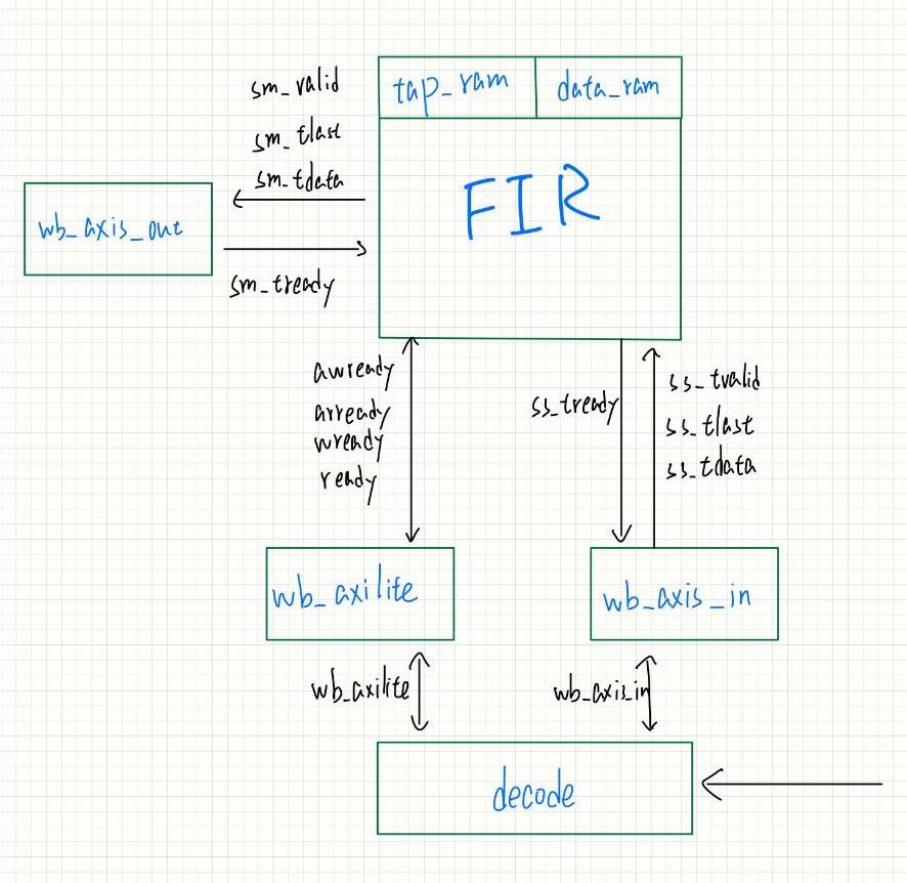
Lab4-2

第九組 紀皓洋 吳承哲 洪啟恩

1. **Design block diagram – datapath, control-path**



2. **The interface protocol between firmware, user project and testbench**

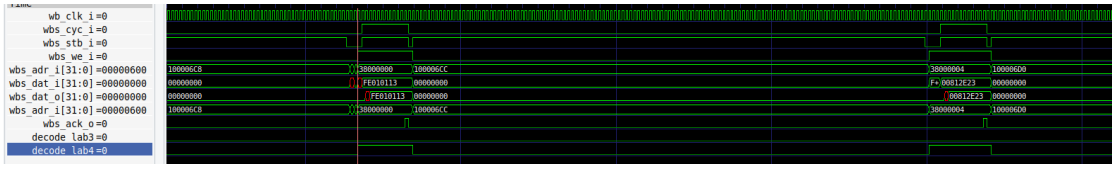
   The firmware code undergoes transmission into the SoC via the housekeeping module, ultimately residing in the user BRAM through the Wishbone interface. Communication between the user project and the external testbench block relies on the mprj_io[15:0] interface. Initialization involves sending the data length to address 0x30000010 and tapping values into addresses 0x300000 to 0x3000007F. Post-transmission, the Wishbone interface facilitates reading values from addresses 0x30000040 to 0x3000007F, with results conveyed back to the testbench through mprj_io[31:15] for validation. A signaling of 0x00A50000 is transmitted to mprj_io[31:15], prompting the testbench to initiate the latency

timer. Verification of system idleness occurs by reading 0x30000000, triggering the transmission of ap_start to 0x30000000 via the Wishbone interface. Following this, X values are sent, and continuous data retrieval is confirmed by reading 0x30000088. Successful retrieval prompts data transmission to 0x30000080, and the status is checked at 0x30000090. Upon successful computation, results are read from 0x30000084. This iterative process repeats thrice, and upon completion, the ap_done status from 0x30000000 is scrutinized. If the operation is complete, 0x005A0000 is conveyed to mprj_io[31:15], signaling the testbench to conclude the latency timer.

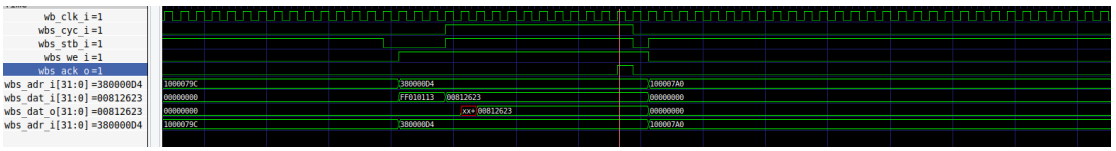
3. **Waveform and analysis of the hardware/software behavior**

```
ubuntu@ubuntu2004:~/Desktop/SoC_Design_Lab4_2/testbench/counter_la_fir$ source run_sim
../../rtl/user/user_proj_example.counter.v:143: warning: Port 4 (wb_we_i) of wb_axistream expects 1 bits, got 4.
../../rtl/user/user_proj_example.counter.v:143:        : Pruning 3 high bits of the expression.
../../rtl/user/user_proj_example.counter.v:169: warning: input port ss_tready is coerced to inout.
../../rtl/user/user_proj_example.counter.v:231: warning: Port 6 (A0) of bram expects 32 bits, got 12.
../../rtl/user/user_proj_example.counter.v:231:        : Padding 20 high bits of the port.
../../rtl/user/user_proj_example.counter.v:240: warning: Port 6 (A0) of bram expects 32 bits, got 12.
../../rtl/user/user_proj_example.counter.v:240:        : Padding 20 high bits of the port.
Reading counter_la_fir.hex
counter_la_fir.hex loaded into memory
Memory 5 bytes = 0x6f 0x00 0x00 0x0b 0x13
VCD info: dumpfile counter_la_fir.vcd opened for output.
LA Test 1 started
LA Test 2 passed
ubuntu@ubuntu2004:~/Desktop/SoC_Design_Lab4_2/testbench/counter_la_fir$
```

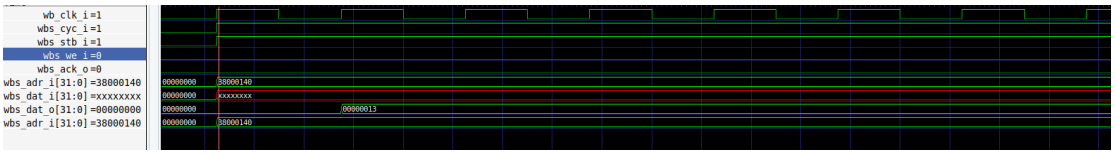
DECODE:



WRITE:



READ:



4. **What is the FIR engine theoretical throughput, i.e. data rate? Actually measured throughput?**
5. **What is latency for firmware to feed data?**

It involves the time taken for the firmware code to transmit and store data in the user BRAM, as well as the subsequent processes such as initiating the latency timer, sending X values, and performing computations. The exact latency can be determined by analyzing the

specific instructions, data transmission rates, and processing times within the described firmware workflow.

6. **What techniques used to improve the throughput?**
   - **6.1 Does bram12 give better performance, in what way?**
     i. We didn’t use bram12