสัญญาณนาฬิกาขาขึ้น แต่เมื่อมีการทำ Latch Base Clock Gating ช่วงเวลาสำคัญจะเปลี่ยนมาอยู่ในช่วงการทำงานของ Latch คือช่วงสัญญาณนาฬิกาขาลง) จึงทำให้ผลที่ได้จากการทดลองอาจไม่เหมือนกับต้นแบบซึ่งมีการทำงานในทันทีในช่วงเวลาสำคัญเดิม ในขณะที่การทำ Latch Free Clock Gating นั้นถึงแม้ว่าจะใช้สัญญาณควบคุมจากภายนอก (การกดปุ่มซึ่งมีรูปแบบไม่แน่นอน) แต่ก็สามารถใช้สัญญาณควบคุมจากภายในซึ่งมีความเกี่ยวเนื่องกับสัญญาณควบคุมจากภายนอกร่วมด้วยทำให้ปัญหาที่ควรจะเกิดไม่ได้เกิดขึ้นและวิธีนี้ยังไม่ทำให้ช่วงเวลาสำคัญของสัญญาณควบคุมเปลี่ยนไปจึงทำให้การทำงานโดยรวมไม่แตกต่างจากการทำงานของวงจรเดิม

จากผลการทดลองในงานวิจัยนี้พบว่าการทดลองที่ 3 ให้ผลทางพลังงานดีที่สุดในส่วนของ Alarm Clock ส่วนใน T80 นั้นผลพลังงานมีปัจจัยเพิ่มขึ้นอีกอย่างหนึ่งด้วยคือโปรแกรมที่รันอยู่ทำให้ผลทางพลังงานที่ดีที่สุดของการรันโปรแกรมแต่ละตัวไม่เหมือนกัน แต่อย่างไรก็ตามมีการทดลองที่ 3 เพียงการทดลองเดียวที่ให้ผลทางพลังงานที่ลดลงในการรันทั้งสามโปรแกรมเมื่อผลการรันถูกต้อง

2. สรุปผลงานวิจัย

ในส่วนการสรุปผลงานวิจัยนี้จะสรุปในสามประเด็นคือสรุปข้อดีข้อด้อยของการทำ Clock Gating ลักษณะการนำรูปแบบการทำ Clock Gating แต่ละรูปแบบไปใช้และส่วนที่ควรพิจารณานำเทคนิค Clock Gating ไปใช้ โดยมีรายละเอียดดังนี้

2.1 ข้อดีข้อด้อยของการทำ Clock Gating จากผลการทดลองในงานวิจัยนี้จะเห็นว่าเทคนิค Clock Gating สามารถลดอัตราการใช้พลังงานลงได้จริงโดยมีข้อดีและข้อด้อยดังนี้

ข้อดี

- สามารถนำไปประยุกต์ใช้กับวงจรดิจิตอลที่มีอยู่เดิมได้ง่าย
- มีการเพิ่มและเปลี่ยนแปลงของวงจรน้อยเมื่อเทียบกับพลังงานที่ลดได้
- ไม่สูญเสียประสิทธิภาพการทำงาน
- การประยุกต์ใช้ทำได้ง่ายแม้ในระดับการออกแบบวงจรด้วยภาษา HDL
- ผู้นำไปใช้ไม่จำเป็นต้องเป็นผู้ที่มีทักษะสูงก็สามารถทำได้

ข้อด้อย

- การทำ Clock Gating อาจทำให้เกิดการ Shift ของสัญญาณนาฬิกา ซึ่งในบางกรณีอาจส่งผลต่อผลการทำงานของวงจรโดยรวม
- อาจทำให้เกิดปัญหาจาก Clock Skew

2.2 ลักษณะการนำรูปแบบการทำ Clock Gating แต่ละรูปแบบไปใช้ โดยจะแยกรูปแบบการทำ Clock Gating ออกเป็น 3 ประเด็นดังนี้

2.2.1 การเลือกทำ Clock Gating แบบ Latch Free หรือ Latch Base

การทำ Clock Gating แบบ Latch Free สามารถใช้ได้ดีกับวงจรที่มีสัญญาณควบคุมเป็นสัญญาณจากภายใน แต่ถ้าจะนำไปใช้กับวงจรที่มีสัญญาณควบคุมจากภายนอกซึ่งมีรูปแบบการเกิดไม่แน่นอนจะต้องมีสัญญาณภายในที่เป็นผลเกี่ยวเนื่องกับสัญญาณภายนอกนั้นร่วมเป็นสัญญาณควบคุมด้วย ส่วนการทำ Clock Gating แบบ Latch Base นั้นใช้ได้ดีกับวงจรที่มีสัญญาณควบคุมจากภายใน แต่ถ้าเป็นสัญญาณควบคุมจากภายนอกจะต้องแน่ใจว่าการทำงานที่ช้าลงจากการทำ Clock Gating จะไม่ส่งผลต่อการทำงานของวงจรโดยรวม

ถ้าหากวงจรที่ต้องการทำ Clock Gating สามารถเลือกใช้ได้ทั้งสองวิธีควรเลือกทำ Clock Gating แบบ Latch Free เพราะวงจรที่เพิ่มเข้าไปมีขนาดเล็กกว่าซึ่งมีผลต่อทั้งพลังงานและการใช้พื้นที่

2.2.2 การเลือกวิธีคงสัญญาณแบบ Logic Low Disable หรือ Logic High Disable

การเลือกใช้แบบใดนั้นควรดูผลทางพลังงานเป็นหลัก เพราะการคงระดับค่าสัญญาณนาฬิกามีผลทางพลังงานต่อ Flip-Flops หรือ Register ไม่เหมือนกัน

2.2.3 การเลือกช่วงการทำงานแบบ Active High หรือ Active Low

การเลือกช่วงการทำงานของวงจรที่จะทำ Clock Gating นี้ควรเลือกในแบบที่ทำให้ไม่เกิดการเลื่อนการทำงาน แต่ถ้าจำเป็นต้องเลื่อนจะต้องแน่ใจว่าจะไม่ก่อให้เกิดการทำงานที่ผิดพลาดใด ๆ ขึ้น แต่อย่างไรก็ตามในการเลือกช่วงการทำงานนี้จะต้องมีการเปลี่ยนการทำงานของวงจรเดิมซึ่งในบางครั้งอาจไม่สามารถทำได้หรือมีความยุ่งยากและไม่คุ้มค่าที่จะเปลี่ยน ทำให้ไม่สามารถเลือกช่วงการทำงานโดยเปลี่ยนการทำงาน (Active Low หรือ Active High) ด้วยวิธีนี้ได้ แต่ก็สามารถใช้การเลือก Latch Base Clock Gating หรือ Latch Free Clock Gating เลือกช่วงการทำงานแทนได้ในกรณีที่ได้เลือกระดับการคงค่าสัญญาณนาฬิกาเอาไว้แล้ว หรือถ้าระดับการคงค่าสัญญาณนาฬิกา (ทั้ง Logic Low และ Logic High) มีผลทางพลังงานต่อ Flip-Flop หรือ Register ที่ใช้ในวงจรไม่แตกต่างกันมากก็สามารถใช้ระดับการคงค่าสัญญาณนาฬิกาเป็นตัวกำหนดช่วงเวลาการทำงานแทนได้

2.3 ส่วนที่ควรพิจารณานำเทคนิค Clock Gating ไปใช้ โดยพิจารณาในกรณีทั่ว ๆ ไปคือ

(1) กลุ่มของ Flip-Flops หรือ Register ที่มีจำนวนมากและใช้สัญญาณควบคุมเดียวกัน เช่น Flip-Flops หรือ Register ที่ใช้เก็บค่าข้อมูลต่าง ๆ เป็นต้น (ตัวอย่างในงานวิจัยนี้ เช่น T80 Register เป็นต้น)

(2) กลุ่มของวงจร Sequential ที่มีการทำงานในช่วงเวลาสั้น ๆ เมื่อมีสัญญาณควบคุมมากระตุ้นและมี Flip-Flops หรือ Register ภายในจำนวนมาก (ตัวอย่างในงานวิจัยนี้เช่น Alarm Controller เป็นต้น)

3. ข้อเสนอแนะ

3.1 ถ้าใช้ Latch Free กับวงจรที่มีสัญญาณควบคุมจากภายนอกซึ่งมีรูปแบบการเกิดที่ไม่แน่นอนจะต้องมีสัญญาณควบคุมซึ่งเกิดจากวงจรภายในที่เกี่ยวข้องกันร่วมด้วยเสมอ

3.2 ควรใช้ Latch Base กับสัญญาณควบคุมจากภายนอกในกรณีที่ไม่ต้องการการตอบสนองทันทีในห้วงสัญญาณนาฬิกาถัดไป

3.3 ควรรวมการทำ Clock Gating ของแต่ละวงจรที่มีการทำงานในช่วงจังหวะเวลาเดียวกันหรือสำพันธ์กันไว้ในวงจรเดียวกัน เพื่อผลทางพลังงานที่ดีขึ้น (เช่นการทดลอง Alarm Clock)

3.4 วงจรการทำ Clock Gating ที่ควรพิจารณาใช้คือวงจรที่ไม่ทำให้เกิดการเลื่อนการทำงาน (ทั้งวงจรที่ทำงานในช่วงสัญญาณนาฬิกาขาขึ้นและขาลง) ดังนี้

(1) ใช้ Logic Low Disable ร่วมกับ Latch Free Clock Gating และเลือกใช้สัญญาณกระตุ้นการทำงานแบบ Active Low

(2) ใช้ Logic High Disable ร่วมกับ Latch Free Clock Gating และเลือกใช้สัญญาณกระตุ้นการทำงานแบบ Active High

(3) ใช้ Logic Low Disable ร่วมกับ Latch Base Clock Gating และเลือกใช้สัญญาณกระตุ้นการทำงานแบบ Active High

(4) ใช้ Logic High Disable ร่วมกับ Latch Base Clock Gating และเลือกใช้สัญญาณกระตุ้นการทำงานแบบ Active Low

3.5 ในบางครั้งการทำ Clock Gating ก็จำเป็นจะต้องมีการเปลี่ยนแปลงวงจรต้นแบบเพื่อให้เข้ากับวงจรการทำ Clock Gating ในบางวิธีเช่น การเปลี่ยนช่วงการทำงาน (Active High และ Active Low) ซึ่งถ้าเปลี่ยนไม่มากและยังให้ผลทางพลังงานที่ดีก็น่าจะทำ

3.6 หากพบว่าการทำ Clock Gating ในแบบที่ทำให้เกิดการเลื่อนการทำงานให้ผลทางพลังงานที่ดีกว่าและจะนำไปใช้ ควรพิจารณานำไปใช้ในกรณีที่มั่นใจได้ว่าจะไม่ก่อให้เกิดทำงานที่ผิดพลาดใด ๆ ขึ้น

3.7 ควรที่จะคำนึงถึงการนำเทคนิค Clock Gating ไปใช้ตั้งแต่ขั้นตอนในการออกแบบ เนื่องจากเป็นเทคนิคที่ทำได้ไม่ยากนักแต่ให้ผลในการลดการใช้พลังที่ค่อนข้างมาก อีกทั้งการนำเทคนิค Clock Gating นี้ไปใช้กับวงจรในภายหลังก็มีความยุ่งยากบ้างในบางกรณี เช่นอาจจะต้องมีการแก้ไขวงจรบางส่วนเพื่อให้สามารถนำเทคนิค Clock Gating นี้เข้าไปประยุกต์ใช้ได้

บรรณานุกรม

1. ASIC [ออนไลน์] ม.ป.ป. [อ้างเมื่อ 1 พฤศจิกายน 2548]. จาก http://astronlogic.com/fpga.html
2. **ทำความรู้จักกับ FPGA** [ออนไลน์] ม.ป.ป. [อ้างเมื่อ 1 พฤศจิกายน 2548]. จาก http://astronlogic.com/fpga2.html
3. Nithya, R., Venkatesh, A., & Smita, B. (1998). Automatic Insertion of gated Clocks at Register Transfer Level. **VLSI for the Information Appliance.** Proceedings of 12th International Conference on VLSI Design.
4. Sanduleanu, Mihai A.T., Tuijl, & Ed A.J.M.van. (2002). Power considerations in sub-micron digital CMOS. **Power Trade-offs and Low-power in Analog CMOS ICs.** (pp. 9-29). London: Kluwer Academic Publishers.
5. Christoph, H., & Ulf Schlichtmann. (2004). Ultra-Low-Power design: Device and logic design approaches. **Low-power Electronics and Design.** (pp. 1-20). Boston: Kluwer Academic Publishers.
6. Soudris, D., Mizas, H., Theodoridis, G., Theoharis, S., Thanailakis, A., & Goutis, C. E. (1998). General Overview of Low-Power Design Techniques. **LPGD project (ESPRIT 25256): Report D1.1R1.** [n.p.].
7. Soudris, D., Mizas, H., Theodoridis, G., Theoharis, S., Thanailakis, A., & Goutis, C. E. (1998). Structure of the Low-Power Design Flow. **LPGD project (ESPRIT 25256): Report D1.2R1.** [n.p.].
8. Mueller, M., Wortmann, A., Simon, S., Kugel, M., & Schoenauer, T. (2004). The impact of Clock Gating schemes on the power dissipation of synthesizable register files. **In Proceedings of International Symposium on Circuits and Systems (ISCAS '04).** (pp. 609-612). [n.p.].
9. Raghunatan, A., Dey, S., & Jha, N.K. (1999). Register transfer level power optimization with emphasis on Glitch analysis and reduction. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**, 18(8), 1114-1131.
10. Emnett, F., & Biegel M. (2000). **Power reduction through RTL Clock Gating.** presented at the Synopsys Users Group (SNUG), San Jose, CA.
11. Benini, L., & Micheli, G.De. (1996). Automatic Synthesis of Low-Power Gated-Clock Finite-State Machines. **IEEE Transaction on CAD**, (june), 630-643.

12. Chao, T.-H., Hsu, Y.-C., Ho, J.-M., Boese, K. D., & Kahng, A. B. (1992). Zero Skew Clock Routing with Minimum Wirelength. **IEEE Transaction on Circuits and Systems-II: Analog and Digital Signal Processing**, CAS-39(11), 799-814.

13. Pangjun, J., & Sapatnekar, S. (1999). Clock distribution using multiple voltages. **In Proceedings of International Symposium on Low Power Electronics and Design**. (pp. 145-151). [n.p.].

14. Gowan, M., Biro, L., & Jackson, D. (1998). Power considerations in the design of the Alpha 21264 microprocessor. **In Proceedings Of 35th Design Automation Conference (DAC)**, (june), 726-731.

15. Bahar, R.I., & Manne, S. (2001). Power and Energy Reduction Via Pipeline Balancing. 28th Annual International Symposium. (pp. 218-229). Göteborg, Sweden: [n.p.].

16. Ian Brynjolfson, and Zeljko Zilic. (2000). Dynamic Clock Management for Low Power Applications in FPGAs. **In IEEE Custom Integrated Circuits Conference**. (pp. 139-142). [n.p.].

17. Qing Wu, Pedram, M., & Xunwei Wu. (2000). Clock Gating and Its Application to Low Power Design of Sequential Circuits. **IEEE Transactions on Circuit and Systems**, 47(3), 415-420.

18. Maria del Pilar Parra Fernandez, Antonio Jose Acosta Jimenez, Raúl Jiménez Naharro, & Manuel Valencia Barrero. (2005). Selective Clock-Gating for Low-Power Synchronous Counters. **Journal of Low Power Electronics**, 1(1), 11-19.

19. Wallner, D. (2002). **T80 cpu: Overview.** Retrieved July 3, 2005, from http://www.opencores.org/projects.cgi/web/t80/overview