

**MODULES TO CREATE**

**POW Position 8-bit flip flop module:**
- **Declaration:** `pow_position(pos,clk,rst,init,d);
- **Inputs:** [7:0]`init`(from random2) and [7:0]`d`(where does this come from?), **Outputs:** [7:0]`pos`(where does this go to?)
- One every clock, load `_init` if `rst` is pressed and load `d` otherwise

**YOUR Position 8-bit shift register module:**
- **Declaration:** `position_shifter(pos,clk,rst,sr,sl);
- Every clock, it should initialize `POS[7:0]=8'b00000001` when `rst` is high
- Otherwise, shift when `shift_right`(where does this come from?) is high and left when `shift_left`(where does this come from?) is high

**LED Display:**
- AND each of the 8 bits from the position register with output of the pulser to create a blinking indicator.
- XOR this result with the output of the POW position flipflop.
- **Pulser:** has duty cycle less then 50% so that the cursor is shown as short blinks of light when not over a POW and inverse when over a POW.
- **Clear Bits:** create the effect of clearing some bits if the cursor position is the same as a POW position
- **Game Over:** use a NOR to figure out when game is over, create some indication
WHAT TO DO IN HOMEWORK #3?

Modules To Add:

• Store POW Positions Module
• Store Your Position Module
• LED Display Modules