**VLSISystemDesign**

**Homework-1**

**Part2**

**學號：N26991784**

**姓名：林宏達**

3. (50 points) Write and verify an ALU which is from exercise 2 in part I combined with a register file as shown in Fig. 3. The 32-bit register file contains 32 registers. This register is shown in Fig. 4 consists of two Read ports and two data output ports (for source register and destination register) and one Write Address Port and one Write Data Port. It shall have the following functions:

※ Working at positive edge of clock

※ When reset, data values in all registers are reset to zero

※ When enable is high and write is asserted, the write\_data is stored into the register as pointed by write\_address

※ When enable is high and read is asserted, the data that are stored in registers as pointed by read\_address1 & read\_address2 are written to src1 & src2

※ It reference code is also attached for your reference.

**a. A summary in the beginning to state what has been done (such as SMILE CPU, synthesis, post-synthesis simulation, additional branch instruction with verification)**

ANS：

我們做了pre- synthesis simulation。一開始MOVE了兩個值到R1跟R2，之後再做R1跟R2的相加。

**b. A block diagram for your completed SMILE CPU indicating all necessary components and I/O pins. Note please use MS Visio that is available in computer center in the university.**

ANS：

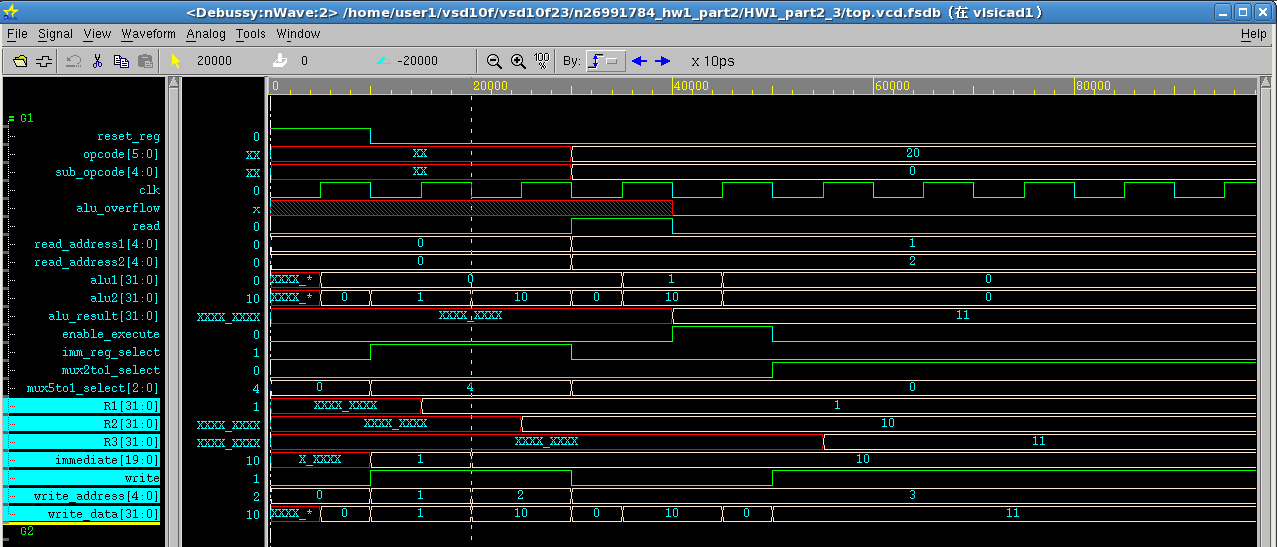


我的電路PORT命名的是與HOMEWORK上的圖示一樣的，所以我就沒有再VISIO上再畫一張一樣的圖片。PORT命名跟圖一樣的好處就是之後比較不會亂，MODULE之後引用直接複製即可，不太需要再做修改。

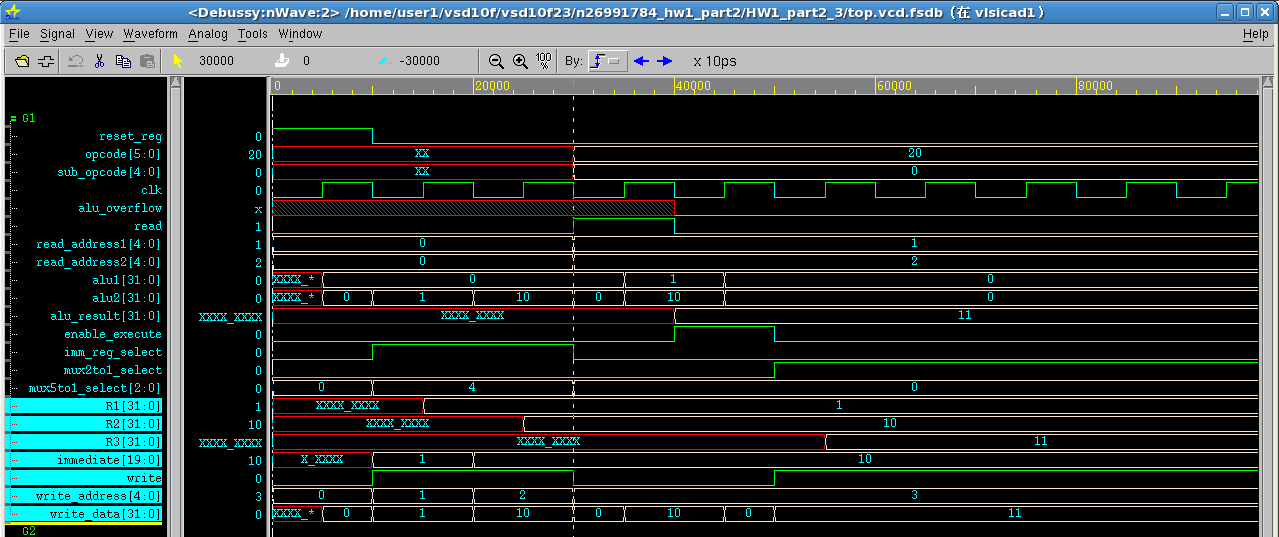
**C. Simulated waveforms with proper explanation**

ANS：

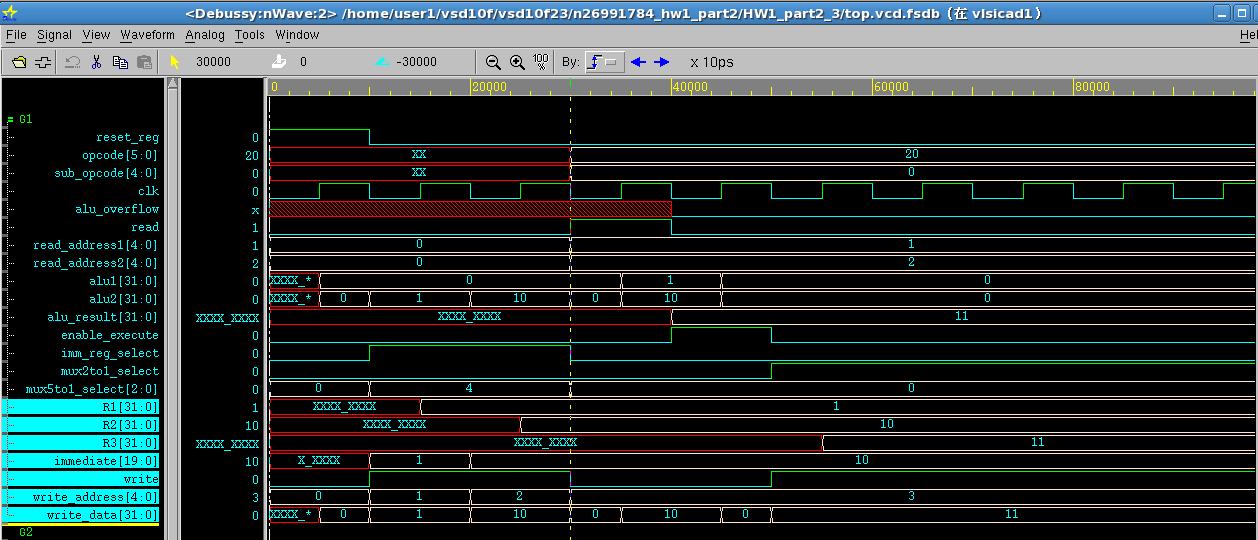
首先先MOVE值0001到R1。



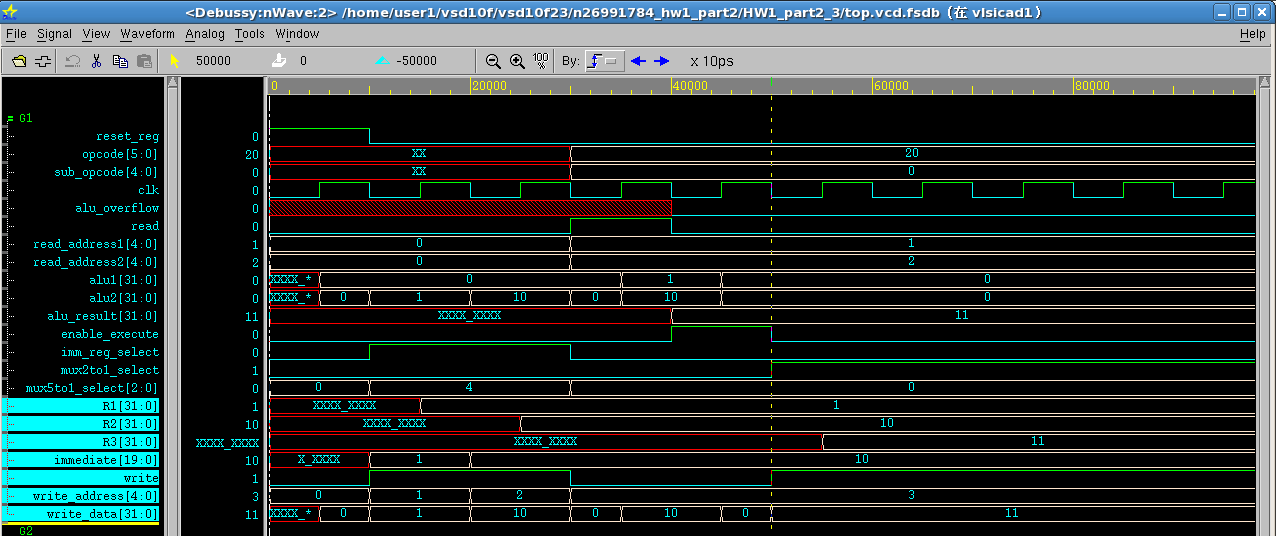
在MOVE值0010到R2。



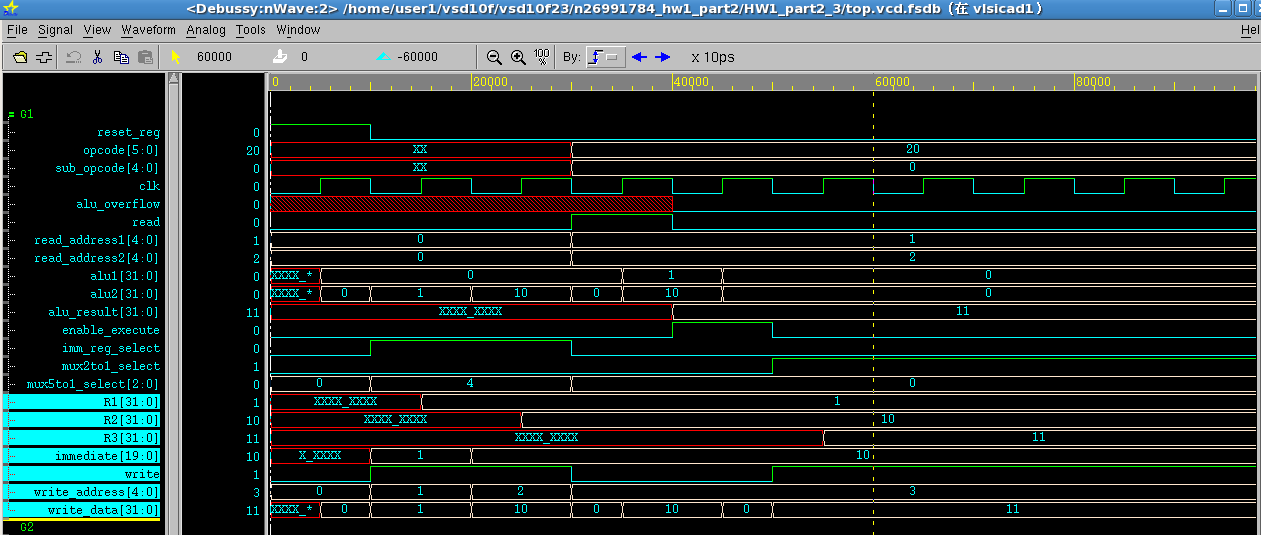
之後再READ暫存器R1與R2做ADD運算。



第4步驟在做R1與R2的ADD這動作EXECUTE。



最後WRITEBACK回暫存器R3。



**D. Learned lesson and Conclusion**

ANS：第3題等於是把自己當作是COMTROLLER，所以坐起來還蠻沒有阻礙，但也了解到CONTROL真正的動作流程，對之後的第4題有很大的幫助。

4. (60 points) Design a controller unit that is able to decode an input instruction as shown in Fig. 5 and control the datapath unit designed in Problem 4 to carry out the designated operation. A reference state diagram is attached in Fig. 6 for your reference. Develop a testbench that includes the instructions with the format as shown in Fig. 7 and verify the overall system (controller + ALU + register file).

**a. A summary in the beginning to state what has been done (such as SMILE CPU, synthesis, post-synthesis simulation, additional branch instruction with verification)**

ANS：

我們做了pre- synthesis simulation。其中我在immediate的選擇多工器的部分，多加了對5BIT與15BIT的ZERO-ENTEND與SIGN-EXTEND，將BIT數擴充到20BIT，之後再經過後面的SIGN-EXTEND擴充到32BIT。

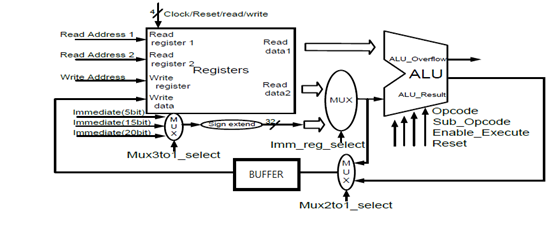
由於電路ENABLE\_EXECUTE的PORT，一旦變為0，我的ALU輸出馬上就轉為0，導致我WRITE\_DATA的值會是寫入0，所以我在MUX2TO1\_SELECT到WRITE\_DATA的中間線路，我又多加了一個REGISTER，讓電路能在ENABLE\_EXECUTE變為0的時候，保持著正確的輸出，這樣我的WRITE\_DATA就是寫入正確的值了。

**b. A block diagram for your completed SMILE CPU indicating all necessary components and I/O pins. Note please use MS Visio that is available in computer center in the university.**

**ANS：**

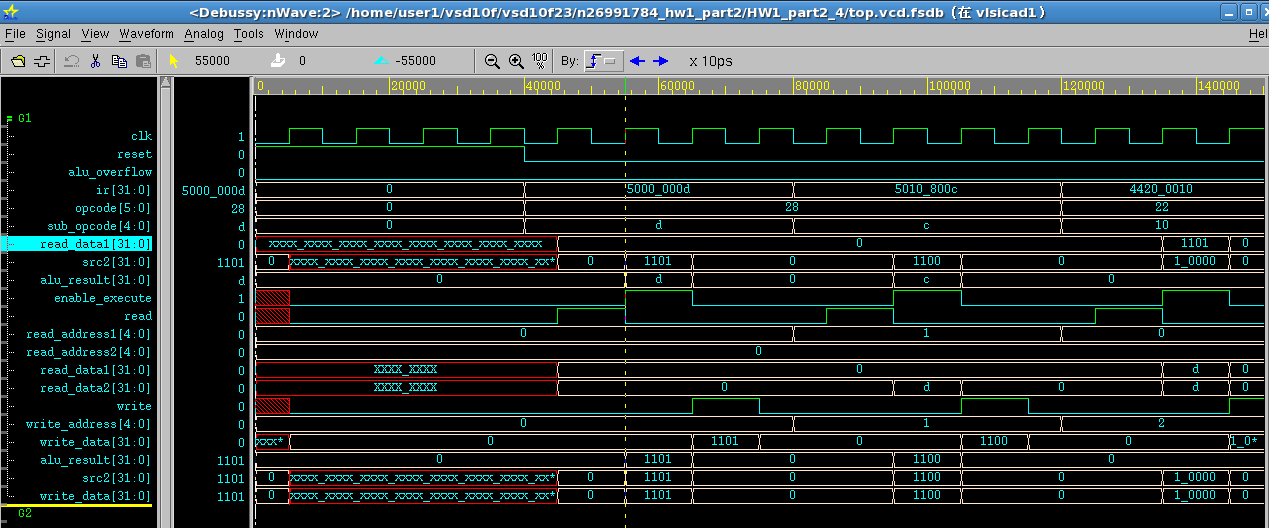


我的電路PORT命名的是與HOMEWORK上的圖示一樣的，這樣在寫CODE的時候比較不會搞混，除了我的READ\_DATA1是跟之前設計ALU時的命名一樣都為SRC1，跟因為我在MUX2TO1\_SELECT的出輸加了BUFFER，所以MUX2TO1\_SELECT的輸出就為MUX2TO1\_RESULT，而BUFFER的輸出就為WRITE\_DATA，所以圖變為下面這張圖片：

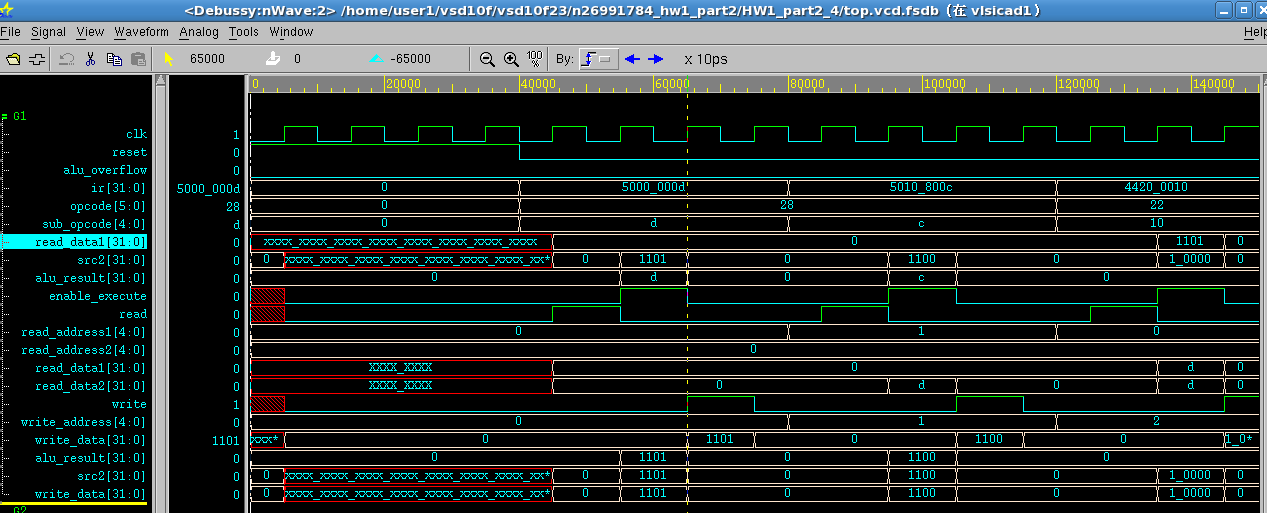


**C. Simulated waveforms with proper explanation**

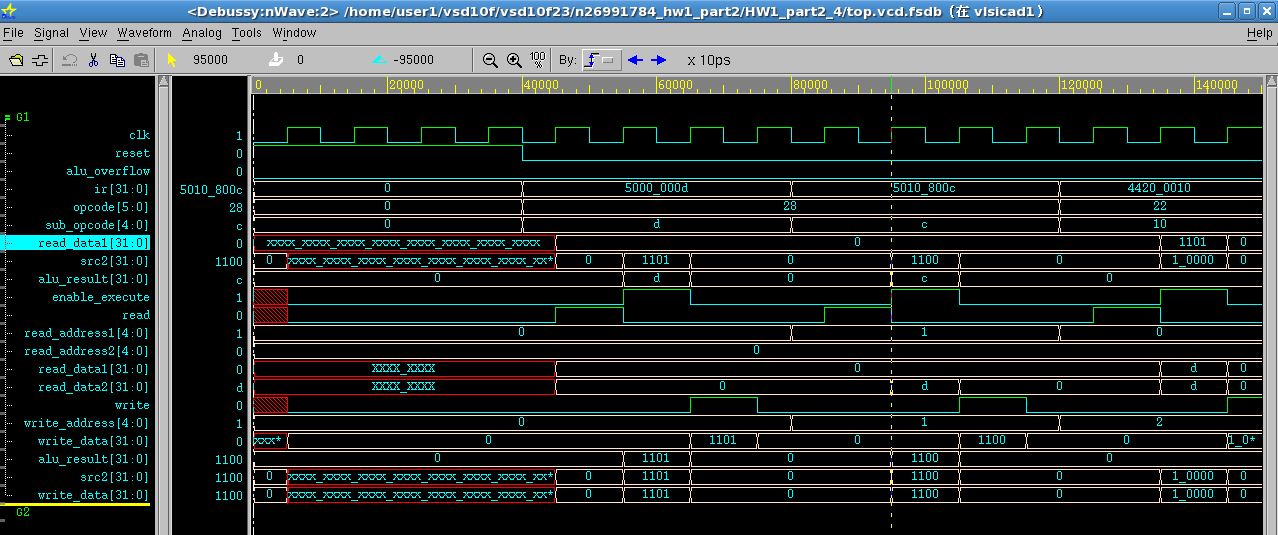
**首先是做R0=R0+5’b01101**

****

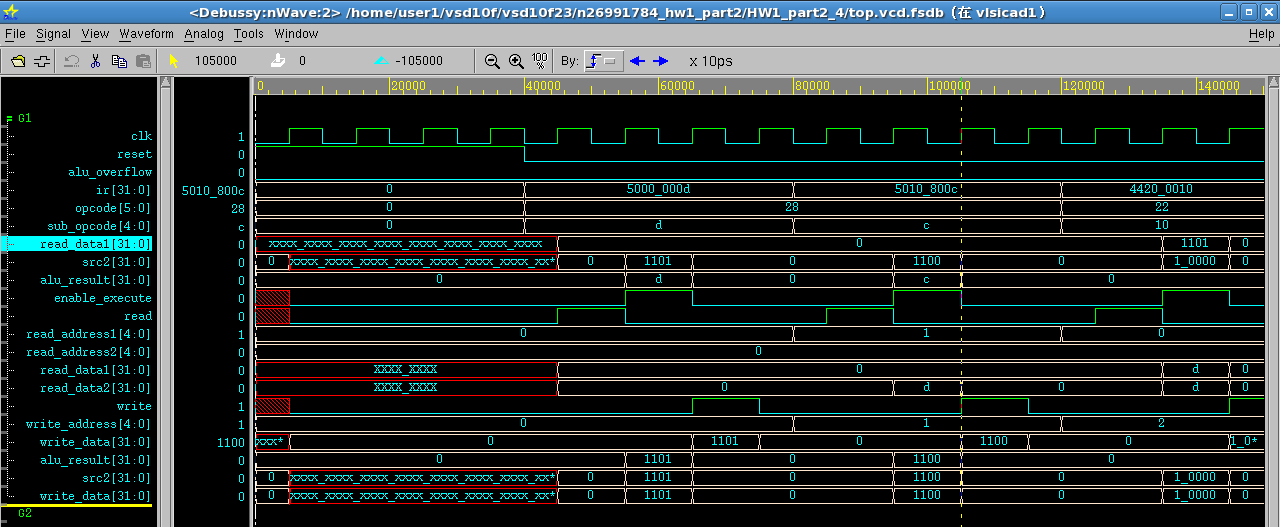
**結果寫入R0**

****

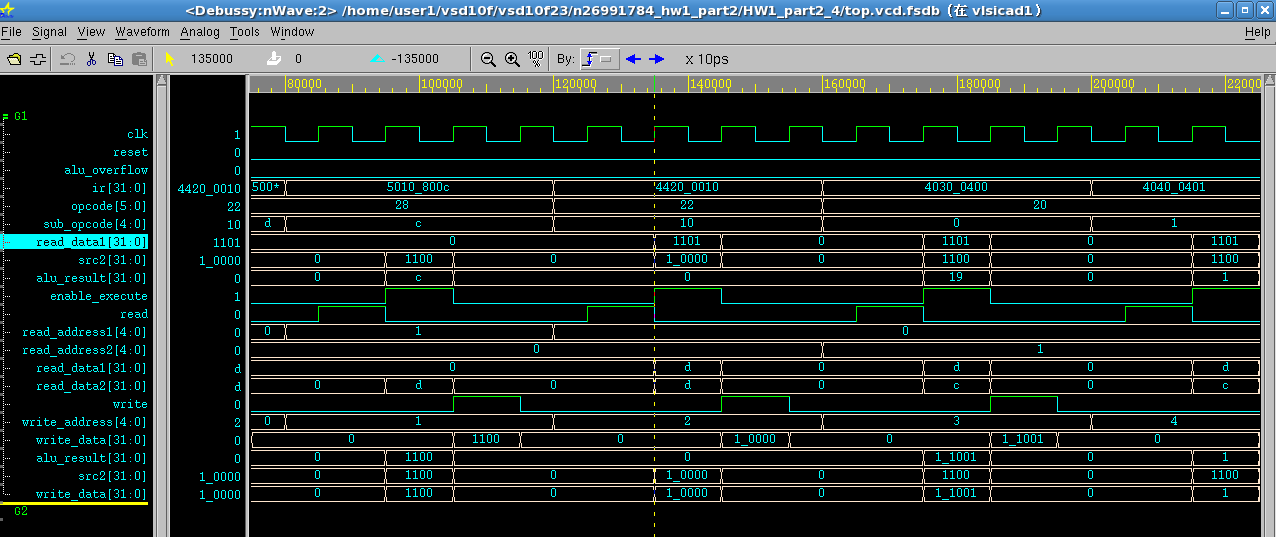
**R1=R1+5’b01100**

****

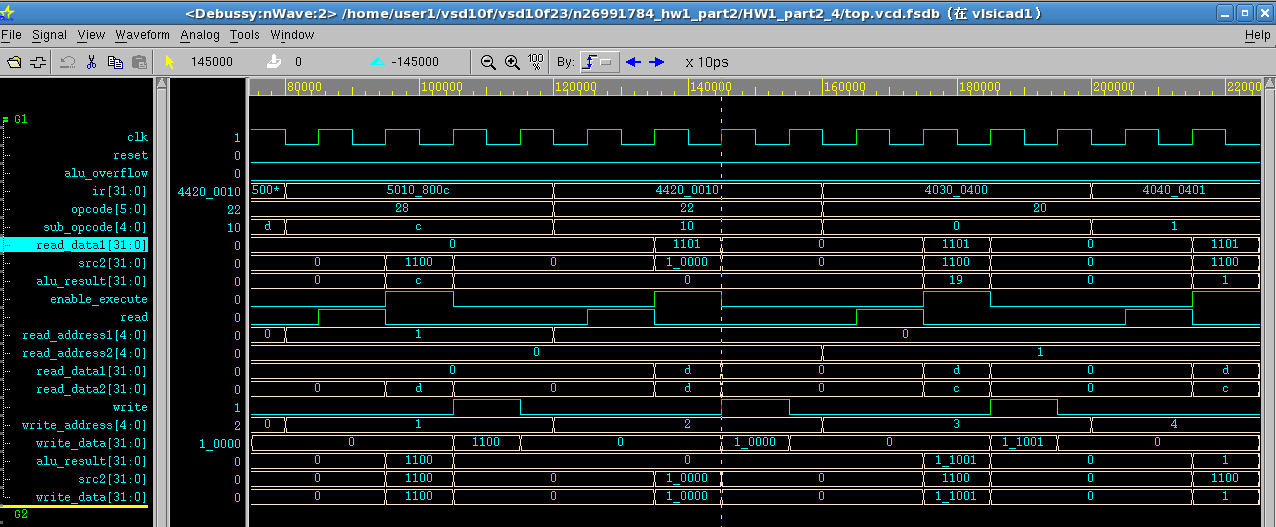
**結果寫入R1**

****

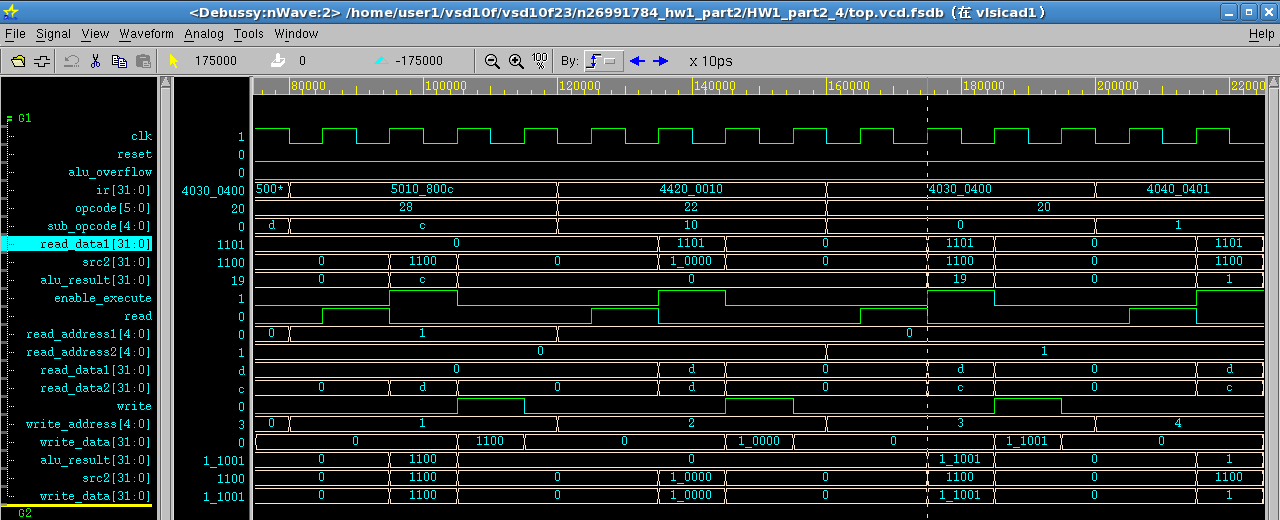
**R2=5’b010000**

****

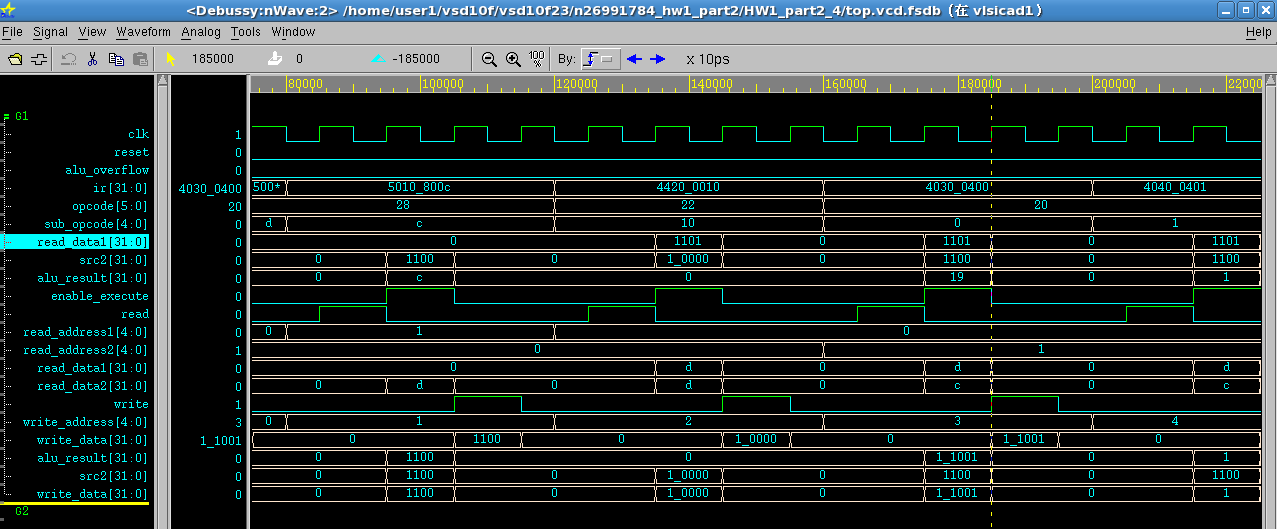
**結果寫入R2**

****

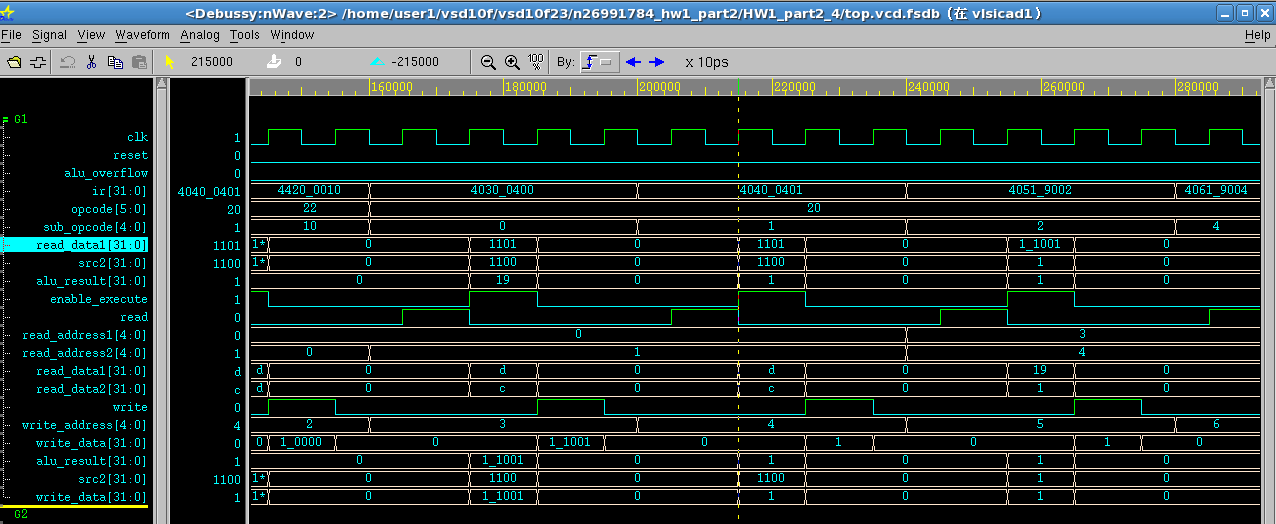
**R3=R0+R1**

****

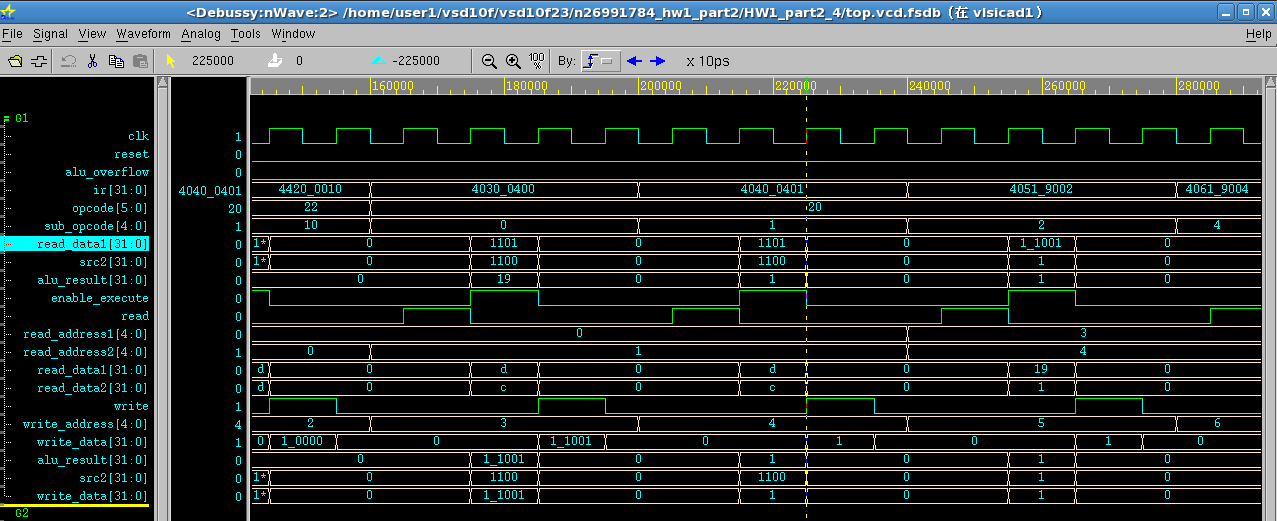
**結果寫入R3**

****

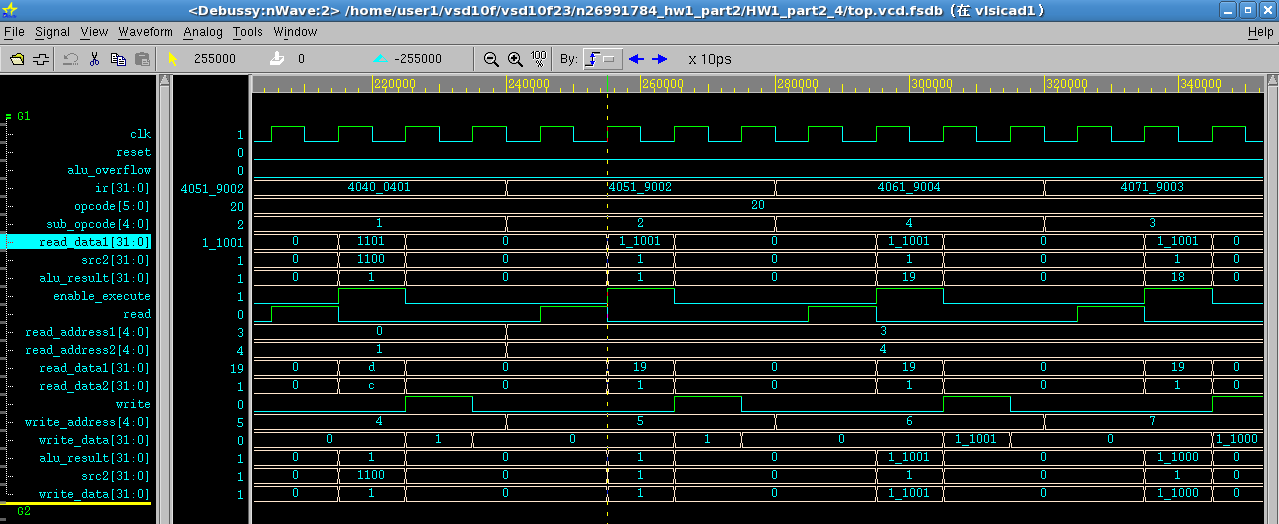
**R4=R0-R1**

****

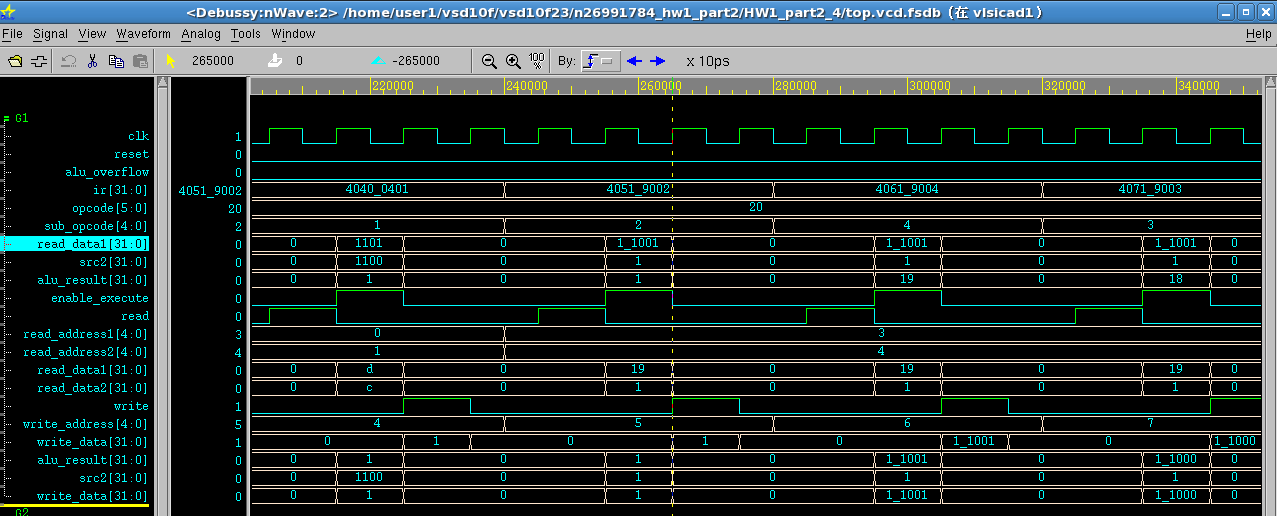
**結果寫入R4**

****

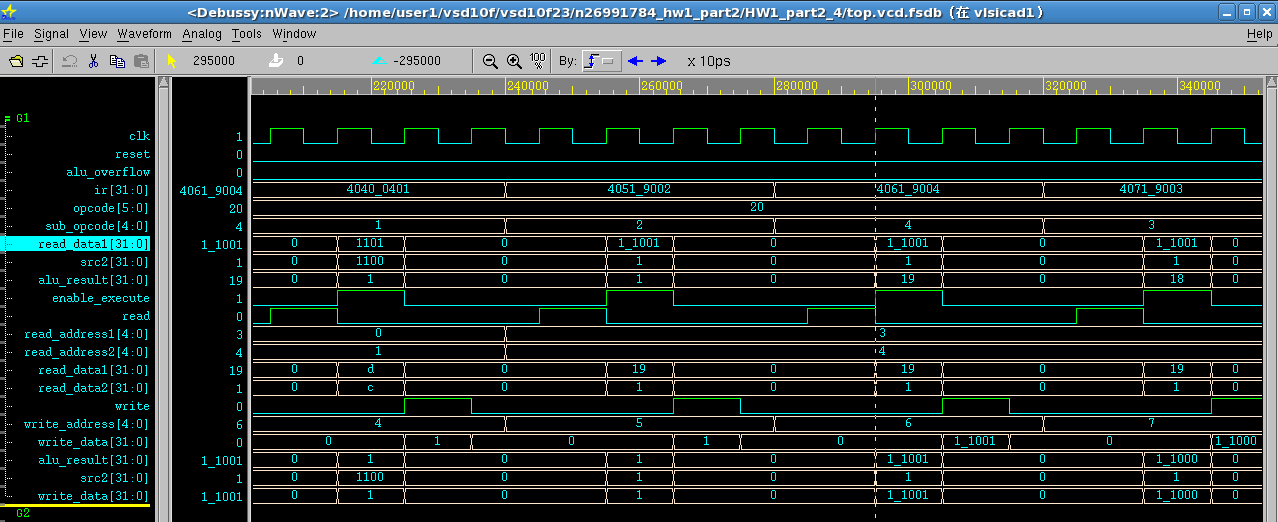
**R5=R3&R4**

****

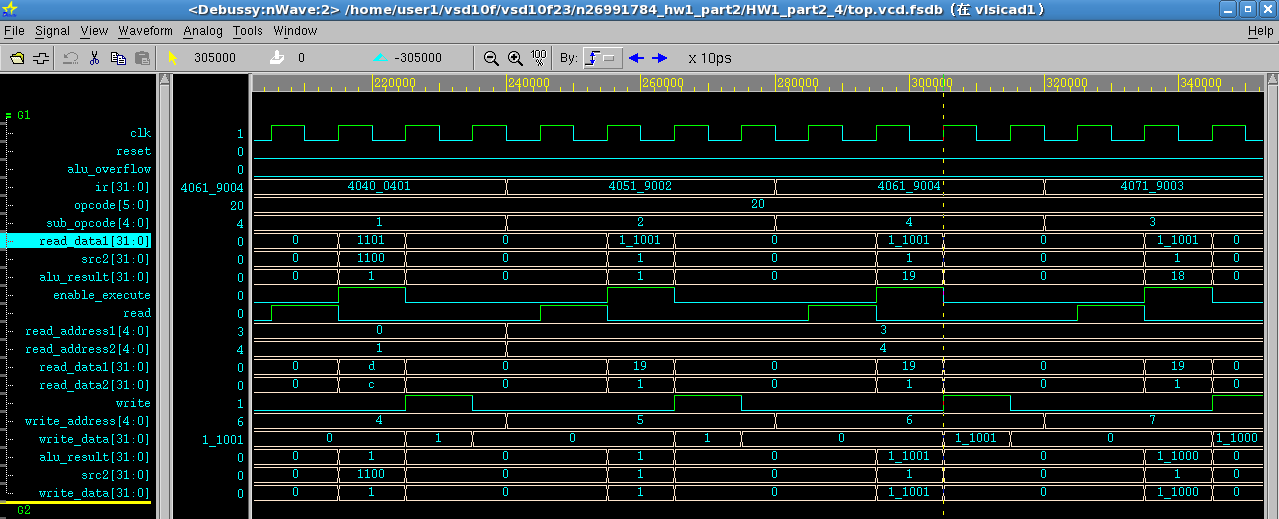
**結果寫入R5**

****

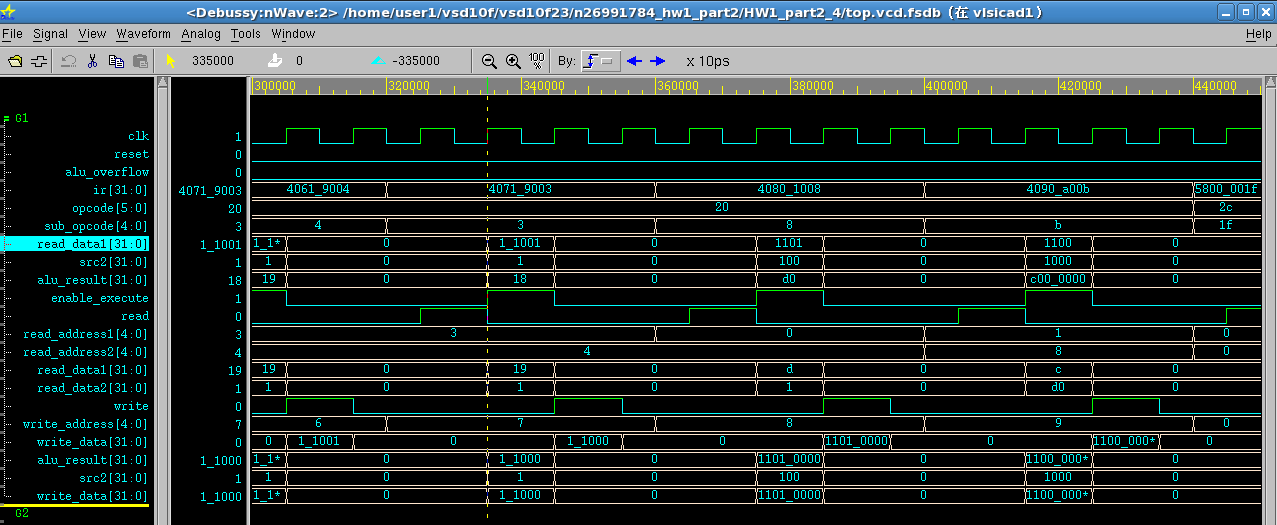
**R6=R3|R4**

****

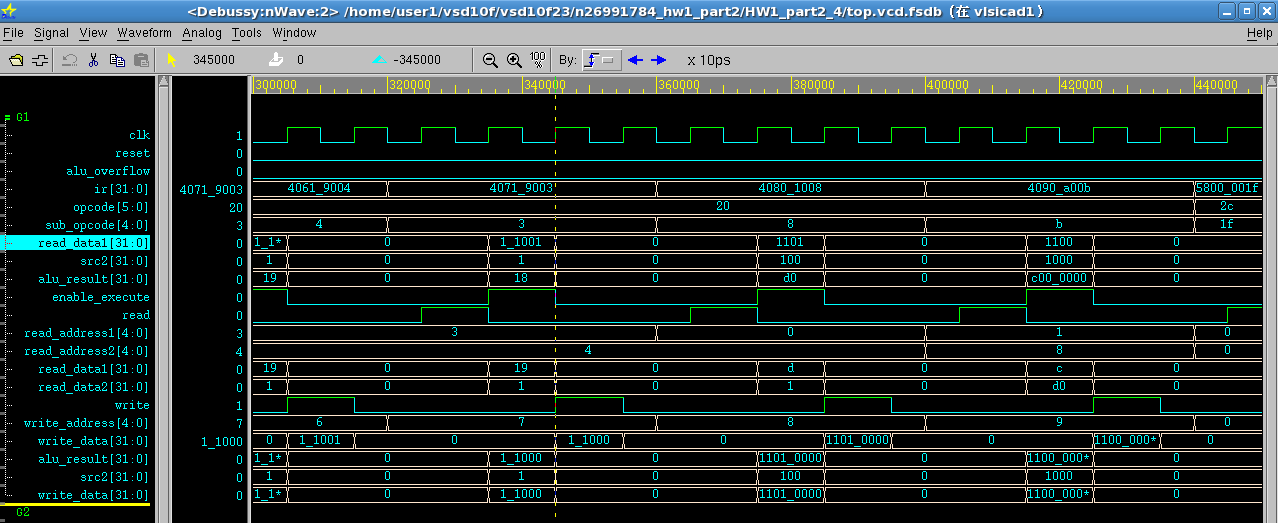
**結果寫入R6**

****

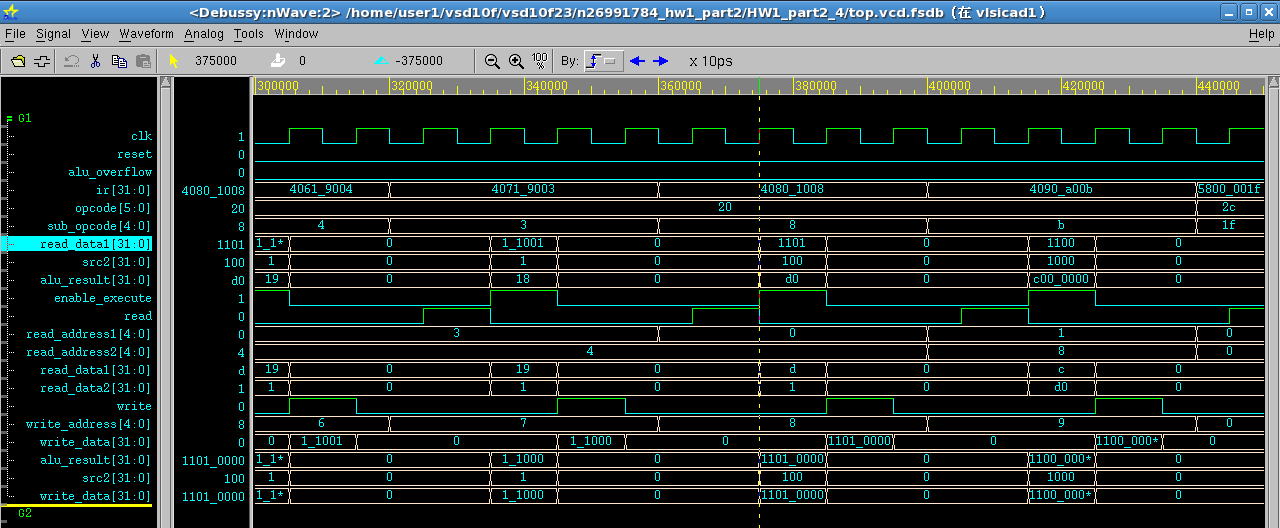
**R7=R3^R4**

****

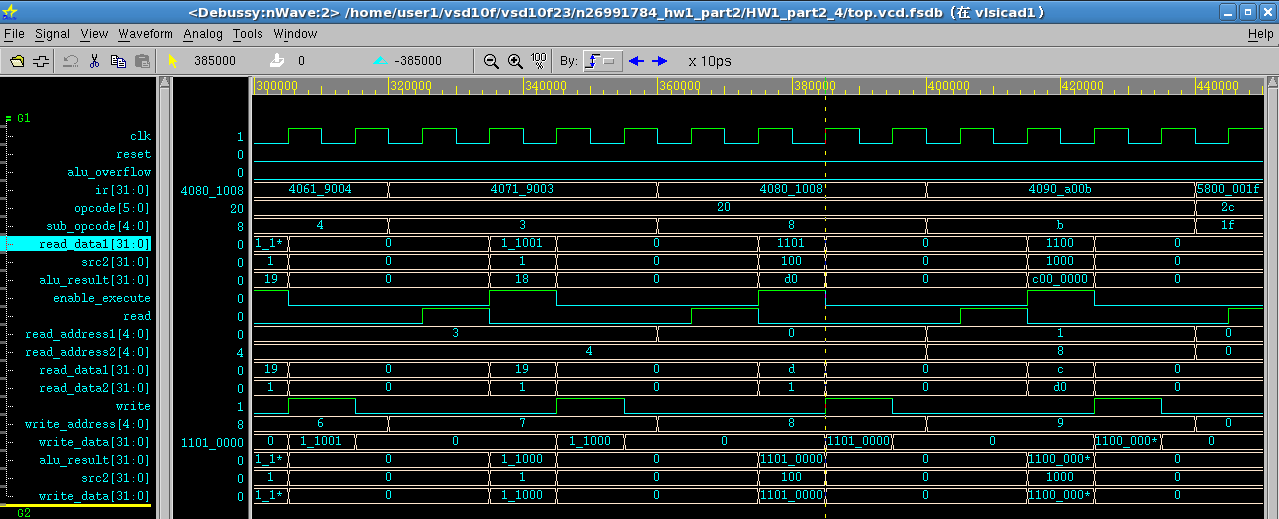
**結果寫入R7**

****

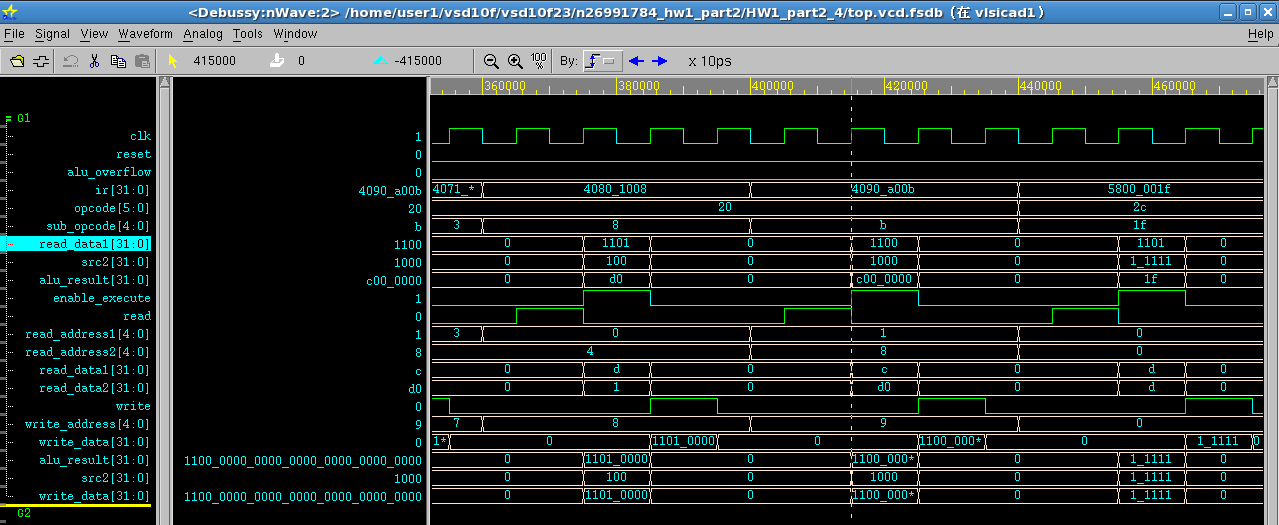
**R8=R0<<5’b00100**

****

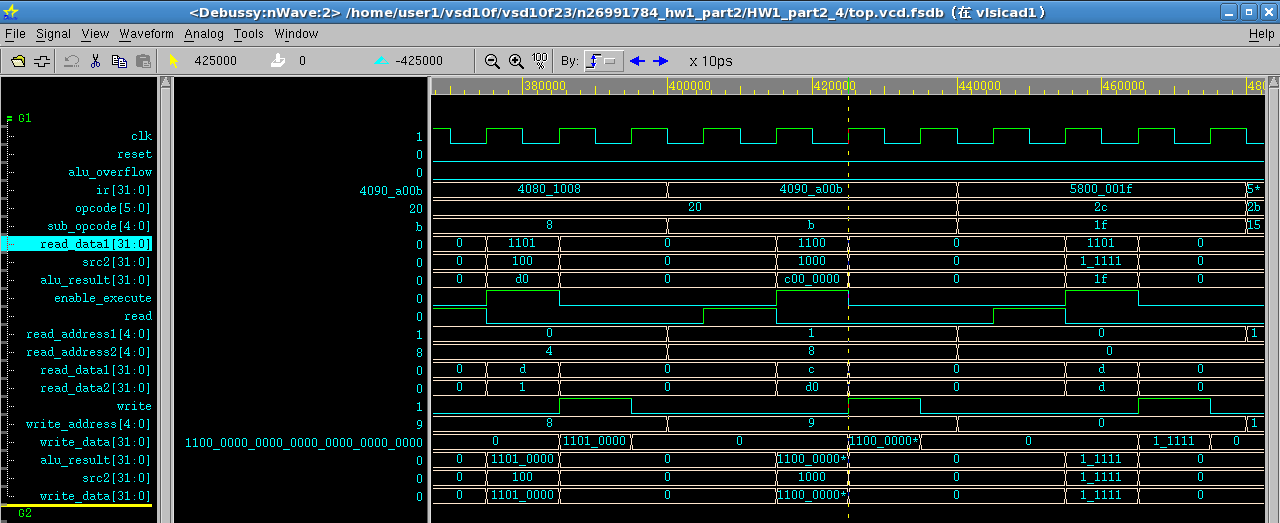
**結果寫入R8**

****

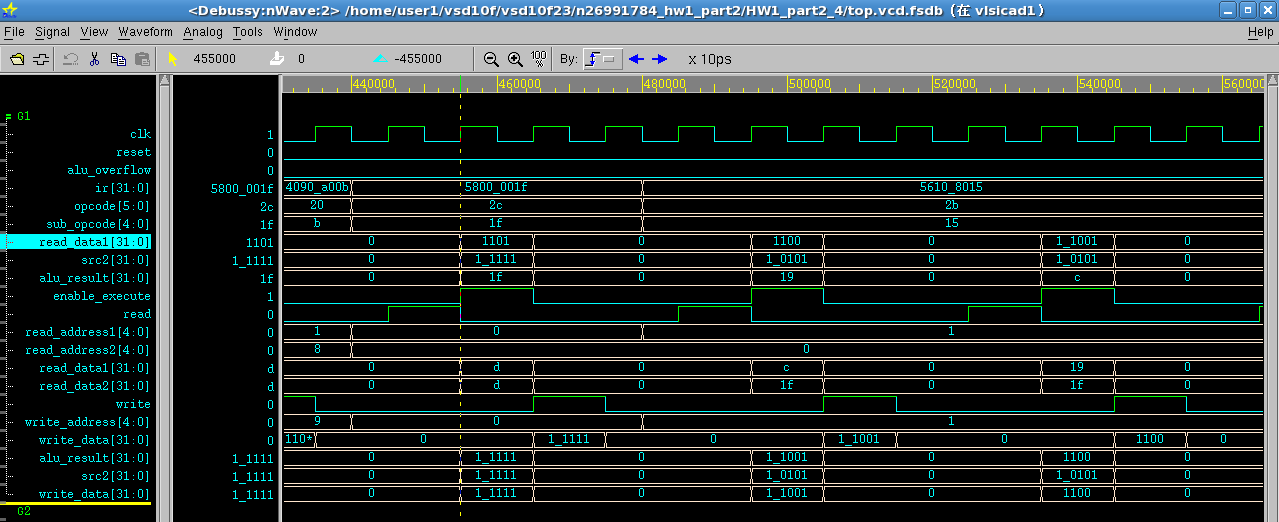
**R9=R1>>5’b01000**

****

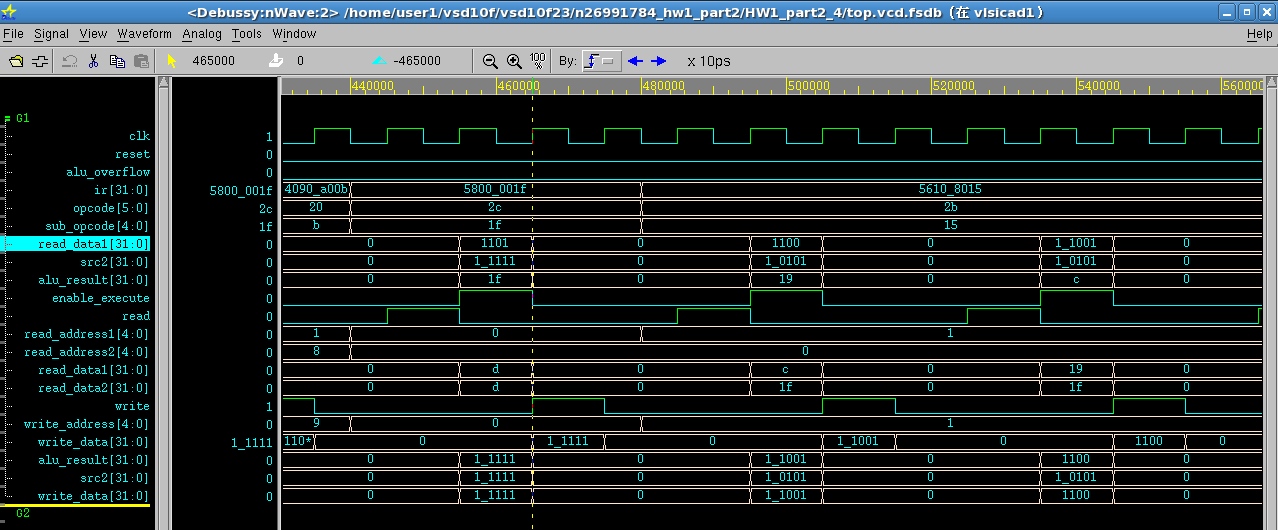
**結果寫入R9**

****

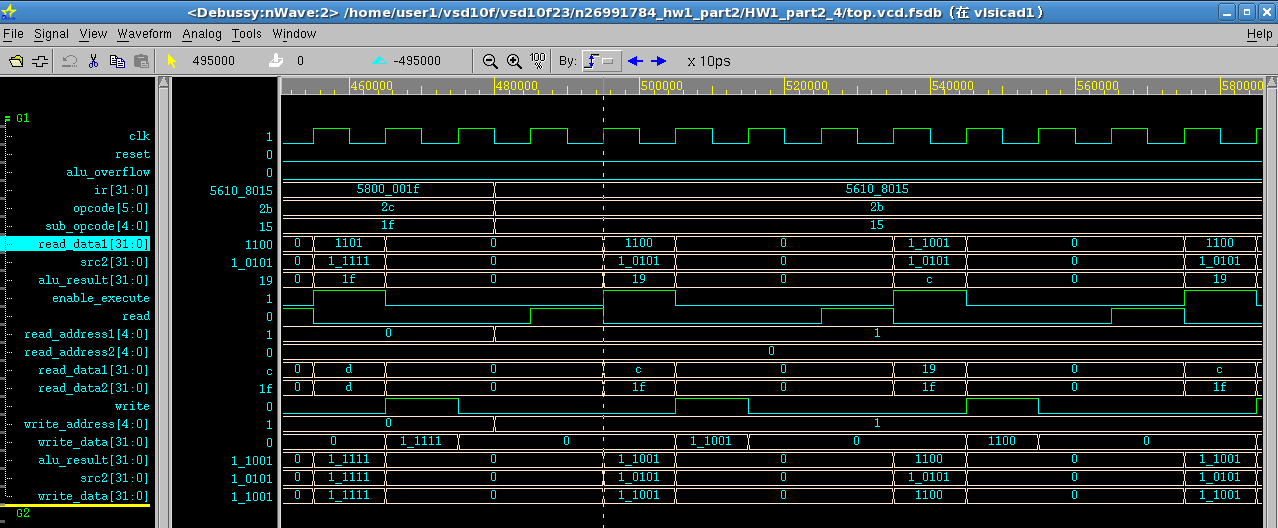
**R0=R0|5’b11111**

****

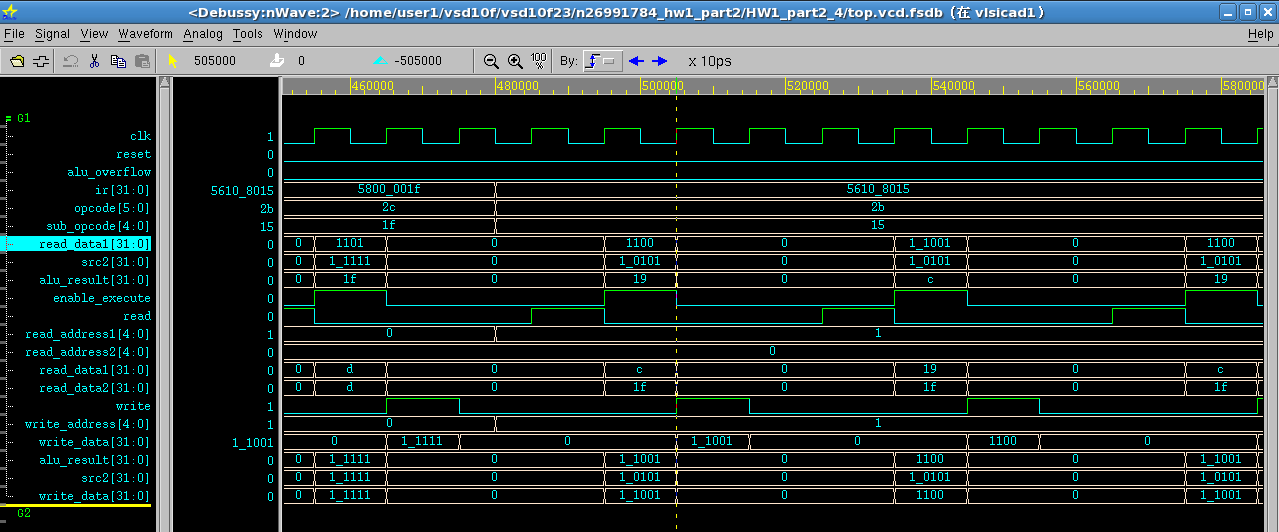
**結果寫入R0**

****

**R1=R1^5’b10101**

****

**結果寫入R1**

****

**全部圖形**

****

**因為我的WRITE\_DATA一開始測的時候有錯誤，加了BUFFER就改進了，所以我在圖片的第2張都附上寫回的圖。**

**D.Learned lesson and Conclusion：**

這次最大的困難點應該就是把CONTROL跟電路合再一起了，前前後後遇到的問題有TIMMING不對，或是接腳接錯地方還是忘了接都發生過，不過也都一一克服了，這次作業也讓我更加了解電路整個再對指令做處裡的詳細流程，也讓我知道電路有那些地方是該注意的有那些地方是可以有很多種寫法，這次作業的做法很多種，我的不一定是最好的，希望可以藉由VSD課程上的學習，讓我不只學習到怎樣解決問題，還能找出最好的方法來解決問題。