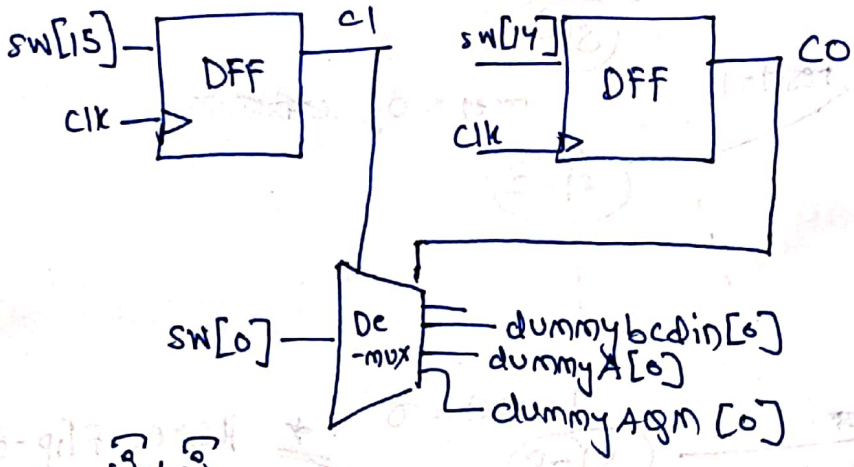
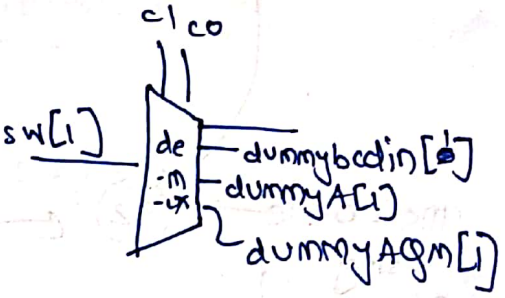
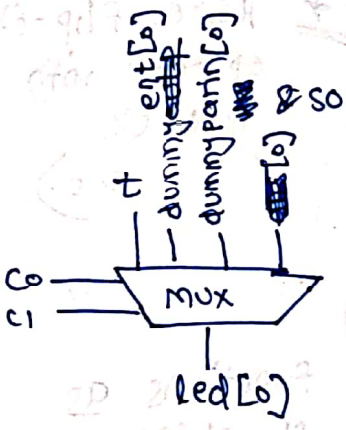
main.v

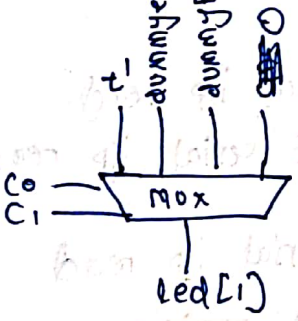
Hansin Ahuja - 2018CSB1094
Paras Goyal - 2018CSB1111





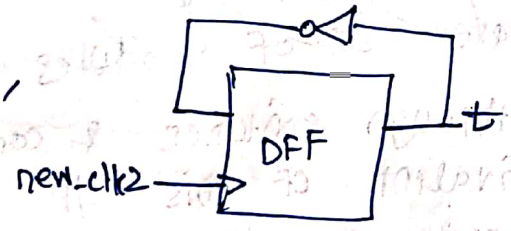
& so on for sw[2....13]





& so on
for led[2...13]

where,

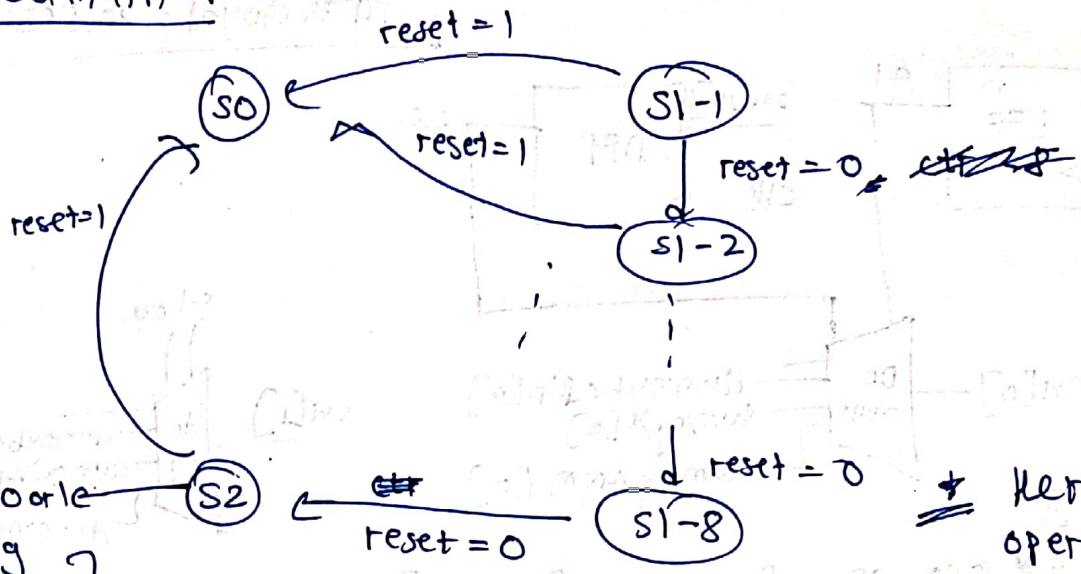


& new_clk2 is a clk of $\leq T = 0.5s$

where i/p = sw[15:14]

& S0: home setting
S1: bcdmain.v
S2: sequence.v
S3: qm.v

bcdmain.v



* Here, flip-flops operate with new-clk (CT = 2s)

Using
- BCD.v
- BCDcheck.v
- & adder.v

S0 : reset state

S1-1 : MSB of serial i/p read

S1-2 : 2nd MSB of serial i/p read

⋮

S1-8 : LSB of serial i/p read

} functions as a shift register

S2 : assigns shift register to a bcd-compater set of modules.

S2 : takes i/p through switches & compares binary equivalent of this i/p with value stored in shift register.

BCDcheck.v

i/p : [7:0] A, [9:0] B, o/p : C

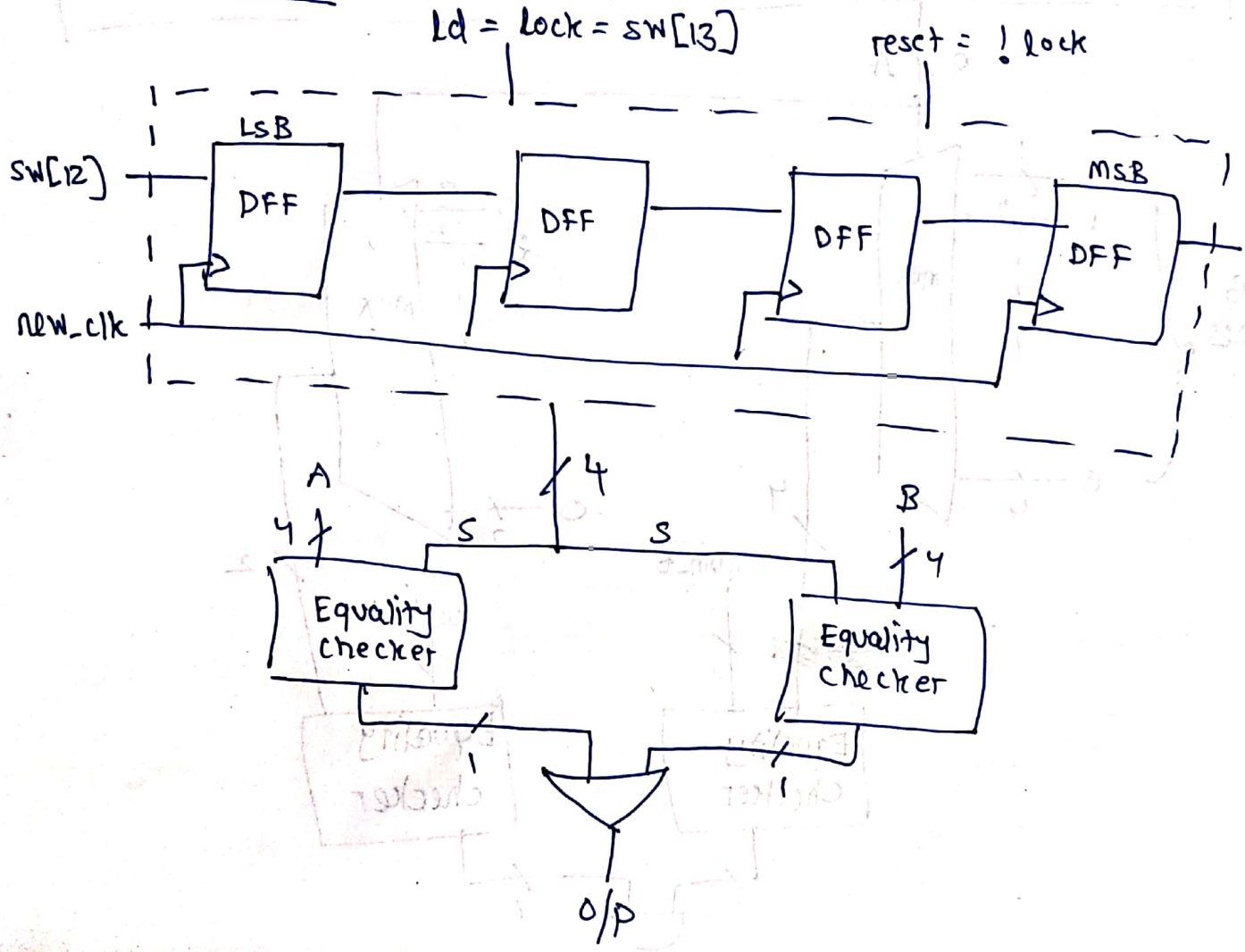
$$C = \begin{cases} 1 & \text{if binary equivalent of } B = A \\ 0 & \text{else} \end{cases}$$

where A = 8 bit binary no.
B = 10 bit BCD no.

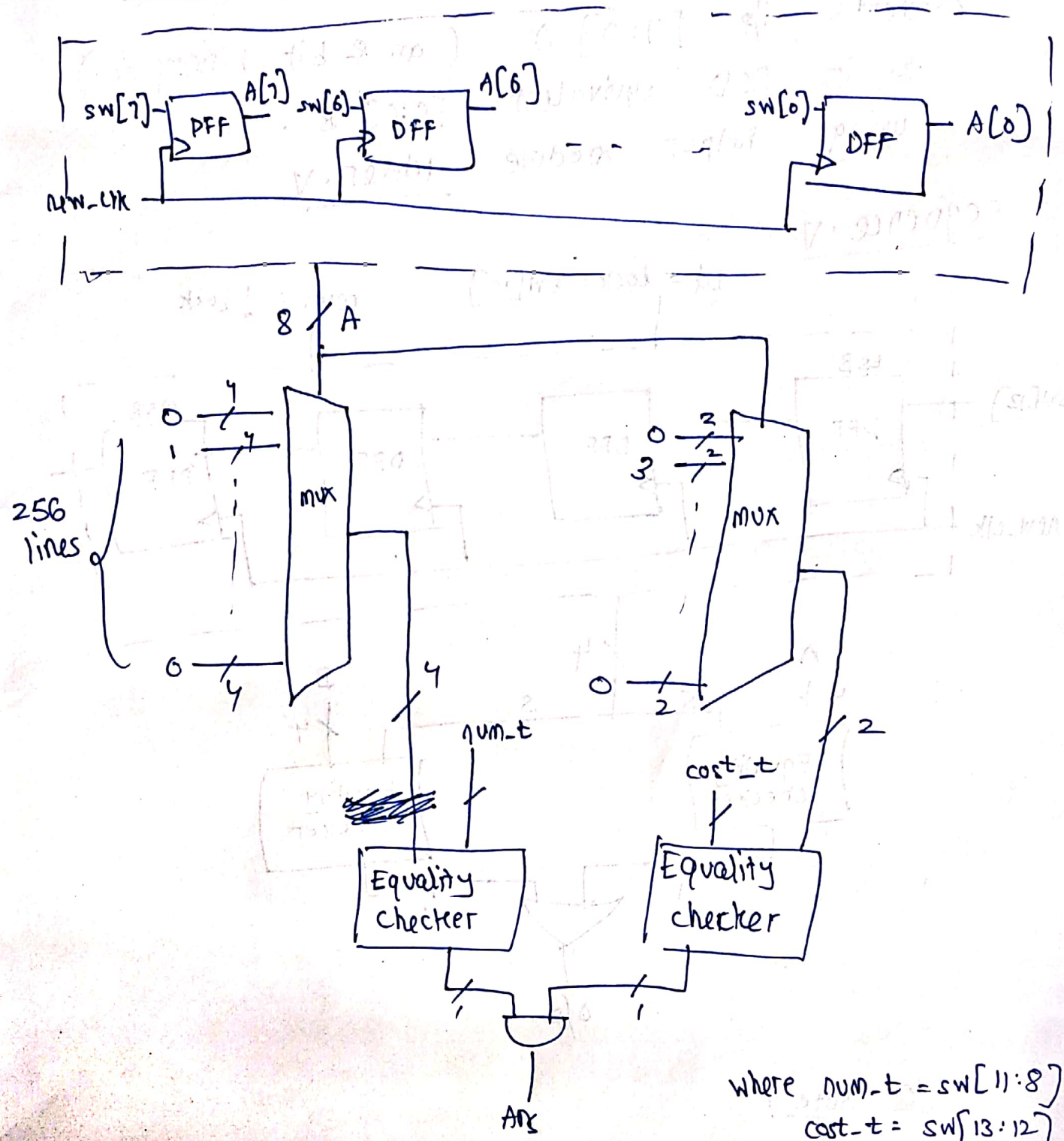
BCD.V

Converts i/p [7:0] A (an 8-bit binary no.) to its BCD equivalent [9:0] B. using helper module adder.V

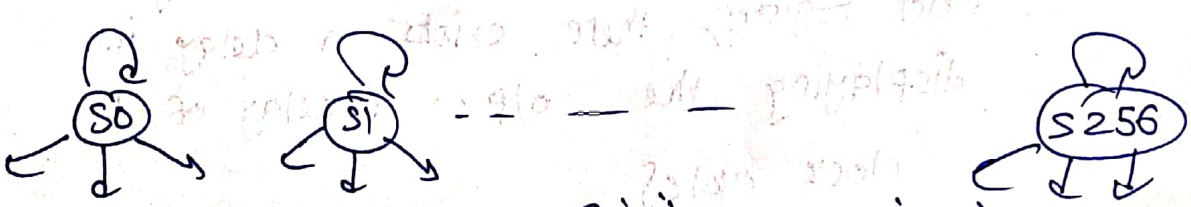
sequence.V



qm.v



where num_t = sw[11:8]
cost_t = sw[13:12]



(similar to main.v)

For state s^i:

i_8 = binary equivalent of i = minterms of 8 variable truth-table

State s_i is responsible for cross-checking the answers of this TT.