Group 1 – Notes about our BST Architecture

BS Cell/Reg

We have four boundary scan cells, and one boundary scan output cell. The boundary scan cells receive a 4-bit vector parallel input, as well as a serial input that is sent through each boundary cell. The resulting outputs result in the serial output “tdo” and parallel output “pout”.

IR Cell/Reg

Four IR cells together in a register. Shifting bits throughout the cells. Sending parallel output to IR Decoder. IR Cell is similar to the BS Cell, but without the rightmost mux in the latch stage.

IR Decoder

Reads the instruction currently in IR Reg and controls the BS or BP cell depending on the instruction.

BP Cell/Reg

The BP cell is built in the same way as the BS-cells, but we have removed the latch stage. The default logic value 0 is hardcoded to be read at the CAPTURE-DR state, given that the current instruction is bypass mode. As the BP reg is made up of a single cell, this is implemented as a single component.

TAP Controller

The TAP controller implements the finite state machine as described in the documentation of SN74BCT8244.

It outputs the following control signals:

- 2 signals to BP-Reg
- 3 signals to BS-Reg
- 3 signals to Override-Reg
- 3 signals to Instruction register.
- 1 signal to the data/instr-multiplexer.

Top Level

Our top level design only contains the two muxes and the other components, no “glue logic”. The Data Mux has three selections, and is controlled by a bus from IR_Decoder.

Core Logic

Takes the pout from the BS Cells and puts them through an AND gate. The result is forwarded to the output BS cell’s pin.

Override Instruction

The override instruction is implemented with an override register, which contains a logic value for each BS cell. The value of the override cell is OR’ed with the instruction signal going into each respective BS cell, and controls the rightmost multiplexer in each BS cell. In capture DR mode, the instruction register will capture “0000”, and in shift DR we will shift values into the override register.

What could be improved?

We could have added the tristate buffer for the outputs, which would also fix the TDO updating on the rising edge and not the falling edge of the clock.