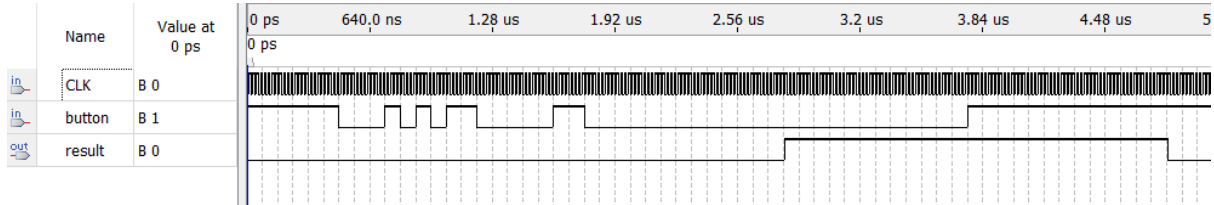


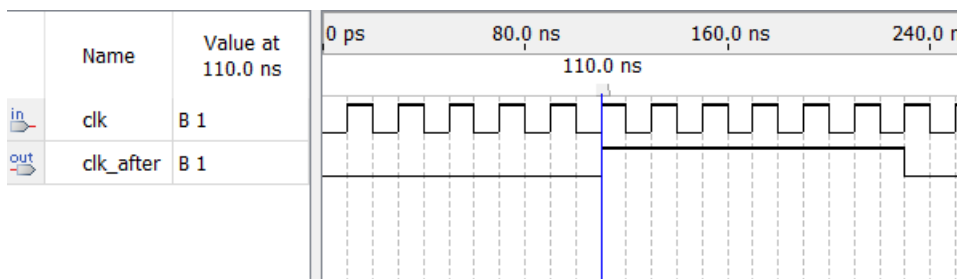
## ผลการจำลอง

### ผลทดสอบในส่วน debounceSwitch



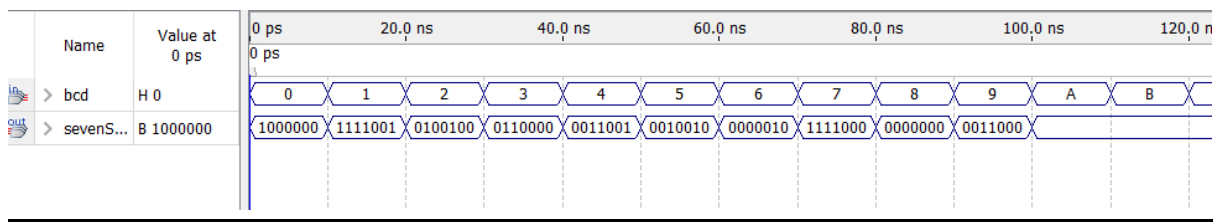
โดยให้ CLK เท่ากับ 50Mhz แต่ว่า ตั้งค่า cnt\_max ไว้ ที่ 50 ซึ่ง ผลที่ได้ นั้น ถูกต้องสัญญาณรบกวนที่เปลี่ยนแปลงไม่ต่อเนื่องถึง 50 clk นั้นถูกตัดออกไป และ ทำการสลับ out จาก low เป็น high และ high เป็น low ตามโค้ดที่ได้ ออกแบบไป

### ผลทดสอบในส่วน divid\_Clk



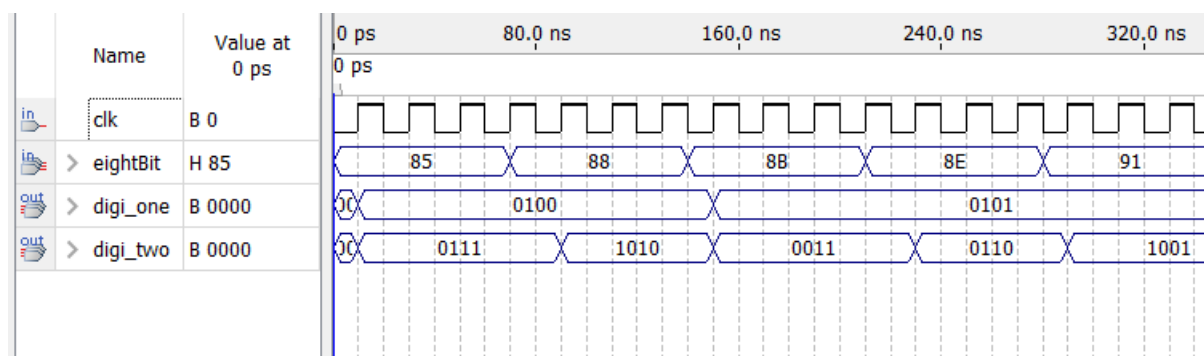
ตั้ง clk เป็น 50Mhz และ ตั้ง Max\_count เท่ากับ 5 ซึ่งก็คือเมื่อนับ clk ได้ 5 ลูกคลื่นแล้ว จะทำการสลับ จาก low เป็น high และ high เป็น low ซึ่ง การที่นับถึง 5 นั้นจะทำให้ ความถี่ลดลง เหลือ 10ลูกคลื่น ต่อ 1 ลูกคลื่น หรือลดลง 10 เท่า

## ผลทดสอบในส่วน bcd\_to\_sevenSeg



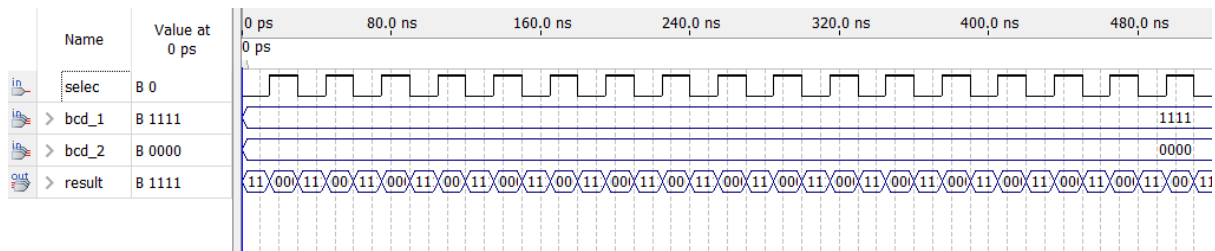
ผลที่ได้คือเมื่อ input เมื่อแปลงจาก bcd เป็นเลขแล้วนำไปแปลงเป็น output สำหรับ 7segment นั้นถูกต้องตามที่ต้องการ และถ้าหาก ไม่อยู่ในช่วง 0 – 9 นั้น จะ ได้ค่าที่ไม่มีความหมาย

## ผลทดสอบในส่วน eightBitToBcd



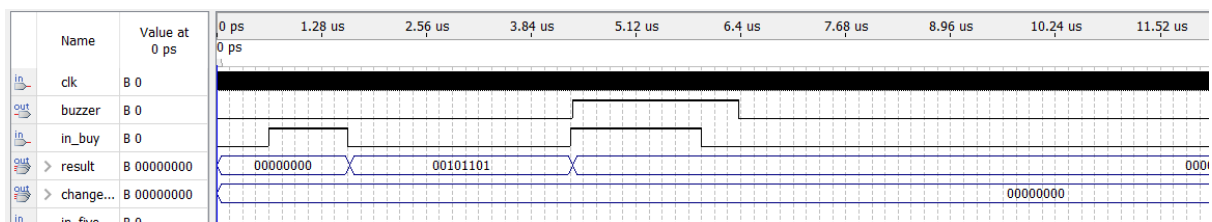
ผลที่ได้คือ เมื่อ input มีค่า integer เป็น 85 output digit หนึ่งจะมี bcd ที่มีค่าเป็น 4 และ difgi สองจะมีค่า เป็น 7 ซึ่ง เป็นค่าที่ถูกต้อง (นับจาก0)

## ผลทดสอบในส่วน MuxTwo

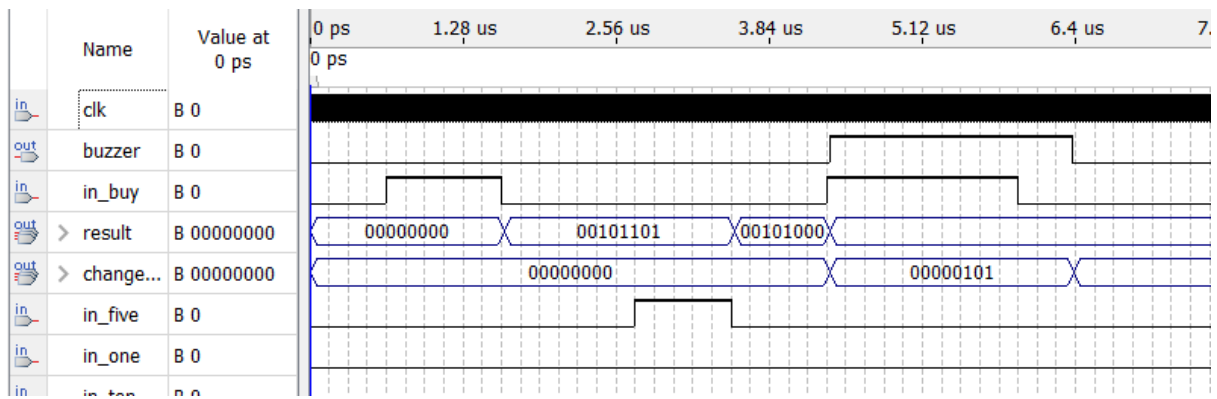


เมื่อให้ input 1 คือ 1111 และ 2 เป็น 0000 จากการจำลองจะเห็นว่า output ขึ้นอยู่กับ ตัว selec ซึ่งตรงตามที่ต้องการ

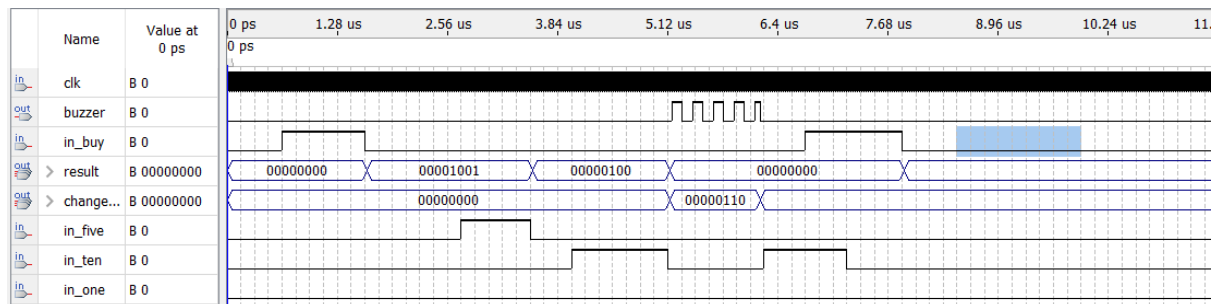
## ผลทดสอบในส่วน processCoin



จำลองในส่วนของการกด คิดเงินแล้วกด ยกเลิก ได้ผลตามที่ต้องการคือ กดครั้งแรก จะให้ผลลัพธ์ราคาสินค้าแสดงออกทาง result และกดอีกครั้งจะทำการทำให้เป็น 0 และมีเสียง buzzer ดัง ประมาณ ค่าที่ตั้งไว้ ซึ่งในการจำลองได้ปรับลงเหลือ 50 clk

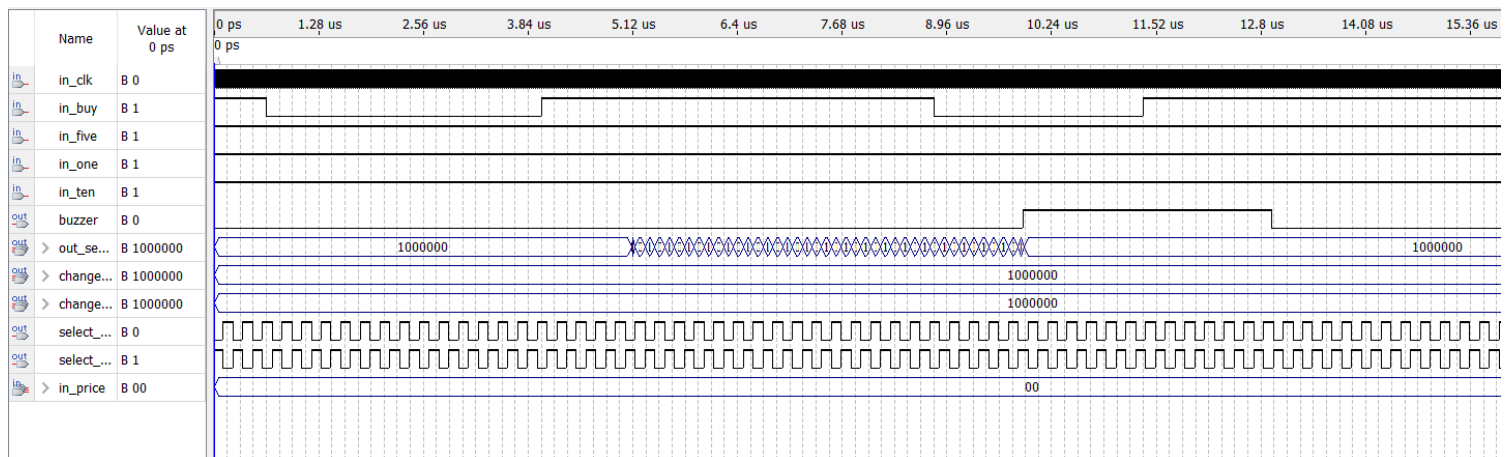


จำลองเมื่อคิดราคาจ่ายเงินแล้วบางส่วนแต่กดยกเลิก จะเห็นได้ว่า จากการจำลองเมื่อหยอดเหรียญ 5 ลงไปแล้วกด ยกเลิก ในช่วงที่ buzzer ดัง จะมีการแสดงส่วนของการทอนเงิน เป็นจำนวน 5 บาท เช่นกัน

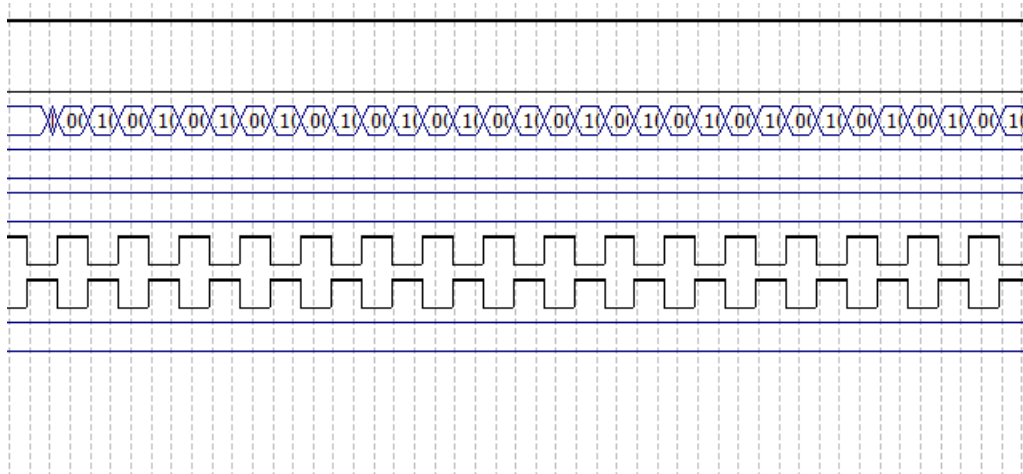


ทดลอง เมื่อจ่ายเงินครบ จะเห็นว่า buzzer จะดังเป็น จังหวะที่คงที่ และมีการแสดง เงินทอนที่จ่ายเกินไป และจากนั้นจะกดจ่ายเงินไม่ได้ เมื่อกด ปุ่ม buy อีกครั้ง ก็จะเริ่มการคิดราคาใหม่อีกรอบตามที่ต้องการ

ผลทดสอบในส่วน ตัวหลัก หรือ vendingMach

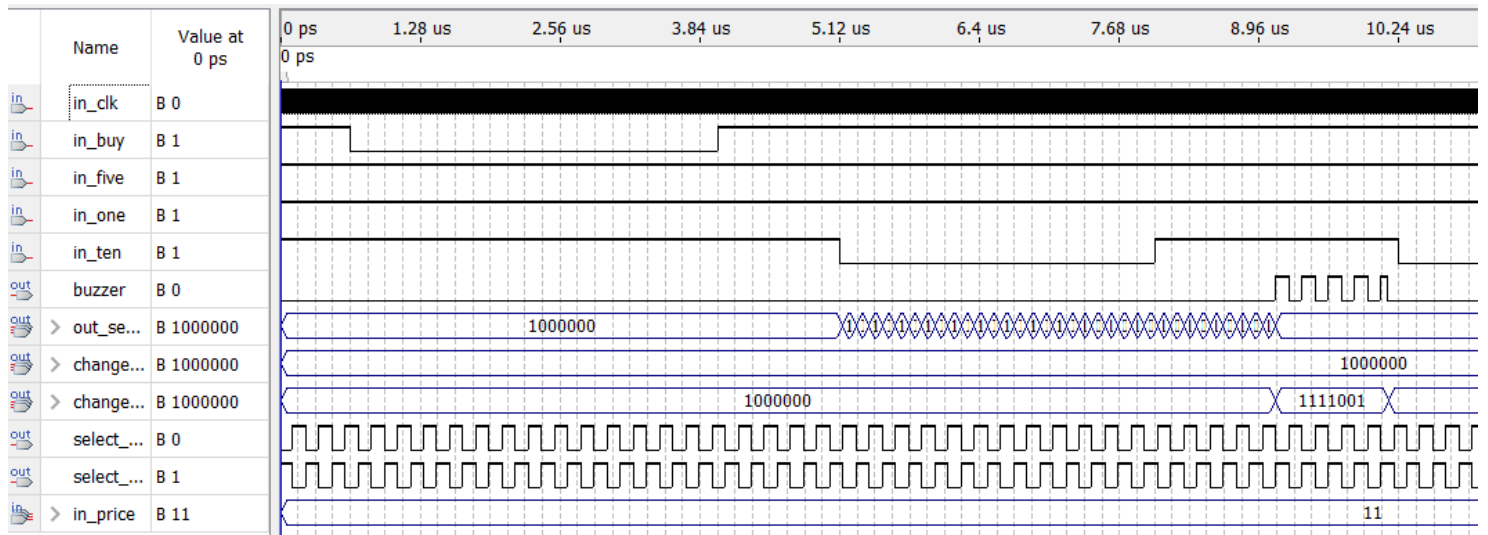


จากการจำลองการกดปุ่ม แบบ **Active Low** จะได้ผลตามที่ต้องการ คือกดครั้ง  
แรกอ่านราคาและครั้งที่ 2 เพื่อยกเลิก ส่วนที่เห็นในส่วนที่มีการสลับค่ากันมากมาย  
ตรง

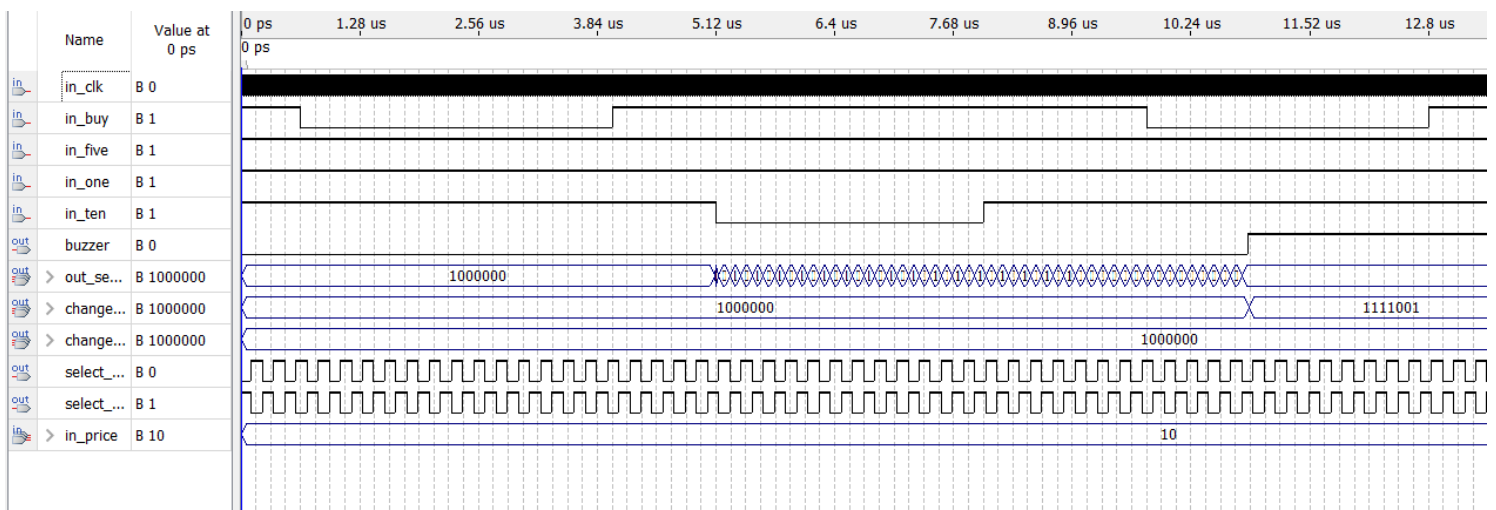


นั่นเป็นผลจากการที่ตัว **selec** ใน **mux** ทำงานเพื่อเลือก **output** ไปให้ 7 **segment** ที่ใช้ **input** ร่วมกันนั่นเอง

ผลการจำลองในสภาวะอื่นๆของโค้ดตัวนี้



การจ่ายเงินเกิน จะมีเสียงbuzzer ดังเป็นจังหวะ



จ่ายเงินไปแล้วแต่กดยกเลิกการซื้อ ทำให้มีการทอนเงินพร้อมมีเสียง buzzer ดังยาวต่อเนื่องไม่เป็นจังหวะ