Compute is applying the algorithm, given in the lab manual shift add multiply

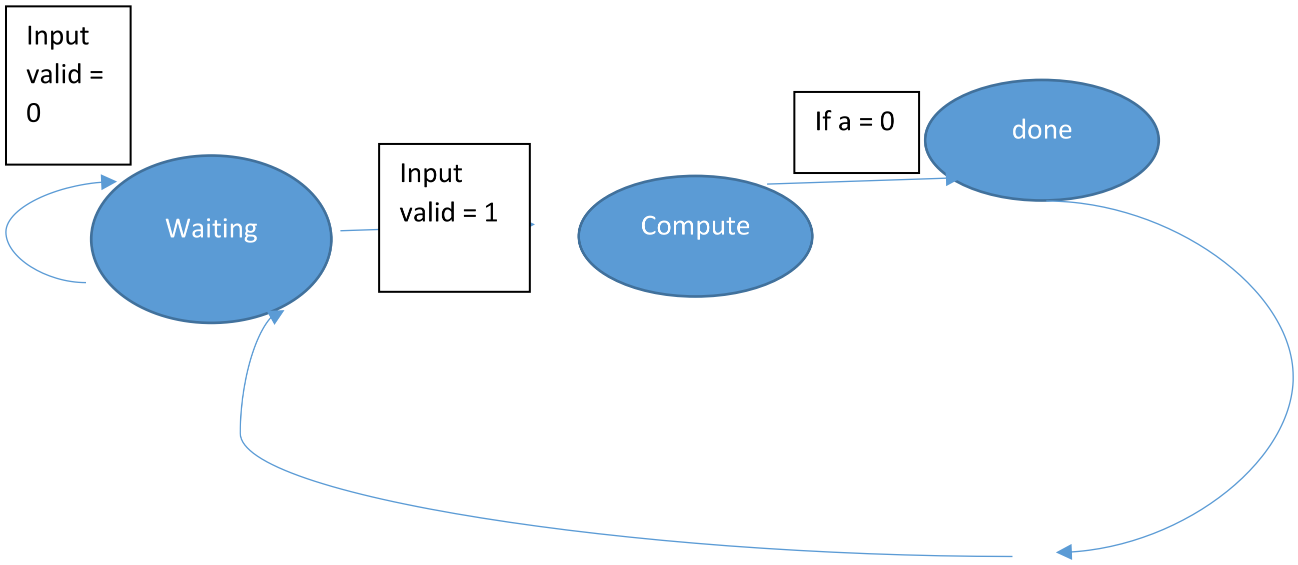
When a goes to zero, move to the done state

Done state sets output valid to high and then goes to waiting state

Waiting state is just checking for a high input valid, as soon as input is valid, it resets output valid to 0 and flops the new values of a and b and goes to compute

We can combine done and waiting in one state, but we separated them to keep it clearer

We changed our state machine from the prelim we had, the prelim had an extra state that was not needed. The extra state was due to having 2 separate states for when a is odd and for when a is even. In this diagram, this is combined in the compute state that applies the whole algorithm until a = 0


Input valid = 0
Waiting
Input valid = 1
Compute
If a = 0
done