TECHNICAL MEMORANDUM
Lafayette College
Department of Electrical and Computer Engineering

Title: Lab 2 - Sequential FPGA Design with SystemVerilog
Author(s): Waseh Ahmad and Geoff Watson
Date: September 12, 2017

Abstract
This memo presents our design for the asynchronous serial transmitter. It describes our design process, simulation results, and implementation results.

Introduction
In this lab, we were tasked with designing an asynchronous serial transmitter that accepts 8-bit data and transmits in a serial format that is compatible with a computer's Serial Port. The transmitter's interface needed to have an 8-bit input for data, a 1-bit input for send, a 1-bit output txd and a 1-bit output rdy, as shown in the figure below.

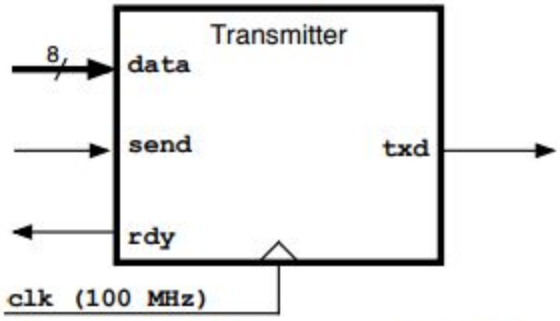


Figure 1 – Serial Transmitter Interface

Design
When designing the transmitter, we started by designing the Baud rate clk generator. In the transmitter module, we parameterized the baud rate, and fed that into the given "clkenb" module as the DIVFREQ. To make sure that only we were only generating this clock when we were transmitting, we had the the divider generator reset when the reset button was pressed, or when the ready signal was high. Next we used the given "single_pulser" module to increment the counter once per 100MHz clock cycle. We then fed that output into the given "bcdcounter" module as an enable signal to increment the counter, which counts up to 10 (1 start bit, 8 data bits, and 1 stop bit). At this point, we needed to create an FSM module that would control what the transmitter would do. We came up three states; idle, transmitting, and ended. In the idle state, the ready output and txd output are both logic high. If the send signal in not asserted high, we stay in the idle state. We leave the idle state and enter the transmitting state when the send signal is asserted high. When we first enter the transmitter, we want to send the start bit (logic low), so we check if the output of our bcdcounter is 0. If it is, we output our start bit, then wait for the counter to increment. The counter increments at the Baud rate, when the counter

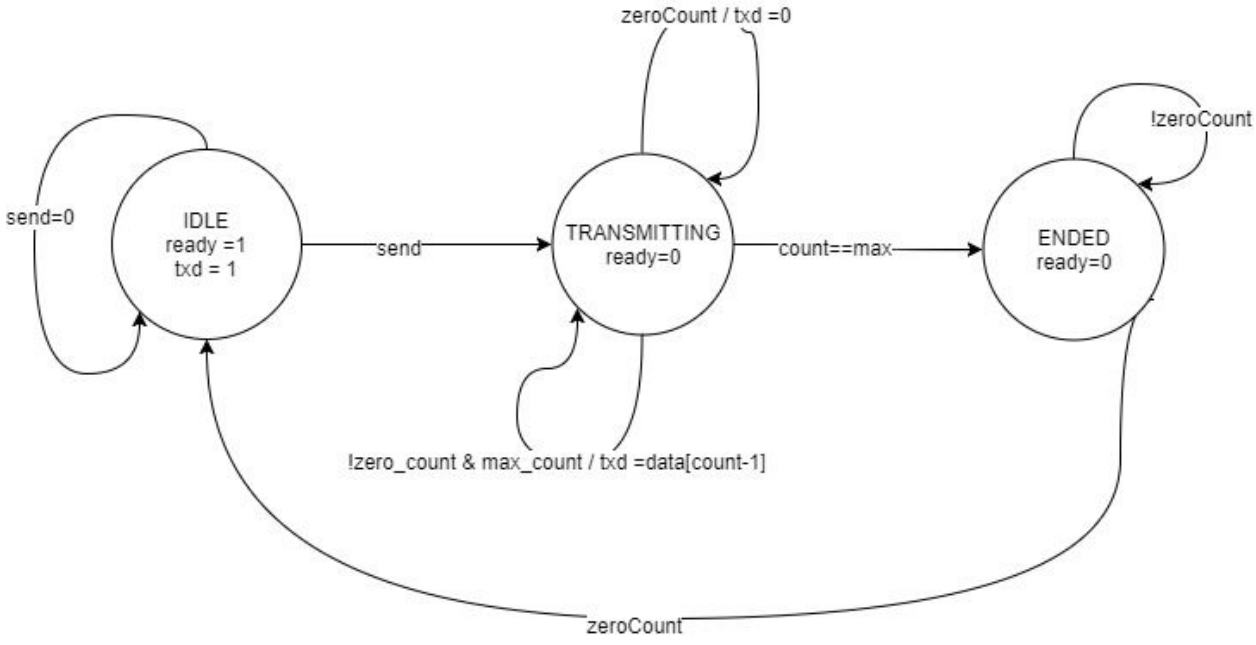
increments, we transmit the data setting txd equal to data[count-1]. Then, when the counter reaches the max value, we enter the ended state and set ready to 0. Then, when the counter rolls back over to zero, we re-enter the idle state and set ready and txd to 1.

FSM diagram

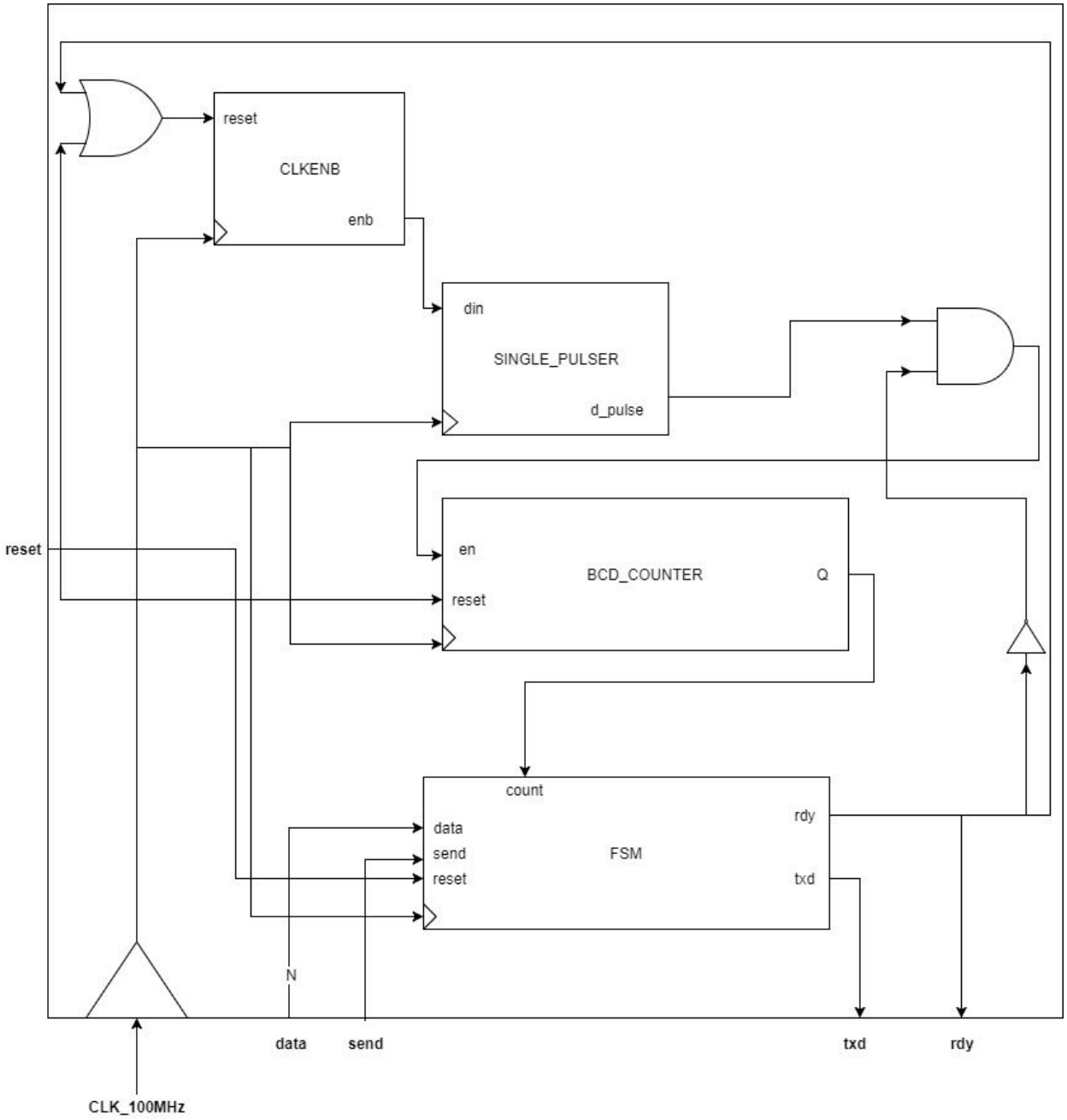
INPUTS
- [N-1 : 0] data
- [Log(N) : 0] count
- send

OUTPUTS
- rdy
- txd

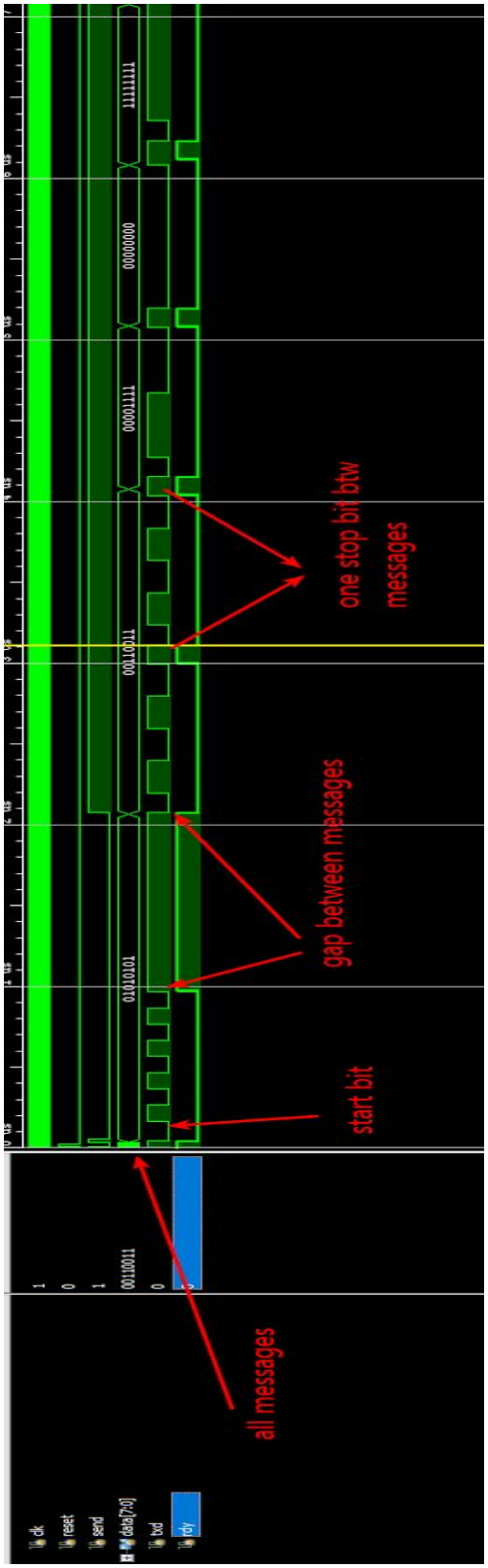
max = N+1
zero_count === (count==0)
not_zero_count
max_count = (count==max)



Module Block Diagram



SImulation Screenshot



Design Verification

Description	Test Method	Detailed Results
1. Module Interface	Code Inspection	Code is neat, orderly and follows the Coding Guidelines
2. Module function: accepts an 8-bit input and 1-bit "send" input and updates a 1-bit output "rdy" and a 1-bit output "txd" that changes at the baud rate	Demonstration in hardware using Nexys 4 DDR board and RealTerm: Properly displays all 8 digits when the send button is pressed When the send button isn't pressed, there is no output in RealTerm	-Bits are correctly shown on Real term. -All signals are correctly shown on the oscilloscope -PASS functionality for Idle state (rdy=1, txd = 1, send = 0) -Time period of one bit transferred is the baud rate.
3. Uses Nexys4 board 100Mhz clock; all flip-flop clock inputs tied directly to this signal	- Check every flip-flop and verify that it is driven by the board 100Mhz clock	PASS
4. Contains no latches	Inspection of Synthesis Report	The synthesis report had no warnings relative to latching
5. Test circuit – show test that test circuit functions properly by connecting it to the oscilloscope.	-Connect the send, rdy, and txd signal to the scope -Trigger the scope on the send signal -Check that the rdy signal goes low when the send signal goes high -Check that the start bit, data, and stop bit display at the correct frequency	Circuit passed all of the mentioned tests
In submitting this checklist as part of our report, I/We certify that the tests described above were conducted and that the results of these tests are accurately described and represented. I/We understand that any misrepresentation of the tests or the results constitutes a violation of the College policy on academic dishonesty.		
Name(s): ***Waseh Ahmad Geoff Watson*** *Date:* ***09/12/2017***		

Conclusion

This Technical Memo has described our design for an asynchronous serial transmitter. Through creating the modules necessary for the application, we learned many things along the way that helped us optimize our design, especially in the design of the FSM. Our approach worked very well and I would recommend to someone doing this lab.