Introduction

In this lab you will develop another register-transfer design: a serial receiver. It is more complex than the serial transmitter because it requires that you synchronize with an incoming signal and sample successive bits. Your design will probably consist of a shift register, one or more counters (used to generate delays), and a control unit that ties it all together.

Lab Preparation and Design

We will design the serial receiver assuming a clock rate that is 16 times the baud rate. The receiver circuit must implement the following steps:

1. Wait for the falling edge of the START bit.
2. Delay to the center of the START bit and sample the current data value. If the data value is not still asserted low, ignore the START bit and return to step 1.
3. Delay to the center of the next data bit. Sample the value of the data value and shift it into a shift register.
4. Repeat step 3 seven more times.
5. Delay to the center of the stop bit and sample. If the value is not correct, indicate a framing error condition.
6. Indicate that valid data is available in the shift register.
7. Go back to step 1.

Your asynchronous serial receiver should have the following inputs and outputs:
RXD is the serial data input. DATA is the parallel data output which should be tied to the manifold LEDs (or your own if you want to to display the 6 least significant bits of received data. RDY should be asserted when a character has been received, and FERR should be asserted when a character has been received but the value of the stop bit was not correct (FERR and RDY should remain asserted until a new start bit is detected).

1. Create a block diagram of your receiver design in terms of primitive building blocks (shift registers, logic blocks, control FSM, etc.) and bring it with you to lab.

2. Create an ASM Diagram which describes the function of your receiver’s control unit.

3. Use the block diagram and ASM Diagram to write Verilog code for your receiver design. Follow the coding guidelines distributed in class –use one file for each module, include title blocks, and use the indentation style described in the guidelines. Bring your code with you to Lab.

**In the Lab**

1. Compile your Verilog code and use the simulator to test its basic operation.

2. Write a sequential testbench that generates the proper sequence of signals to exercise your receiver design and checks that its outputs are received correctly. It should include test inputs that are
   
   a. Generated using Verilog behavior code.

   b. Generated using an instantiation of your transmitter design from last week.

   In both cases, the testbench should check that the values are correct and otherwise write an error message.

3. Synthesize your circuit using the Synopsys tools and then use `db2mag` to generate a layout. Plot the resulting layout and note its size.

**Report**

For your lab report hand in the following:

1. A short technical memorandum which describes (a) what was done, (b) what you learned, and (c) what difficulties you encountered.

2. The block diagram of your serial receiver design.

3. The ASM Diagram of your serial receiver control unit.

4. Verilog listings for your serial transmitter design.

5. A Verilographer timing diagram printout showing that it works for a single byte.


7. A printout of the “verilog.log” output log showing the messages printed by your testbench.