EE417 Advanced Digital Logic with Verilog HDL Date: 08/03/24

Group 3: Lamin Jammeh/Ron Kalin

Lesson 12 **Final Project**

Finite Impulse Response Multiply-Accumulate FIR_MAC Project

Objective

In Digital Signal Processing (DSP), an input signal can be filtered using a Finite Impulse Response with Multiply-Accumulator (FIR_MAC) filter. The filter coefficient can be calculated using MATLAB using the fir1 function giving the design specifications (cut-off frequency, Max-frequency, filter order, and filter Type) are already determined. The filter function is used in MATLAB with a sample signal s to determine if the given coefficient can filter the sample signal s. The coefficients and the filter order are then used to implement the design of the FIR_MAC in DSP.

The Finite Impulse Response with Multiply-Accumulator (FIR_MAC) filter in this design will take in an input signal and multiply the signal with coefficient. The output of the multiply is temporarily stored in a pipeline register (PR) to reduce hardware idle time, thereby improving the latency of the clock signal. A similar PR is implemented at the input of the adder which performs the Accumulator section. The output of the Multiplier and the input of the adder and summed to delivered to input of the next adder on the next filter order. The number of adders and multiplexers is determined by the filter order of the design. To maintain the signal coherency a PR is added at the output of each Multiplier and input of each Accumulator [2].

The filter in this design has two submodules and a main or top module. The two submodules are the controller and the Datapath modules. The controller module enables the clock and reset signals. The Datapath looks for the posedge of the clock signal and the state of the reset signal. When reset is high; all registers and output of the FIR_MAC goes to zero, and at reset low the Datapath module start processing input signals on the next posedge of the clock. To make sure the time requirement for the hardware is met the clock period was change from 1.0ns to 4.0ns, this changes the slack time from negative to positive tog give the hardware some leeway in case of propagation delay[3].

Application

Processing analog signals can come with lots of challenges since the analog signal are prone to noise which can be very difficult and expensive to clean. These can lead to wrong data especially from biomedical devices. The application of FIR_MAC filters in digital signal processing suppresses the Powerline Interference (PLI)[1]. When FIR_MAC filters are implemented in FPGA devices, the filter can be used for multiple DSP applications without the need for hardware change. The filter coefficients and filter order can be changed at the programming level. This is why FIR_MAC filters are preferred in instrumentation and measurement devices.

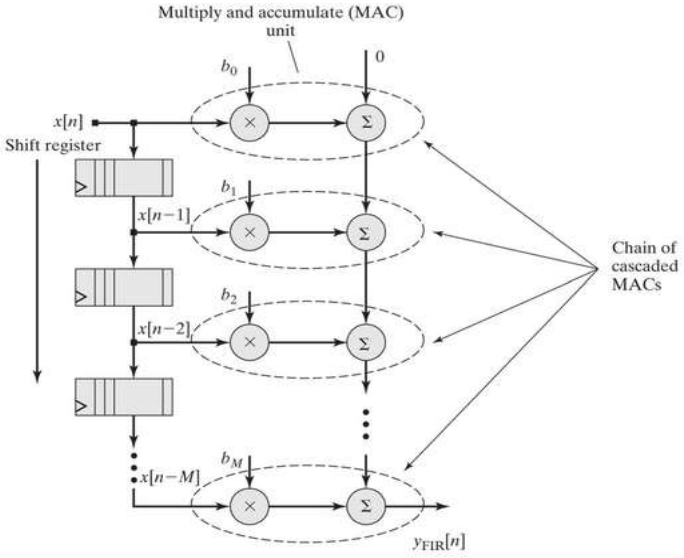

Figure 12.1 - FIR MAC diagram

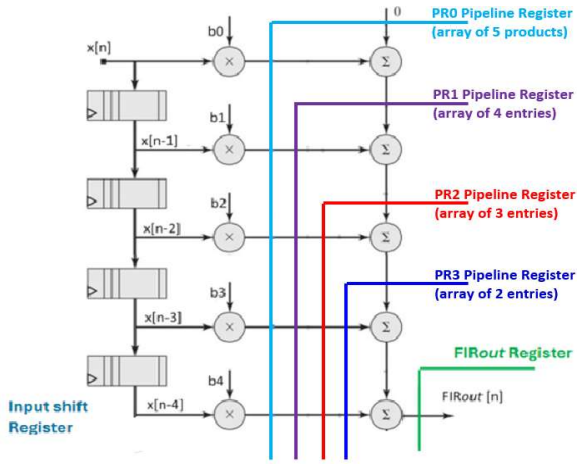

Figure 12.2 -
FIR MAC diagram, 4th order, showing pipelining cutlines