Chapter 4

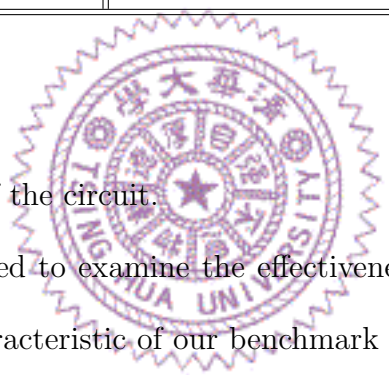
Experimental Results

Our power reduction algorithm is implemented in C language. The experimental process proceeds as follows. First of all, circuit are synthesized with low Vth library by *DesignCompiler*. After the circuits are synthesized, we obtain the critical paths of each circuits using *PrimeTime*. This critical timing will be used as timing constraints. Then, we change Vth from low to high and form a new circuit. Next, we use *PrimeTime* again to get the output capacitance of cells. Besides, we randomly generate some input patterns which are stored in *VCD* file. By reading such *VCD* file, *PrimePower* is used to report the toggle rate of every cell.

With the information mentioned above, we execute our program and output a *Verilog* file using cells from high Vth and low Vth cells from *TSMC 0.13*μm library. During executing our program, we use *PrimeTime* to iteratively check the timing. Finally, *PrimePower* is used to report the total

Table 4.1: Circuit Descriptions

Cir.	CN	Characteristics
TOP	463	An Alarm Clock
MAC	2425	Multiplier and Accumulator
AVG	6361	Average Number Calculator
GCC	8204	Gravity Center Calculator
RSA	14815	Asymmetric Crypto-Processor
AES	16824	Advanced Encryption Core



power consumption of the circuit.

Six circuits are used to examine the effectiveness of our algorithm. Table 4.1 shows the characteristic of our benchmark set. The columns labeled **CN**, and **Characteristics** are the cell number of the design, and the characteristics of the design, respectively.

The following tables show the power consumption and reduction after performing our algorithm for different cases where the fractions of active time and idle time are different. Let the fraction of active time and idle time be α. α is computed by

$$\alpha = \frac{active\_time}{active\_time + idle\_time} * 100\%$$

Table 4.2, Table 4.3 and Table 4.4 are the results that our algorithm is performed under the situation that α is 100%, α is 50%, and α is 10%, respectively. The columns labeled P_O, P_A and P_B are the original power consumption, power consumption from executing **Step2+Step3.1** and power consumption from executing **Step3.2**, respectively. **Step2**, **Step3.1** and **Step3.2** are steps shown in Figure 3.1 and Figure 3.2, and are described in Chapter 3.

The column labeled **Original** is the results of the circuit with low Vth cells mapped by **DesignCompiler**. The column labeled **A (Step2 + Step3.1)** is the results of the circuit that we change the Vth of cell from low to high and repair the timing violation problem on critical path. The column labeled **B (Step3.2)** is the results of the circuit that we utilize the remaining slack of circuit on non-critical path to minimize the power consumption of cell. The column labeled **Total** is the results after performing our algorithm.

The columns labeled Red_A , Red_B and Red_{total} are the power reduction rate and is computed by

$$Red_A = \frac{(P_O - P_A)}{P_O} * 100\%$$

$$Red_B = \frac{(P_O - P_B)}{P_O} * 100\% - Red_A$$

$$Red_{total} = \frac{(P_O - P_{total})}{P_O} * 100\%$$

On the average, the power reduction rate of **A (Step2 + Step3.1)** of Table 4.2, Table 4.3 and Table 4.4 are 8.84%, 11.15% and 21.06%, respectively. The power reduction rate of **B (Step3.2)** of Table 4.2, Table 4.3 and Table 4.4 are 7.42%, 7.38% and 5.64%, respectively. Total power reduction rate of Table 4.2, Table 4.3 and Table 4.4 are 16.26%, 18.53% and 26.70%, respectively.

Table 4.5 shows the circuit timing information. The column labeled **New_T** represents the circuit timing after performing our algorithm. The column labeled **T_penalty** represents the timing penalty caused by our algorithm and is calculated as

$$T\_penalty = 1 - \frac{(New\_T)}{Original\_T}$$

It represents the timing relation between original timing and the resultant timing. If **T_penalty** is less than *0*, it means that timing becomes better than the original circuit. From the table, we can see that our algorithm do

Table 4.2: The fraction of active time (α) is 100%

Circuit	Original	A (Step2 + Step3.1)		B (Step3.2)		Total
	P_O(W)	P_A(W)	Red_A	P_B(W)	Red_B	Red_{total}
TOP	4.13E-04	3.83E-04	7.10%	3.63E-04	4.85%	11.95%
MAC	9.70E-04	8.43E-04	13.06%	7.90E-04	5.50%	18.56%
AVG	1.75E-03	1.70E-03	2.86%	1.65E-03	3.07%	5.75%
GCC	8.06E-04	7.64E-04	5.19%	7.53E-04	1.29%	6.48%
RSA	3.49E-03	2.97E-03	14.82%	2.12E-03	24.38%	39.20%
AES	1.59E-02	1.43E-02	10.18%	1.34E-02	5.42%	15.60%
Average	_	_	8.84%	_	7.42 %	16.26%

Table 4.3: The fraction of active time (α) is 50%

Circuit	Original	A (Step2 + Step3.1)		B (Step3.2)		Total
	P_O(W)	P_A(W)	Red_A	P_B(W)	Red_B	Red_{total}
TOP	2.09E-04	1.91E-04	8.39%	1.79E-04	5.85%	14.24%
MAC	5.04E-04	4.24E-04	15.89%	3.97E-04	5.19%	21.09%
AVG	9.16E-04	8.62E-04	5.87%	8.35E-04	2.92%	8.79%
GCC	4.51E-04	4.18E-04	7.39%	4.12E-04	1.30%	8.69%
RSA	1.84E-03	1.50E-03	18.16%	1.08E-03	23.24%	41.40%
AES	8.07E-03	7.16E-03	11.19%	6.70E-03	5.79%	16.99%
Average	_	_	11.15%	_	7.38%	18.53%

Table 4.4: The fraction of active time (α) is 10%

Circuit	Original P_O(W)	A (Step2 + Step3.1) P_A(W)	A (Step2 + Step3.1) Red_A	B (Step3.2) P_B(W)	B (Step3.2) Red_B	Total Red_{total}
TOP	4.59E-05	3.90E-05	15.05%	3.71E-05	4.07%	19.12%
MAC	1.31E-04	8.92E-05	31.75%	8.37E-05	4.22%	35.97%
AVG	2.48E-04	2.18E-04	12.23%	2.11E-04	2.61%	14.84%
GCC	1.68E-04	1.44E-04	14.36%	1.42E-04	1.10%	15.46%
RSA	5.13E-04	3.27E-04	36.35%	2.39E-04	1.71%	53.50%
AES	1.80E-03	1.50E-03	16.63%	1.41E-03	4.69%	21.33%
Average	_	_	21.06%	_	5.64%	26.70%

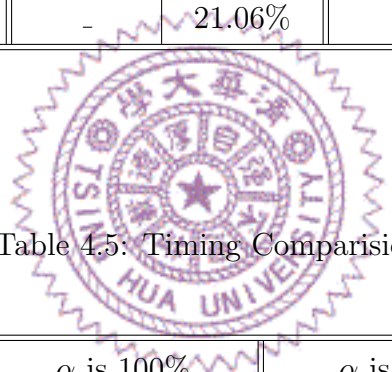


Table 4.5: Timing Comparisions

Circuit	Original T	α is 100% T'	α is 100% T_penalty	α is 50% T'	α is 50% T_penalty	α is 10% T'	α is 10% T_penalty
TOP	1.43	1.37	-4.2%	1.39	-2.8%	1.40	-2.1%
MAC	3.30	3.33	0.8%	3.33	0.8%	3.33	0.8%
AVG	23.78	23.13	-2.7%	23.46	-1.3%	23.54	-1.0%
GCC	26.30	26.65	1.3%	26.73	1.6%	26.34	0.2%
RSA	10.00	10.08	0.8%	10.03	0.3%	10.10	1.0%
AES	2.29	2.21	-3.5%	2.27	-0.9%	2.27	-0.9%

not cause serious timing violation problem. The largest timing penalty is 1.6% (Circuit GCC , $\alpha = 50\%$)

From the table above, we found that our algorithm can effectively decrease the power consumption with tolerable timing penalty.