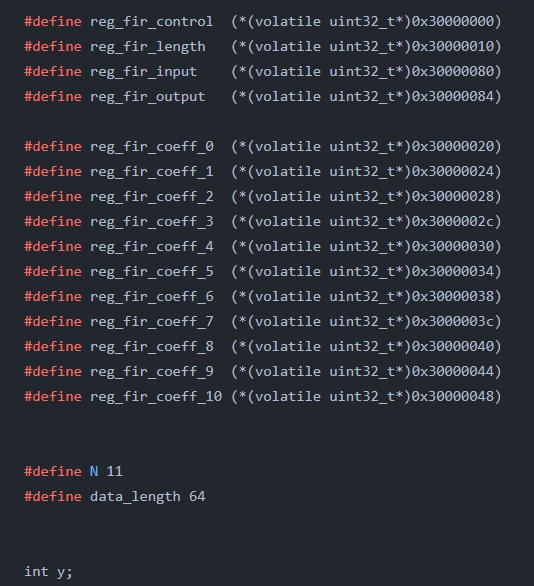
我們將wb-axi interface 與 lab 4-1的exmem一同寫入user\_proj\_counter.v中

，並使用了bram11作為tap\_bram及data\_bram。

1. firmware

我們在fir.h中定義了

ap\_control、data\_length、x\_input、y\_output、及11個coeff各自對應的address，coeff 為了對應lab3的address因此採用0X20-0X48而非講義的0X40-7F。

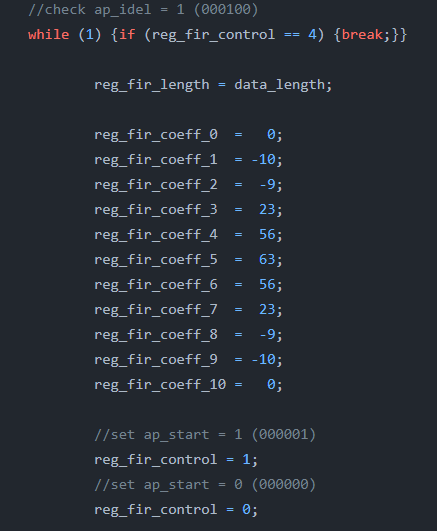


在fir.c中

Firmware會先check ap\_idle = 1 後才開始動作。

當ap\_idle = 1 之後開始向slave透過axi\_lite送入datalength及coeff的data

完成後送出ap\_start訊號

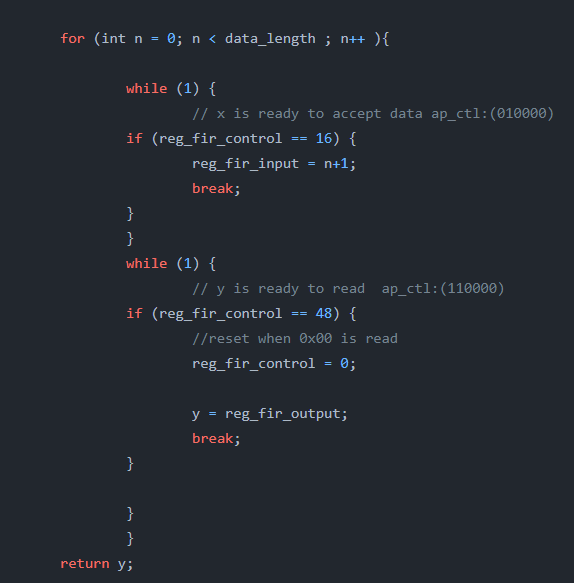


透過for迴圈送出x[n]的data

並設置迴圈使ap\_control[4] = 1時(slave is ready to accept input)才送出x[n]

同理使ap\_control[5] = 1時(y[n] is ready to read)master才接收y[n]

最終fir回傳y的最後一筆data



在counter\_la.c中

Output hA5至mprj[23:16] to start latency-timer

再output fir return 的 data 其 [7:0] 的部分 (00101101)至mprj[31:24]

【第64筆計算結果為d10797 其 [7:0] 的部分為00101101】

最後output h5A至mprj[23:16] to end latency-timer

總共repeat 3次

結束後output hAB51表示finish



1. decode

exem\_fir:3800xxxx

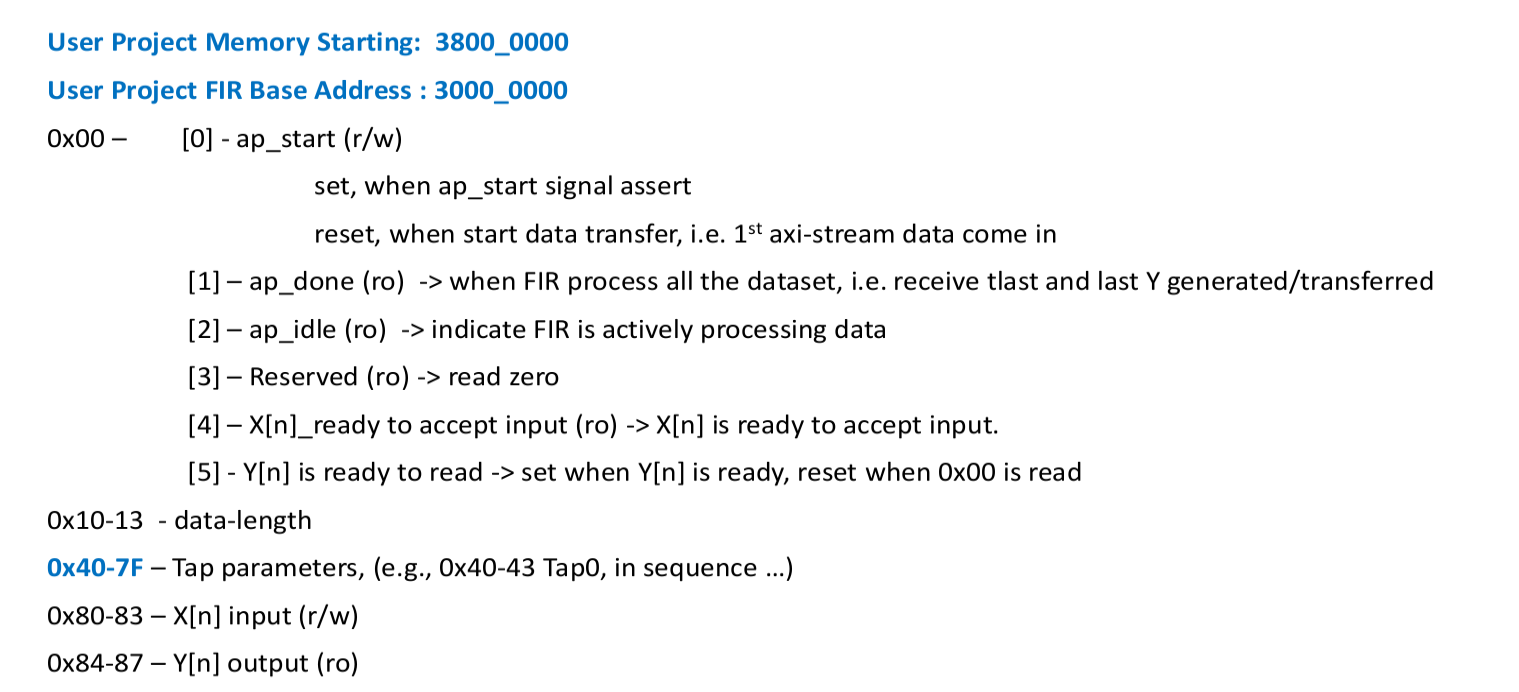


Axi\_lite:



Axi\_stream:





Axi stream

Axi lite

20-48

20-48

Axi lite(b): 0110 0000 xx….xx 0000 0000 ~ 0110 0000 xx….xx 0111 1111

Axi stream(b): 0110 0000 xx….xx 1000 0000 ~ 0110 0000 xx….xx 1000 0100

axi\_l\_decoded = {wbs\_adr\_i[31:24],wbs\_adr\_i[7] } == 9'b1100000 ? 1'b1 : 1'b0

axi\_s\_decoded = {wbs\_adr\_i[31:24],wbs\_adr\_i[7] }== 9'b1100001 ? 1'b1 : 1'b0;

1. exmem-fir : 同lab 4-1
2. wb-axi lite:

wbs\_cyc\_i && wbs\_stb\_i == 1 可視為valid

而wbs\_sel\_i && wbs\_we\_i == 1 為 write enable

當axi\_l\_decoded true 時為axi lite

故**wvalid = wbs\_cyc\_i && wbs\_stb\_i && axi\_l\_decoded && wbs\_sel\_i && wbs\_we\_i;**

wbs\_sel\_i && !wbs\_we\_i == 1 為 read only

故**arvalid = wbs\_cyc\_i && wbs\_stb\_i && axi\_l\_decoded && wbs\_sel\_i && !wbs\_we\_i;**

Address map 與lab3相同，因此只需直接保留wbs\_adr\_i[11:0]即可作為axi\_lite的address

**awaddr = wbs\_adr\_i[11:0];**

**araddr = wbs\_adr\_i[11:0];**

1. wb-axi stream:

Synthesis report

