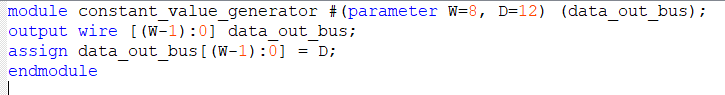
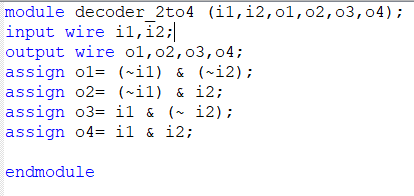
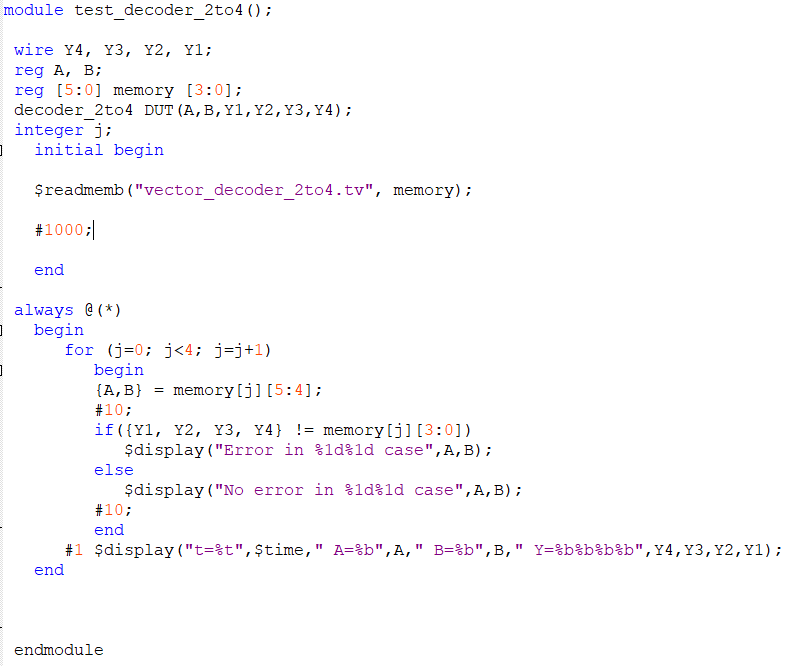
**Constant Value Generator**



**2to4 Decoder**



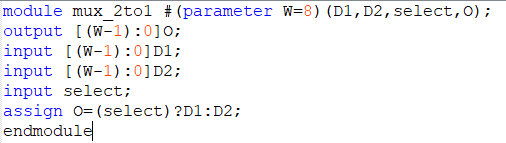
**TB**



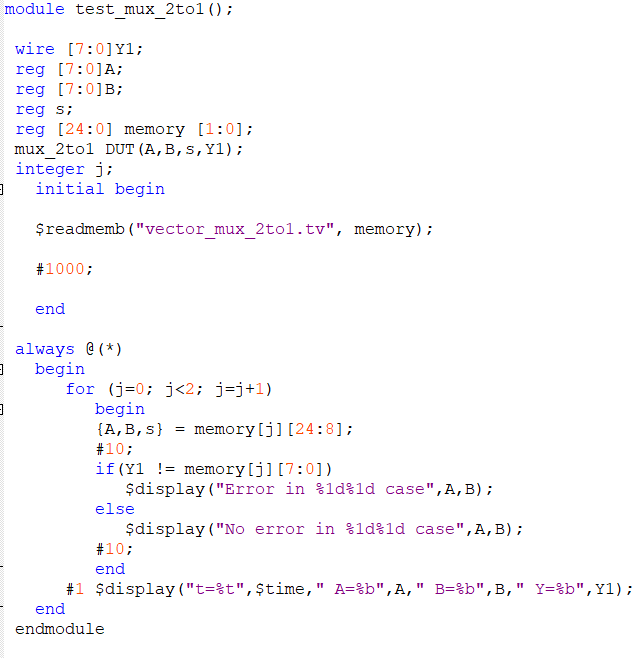
**TV**



**W-bit 2to1 MUX**



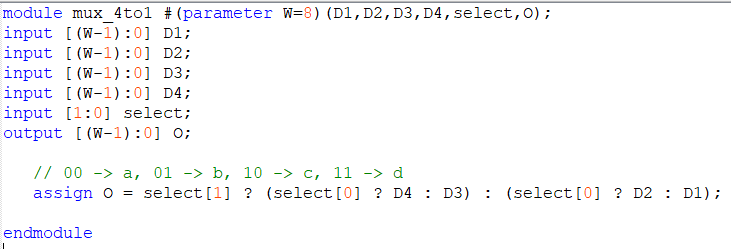
**TB**



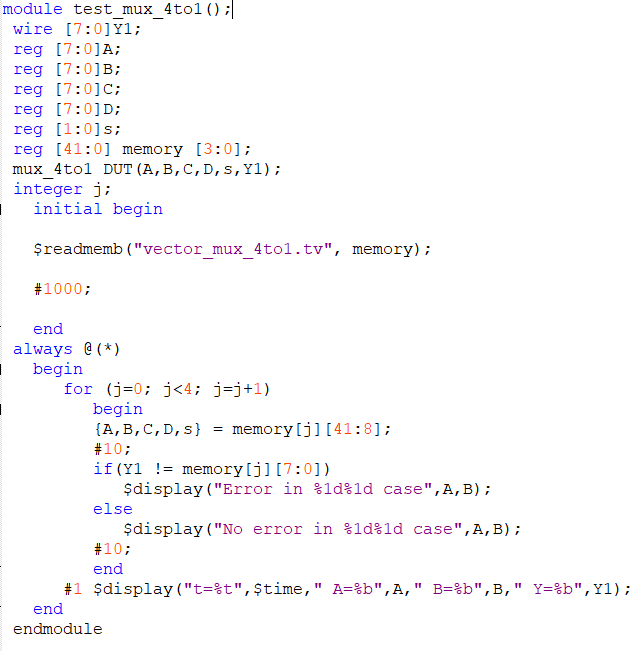
**TV**



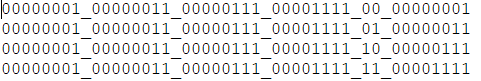
**W-bit 4to1 MUX**



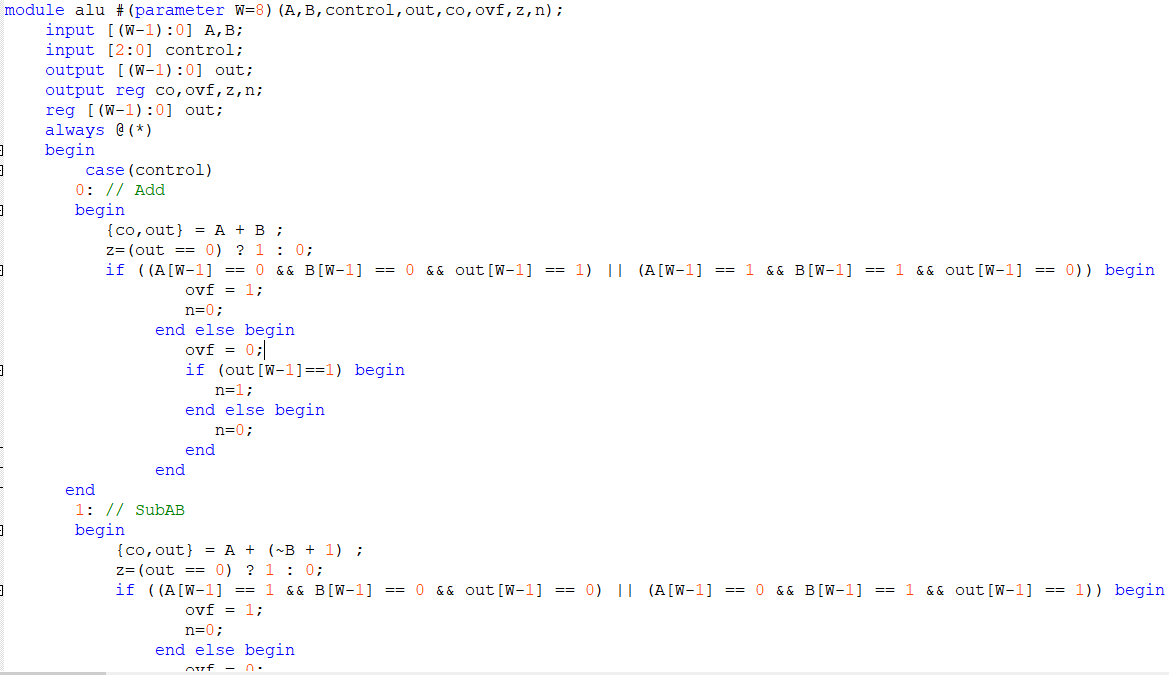
**TB**

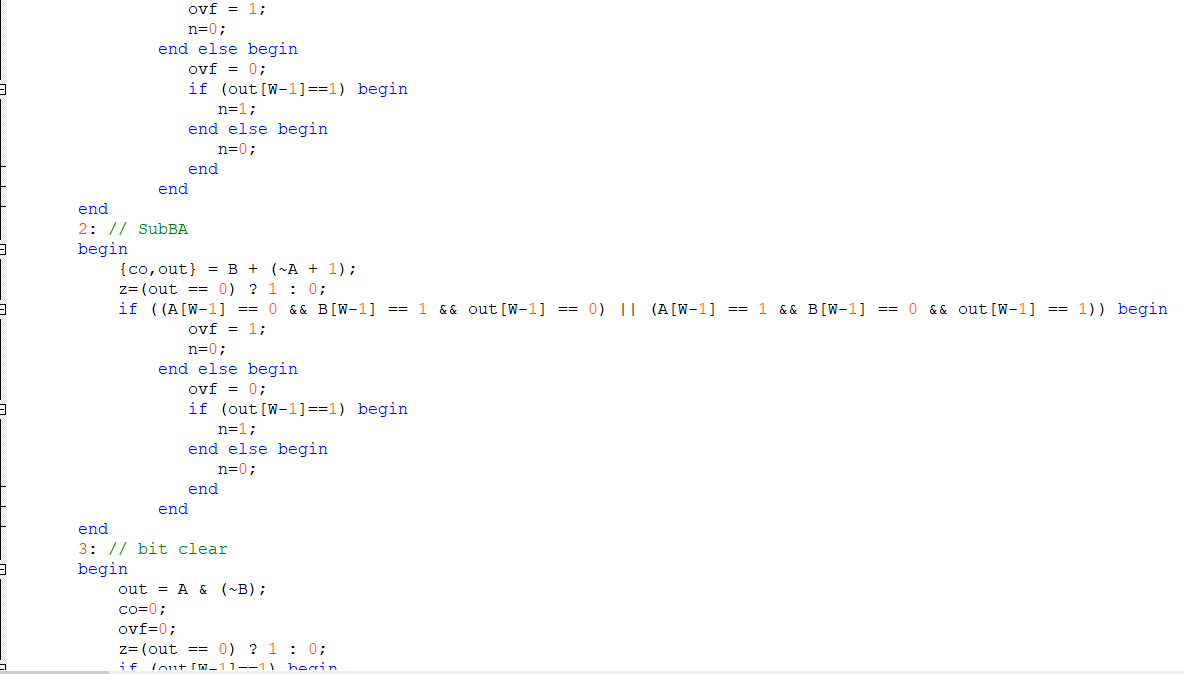


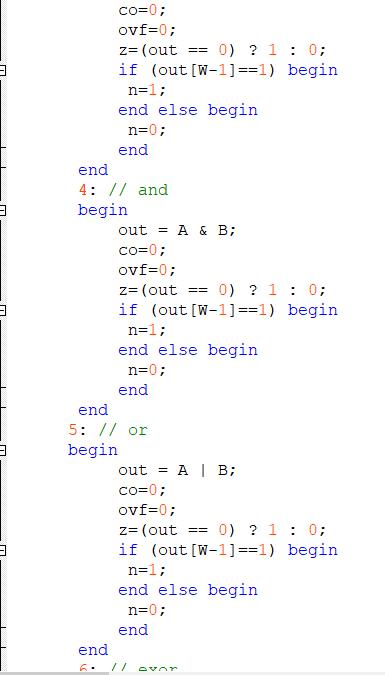
**TV**

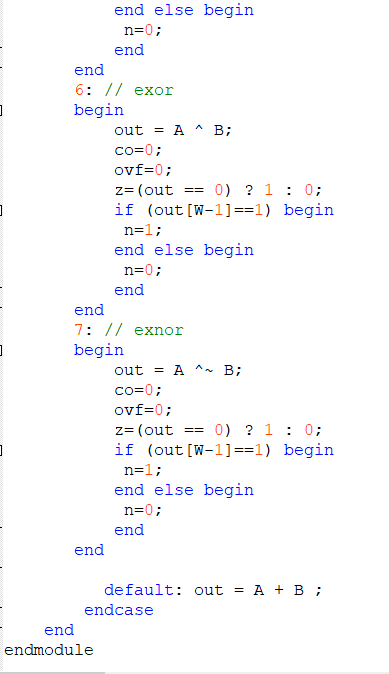


**ALU**









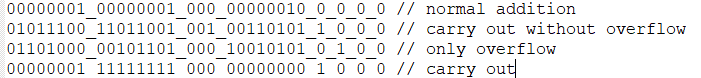
To detect overflow I ANDED the cases which cause overflow. They are:

(Wth bit of inputs 0 ANDED Wth bit of output 1) OR (Wth bit of inputs 1 ANDED Wth bit of output 0)

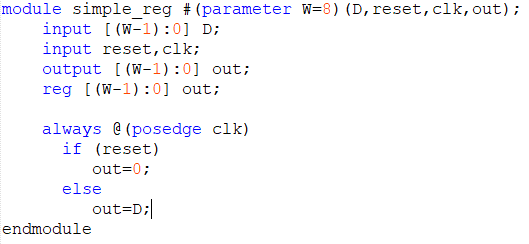
**TB**



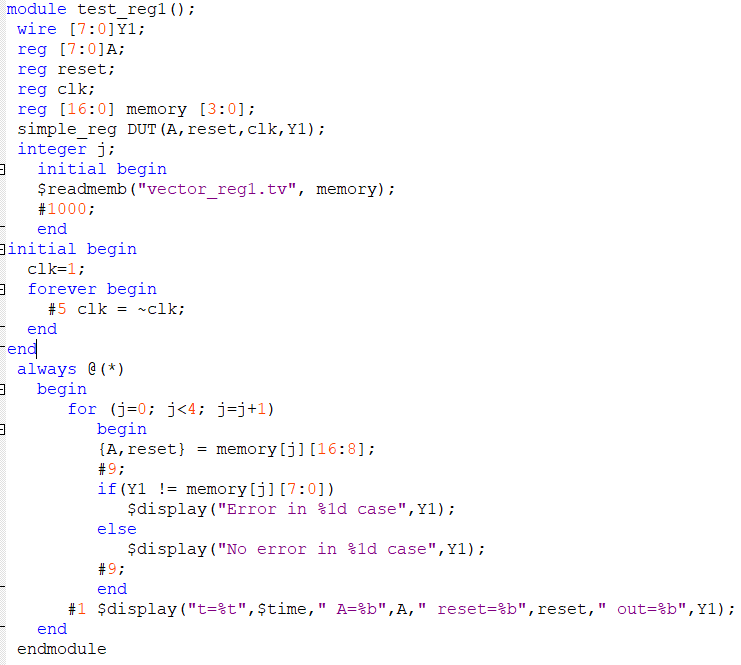
**TV**



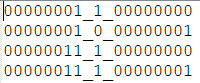
**Simple Register with synchronous reset**



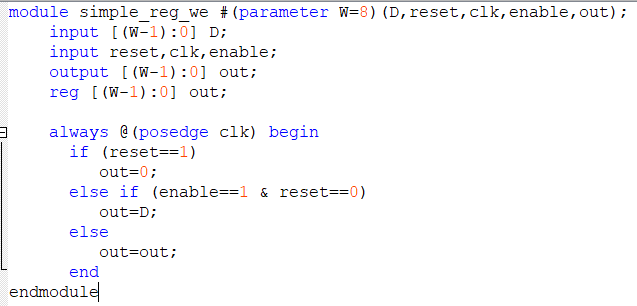
**TB**



**TV**



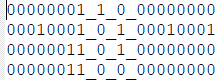
**Register with synchronous reset and write enable**



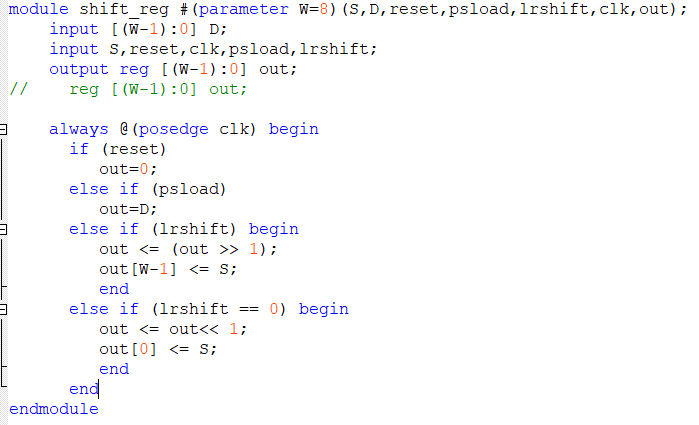
**TB**



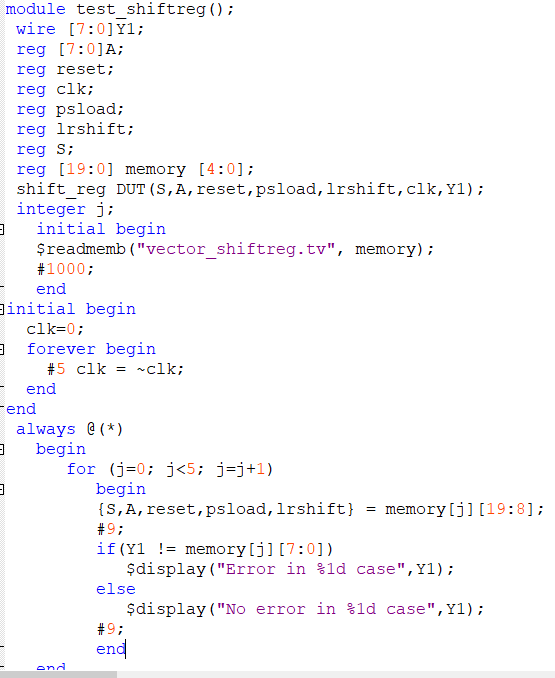
**TV**



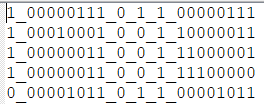
**Shift Register with parallel and serial load**



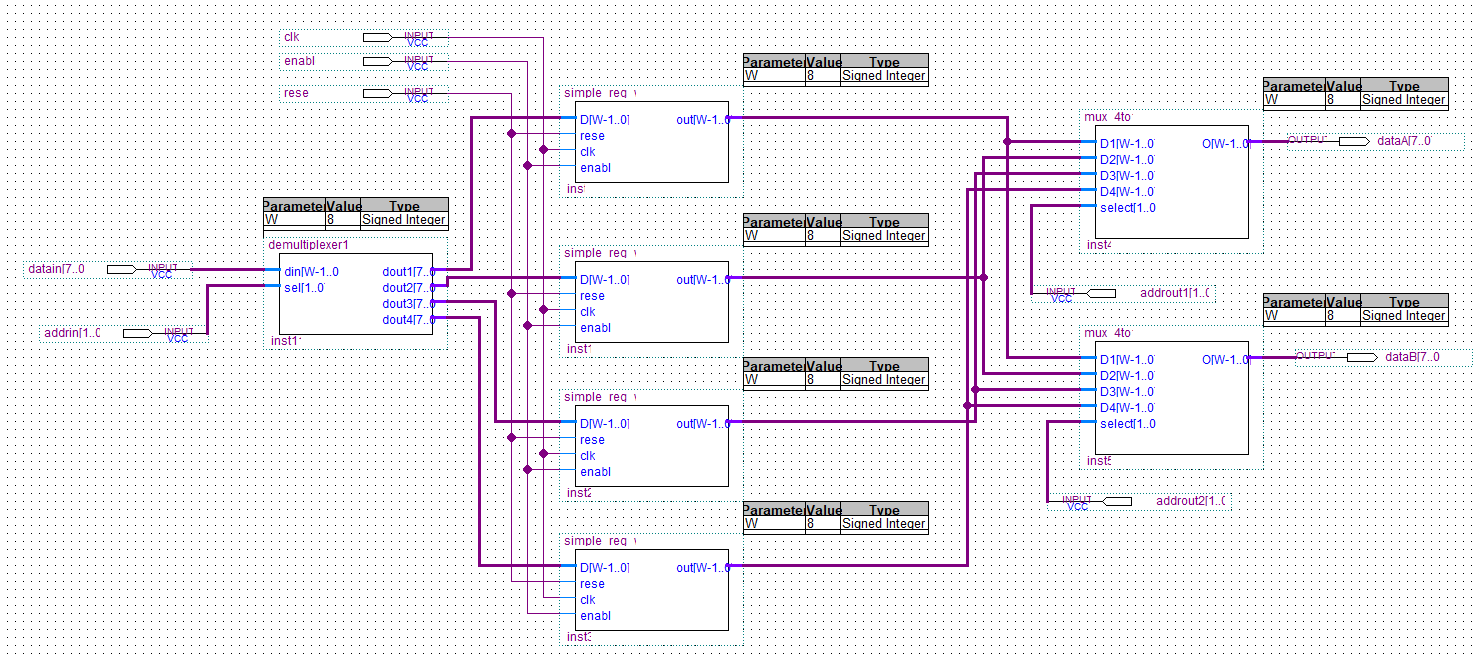
**TB**

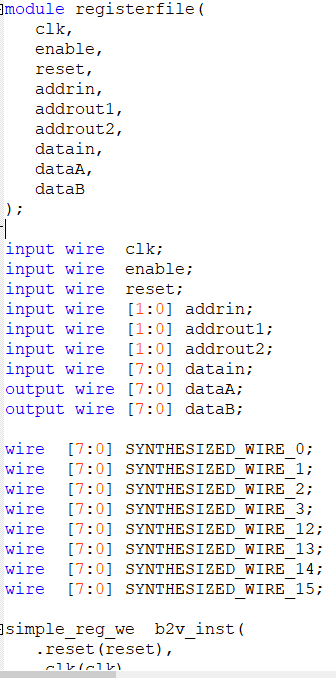


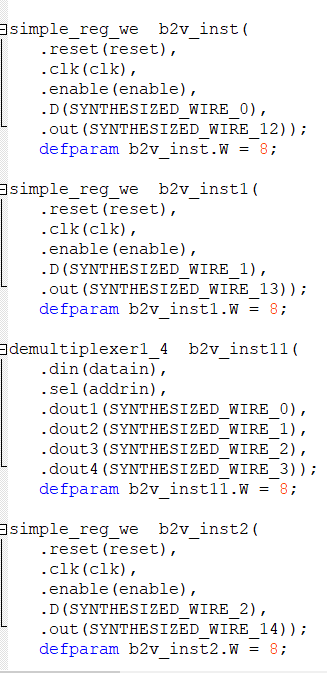
**TV**

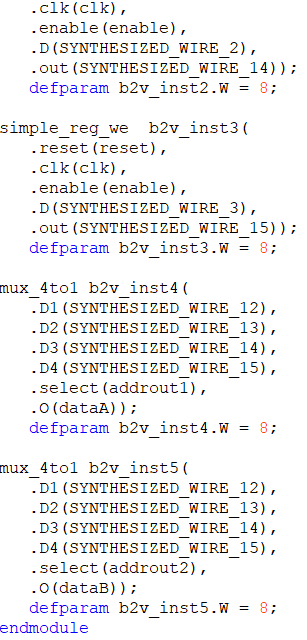


**Register** **File**

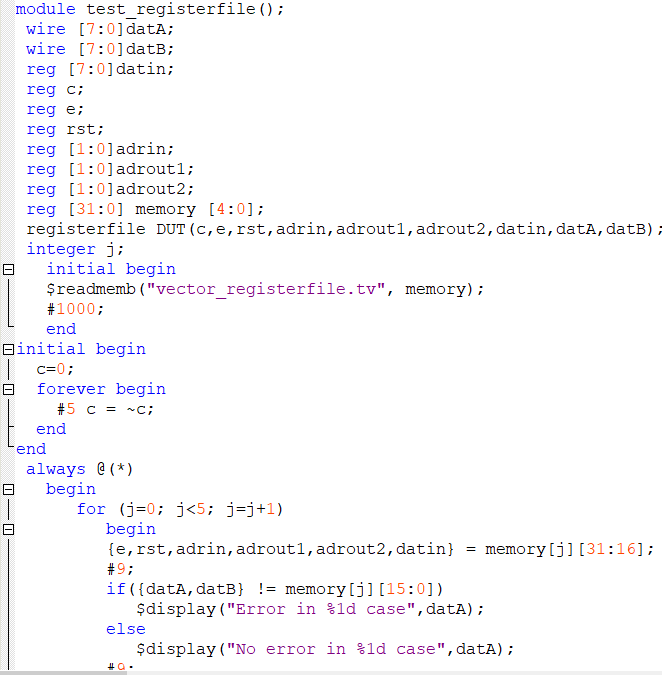


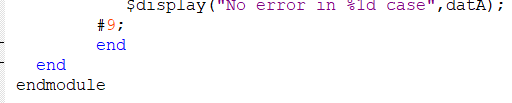




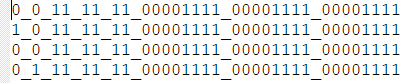


**TB**

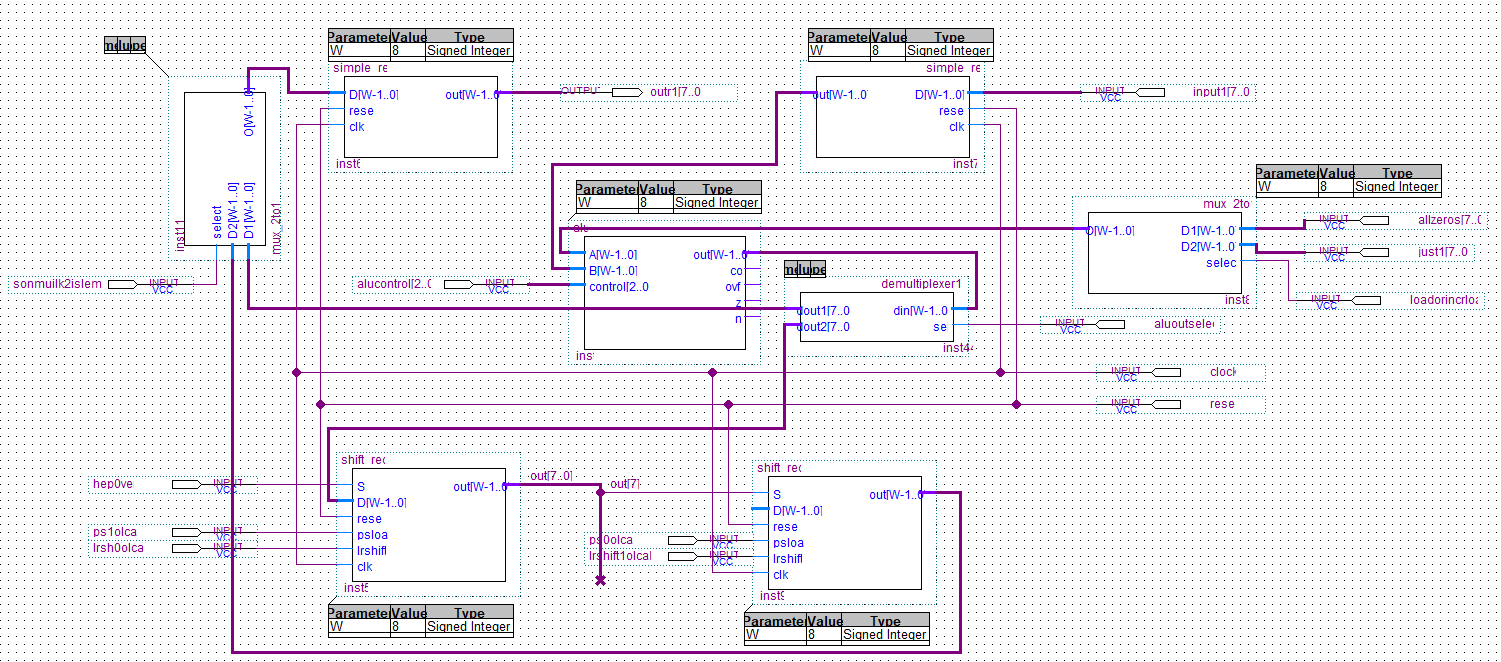




**TV**



**Datapath Design for an Architecture**



In my design I have 3 control signals. They are one-bit signals. One signal is selecting the ALUs A input to have a 0 or 1. The other control signal forwards the data coming from ALU to Acc or R1. The last signal is selecting R1s data input. ALU control is always 000 which is add. Acc shifts left, Q shifts right.

Control pin numbers can not be decreased since each are one-bit signals.

Signals are created in every clock cycle.

Clock 1: R0 is loaded with input1[7..0]. Loadorincrload signal is 1. Aluoutselect is 1.

Clock 2: Acc is loaded with data.

Clock 3: Acc is shifted left. Q is shifted right.

Clock 4: Acc is shifted left. Q is shifted right.

Clock 5: Acc is shifted left. Q is shifted right.

Clock 6: Acc is shifted left. Q is shifted right.

Clock 7: Acc is shifted left. Q is shifted right.

Clock 8: Acc is shifted left. Q is shifted right.

Clock 9: Acc is shifted left. Q is shifted right.

Clock 10: Acc is shifted left. Q is shifted right.

Clock 11: Last mux selects the Qs output to write to R1.

I simulated all with ModelSim and got correct results.