Chapter 4

Digital Processing Elements: Programmable Coefficients and Clock-Phase Generation

The quality of the analog signal-processing blocks, described in the previous chapter, determines the majority of the SI-FIR filter-performance limits, such as: the maximum input-bandwidth, maximum clock-rate, linearity, white-noise-level and dynamic-range. In contrast, the digital blocks, discussed in this chapter, will determine the ease of use and level of programmability of the filter. Ideally, the digital circuitry should not influence the analog performance. However, care must be taken to ensure that interference from the digital signals will not corrupt the analog waveforms. The largest source of cross-coupling is *clock feed-through*, caused by charge injection of a clock signal through parasitic capacitors to sensitive analog nodes. In this chapter, the digital circuits for clock-phase generation and programmable-coefficient control will be discussed and the effort needed to minimize the cross-coupling of digital switching noise into the output will be considered.

4.1 Digitally-Programmable Coefficients

The primary signal-processing element of the SI-FIR filter is the current-division ladder, which acts as a multiplier at every filter tap. Multiplication is performed by steering binary-weighted currents based on the value of a digital coefficient. A simple differential register can store the coefficient and its inverse, which drives the current-division ladder directly. Using digital coefficients allows significant flexibility in this architecture. Aside from making the filter fully-programmable, the digital coefficients can be easily stored, rotated and shifted; this allows the analog sampled signal to be kept static, eliminating errors due to resampling, while the digital coefficients circulate as shown in Fig. 4.1.

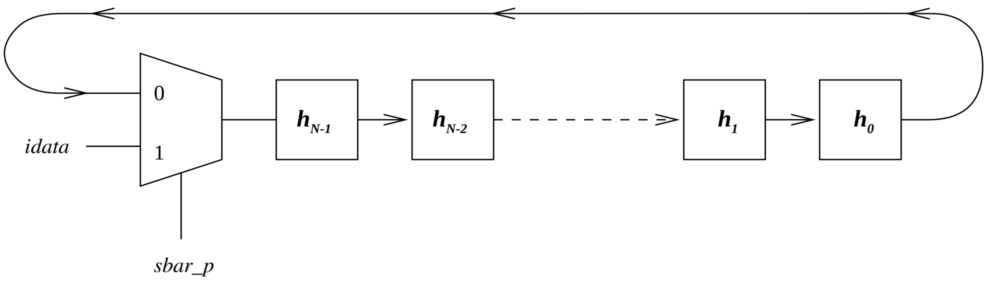


Figure 4.1 Coefficients circulating to emulate the propagation of samples

Filter-Mode vs. Program-Mode: A multiplexer is placed in the front-end of the coefficient register to select either the *filter-mode* or the *program-mode*. In the filter-mode, the coefficients circulate within the register bank while normal filtering is in progress. In the program-mode, a new set of coefficients is applied to the FIR filter. The total number of programming clock cycles is equal to the number of taps of the FIR filter. The mode-control signal is determined by the control-bit *sbar_p* as listed in Table 4.1.

$sbar\_p$	mode
0	filter
1	program

Table 4.1 Different modes of operation of the FIR filter

Gate-Level Implementation The corresponding gate-level implementation of Fig. 4.1 is depicted in Fig. 4.2. This circuit is a bit-slice of all N filter coefficient registers, showing one bit for each of the N taps. A filter of N taps, with coefficients consisting of 8-magnitude-bits and 1-sign-bit, needs a total of 9 identical bit-slices of length N. Therefore, the circuit of Fig. 4.2 is repeated 9 times to generate the programmable coefficient registers. It is worthwhile noting that the signals *idata* (input coefficient) and *odata* (output coefficient) could share the same bus. However they are retained as shown in the present configuration for ease of testing.

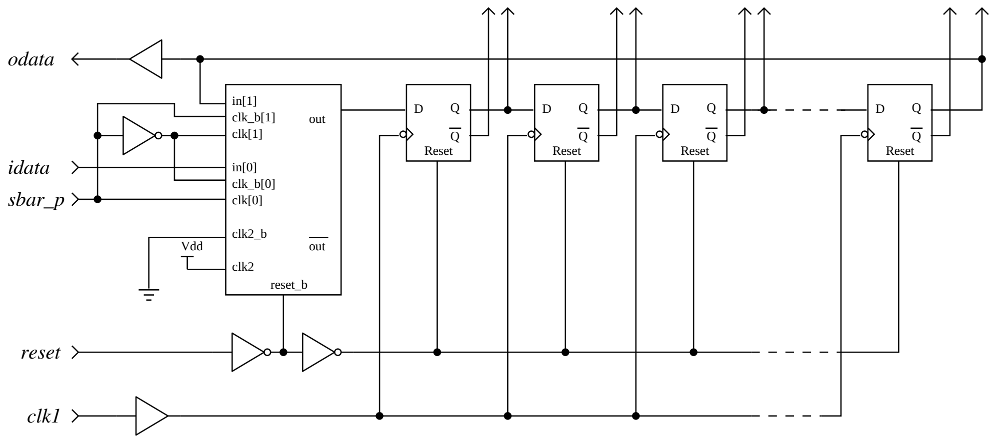


Figure 4.2 A bit-slice of the coefficients

Standard-Cell Approach: The digital circuits for this prototype utilized a standard-cell approach.[1] Layout considerations of these cells is the main focus of Chapter 5, and will be discussed later. Since the standard-cell approach is primarily designed to implement random-logic sections of VLSI circuits, it is not quite ideal for data-path applications such as this one. Therefore, a standard-cell realization resulted in a design that was not as optimized as a full-custom approach would have been. Yet, with the exception of a small increase in wiring space, the standard-cell layout does not have an adverse effect on the chip-area or filter performance.

Considering the advantages of an automated design procedure, and place-and-route capabilities that standard-cells offer, this approach was chosen, due to ease of design and reduced layout time. In addition, the standard-cell library includes certain desirable features that make the design more flexible, such as :

- Variable power and ground bus widths,
- Independent p- and n-channel transistor sizing,
- Optional second-level-metal inputs and/or outputs,
- Variable terminal spacing and/or grid terminal placement,
- Automatic pitch matching,
- Ability to create technology-portable designs.

4.2 Clock Generation

[1]The *GDT* library of standard cells from *Mentor Graphics* was used for all the digital circuits of this prototype chip.

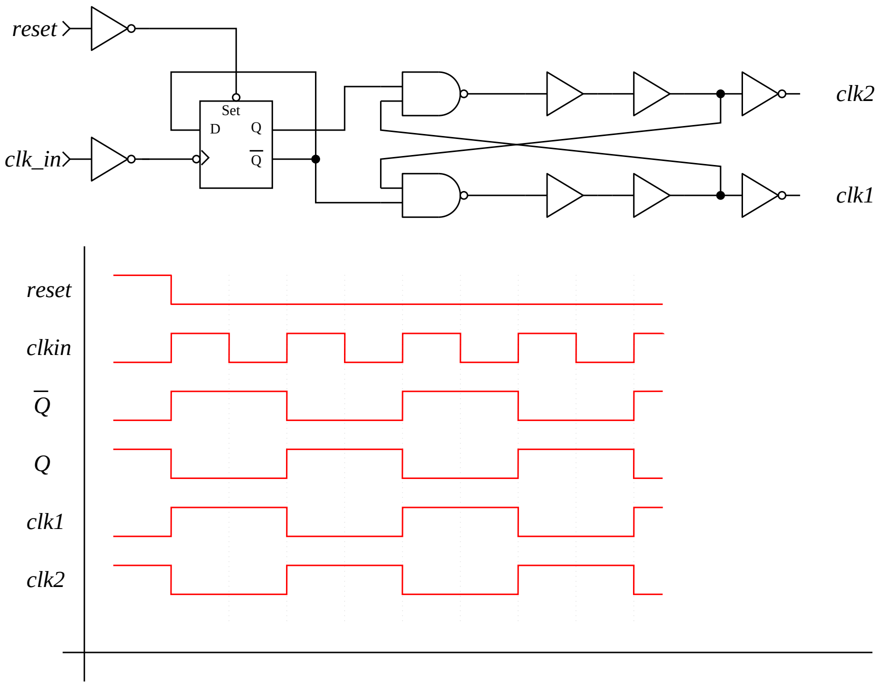


Figure 4.3 Non-overlapping-clock generator and the timing diagram

Non-overlapping clocks are necessary in this design to ensure that the switches in the sampling path are turned off completely before the hold phase, and vice versa. Overlapping clock phases will cause signal contamination. Fig. 4.3 shows the circuit of the non-overlapping clock generator and the corresponding timing diagram. The circuit divides an external clock by 2 to derive a 50% duty-cycle internal clock. Timing simulations of this circuit, and all other digital blocks, were performed using *LSIM* from *Mentor Graphics*. Final simulations using *HSPICE* were undertaken at the top-level for higher accuracy. Simulation results of Fig. 4.4, show a 2-ns non-overlapping dead time between the two clock phases.

Rotating-Clock-Phase Generation The clock signal *clk1* is used to activate the coefficient storage registers. Rotation of the coefficients is initiated by the falling-edge of *clk1*. Various track-and-hold clock phases for the N

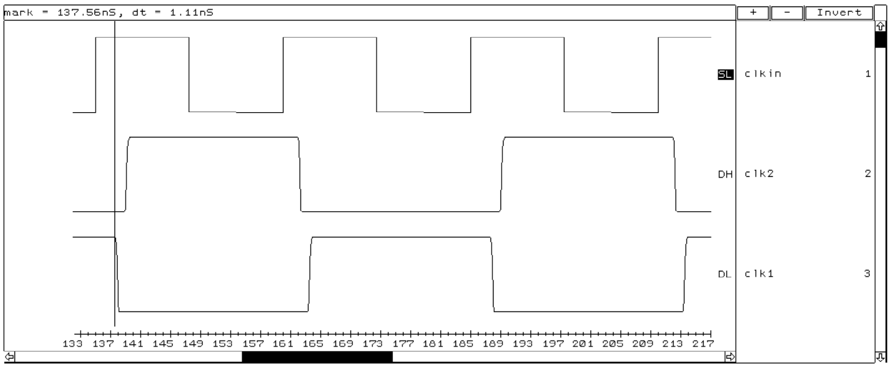


Figure 4.4 Simulation shows 2ns non-overlapping of $clk1$ and $clk2$ with an external clock $clkin$

FIR filter taps are derived from the clock signals *clk2* and *clk1*, as depicted in Fig. 4.5. The signal $phi1[0]$ is the track clock phase for tap 1, while $phi2[0]$ is the hold clock phase for tap 1. A pair of such $phi1$ and $phi2$ clocks is generated for each filter tap. The N track-and-hold states are listed in Table 4.2, where $x = 0, \cdots, N-1$.

$phi1[x]$	$phi2[x]$	State
1	0	Track
0	1	Hold

Table 4.2 Track-and-hold states

4.2.1 Glitch Avoidance and Noise-Coupling Reduction

The gate-level circuit used for clock-phase generation is shown in Fig. 4.6. Falling-edge-triggered flip-flops are used to avoid possible glitches in the sampling clock phases. This also ensures that the hold-period starts midway through the clock cycle, after the transition of the coefficients has occurred.

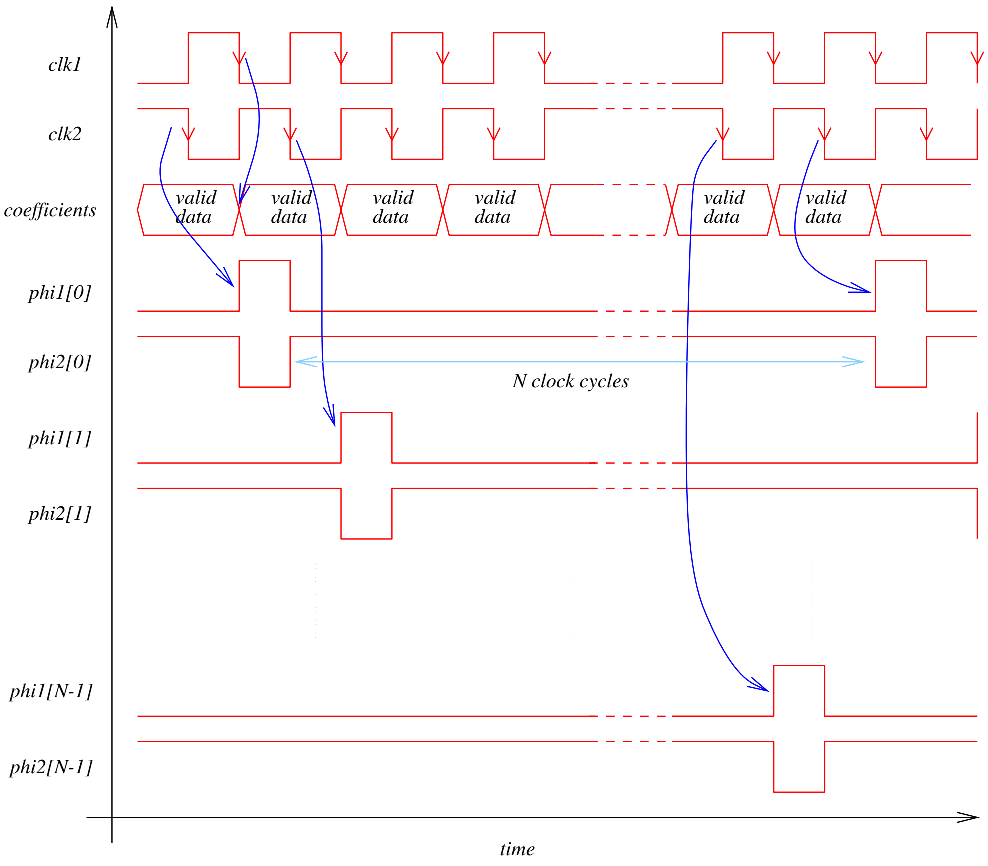


Figure 4.5 Clock $clk1$ triggers the rotation of the coefficients and $clk2$ serves as the clock source for sample-and-hold phases for an N-tap FIR filter

Delaying the hold-phase is necessary to reduce noise-coupling to the analog signal path from the transition of digital signals, which can be explained with the aid of Fig. 4.7a.

Possible Glitches with Positive-Edge-Triggered Flip-Flops: Suppose a rising-edge-triggered flip-flip were used in the circuit of Fig. 4.6. The output *phi1* is derived from the Q output of the D-flip-flop ANDed with the clock signal *clk2*. Note that the gate-delay of the flip-flop causes a time delay t_d in Q. At time t_2, the second clock cycle *clk2* goes to a logic *HIGH*. However, Q remains *HIGH* because of the gate-delay. A glitch arises at that

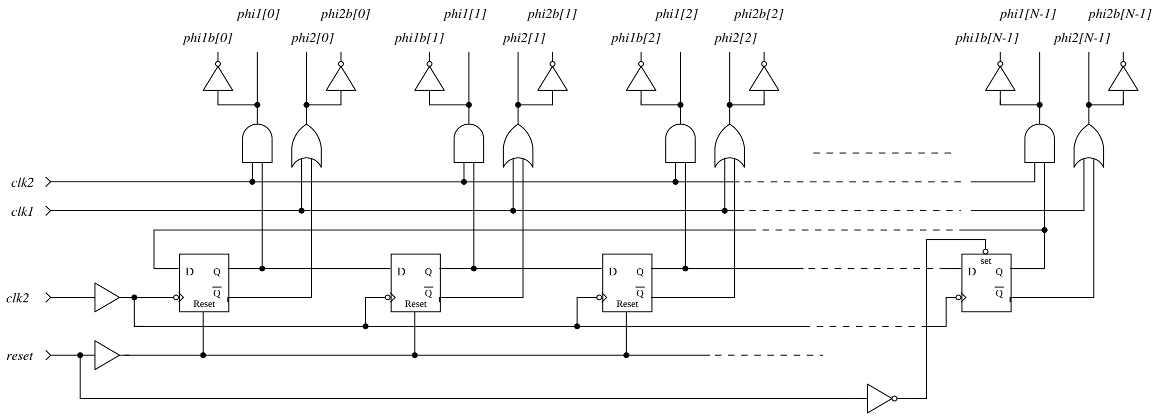
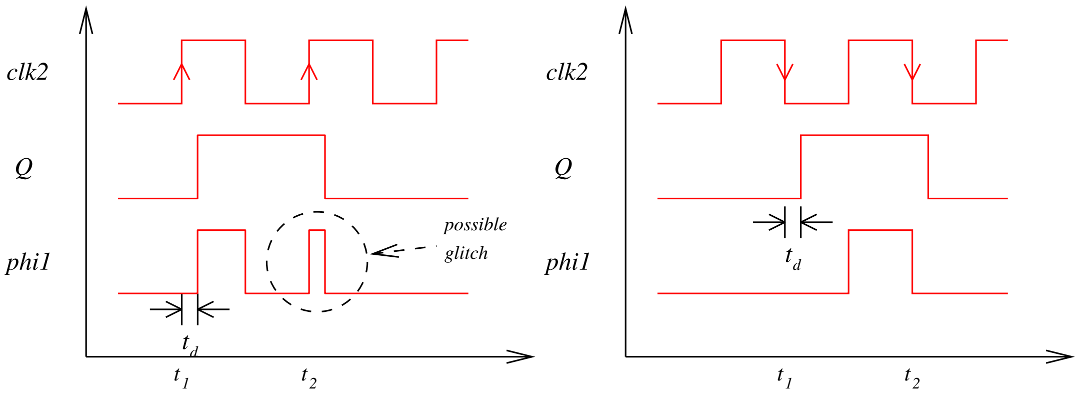


Figure 4.6 Sampling clock for individual taps



(a) Possible glitch generation

(b) No glitch generated using falling-edge triggered flip-flops

Figure 4.7 Use of falling-edge-triggered flip-flop to reduce the chance of glitch generation

time because Q and *clk2* are both *HIGH*.

No Glitch with Negative-Edge-Triggered Flip-Flops: The previous situation can be contrasted with the case where negative-edge-triggered flip-flops are used. Referring to Fig. 4.7b, the falling-edge of *clk2* activates the output, Q, after the same delay t_d. At time t_2, the next period of *clk2* is now *LOW*; the resultant *phi1* signal remains *LOW* regardless of the gate-delay. Also, the *HIGH* period of *phi1* is coincident with that of *clk2*, while this period would be reduced by t_d in the case of using rising-edge-triggered flip-flops.

4.3 Summary of Digital Element Design

Although the digital portions of the SI-FIR filter are straightforward, a sloppy design could severely degrade overall performance by injecting switching-noise. Care was taken in this design to minimize digital interference by using non-overlapping clocks and by triggering the flip-flops such that harmful glitches can not occur.

A certain degree of cross-coupling between circuits sharing a common substrate can not be avoided. Unwanted signals will be transmitted to sensitive analog nodes through parasitic capacitances and through the substrate. Therefore, the circuit layout is just as important as the functional design in minimizing interference. Digital-noise immunity has been considered in the functional design reported in this chapter. In the next chapter, physical mechanisms for noise coupling in the circuit layout will be considered.

The saying: "The cultivation of the individual consists in the rectification of the heart or mind" may be illustrated in this way — If a man is filled with anger, then he will not be able to behave correctly; nor can be behave correctly, if he is filled with terror, or if he delights excessively in something, or if he is full of worry and grief. Thus when the man is full of anger, terror, joy, worry and grief, it would appear that his mind no longer resides in him..... When a person becomes so absent-minded, he will look but not see; he will listen but not hear; he will eat but not taste. That illustrates what is meant by "the cultivation of the individual consists in the rectification of the mind".

Commentary VII — Rectifying the heart or mind and cultivating the individual.
The Great Learning, Confucian Classic
Translated by Dr. Mary Ng En Tzu