

การทดลองที่ 4 วงจรบวกเลขโดยใช้วิธี Schematic บนบอร์ด FPGA

วัตถุประสงค์

1. เพื่อให้นักศึกษาฝึกใช้งานโปรแกรม툴ช่วยการออกแบบวงจรดิจิทัลเป็น
2. เพื่อให้นักศึกษาฝึกการใช้งาน FPGA

หมายเหตุให้อ่านเอกสารการทดลองและเอกสารประกอบให้ครบก่อนเริ่มทำการทดลอง

บทนำ

การทดลองนี้แตกต่างจากการทดลองที่ผ่านมา นักศึกษาจะได้รู้จักการออกแบบวงจรด้วยการวาดรูป (Schematic Circuit) และบันทึกโปรแกรมลงในชิป Field Programmable Gate Array (FPGA)

โปรแกรม ISE Xilinx WebPACK

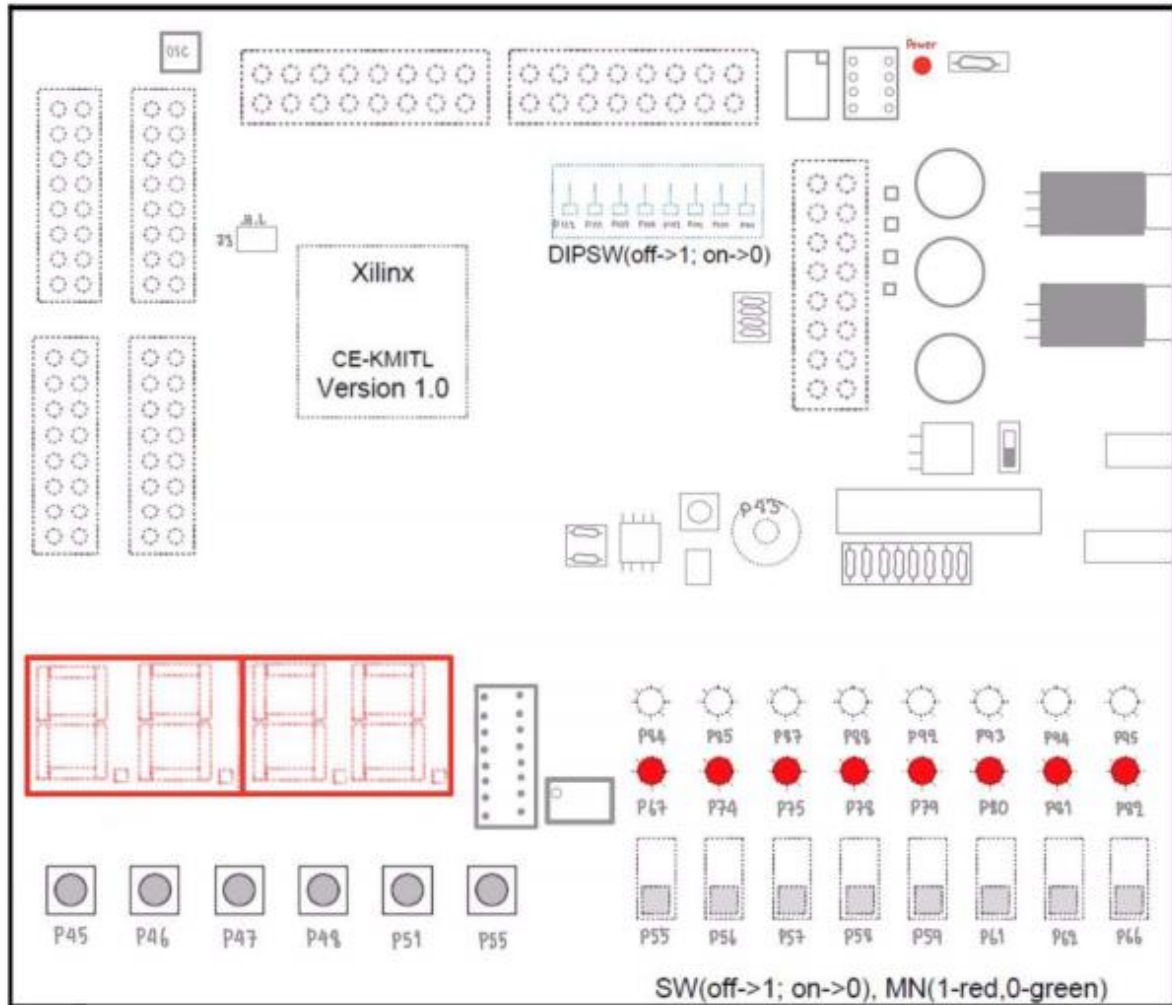
นักศึกษาที่มีหนังสือ “ออกแบบไอซีดิจิทัลด้วย FPGA และ CPLD ภาคปฏิบัติ โดยใช้วิธี Schematic ซอฟต์แวร์툴 ISE WebPACK”ให้อ่านหนังสือก่อนการทดลองในบทที่ 1 และ 2 ก่อน

บอร์ดทดลอง

วงจรดิจิทัลในปัจจุบันมีความซับซ้อนสูงทำให้การปรับแก้วงจรทำไม่สะดวก การลากสายไฟการติดตั้งไอซีเกตเพิ่มเติมเพื่อปรับเปลี่ยนการทำงานของวงจรทำได้ยาก จึงมีการใช้ไอซีที่สามารถโปรแกรมวงจรได้ ซึ่งมีหลายแบบไม่ว่าจะเป็น Generic Logic Array (GAL), Programmable Array Logic (PAL), Programmable Logic Device (PLD), Complex Programmable Logic Device (CPLD) และ Field Programmable Gate Array (FPGA) ซึ่งล้วนแต่มีลักษณะที่แตกต่างกัน ทั้งขนาดจำนวนเกต ภาวการณ์คงอยู่ของโปรแกรม/ข้อมูล การโปรแกรม โดยการทดลองนี้ใช้งาน FPGA

FPGA มีหลายบริษัทเป็นผู้ผลิต แต่มีสองบริษัทใหญ่คือ Xilinx กับ Altera โดยตระกูลผลิตภัณฑ์ของ Xilinx ที่นิยมกันคือตระกูล Spartan รุ่นเทคโนโลยีขนาด 45nm และตระกูล Virtex รุ่นเทคโนโลยีขนาด 28nm ทางด้าน Altera ตระกูลผลิตภัณฑ์ที่นิยมคือตระกูล Stratix รุ่นเทคโนโลยีขนาด 40nm และ 28nm การทดลองในวิชานี้ใช้ Xilinx Spartan-6 รุ่นเทคโนโลยีขนาด 45nm

ทั้งนี้ Spartan-3 ที่ใช้ในการทดลองคือรุ่น XC6SLX9 มีความจุขนาด 400,000 เกต จำนวนเกตยิ่งมามากยังสามารถสร้างวงจรที่มีขนาดใหญ่และซับซ้อนได้ จุดอ่อนที่ต้องระมัดระวังของ FPGA คือมีอายุการใช้งานจำกัด กล่าวคือสามารถเขียนโปรแกรมลงได้ไม่เกิน 20,000 ครั้ง จากนั้นต้องเปลี่ยนไอซี FPGA ใหม่ การเขียนโปรแกรมลง FPGA ทำได้โดยใช้สายซึ่งใช้มาตรฐานการเชื่อมต่อแบบ Joint Test Action Group (JTAG) และส่งผ่านโปรแกรมด้วย Parallel Port to JTAG (ปัจจุบันนิยมใช้ USB to JTAG มากกว่า)



รูปที่ 1 เลย์เอาต์ของบอร์ด FPGA รุ่น XC6SLX9

บอร์ด FPGA ที่ใช้ในการทดลองดังรูปที่ 1 นั้นหมายเลข Pxx ต่างๆ ระบุถึงขาที่เชื่อมกับตัวชิป FPGA ฉะนั้นจึงไม่ควรสัมผัสตัวไอซี ขา และส่วนประกอบในบอร์ด เนื่องจากไฟฟ้าสถิตจากร่างกายอาจก่อให้เกิดความเสียหายแก่ไอซี อีกทั้งยังทำให้เกิดอ็อกไซด์ที่ขาและส่วนประกอบโลหะในวงจร

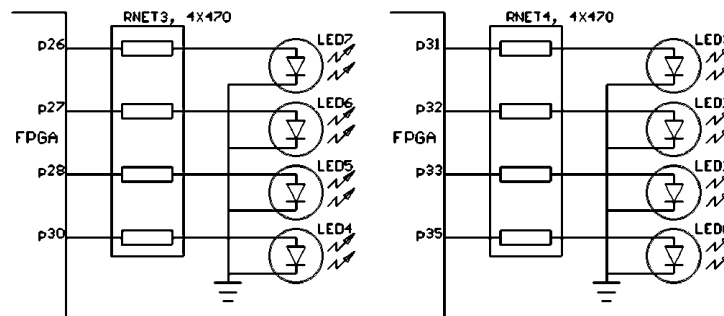
วงจรภายในของบอร์ดทดลอง

- **สวิตช์ (SW0 – SW7)** ลักษณะการต่อภายในวงจรเป็นการต่อแบบ R - Pull Up ดังรูปที่ 2 ซึ่งตัวต้านทานต่อเข้า VCC เพื่อรักษาระดับของแรงดันให้คงที่ วงจรจึงอยู่สถานะลอจิก "1" ตลอดเวลา และเมื่อเลื่อนสวิตช์จะให้สถานะลอจิก "0" หรือที่เรียกว่า Active Low เพราะจะทำงาน (ON) เมื่อกระแสไฟฟ้าไหลลงกราวด์ วงจรแบบ Pull Up ได้รับความนิยมมากเพราะกันสัญญาณรบกวนได้ดี ฉะนั้นเมื่อเลื่อน SW = OFF ให้ "1" และ ON ให้ "0"



รูปที่ 2 วงจร R-Pull Up (ซ้าย) และวงจร R-Pull Down (ขวา)

- หลอด LED0 – LED7 ลักษณะวงจรเป็นดังรูปที่ 3 สัญญาณลอจิกจาก FPGA ผ่าน R ความต้านทาน



470Ω เพื่อจำกัดกระแสป้องกัน LED เสียหาย ฉะนั้นเมื่อกำหนดให้ Pxx ดังกล่าวเป็น “1” หลอด LEDx จะสว่างเป็นสีแดง และเมื่อกำหนดให้ Pxx เป็น “0” LED จะไม่สว่าง

รูปที่ 3 วงจรขับ LED สีแดง

- หลอด LED สองสี (MN0 – MN7) ลักษณะวงจรขับแสดงดังรูปที่ 4 ลักษณะวงจรมีอินเวอร์เตอร์ทำหน้าที่กลับบิตโดยต่ออนุกรมกับ LED กับ R ความต้านทาน 470Ω เพื่อจำกัดกระแสป้องกัน LED เสียหาย หากลอจิกที่ขา P36 (รุ่นที่ใช้ในการทดลองคือ P84) เป็น “1” จะได้ผลลัพธ์ไปยัง LED เป็น “0” กระแสย่อมไหลผ่าน LED ดวงซ้ายมือทำให้ติดสว่าง แต่หากขา P36 เป็น “0” กระแสจะมีทิศทางกลับกัน LED ดวงขวามือจึงติดสว่างแทน ฉะนั้นเมื่อกำหนดให้ Pxx ดังกล่าวเป็น “1” หลอด MNx จะสว่างเป็นสีแดง หากเป็น “0” จะเป็นสีเขียว และกำหนดเป็นสัญญาณนาฬิกาจะติดสลับไปมาเป็นสีส้ม

รูปที่ 4 วงจรขับ LED สองสี

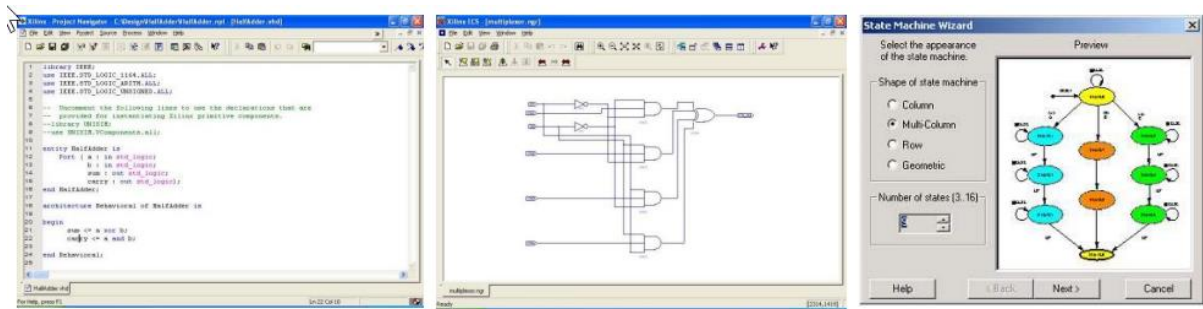
ขั้นตอนการออกแบบวงจรอย่างย่อของ version 14.7(ฉบับเต็ม version 8.1i ดูในหนังสือ)

กระบวนการออกแบบวงจรดิจิทัลด้วย FPGA มีทั้งหมด 5 ขั้นตอนดังนี้



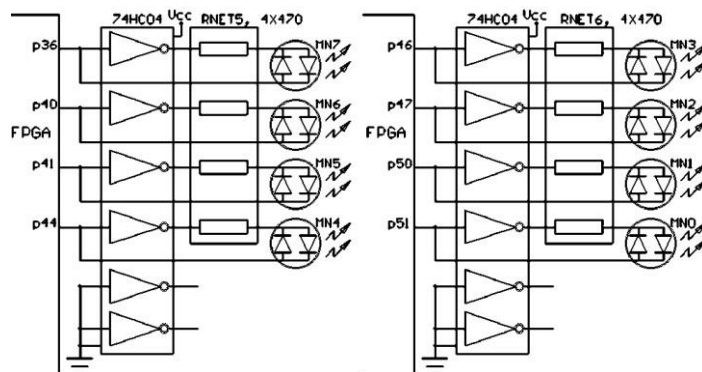
รูปที่ 5 กระบวนการออกแบบ 5 ขั้นตอน

1. **Design Entry** การสร้างวงจรด้วยวิธีการต่างๆ เช่น สร้างด้วยภาษา VHDL หรือ Verilog สร้างด้วยวิธีการออกแบบผังวงจร (Schematic Design) และสร้างด้วยแผนภูมิสถานะ (State diagram) ดังรูปที่ 6 ซึ่งการทดลองครั้งนี้สร้างวงจรด้วยวิธี Schematic Design



รูปที่ 6 ขั้นตอนการสร้างวงจรดิจิทัลทั้ง 3 แบบ

2. **Design Verification** การตรวจสอบวงจรที่ออกแบบเป็นการนำวงจรที่ออกแบบมาตรวจสอบ



ความถูกต้องด้วยโปรแกรมจำลองการทำงาน โดยทั่วไปไม่มีการจำลองการทำงานใน 3 ระดับคือ

- **Behavioral Simulation** การจำลองเฉพาะพฤติกรรมของวงจรโดยยังไม่คิดถึงโครงสร้างภายใน เพื่อได้แบบจำลองการทำงานเบื้องต้น
 - **Functional Simulation** การจำลองการทำงานของโค้ดในระดับ Register-Transfer-Level (RTL) เพื่อตรวจสอบการทำงานของวงจรถ้าก่อนนำไป Synthesis
 - **Timing simulation** การจำลองการทำงานที่ใกล้เคียงกับฮาร์ดแวร์มากที่สุดเนื่องจากนำข้อมูลเชิงเวลา (Timing) ที่เกิดขึ้นใน FPGA มาประกอบด้วย มักใช้หาข้อผิดพลาดในกรณีที่การทำงานจริงบน FPGA ไม่เหมือนผลจาก Functional Simulation
3. **Design Synthesis** การสังเคราะห์วงจร คือ การแปลงโค้ดให้เป็นวงจรในระดับเกต (Gate - level) ซึ่งเป็น Netlist ของเกตต่างๆ ในวงจร ผู้ออกแบบต้องระบุถึงเทคโนโลยี ผู้ผลิต และหมายเลขรุ่นของฮาร์ดแวร์ที่ใช้แก่โปรแกรม Synthesis ซึ่งบางโปรแกรมเปิดโอกาสให้เลือกวิธีการสังเคราะห์วงจรได้ว่าต้องการออปติไมซ์แบบใดระหว่างความเร็วสูงกับประหยัดเนื้อที่
 4. **Design Implementation** ขั้นตอนนี้ประกอบด้วย การแบ่งวงจร การวางอุปกรณ์ และการเชื่อมต่อสัญญาณ (Partitioning, Placement and Routing) ตามลำดับ โดยแบ่งวงจรที่ได้จากการสังเคราะห์แล้วแยกเป็นวงจรย่อยๆ เพื่อวางลงในโครงสร้างของอุปกรณ์ชนิดต่างๆ ตามชิปที่เลือกเป็นเป้าหมาย จากนั้นจึงวิเคราะห์ความเหมาะสมถึงการวางวงจรย่อยๆ เหล่านั้น ณ ตำแหน่งภายในชิปดังกล่าวท้ายสุดเป็นขั้นตอนการเชื่อมต่อสัญญาณภายในชิปเข้าด้วยกัน
 5. **Download and Test** เป็นการนำข้อมูลวงจรบันทึกลงชิป โดยใช้ไฟล์บิตสตรีม (.JED) ที่ได้จากขั้นตอน Design Implementation ก่อนหน้า

หมายเหตุ

- การใช้งานโปรแกรม Xilinx ISE WebPack นักศึกษาสามารถศึกษาได้จาก การทดลองที่ 0
- จากการทดลองที่ 0 นักศึกษาจะเห็นว่าการดาวน์โหลดข้อมูลวงจร หรือการโปรแกรมลง FPGA 2 รูปแบบ คือการโปรแกรมลง PROM ซึ่งเป็นหน่วยความจำของ FPGA ทำให้การใช้งานบอร์ด ครั้งต่อไปไม่ต้องดาวน์โหลดข้อมูลวงจรจากเครื่องคอมพิวเตอร์อีก (เมื่อปิดเครื่อง และเปิดใหม่วงจรที่โปรแกรมไว้ยังคงอยู่) ซึ่งมีอายุการใช้งานจำกัด (สามารถเขียนโปรแกรมลงได้ไม่เกิน 20,000 ครั้ง) ส่วนการโปรแกรมลง Volatile memory ของ FPGA ต้องมีไฟเลี้ยงตัวชิปอยู่ตลอดเวลาหากไฟดับวงจรจะหายไปทันทีแต่สามารถโปรแกรมได้ไม่จำกัด

ดังนั้นในระหว่างการทดลองให้นักศึกษาโปรแกรมวงจร Volatile memory ของบอร์ด FPGA เพื่อทดสอบและเมื่อต้องการส่งการทดลอง จึงโปรแกรมลง PROM ของบอร์ด FPGA แล้วนำมาส่งได้

การทดลอง

1. ให้นักศึกษาออกแบบพร้อมแสดงวิธีการออกแบบวงจรสร้างสัญญาณเตือนในรถยนต์ซึ่งต่ออยู่กับ Buzzer เพื่อเตือนผู้ขับขี่รถยนต์ในกรณีต่างๆ ดังต่อไปนี้
 - 1.1 เมื่อประตูผู้ขับขี่เปิดและมีการติดเครื่องยนต์
 - 1.2 เมื่อมีผู้ขับขี่นั่งในเบาะผู้ขับขี่แต่ไม่มีการคาดเข็มขัดนิรภัยและมีการติดเครื่องยนต์
 - 1.3 เมื่อเปิดไฟหน้าแต่ไม่มีการติดเครื่องยนต์
 - 1.4 เมื่อไม่มีผู้ขับขี่นั่งในเบาะผู้ขับขี่แต่มีการคาดเข็มขัดนิรภัยและประตูผู้ขับขี่ปิด

โดยกำหนดให้เซนเซอร์แต่ละจุดส่งสัญญาณดิจิตอลมาดังนี้

- H = Headlights คือไฟหน้า จะเป็น 1 เมื่อเปิดไฟ และเป็น 0 เมื่อดับไฟ
- S = Seat คือเบาะผู้ขับขี่ จะเป็น 1 เมื่อมีคนนั่ง และเป็น 0 เมื่อไม่มีคนนั่ง
- B = SeatBelt คือเข็มขัดนิรภัย จะเป็น 1 เมื่อคาด และเป็น 0 เมื่อไม่ได้คาด
- D = Door คือประตูผู้ขับขี่ จะเป็น 1 เมื่อปิด และเป็น 0 เมื่อเปิด
- E = Engine คือเครื่องยนต์ จะเป็น 1 เมื่อติดเครื่องยนต์ และเป็น 0 เมื่อดับเครื่องยนต์

และกำหนดให้รับอินพุต H S B D E ทางสวิตช์โยกบนเครื่อง Logic Trainer และต่อเอาต์พุตเข้ากับ Buzzer บน Logic trainer

2. ให้นักศึกษาวาดไดอะแกรมของวงจรบน ISE WebPack
3. ทดสอบการทำงานโดยการ Download ลงบอร์ด FPGA
4. โปรแกรมลง PROM ของบอร์ด FPGA แล้วนำมาส่งกับอาจารย์ผู้ดูแลการทดลอง