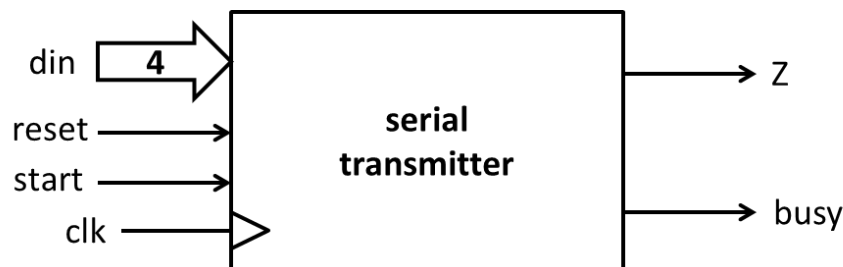# CPE 487: Digital System Design

HW7
Due: 5/3/18

1: FSM Design (25 points):

Assume you are asked to design a serial transmitter for a digital processor. The transmitter takes as input a clock, a *reset*, a *start* signal and a 4-bit data word to be transmitted. Outputs of the transmitter are a serial output bit Z and a *busy* flag.

When not transmitting, the serial transmitter sits in an *idle* state waiting for a *start* signal. The *reset* input asynchronously sets the transmitter into the *idle* state. When in the *idle* state, the output Z and the *busy* signal are both set to '0'.

When *reset* is not set, the transmitter samples the *start* signal on the positive edge of the clock. When *start='1'*, the transmitter reads the input data *din*, outputs a *z='1'* for one clock cycle and sets *busy='1'*. It then transmits the 4-bit word, one bit per clock cycle starting with the LSB and ending with the MSB. Once it has transmitted all four bits, it returns to the *idle* state for at least one cycle and sets *busy='0'*. (You can assume the processor raises the *start* signal for one cycle only and does not issue another *start* pulse until *busy* has been '0' for at least one clock cycle)

(1) Draw a Moore type FSM state diagram for the transmitter* (10 points)
   (Hint: In a Moore FSM, the output depends only on the state and not on the inputs. Augment the state with the value of the data to be transmitted. Set this value equal to din when start goes to '1')

(2) Write VHDL code to implement your design. Simulate your design transmitting the data word "1010" (15 points)

* With only 4 data bits to transmit, it is easy to explicitly enumerate the four states which output these data bits. If, however, the data word was longer (e.g. 12 bits) this could become tedious. A solution would be to also include a bit count as part of the augmented state. There are 3 bonus points for also including the bit count as part of the augmented state (still keep your design, however, at 4 bits of data)