Introduction

Cyclic redundancy check (CRC) is an error detecting code designed to detect changes in transmitted data and is used in digital networks and storage devices. CRC’s are easy to implement in binary hardware, analyze mathematically and are used to detect errors due to noise in the transmission channel. The message is treated as a polynomial in GF(2). For example, the message 1111001 is represented as a polynomial as \( x^6 + x^5 + x^4 + x^3 + 1 \). Specification of a CRC code also requires a pre-defined generator polynomial. The sender and receiver agree on a certain fixed polynomial. The remainder is calculated as by dividing the GF(2) polynomial by the generator polynomial to generate a reliable check sum. Division of polynomials over GF(2) can be done in much the same way as long division of polynomials over the integers. This can be implemented using an exclusive OR gate. At the receiver, the integrity of the data is checked similarly. The data is said to be valid if the check sum at the receiver yields all zeros.

Design progress

The design component of the project involved implementing a cyclic redundancy check circuit using Linear Feedback Shift Registers (LFSR). We use a generator polynomial \( G(x) = x^6 + x^5 + x^4 + x^3 + 1 \). The CRC-6 is a method of performance monitoring that is contained within the F-bit position of frames 2, 6, 10, 14, 18 and 22 of every multiframe in Synchronous Frame structures used at 1544, 6312, 2048, 8448 AND 44 736 kbit/s Hierarchical levels.

We implemented three different transistor topologies for a positive edge triggered D flip flop for power reduction in the overall CRC circuit. The transmission gate master-slave flip flop, TSPC D-flip flop and a novel TSPC flip flop proposed in [1]. We worked on effective sizing strategies of these topologies for power reduction and reduced glitching. Table 1 shows the power drawn by the circuit for different sizing of the transistors in the TSPC D-flip flop topology.
Table 1: Average power consumption

<table>
<thead>
<tr>
<th>Size</th>
<th>Power (uW)</th>
<th>Reasoning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial Wp/Wn=2</td>
<td>21.7</td>
<td>Greater power consumption as optimal sizing strategy was not used</td>
</tr>
<tr>
<td>Optimal sizing</td>
<td>4.6</td>
<td>With proper sizing, the TSPC flip flop topology showed considerable power decrease.</td>
</tr>
<tr>
<td>Optimal sizing (x1/4)</td>
<td>10.6</td>
<td>Validates that the power drawn by the circuit is a proportional to the area of the circuit. It is a convex function.</td>
</tr>
<tr>
<td>Optimal sizing (x1/2)</td>
<td>5.56</td>
<td></td>
</tr>
<tr>
<td>Optimal sizing (x2)</td>
<td>6.2</td>
<td></td>
</tr>
<tr>
<td>Optimal sizing (x4)</td>
<td>7.2</td>
<td></td>
</tr>
</tbody>
</table>

Figure 1: Power consumption as a convex function of transistor scaling (TSPC topology)
Challenges faced during simulation of flip-flop topologies

Three different flip flop topologies are simulated and their individual total power measured. We found that the power drawn from the source for flip flop function was higher than the expected value by a factor of about 4. The issue with the topologies were the sizing of the transistors implemented in the circuits. The overall power consumed by the CRC circuit (implemented using these individual topologies) was also in the order of mWs (for dynamic TSPC topology for glitch reduction) and in higher uWs (for master-slave flip flop configuration). After adequate research and literature reading we identified the issue and realized that logical effort cannot be used to size the TSPC flip flops as it is not CMOS. After adequate sizing based on the TSPC flip flop functionality, we observed power reduction by a factor of about 4.

Also, the overall output of the CRC shows reduced glitching as compared to high glitching activity of the circuit sized previously with a ratio of Wp/Wn=2. Figure 2 and shows the reduced glitching activity for CRC implemented using TSPC topology.

Secondly, when we examined the leakage power of the overall CRC circuit implemented using TSPC D flip flops, we observed leakage in the order of high uW. We looked into the circuit and isolated the issue by debugging the overall system to see if there were any possible floating voltages in the output node. We found voltage drop at the output node and implemented a balloon latch to mitigate the issue. Figure 3 shows the drop in voltage and the mitigated dissipation after a balloon latch was implemented. Further, the leakage power of the overall circuit was 3.8uW for optimal sizing.
Figure 2: TSPC D – Flip flop; effect of transistor size optimization
Figure 3: Effect of balloon latch; examining leakage power of the circuit.

Figure 4 shows the circuit schematic of the TSPC flip flop implemented using SPICE.

**Questions/Obstacles/Issues**

During the implementation of the proposed TSPC topology in the CRC circuit, we noticed increased power consumption (in the order of mWs). Sizing provided in [1], according the the 0.25um technology did not help reduce power in the SPICE simulation. Going forward we plan on examining a different sizing strategy and also aim to isolate floating nodes (if any) to identify the issue. Although the topology functions effectively as a flip flop (as shown in Figure 5 and 6), its overall functionality with the CRC circuit will be re-examined. Further, we plan on calculating the power consumed by the circuit vs. scale (area) and perform analysis similar to TSPC D- flip flop implementation.

**Remaining Tasks**

- Re-examine the proposed [1] TSPC flip flop topology for optimal sizing and reduced power consumption.
- Examine the leakage power of the overall CRC circuit implemented using the proposed topology.
• Power consumption vs. scale (area) curve generation for the same.
• Examine both these topologies based on the Pareto curve (energy vs. delay)

Paper summaries


The most important step in the clock subsystem design is the optimization of register elements, which involves the selection of energy-efficient flip-flop topologies. At the same time, energy consumed by the clock distribution network is reduced when register elements are able to relax the clock distribution constraints. The most commonly used flip-flop design techniques are conventional master-slave latch-pairs and pulse-triggered latches. Other low-energy designs often derived from the conventional techniques, use double-edge-triggering, reduced swing clock, or internal clock gating. Characterization metrics, relevant to low-energy systems are discussed, providing insight into timing and energy parameters at both the circuit and system levels. Transistor sizes are optimized for minimal delay under constrained energy consumption. This methodology is applied to characterization of various flip-flop styles and their comparison in 0.25µm CMOS technology under scaled supply voltages. The optimal flip-flop topology and size is dependent upon the particular operating condition. However, among the presented flip-flops, the transmission-gate flip-flop with input gate isolation is the best overall choice for low energy digital design due to its good energy-delay trade-off, large race margin, sufficient noise robustness, and small energy required to drive data and clock inputs. Internal clock gating is effective for low input switching probabilities, added to the transmission-gate flip-flop with input gate isolation.

D-flip-flops (DFF’s) are one of the major functions in finite-state machines (FSM), which in turn form a critical part of control logic. Enhancing DFF speed can either lead to a higher clock rate or allow more logic depths between two pipeline registers. In this paper, the authors compare the area, speed, and power dissipation of existing DFF implementations with a conventional low-risk DFF, a low-area DFF, and a low-power DFF. They also propose two energy-efficient DFF designs: a push–pull DFF for performance and, a push–pull isolation DFF (PPI-DFF) for performance and energy efficiency. These are then compared with the existing designs and demonstrated to be better for high-performance, energy-efficient applications. Though the low-area DFF uses up to 33% fewer transistors, the internal voltage contention consumes up to 122% more energy than the rest of DFF’s. Compared to a conventional DFF, a low-power and a push–pull DFF improve power dissipation by 1% and delay by 31%, respectively, but end up with comparable energy efficiency. The proposed PPI-DFF improves speed by 56% at the expense of only 6% more power, when compared to a conventional DFF. Energy efficiency of this PPI-DFF is 45–122% better than that of the other DFF’s. Compared to the existing low-power DFF, the PPI-DFF uses 47% less energy. This may result in a 30–40% reduction in the overall energy consumption of control logic. On the issue of metastability, the lack of feedback transistors in both low-area and push–pull DFF’s causes them to have a metastability window one to two orders of magnitude larger than the other three DFF’s. In the applications where synchronizing circuits are required, the inclusion of both nMOSFET and pMOSFET devices in the feedback path of a DFF can minimize the metastability window and the resolution time constant, hence reducing their susceptibility to failure.


Various high-speed FFs have been proposed in the past, mainly belonging to the Pulsed and Differential classes. Usually, they are featured by a transparency window, leading to clock-uncertainties
absorption properties but also to reduced race immunity. However, both setup and hold time values can be arranged regardless of the FF delay value, since they depend on the sizing of gates that do not belong to the FF critical path. Therefore, the real figure of merit concerning the timing of such FFs is the minimum data-to-output delay, measuring the impact of FF speed on the clock cycle. Given the presence of precharged nodes and the high switching activity in the pulse generator stages, high-speed FFs are distinguished by an high dissipation (e.g., compared to low-energy FFs, such as Master-Slave ones). Therefore, given that CMOS technology has entered a power-limited regime, identifying the most energy-efficient high-speed FFs is nowadays a decisive issue. In this paper, the ranking of the most representative high-speed FFs in a 65-nm CMOS technology is reconsidered by including the the increasing impact of layout parasitics associated with local interconnects, in order to reach the very optimum FFs sizings corresponding to the energy efficient designs in the Energy-Delay space. As a general remark, simpler basic structures are rewarded in nanometer technologies because of the strong impact of layout parasitics. In particular, the Explicit Pulsed circuits, which employ a pulse generator providing an actually pulsed clock, and the Transmission Gate Pulsed Latch circuits, have been recognized as the best high-speed FF topologies in a very wide range of applications.