Ternary Logic Synthesizer

User Guide v.1

****

Contact: syun.lee@postech.ac.kr

****

1. Run Program

- **Download the program in the following link.**
  - csdl.postech.ac.kr/downloads/public/TQMSyn.tar.gz

- **Unzip the file using TAR in Linux.**
  - tar –xvzf TQMSyn.tar.gz

- **There are a binary program and some benches described in the paper (Appendix B).**
  - [IEEE Transactions on Circuits and Systems-I: Regular Papers] “A Logic Synthesis Methodology for Low-Power Ternary Logic Circuits”

- **Run the program with the following command.**
  - ./TQM.exe –input [inputFile] –output [outFile] | tee [logFile]
    - [inputFile] is a benchmark such as ./bench/balanced/0_cons.txt
    - [outFile] is a output file that contains SOP formulation for each network.
    - [logFile] is a log file of the program. (if you don’t want to print, use “>” instead of “| tee”.
  - **Ex)** ./TQM.exe –input ./bench/balanced/0_cons.txt –output ./output/balanced/0_cons.out | tee ./log/balanced/0_cons.log

2. Check Output File

- **Check the output file in output directory.**

- **The Sum-of-Product expressions for each network are displayed.**

- **Match the proper device with the SoP expression, and construct SPICE netlist.**

3. Device Matching

- **There are four networks in our static logic gate based ternary logic circuit as shown in the Figure 1.**

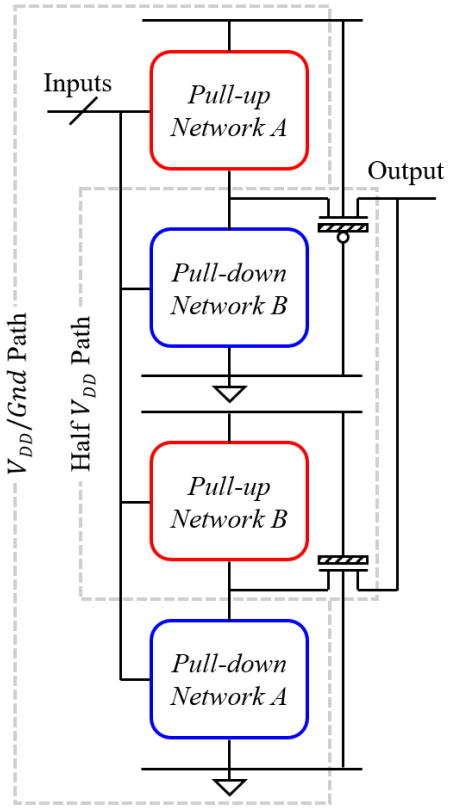
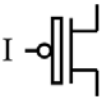
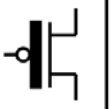
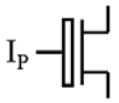
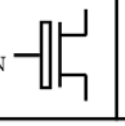
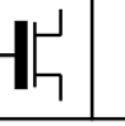
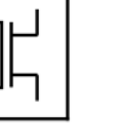


Figure 1.

- **We have got the SoP expression of each network from the output file. As a result, we can get the transistor devices of each network from SoP expression.**

- **The transistor device for SoP expression is based on Figure 2.**

Input	I_0+I_1	I_0	I_1+I_2	I_2
Pull-up Network				
Pull-down Network				

(n,m)	D_{CNT} *[nm]*	$\lvert V_t \rvert$ *[V]*	*N type*	*P type*
(17,0)	1.331	0.323		
(13,0)	1.017	0.428		
(8,0)	0.626	0.687		

Figure 2.

- **As mentioned in the paper, the product term will be connected in series, and the summation term will be connected in parallel.**
  - Some summation terms can be matched with only one device, not parallel connection.
    - Some summation terms indicate (I_0+I_1) and (I_1+I_2) as shown in the Figure 2.

4. Example

- **Example is based on benchmark “0_cons”.**
  - The truth table (function) of “0_cons” is as Figure 3.

Input		Output
A	B	0_cons
-	-	-
-	0	0
-	+	0
0	-	0
0	0	0
0	+	0
+	-	0
+	0	0
+	+	+

Figure 3.

- **./TQM.exe –input ./bench/balanced/0_cons.txt –output ./output/balanced/0_cons.out | tee ./log/balanced/0_cons.log**

- **From “SoP expressions” in output file to “transistor device connections”**
  - SoP for half-V_{DD} pull-up network : $(A_1+A_2) + (B_1+B_2)$

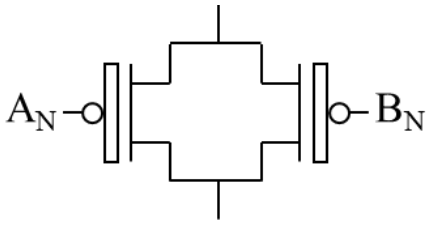


Figure 4.

- SoP for half-V_{DD} pull-down network : $(A_0+A_1) + (B_0+B_1)$

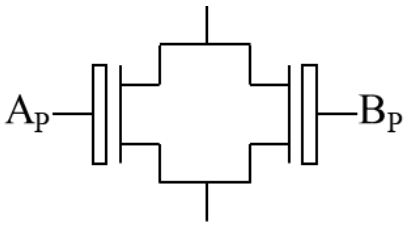


Figure 5.

- SoP for V_{DD}/Gnd pull-up network : A_2B_2

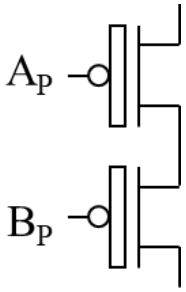


Figure 6.

- SoP for V_{DD}/Gnd pull-up network : A_0B_0

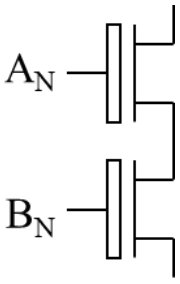


Figure 7.

- **The final transistor-level schematic of ternary logic gate of “0_cons”. You can generate SPICE code based on the schematic for SPICE simulation.**

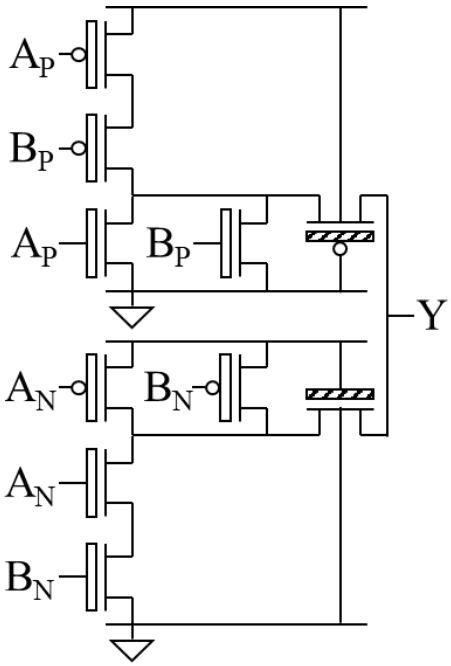


Figure 8.