SoC Lab4-1

組別: 第十六組 成員:周聖平、蔡以心、張煒侖

1. Explanation of your firmware code

* How does it execute a multiplication in assembly code

FIR是輸入訊號跟FIR係數進行摺積，公式可以表示為

y[n] = [n-k] \* bk。

y[0] = b0 \* x[0]

y[1] = b0 \* x[1] + b1 \* x[0]

…

可以用兩層的for迴圈實現，外層的for迴圈是計算第i個輸出y[i]，內層的for迴圈是將各項bj \*x[i-j]累加起來。

for(i=0; i<N; i++){

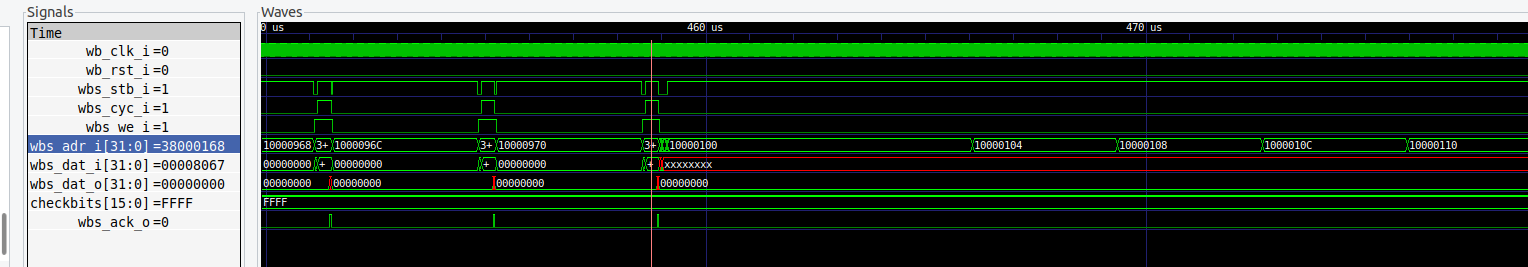
for(j=0; j<=i; j++){

outputsignal[i] = outputsignal[i] + inputsignal[i-j] \* taps[j];

}

}

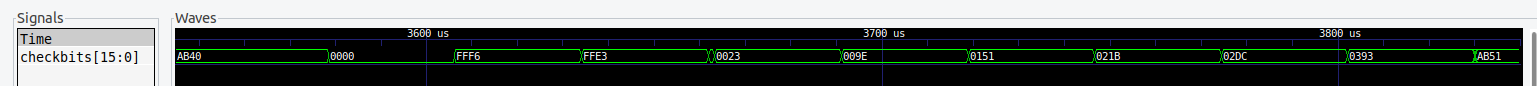
* What address allocate for user project and how many space is required to allocate to firmware code



從波形中可以看到wbs\_adr\_i最多到0x38000168，其中0x168 = 360 bytes = 90 words，這就代表在BRAM中，最少需要2^7(128)個word來去儲存firmware code，因此BRAM的N 需要 >= 7。

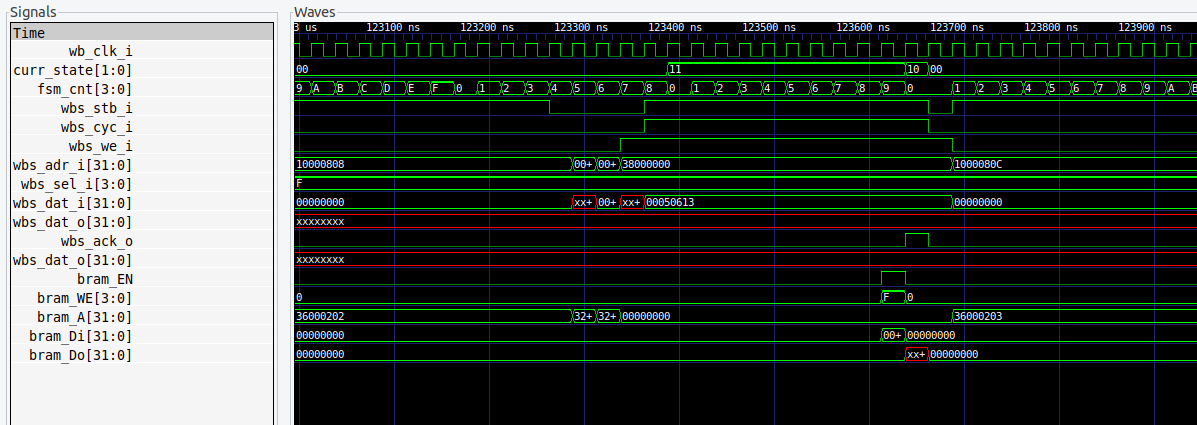
二、Interface between BRAM and wishbone

* Waveform from xsim

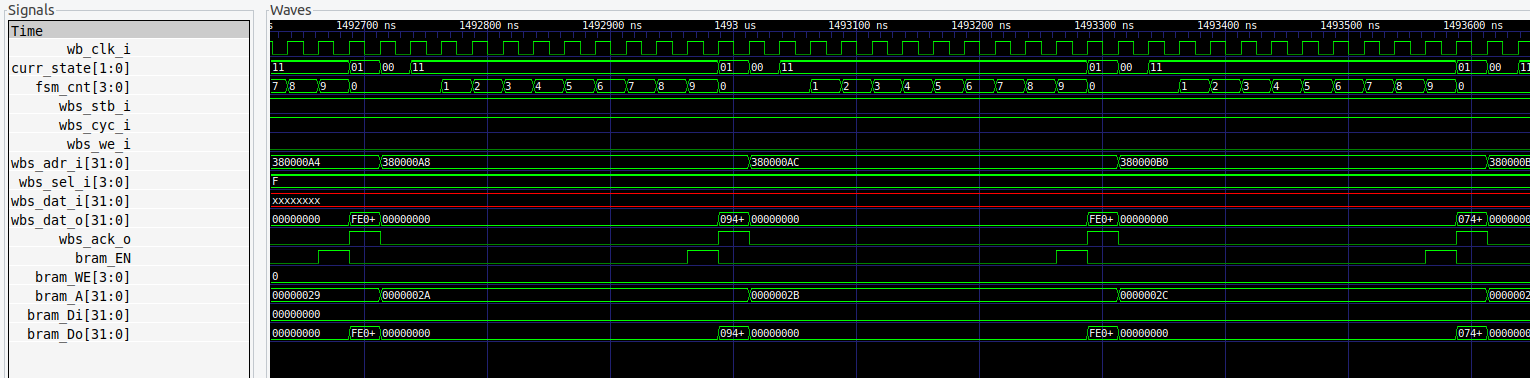


* FSM

WRITE



READ



FSM共有四個State，RECV(00)、RD(01)、WR(10)、DELAY(11)。

RECV : 當RECV接收到wbs\_stb\_i && wbs\_cyc\_i 時，就會進入到DELAY。

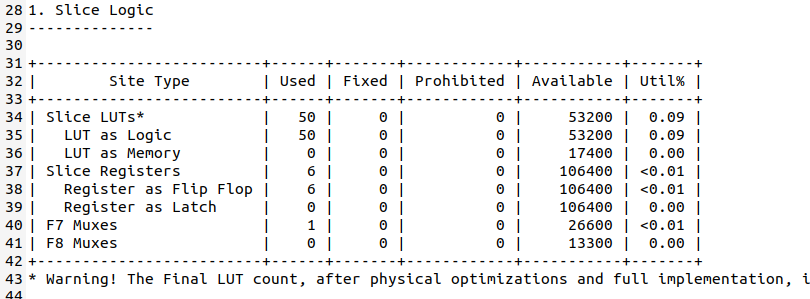
DELAY: DELAY狀態是為了要去滿足Parameter Delay，當DELAY到最後一個clock時，就會去Access BRAM，也就是去對BRAM做讀寫的動作。而接下來，將會由wbs\_we\_i來決定是要跳去RD 還是 WR。

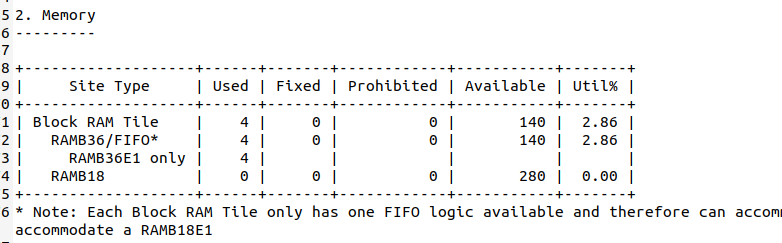
RD: RD狀態會將bram\_Do 輸出到 wbs\_dat\_o，並將wbs\_ack\_o拉為H，接下來，就會跳回RECV等待下個Request。

WR: 進到WR狀態時，會將wbs\_ack\_o拉為H，由於在DELAY狀態的最後一個clock就會去完成寫入BRAM的動作，因此接下來，只要跳回RECV等待下個Request就好。

1. Synthesis report

LUT: 50 Register: 6



BRAM:4

1. Other discoveries

五、Github Link

<https://github.com/PatriChou/lab-exmem-fir.git>