Compression Logic

Report:

The schematic of the compression logic is shown below. As shown in figure, in this logic, the data inputs are coming in serially and the outputs are taken out parallel in 11 output lines. A byte of input is taken serially, bit-by-bit. To accommodate this type of data-in, I have considered an 8-bit shift register. The LSB of the input first moves into first D-Flip flop. When the second bit comes into the shift register, the bit in the first D-Flip flop moves to the second D Flip flop and so on. Finally, after 8 clock cycles, all the bits of a byte are stored in the shift register and ready to get processed in the combinational logic.

Combinational logic is a processing block where the data inputs are processed and accordingly compression takes place. In my code, the compression block consists of bunch of If and If-else statements which will determine whether the output has to be compressed to 2-bits or the output have to be 9-bits. My code analyses the data-in and if all the bits are 1's, then 01 is output through a 9 bit line. If the data-in consists of all 0's, then 00 is output through the same 9 bit line. In any other case (mixture of 1's and 0's) 1 followed by the same input is output through all the available 9 bit output line.

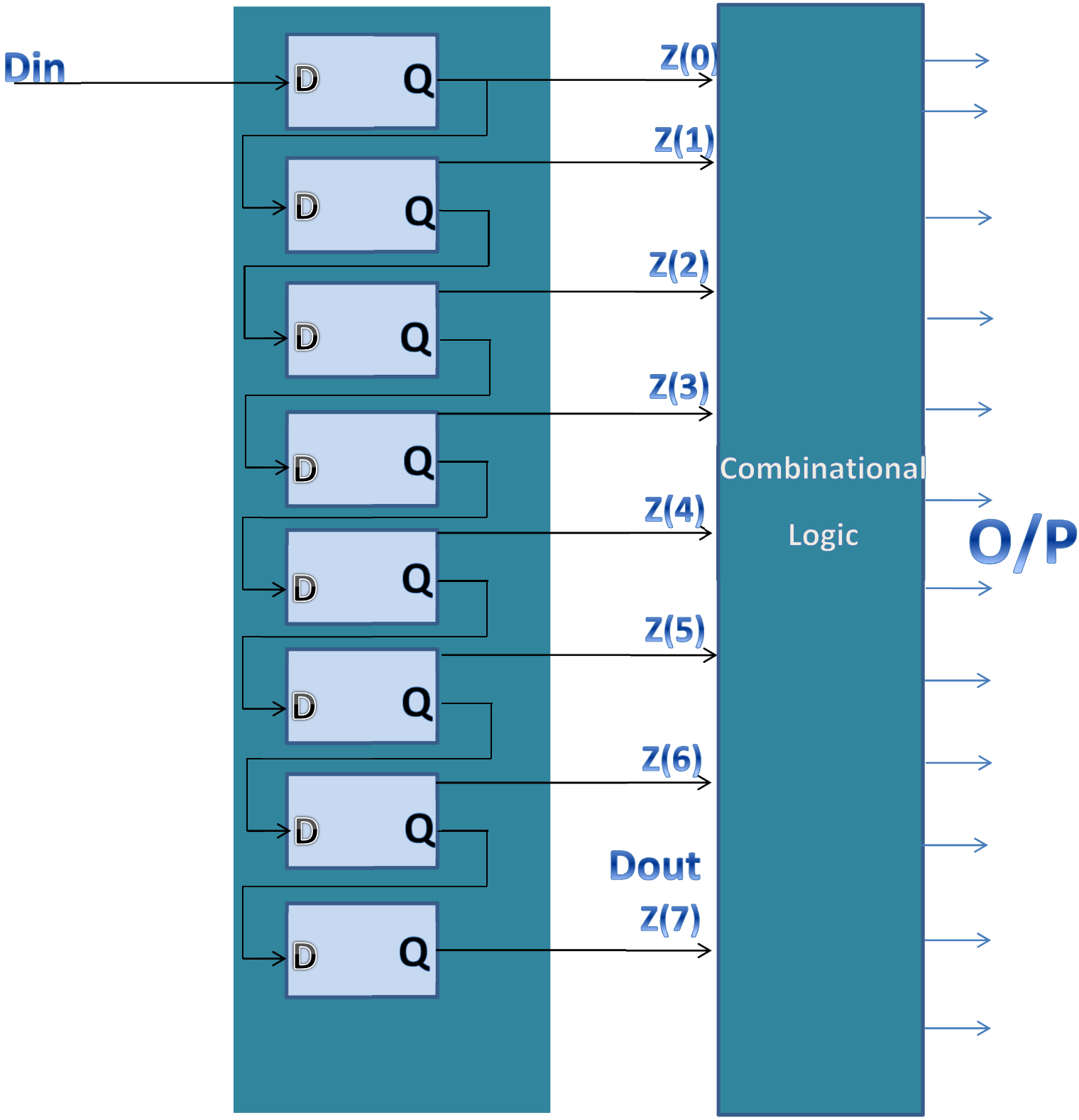
These outputs of the combinational logic go to 16 X 1 mux that has 4 select lines which are controlled by a counter. This counter will select either 2 lines or all the lines of the mux input depending on the output of the combinational logic block. If there are only 2 bit outputs coming through the combinational logic, the counter starts from 0000 and stops at 0001. If there are 9 bits coming out of the combinational logic, the counter starts from 0000 and stops at 1000.

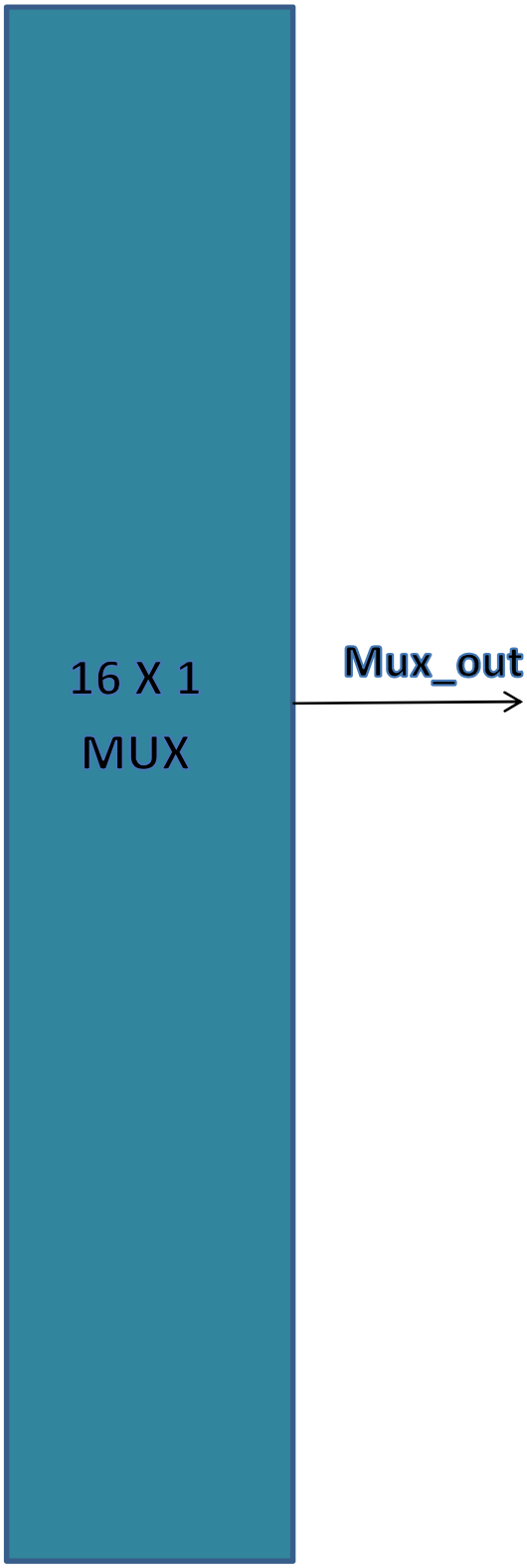
Thus the final outputs are taken out serially.

SHIVA CHAITANYA V C
svishwak@uncc.edu

Advanced system design using HDL
Project-1

Due date: 09/12/2012




16 X 1
MUX
Mux_out