EE382m: Homework 5
Multi-level logic optimization

Professor Adnan Aziz

For this assignment, when you are asked to use SIS for optimization, restrict yourself to the following operations: [1]

sweep (performs constant propoagation and inverter minimization)

el (performs elimination)

fx (performs kernel–based common logic extraction)

decomp (performs a decomposition of the network nodes)

For you own edification, you may want to read about/play with the following commands: *simplify, full_simplify, print_delay, reduce_depth, tech_decomp, tech_map.*

You can see the results of each operation by typing *print_stats* at the prompt; you may want to *set auto_exec print_stats* initially, which will run *ps* automatically after each command. The network can be printed using *write_blif, write_eqn, write_pla, print_factor.*

Report the best result (i.e., least number of literals) and the steps you took to get there.

1. (Problem 15, Chapter 10 Hachtel & Somenzi) Perform weak division of $F = abrs + abrt + abd + abe + abu + ghrs + ghrw + ghd + ghe + ghu + dp + ep + rstuw$ by $D = ab + gh$.

   10 marks

2. (Problem 12, Chapter 10 Hachtel & Somenzi) Run `SIS` on the function F given above. You may input the function in either `eqn` or `blif` format, but the output should be in `blif`. (The optimum result contains 23 literals; show your steps.)

   10 marks

3. For the function $F = uwxy + uvxy + tx + tz + rsw + rsv + vwxy$, compute all the level-0 kernels.

   10 marks

[1] Help is available for all these commands; you can save yourself a lot of typing by making use of aliases and paths. Run *alias* at the prompt, also take a look at `adnan/.sisrc`.

4. (Problem 4, Chapter 8 deMicheli's book.) Consider the logic network defined by the following expressions.

$$\begin{aligned} x &= ad' + a'b' + a'd' + bc + bd' + ac \\ y &= a + b \\ z &= a'c' + a'd' + b'd' + e \\ u &= a'c + a'd + b'd + e' \end{aligned}$$

Draw the logic network graph. Outputs are $\{x, y, z, u\}$. Perform the weak division f_x/f_y and show all steps. Substitute y into f_x and redraw the network graph. Compute all kernels and co-kernels of z and u. Extract a multiple-cube subexpression common to f_z and f_u. Show all steps. Redraw the network graph.

25 marks

5. In this problem, you are to devise an 8-bit ALU in Verilog and run it through algebraic optimizations available in SIS.

The ALU should take two 8-bit vectors data arguments, and a two bit control argument, corresponding to the operation to be performed. The output is a single 8-bit vector. The functionality is given in Table 1.

ALU Operation Mode	*Output*
00	*in*1 AND *in*2
01	*in*1 XOR *in*2
10	*in*1 OR *in*2
11	*in*1 PLUS *in*2

Write this design in the Verilog HDL. Compile it to the *blif_mv* format using the `vl2mv` compiler. Read it in `VIS` using the *read_blif_mv* command, and write it out in the *blif* format using the *write_blif* command. (You may want to verify your design works by using the *sim* command in VIS.)

SUN binaries for SIS, vl2mv and VIS are available at

(a) /home/projects/ece/verif/vis-1.2/vis

(b) /home/projects/ece/verif/vis-1.2/vl2mv

(c) /home/projects/logic_synthesis/solaris/bin/sis

50 marks