

# A HIGH PERFORMANCE, HYBRID WAVE-PIPELINED LINEAR FEEDBACK SHIFT REGISTER WITH SKEW TOLERANT CLOCKS

*Jabulani Nyathi, José G. Delgado-Frias, and Jeff Lowe  
School of Electrical Engineering and Computer Science  
Washington State University, WA 99164*

**Abstract**– *Clock skew and clock distribution are increasingly becoming a major design concern in synchronous pipelined systems. We present a novel high-speed hybrid wave-pipelined linear feedback shift register that manages clock skew by permitting the clock to travel with its associated data through the pipeline. The wave-pipelined clock has a skew 8.34 times lower than that of a buffered clock and is 1.2 times faster.*

**Index Terms**– *Hybrid wave-pipelining, clock skew, clock distribution, high-performance, linear feedback shift register, delays clock network, power*

## I. INTRODUCTION

Technology scaling facilitates high clock frequencies and provides the ability to integrate many devices on a chip. Clocks are usually subjected to larger loads resulting in increased clock skew and longer delays in distributing these clocks and other global signals. In this paper, we discuss a high performance hybrid wave-pipelined linear feedback shift register with a skew tolerant wave-pipelined clock. The hybrid wave-pipelining scheme takes into consideration interconnects delays and data path delays to improve system performance and optimize clock skew [1]. In this scheme, the system's clock in conjunction with pipeline stage delays is used to generate wave-pipelined clocks that have short cycle times and accompany the data through the pipeline. Short clock cycle times are achieved by reducing the delay difference between the critical path and the shortest path of a system [2, 3]. In hybrid wave pipelining, the intermediate latches allow for delay balancing and the system's delay variations are minimized per stage. The stage with the largest delay variation sets the system's clock cycle time [4].

Linear feedback shift registers (LFSR) in general are constructed with D-type flip-flops in the forward path and linear XOR or XNOR logic in the feedback path. The initial value of the shift register, shift register taps and feedback logic determine the output sequence [5]. The initial values cannot be set to all zeros if XOR gates are used and they cannot be all ones if XNOR gates are used in the feedback. These conditions constitute lock-up states and if entered the linear feedback shift register's output sequence will never change. The taps when selected properly allow the

output sequence to have a longer period. An LFSR with  $n$  stages will have a period  $2^n - 1$ . Maximal sequences (sequences with the highest attainable period) are preferable since the properties of non-maximal sequences are generally of little importance. Linear feedback shift registers find wide spread applications in pseudo-random number generator, random pattern generator and analyzer, encryption/decryption and direct sequence spread spectrum for digital signal processing [6]. In some of these applications it is necessary to have large sequences and for proper operation a system clock must drive all the entries with minimal skew. In this paper we present a linear feedback shift register that has skew tolerant clocks at frequencies in excess of 1 GHz in a modest 0.5micron technology.

Section 2 discusses linear feedback shift registers and general flip-flop clocking, Section 3 presents the circuits designed to achieve hybrid wave-pipelining and high performance along with the associated traces. Some concluding remarks appear in Section 4.

## II. LINEAR FEEDBACK SHIFT REGISTERS

There are primarily two approaches that have been employed in the implementation of linear feedback shift registers namely, the Fibonacci and the Galois implementations. Existing literature indicate that the Galois implementation yields better performance since there is no logic in the feedback path. This leads to less performance degradation in the event that the length of the sequence is increased. The Fibonacci approach on the other hand has logic in the feedback path and this leads to performance degradation for long sequences. Our design uses the Fibonacci approach to enable an investigation of how hybrid wave pipelining could reduce delays associated with logic in feedback paths. A common practice in the implementation of Fibonacci based linear feedback shift registers is to place the feedback logic in series. This adds considerably to the delay associated with the feedback path. In our design the feedback logic has been placed in parallel in order to reduce the feedback path delays that result from placing these circuits in series. Figure 1 is a block diagram of an  $n$ -bit LFSR showing how the feedback logic is configured in parallel.

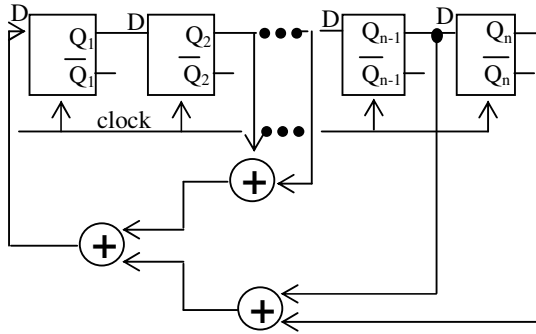


Figure 1. Two level tap LFSR

Linear feedback shift registers sequence through  $2^n - 1$  states, where  $n$  is the number of registers in the LFSR. At the edge of each clock cycle once loaded, the contents of the registers are shifted one place to the right. A 3-bit linear feedback shift register will go through seven clock cycles before the initially loaded pattern repeats again. There are  $2^3 - 1 = 7$  states for this 3-bit LFSR. The  $Q_1 = Q_2 = Q_3 = 0$  state is illegal as it constitutes a lock-up state when XOR gates are used in the feedback path. Table 1 shows the sequence of states when  $Q_1, Q_2$  and  $Q_3$  are initially loaded with the pattern  $100_2$ . The selected example of a 3-bit LFSR provides for only two taps. The outputs  $Q_2$  and  $Q_3$  are inputs to the XOR gate whose output becomes input to the first register. The seed (initial values) generally has no bearing on the length of the sequence and all the possible states will eventually be entered regardless of the starting values. The linear feedback shift register's size, implementation and tap position determine how the unique values vary. A poor selection of taps can lead to a LFSR that has a cycle time much less than  $2^n - 1$ . The LFSR counter is fast since it does not require carry signals and can be used in performance critical applications where the count sequence is unimportant as in first in first out (FIFO) buffers.

Table 1. Possible states a 3-bit LFSR

$Q_1$	$Q_2$	$Q_3$
1	0	0
0	1	0
1	0	1
1	1	0
1	1	1
0	1	1
0	0	1
1	0	0

To further reduce the delays associated with the feedback path clock skew must be tightly controlled. For an LFSR with 16 registers and taps at nodes 4, 13, 15 and 16, it becomes imperative that the output at node 4 and node 16 be available at the same time when the clock makes a transition. We present a hybrid wave-pipelined LFSR that manages clock skew and reduces the delays associated with the feedback path.

### III. HYBRID WAVE-PIPELINED LSFR

The high performance, hybrid wave-pipelined linear feedback shift register has been designed to be flexible. The user has the ability to choose the length of the sequence, select taps, disable the clock, and perform initialization within a single clock cycle. The design uses a six-transistor D-type flip-flop as a basic cell in the forward path. The basic cell is clocked on both levels of the clock in an effort to reduce idle time within the logic by admitting new data into the flip-flop before the current data is latched at the output latch. This approach allows for at least two unrelated data waves within the flip-flop (a characteristic that distinguishes wave-pipelining from conventional pipelining).

The hybrid wave-pipelined LFSR is designed for high-speed applications such as built-in self test, encryption/decryption and direct spread spectrum just to name a few. The LFSR design accommodates environmental parameter variations and the reported maximum clock frequencies can be maintained over the range of temperatures supported by the technology. It is recognized that the basic register cell appearing in Figure 2 could dissipate considerable static power and might have poor noise margins, we therefore point out that the focus of the design was on the clocking scheme instead. If density is crucial this becomes an ideal cell.

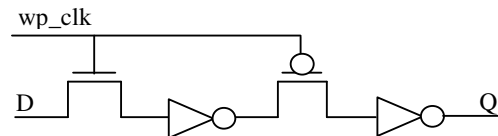


Figure 2. The basic register cell

The wave-pipelined clock experiences the same delays as the data resulting in the design's operation frequency being limited by the data path delays, the register set-up and hold times and wire delays. The clock generating circuitry is designed to mimic the data path, setting the clock's amplitude and pulse width. Use of logic to determine the amplitude of the clock allows the design to detect logic levels resulting in possible reduction of current sourcing or sinking. Dynamic nodes can thus experience brief switching activities. The wave-pipelined clock is designed based on this data path to accommodate the clock skew. In the

register of Figure 2, the data path is nothing more than two pass transistors and two inverters all in series and therefore easy to mimic. Figure 3 shows the clock generation circuitry. The devices in this circuit are designed such that they determine the pulse width and amplitude of the clock as aforementioned. The clock frequency is not set based on the knowledge of the system's worst case operation, but determined by the logic of the system's data path. Allowing the clock to accompany the data through the stages of a pipeline reduces clock skew since the clock now experiences the same delays as the data.

The delayed system clock labeled  $\bar{A}$  in Figure 3 provides a path to ground whenever it is high and places the *wp-clk* node at logic 1 whenever it is low through transistors  $N_2$  and  $P_1$  respectively. The system clock (*ref\_clk*) is input to transistor  $N_1$  preventing the *wp-clk* signal from being the exact inverse of the delayed clock and serving to charge up node *B* just before evaluation occurs. It must be pointed out that the wave-pipelined clock (*wp\_clk*) node has a brief window during which it floats. This occurs when signal *ref\_clk* goes to logic 0 while the signal driving pass transistor  $N_2$  is at logic 1. This is a potential problem particularly for the physical chip's proper operation. The circuit therefore requires some minor modifications, as critical designs would not tolerate floating clocks.

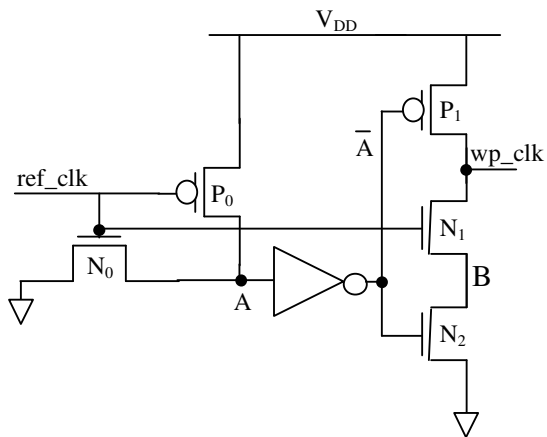


Figure 3. Wave-pipelined clock generating circuit

The wave-pipelined signal generated by the circuit above appears in Figure 4. The inverse of the wave-pipelined clock is also shown. The wave-pipelined clock has 2.5 volts as its highest value denoting logic 1 and its pulse width is far much shorter than that of the reference clock (*ref\_clk*). Permitting the logic to determine the signal pulse widths and amplitudes results in a clock pulse width reduction of 55 percent. This implies that the clock cycle time can be improved considerably. Having the logic determine pulse widths and amplitudes of the clock permits the feedback logic

to receive inputs early, thus reducing the delays associated with the feedback path to just the sum of the XOR gate delays in the path. This is a direct result of the hybrid wave-pipelining approach, where the delay differences are reduced per stage and the intermediate latches used to balance the delay paths. Wave-pipelining the clock allows the system to maintain high clock speeds even with added logic in the feedback path. An increase of logic stages in the feedback path can occur when the number of taps within the shift register is increased.

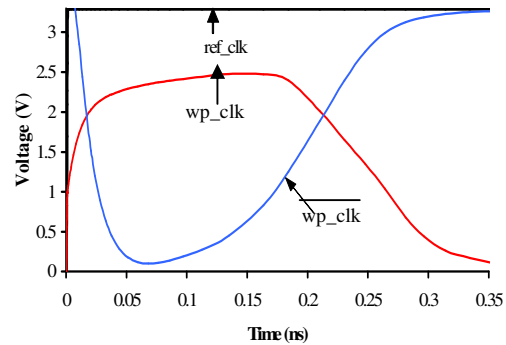


Figure 4. Reference and wave-pipelined clocks

Simulation results of a 16-stage hybrid wave-pipelined LFSR showing the wave pipelined clock and the propagation of logic 1 through the stages are shown in Figure 5. Figure 6 depicts the skew measurements with the fourth stage (flip-flop) receiving the clock edge  $8.061ps$  before the last stage receives the same clock edge. This is considerable improvement compared with  $67.221ps$  measured for a buffered clock. The  $Q_4$  and  $Q_{16}$  outputs were considered because they are the two tap points far apart and having larger parasitic capacitance than the outputs without taps. The flip-flop outputs ( $Q_4$  and  $Q_{16}$ ) for the system with a wave-pipelined clock (Figure 6) need to be magnified in order to make the skew more visible since  $8.061ps$  is such a small duration. These measurements represent the worst-case skew for both cases. The hybrid wave-pipelined clock operates at a frequency 1.2 times higher than that of a buffered clock.

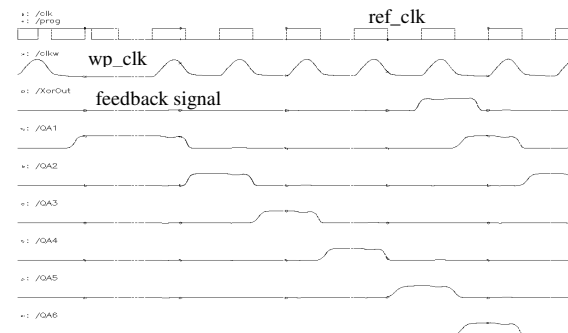


Figure 5. Logic 1 propagation for 16-bit LFSR

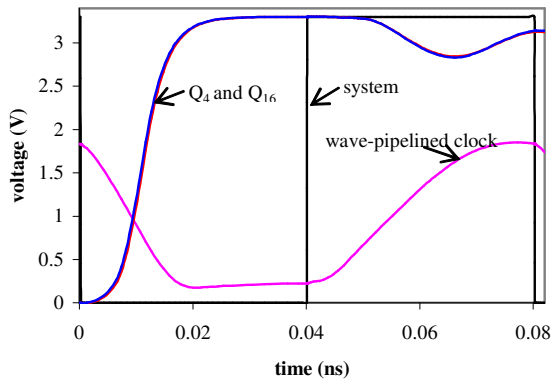


Figure 5. Worst-case clock skew measurements

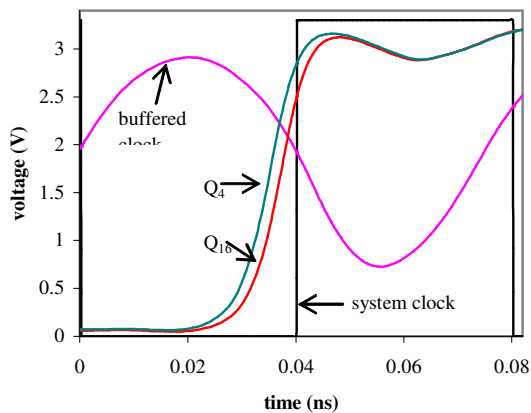


Figure 6. Worst-case clock skew (buffered clock)

Measurements to determine performance degradation with an increase in the number of registers have been carried out. Feedback delays were monitored as the register and tap numbers were increased. There is a 20 percent decline in clock frequency as the number of taps increases from two to four while the number of registers is increased from four to sixteen. This degradation of clock frequency would remain the same for a large number of stages (32, 64 etc) as the number taps would remain constant (four) even as more registers are added. The LFSR driven by a buffered clock showed a 30 percent decline in performance and could not be run at comparable frequencies with sixteen stages.

#### IV. CONCLUDING REMARKS

In this paper we presented a linear feedback shift register with skew tolerant clocks. The hybrid wave-pipelining scheme permits for skew reduction by pipelining the clock and allows for a decrease in clock cycle time by reducing delay variations per system stage. In addition the scheme can sustain at least two unrelated data waves within a stage resulting in an increase in throughput. The scheme is based on wave

pipelining and strives to reduce a design's delay variation between the critical path and the shortest path per stage. The hybrid wave-pipelining scheme also reintroduces the intermediate pipeline latches to ensure ease of delay balancing.

Using the hybrid wave-pipelining scheme we have demonstrated that we can manage clock skew and reduce clock network design effort. The design of the linear feedback shift register presented shows 8.34 times improvement on clock skew when compared to the traditional buffered clock subjected to the same load as the pipelined clock. Pipelining the clock entails having the clock circuitry mimic the data path, resulting in the clock experiencing the same delays as those of the data path. The fact that the data path circuitry needs minor modifications for clock generation translates into reduced design effort for the clock network. It is envisioned that this design approach can be extended to alleviate the clock distribution issues in the nanometer regime by designing the clock to accompany the data from one module to the next.

It has also been shown in this paper that clock cycle time can be reduced by 20 percent when the hybrid wave-pipelining scheme is employed. This paper considers a configuration of the LFSR's feedback logic that enables for the reduction of delays associated with the feedback path. Some logic gates in the feedback path are placed in parallel instead of the typical series configuration. This contributes to the design's capability of maintaining high-speed operations with minor degradation in performance even when more logic gates are added in the feedback path. The recorded results associated with the feedback delays would be more meaningful when compared to those of an implementation without logic in the feedback path. Further studies are underway to evaluate the scheme's potential in reducing the power associated with the clock networks.

#### V. REFERENCES

- [1] J. Nyathi and J. G. Delgado-Frias, "A Hybrid Wave-Pipelined Router for Interconnection Networks," *IEEE Transactions on Circuits and Systems*. Vol. No. December 2002.
- [2] W. P. Bursleson, M. Ciesielski, F. Klass and W. Liu, "Wave-Pipelining: A Tutorial and Research Survey," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, Vol. 6, No. 3, pp. 464-474, September 1998.
- [3] C. Thomas Gray, Wentai Liu, Ralph K. Cavin, III, *Wave Pipelining: Theory and CMOS Implementation*, Kluwer Academic Publisher 1994.
- [4] J. G. Delgado-Frias and J. Nyathi, "A Wave-Pipelined CMOS Associative Router For Communication Switches," *2000 IEEE International Symposium on Circuits and Systems*, Geneva, Switzerland, pp. 391-394, May 2000.
- [5] M. George and P. Alfke, "Linear Feedback Shift Registers in Virtex Devices (application note)" <http://www.xilinx.com/bvdocs/appnotes/xapp210.pdf>.
- [6] <http://www.sss-mag.com/pdf/lfsr.pdf> "Linear Feedback Shift Register Megafunction," Version 1, December 1996.