During synthesis, the addition of optimization passes such as opt_clean and opt_reduce after ABC optimization has resulted in errors in register assignment

I have encountered an inconsistency issue while using Yosys. During synthesis, we did not use the default synthesis flow of Yosys. After ABC optimization, we applied additional optimization passes such as opt_clean and opt_reduce. The specific synthesis flow we followed is as follows:

read_verilog rtl.v

hierarchy

proc; opt_merge; opt_clean; opt_clean; opt_clean; opt_reduce; opt_expr;

memory; opt_clean; opt_clean; opt_reduce;

techmap; opt_clean; opt_clean; opt_reduce;

abc; opt_demorgan; opt_clean; opt_merge; opt_clean; opt_clean; opt_clean; opt_reduce; opt_expr; opt_lut_ins; opt_expr; opt_expr; opt_clean; opt_dff; opt_lut; opt_reduce; opt_clean; opt_share;

write_verilog syn_yosys.v

We aimed to optimize the logical structure by adding these passes, but these modifications should not affect the code consistency. Using the provided design file (rtl.v), we performed synthesis separately with Yosys's default synthesis flow and our modified synthesis flow with additional passes. Subsequently, we conducted simulation verification using the Iverilog simulator. During simulation, we observed inconsistencies in the output signals (at 0sec). This inconsistency arose from erroneous register assignments in module5 of the design file (starting from line 50 in the attached rtl.v file). The waveform comparison chart is presented below.

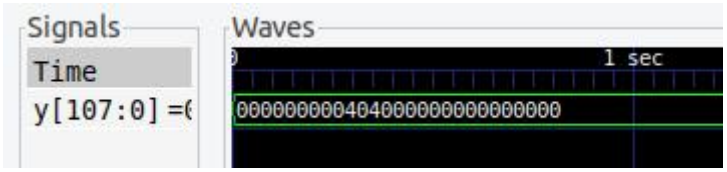

Figure 1: Before Modification

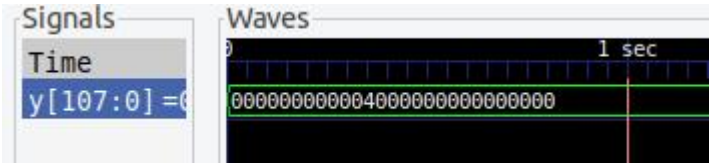

Figure 2: After Modification