UART Receiver:

- Receiver looks for falling edge of start bit on **RX** line. (This does **not** mean you need an **always** block that is sensitive to **negedge RX**!)
- Once it sees falling edge of start bit it will count off ½ a baud time (1302 clocks in our case). Then it simply right shifts a 10-bit shift register sampling **RX** into the MSB position
- Now for the next 9 times the baud counter reaches 2604 it will right shift the shift register sampling a new value of **RX** into the MSB of the shift register.

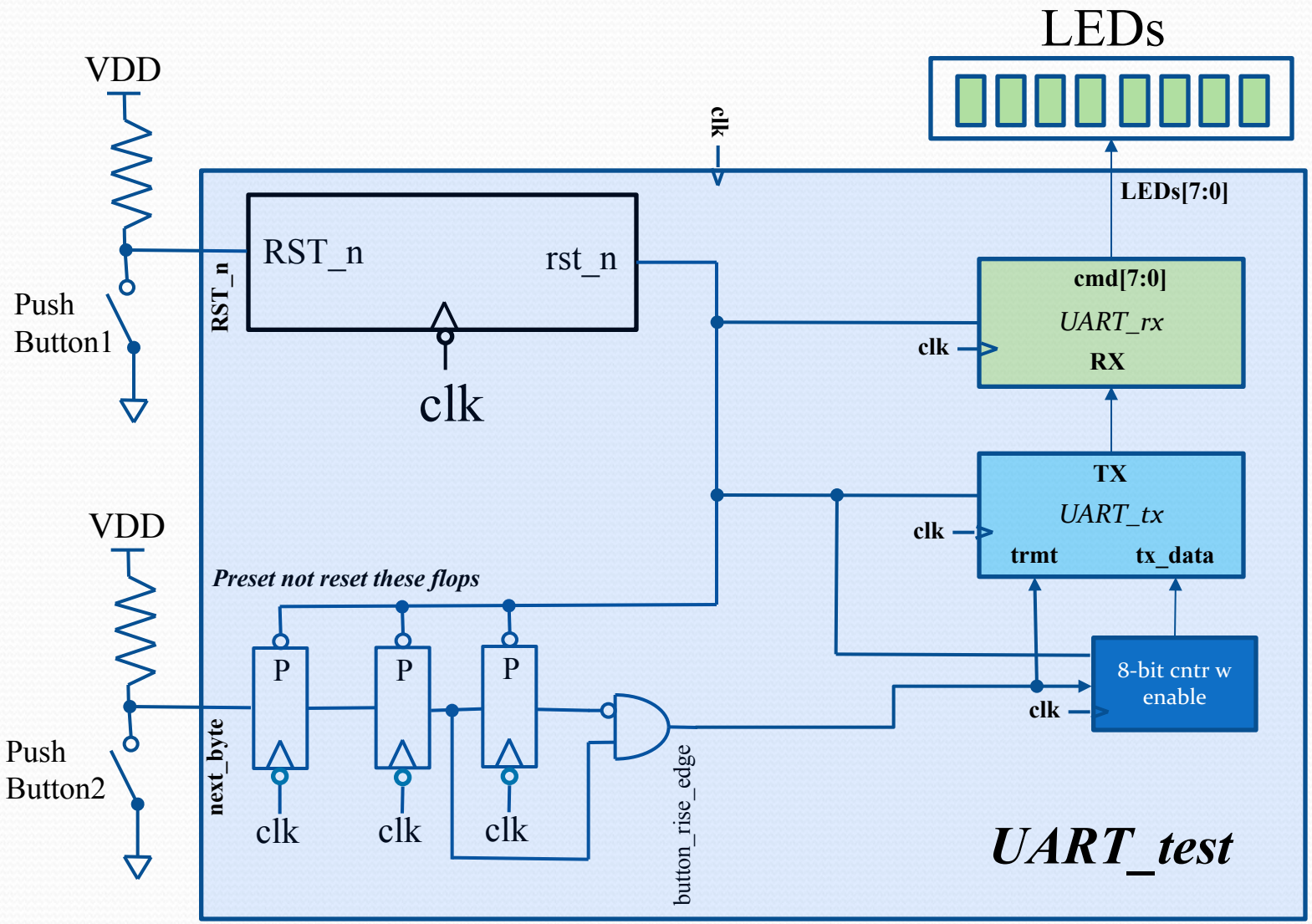
UART Receiver:

What does UART stand for?

What does the **A** in U**A**RT stand for?

Hmm… do I really want to sample **RX** into the MSB of my shift register?

Exercise 09 UART mapped to FPGA:



Exercise 09 UART mapped to FPGA:

- Download UART_test.qsf and UART_test.qpf from the website. Also download shell for UART_test.v…you will flush this out.
- Make a top level verilog block called UART_test.v that instantiates both your UART transmitter and UART receiver. Also include a small counter that will be used to send data to the transmitter.
- You will also need some logic off of push button2 hooked to next_byte.
- Every time you push button2 the counter should increment and sent its value over the UART to the receiving UART which displays its received byte on the LEDs.
- Submit UART_rcv.v and UART_rcv_tb.v
- Either demonstrate proper behavior to Prof. Kim or TAs, or submit a short video demonstrating proper behavior.