Design an 8 bit serial 2’s complementer with an 8 bit shift register and a single flip-flop (to hold the carry). On each rising clock edge the binary number should be shifted out from lsb or right side, y should take on the value of the complemented bit which should also be shifted into the msb, ie left side, so that after 8 shifts the register holds the complemented value.

module Serial_Twos_Comp (output y, input [7: 0] data, input load, shift_control, Clock, reset_b);

Test data will change on the falling edge of the clock and your register should shift on the rising edge.

y is the serial output (ie the 2’s complement bits, least significant bit first)
data is a number to load the shift register with,
when load is high this data is stored on rising clock edge.

when load is low shift_control high tells the register to shift and output the next bit in y on rising clock edge. when shift_control is low y should be 0.

Your design should store 0 on the negative edge of the reset signal.