

# Power / Delay Estimation of Auto Generated Circuits

K. P. Ghosh, S.Subramanian

Department of Electrical Engineering, IIT, Bombay, India

## Abstract

The goal of the research project was to characterize RTL generated by AHIR flow. We had taken a set of examples, used AHIR to generate RTL, mapped the RTL to layout, and finally performed power, area and delay estimation for the above examples. For all our circuit examples we have used OSU Standard Cell library for TSMC 0.18 technology and Synopsis Design Compiler to perform synthesis. The netlist thus obtained is used as an input to Cadence SOC Encounter to generate the layout. The layout being closest to the actual hardware is most suitable for estimation of power consumed in the design for a particular input sequence. Hence, we present an alternate approach of implementing memory in our circuits as Blackboxes. Cacti 3.2[1] was used to model the power and delay of the memory blackboxes. Five examples viz. A5, AES, Red Black trees, Linpack and FFT were selected, six memory architectures were explored for each and results are summarized in final section.

## I. INTRODUCTION

AHIR provides a platform to describe hardware in a high level programming language like C and translates it to its equivalent hardware description. When we use this description of hardware, followed by the RTL synthesis and then its layout, we would like to characterise the hardware thus generated to meet certain physical and performance criteria. Area, power and delay are important criteria that mark the quality of the hardware. The current approach followed by us to obtain layout from a VHDL description is as shown in the figure 1. Memories in the hardware have been treated as blackboxes with suitable power and area models obtained from Cacti 3.2.

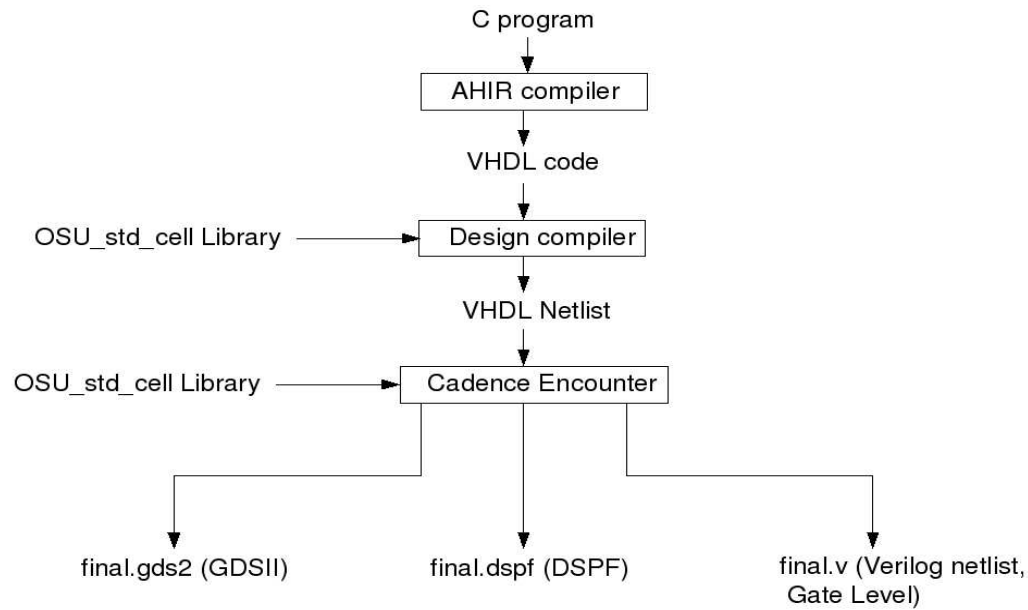


Fig. 1. Flow describing VHDL Description to Layout

At each level, a different tool is used for a specific purpose. At first, the VHDL code is simulated and checked for its functionality using ModelSim 6.3a. The next step is to perform synthesis and obtain a gate

TABLE I  
ARCHITECTURE CHOICES FOR EACH EXAMPLE

ARCHITECTURE CHOICES		MEMORY BANK CHOICE		
		1	2	4
Example	Memory Size	Base Bank Address Width		
A5	16	4	4	4
LINPACK	16K	12	12	12
R-B TREES	16k	12	12	12
FFT	512	8	8	8
AES	1024	8	8	8

level netlist. For all our circuit examples we have used OSU Standard Cell library for TSMC 0.18 technology and Synopsis Design Compiler (DC) to perform synthesis. The netlist thus obtained is used as an input to Cadence SOC Encounter to generate the layout. The layout being closest to the actual hardware is most suitable for estimation of power consumed in the design for a particular input sequence. Power has been calculated using Cadence SOC Encounter.

The memory could either be built up as a single bank or as a combination of multiple banks. The later would aid parallelism in the circuit. Also, the memory could be pipelined. We have tried to explore 6 different memory architectures for different examples. This would help us compare the performance of the circuits with multiple banks versus single banks, pipelined as opposed to non-pipelined memory.

## II. MEMORY MODELLING

For encounter to estimate power, delay and area of the entire hardware, the memory black boxes must have the respective information. We use Cacti 3.2 to estimate power and delay in the memory black box. The area is modelled using [2].

$$\text{area(in mm}^2\text{)} = (0.001) * (\text{tech})^{2.07} * (\text{bits})^{0.9} * (\text{port})^{0.7} + 0.0048$$

The values of power, area and delay obtained from the above models are used to generate the necessary information for the blackboxes in timing library file (TLF) and library exchange file (LEF) formats. The vital timing information is also obtained from delay models. To describe a memory element as a black box, we instruct Synopsys DC to turn off its synthesis when it encounters the architecture for the memory. Thus for synopsys, the memory is an entity definition with well defined input output ports but a blank architecture. Similarly, Cadence SOC encounter is made to infer black boxes whenever a memory component is instantiated from the TLF and LEF files.

## III. POWER ESTIMATION

Using the final gate level verilog netlist provided by encounter we perform a post -layout simulation and dump all the signal information to a vcd file. This vcd file which now contains all the switching activity information is used by SOC Encounter to generate a power report. One is also required to mention the top level entity name for which power is to be estimated. For examples like linpack, Redblack which required huge size of memories and also which took a long time to simulate, a sampling technique was followed, in order to reduce the size of VCD files (which was around 25Gb). In this technique, the final post layout netlist is simulated and the switching activities are captured at 10 random intervals of time. It was observed that, when power is calculated using these samples, the variance in power estimates for all the samples is in the orders of 1e-5.

## IV. RESULTS

Five examples viz. A5, AES, Red Black trees, Linpack and FFT were selected. The memory architecture choices are shown table 1:

For each of memory architecture choices, mux and demux degree of 2 and infinity was selected. The range of frequencies tried for each architecture is shown in table 2:

TABLE II  
RANGE OF FREQUENCIES FOR EACH ARCHITECTURE

	ARCHITECTURE CHOICES					
	1x2	1x0	2x0	2x2	4x2	4x0
Example	Frequency of Operation (in MHz)					
A5	71.42	71.42	83.33	71.42	83.33	71.43
LINPACK	45.45	41.67	41.67	41.67	41.67	38.4
R-B TREES	41.67 and 62.5	41.67 and 55.56	71.42	50	71.42	38.36
FFT	41.66	41.66	41.66	45.45	45.45	45.45
AES	45.45 and 71.42	45.45 and 71.42	71.42	71.42	71.42	45.45

Note : Architecture choices are in the form [memory bank x mux/demux degree]. For eg. 4 x 2 indicates an architecture with number of memory banks as 4 and mux/demux degree of 2. 0 indicates a mux/demux degree of infinity.

### A. Reports

The energy, delay and area estimates for each example with their respective 6 architectures are as follows:

1) A5/I: The energy, delay and area estimates for each architecture / frequency combination is given in table 3. Figure 2 and 3 shows the Power-Delay and Area-Delay plot for A5/I.

TABLE III  
ENERGY / DELAY / AREA FOR EACH ARCHITECTURE/FREQUENCY COMBINATION

A5/I	1x2	1x0	2x0	2x2	4x2	4x0
Energy (nJ)	25.26	20.44	29.484	22.078	34.56	23.436
Delay (ns)	308	280	252	266	240	252
Area (mm <sup>2</sup> )	1.45	1.3	1.77	1.52	2.3	1.9

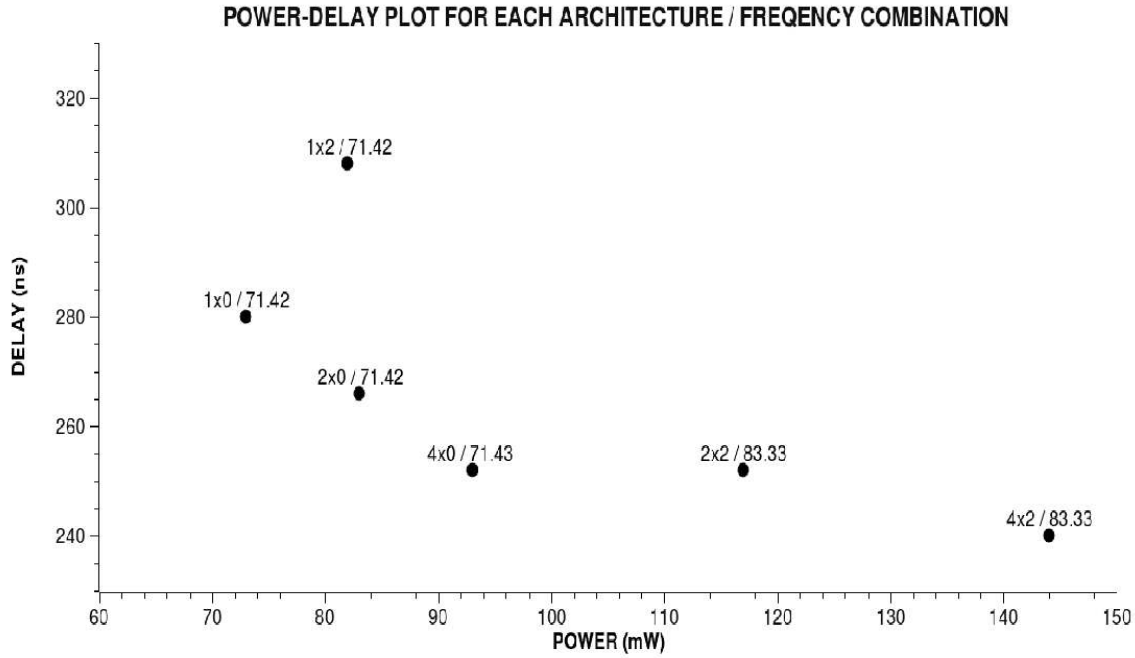


Fig. 2. Power-Delay plot for A5/I

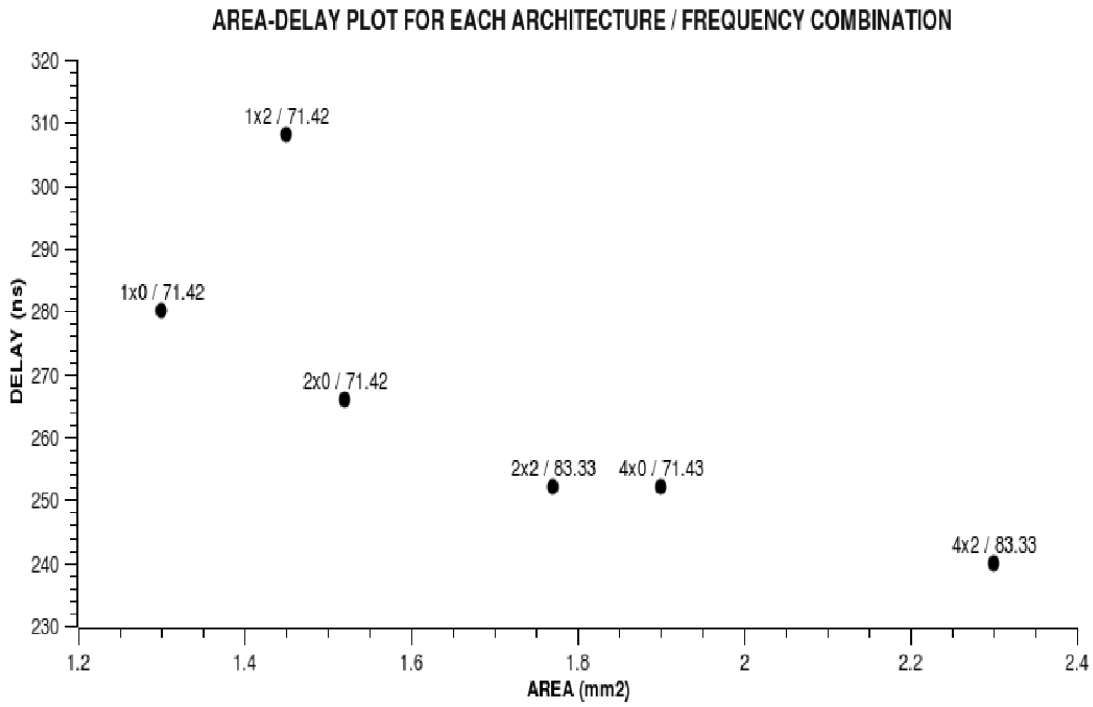


Fig. 3. Area-Delay plot for A5/1

2) *LINPACK*: The energy, delay and area estimates for each architecture / frequency combination is given in table 4. Figure 4 and 5 shows the Power-Delay and Area-Delay plot for Linpack.

TABLE IV  
ENERGY / DELAY / AREA FOR EACH ARCHITECTURE/FREQUENCY COMBINATION

LINPACK	1x2	1x0	2x0	2x2	4x2	4x0
Energy (mJ)	18.92	11.47	22.75	10.28	29.53	11.07
Delay (ms)	56.48	43.65	55.63	37.68	51.64	38.71
Area (mm <sup>2</sup> )	27.5	25.5	30.8	27	36.2	28.9

3) *R-B TREES*: The energy, delay and area estimates for each architecture / frequency combination is given in table 5. Figure 6 and 7 shows the Power-Delay and Area-Delay plot for Red-Black Trees.

TABLE V  
ENERGY / DELAY / AREA FOR EACH ARCHITECTURE/FREQUENCY COMBINATION

R-B TREES	1x2	1x0	2x0	2x2	4x2	4x0
Energy (mJ)	3.935 and 7.062	1.511 and 2.54	8.07	2.851	11.47	3.23
Delay (ms)	18.74 and 12.5	9.88 and 7.41	10.9	8.17	10.87	10.57
Area (mm <sup>2</sup> )	19 and 23	18 and 21	26	22	32	25

4) *FFT*: The energy, delay and area estimates for each architecture / frequency combination is given in table 6. Figure 8 and 9 shows the Power-Delay and Area-Delay plot for FFT.

5) *AES*: The energy, delay and area estimates for each architecture / frequency combination is given in table 7. Figure 10 and 11 shows the Power-Delay and Area-Delay plot for AES.

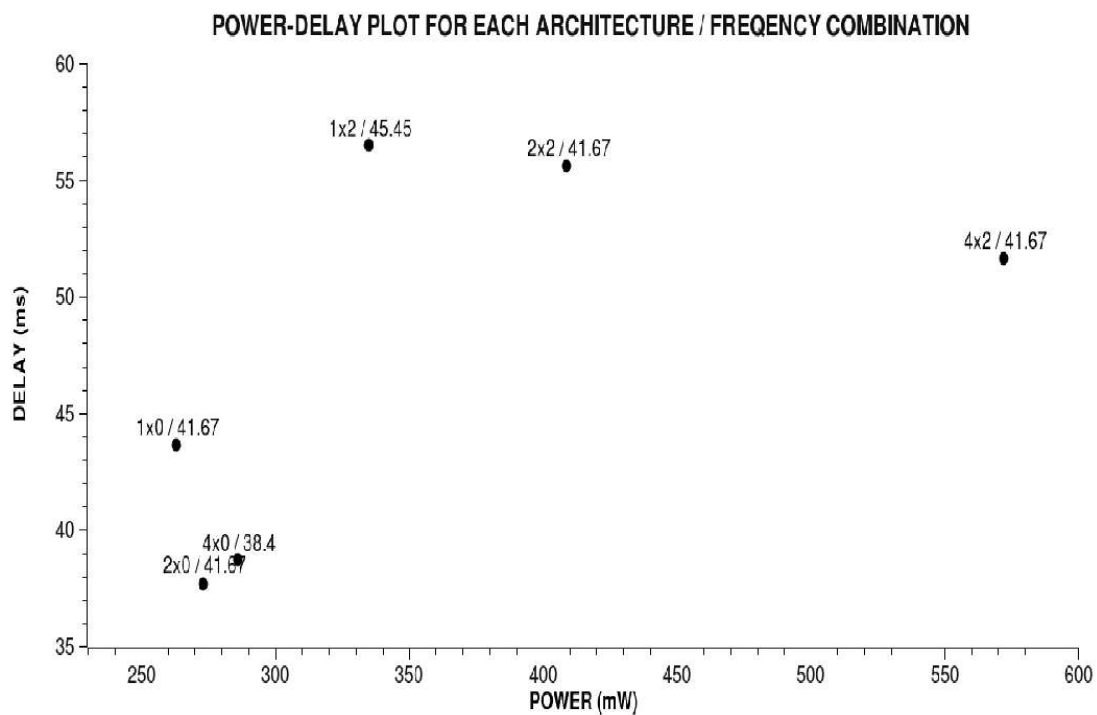


Fig. 4. Power-Delay plot for LINPACK

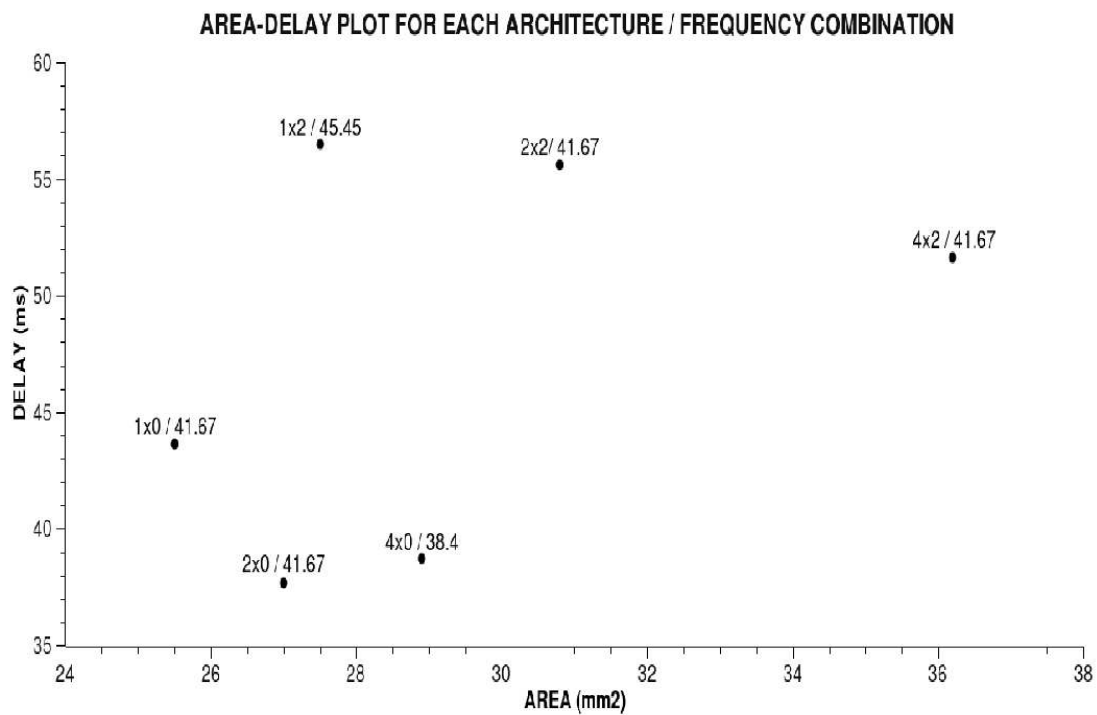


Fig. 5. Area-Delay plot for LINPACK

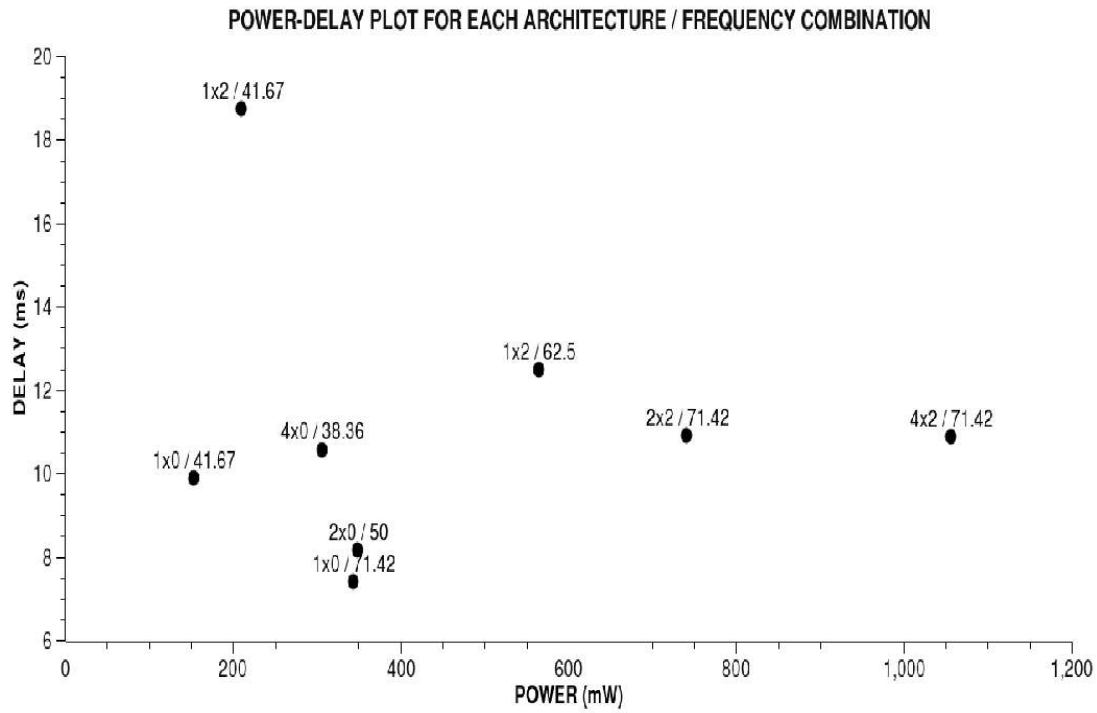


Fig. 6. Power-Delay plot for RED-BLACK TREES

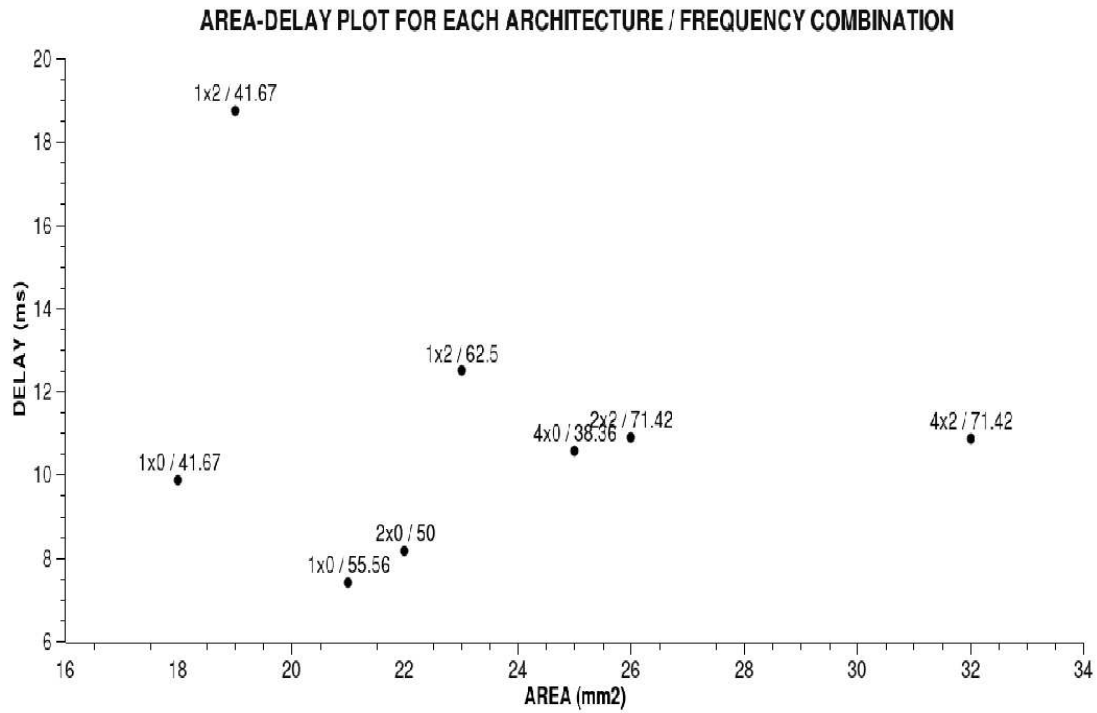


Fig. 7. Area-Delay plot for RED-BLACK TREES

## V. CONCLUSION

From the results, tabulated earlier, we can conclude the following: The circuits generated showed very limited parallelism in memory access patterns. Also, there is no performance boost in pipelined memory as opposed

TABLE VI  
ENERGY / DELAY / AREA FOR EACH ARCHITECTURE/FREQUENCY COMBINATION

FFT	1x2	1x0	2x0	2x2	4x2	4x0
Energy (uJ)	24.65	14.71	31.04	16.7	41.38	18.55
Delay (us)	203.792	154.88	189.27	141.606	172.453	140.55
Area (mm <sup>2</sup> )	5.4	5.07	6.6	5.7	9.2	6.7

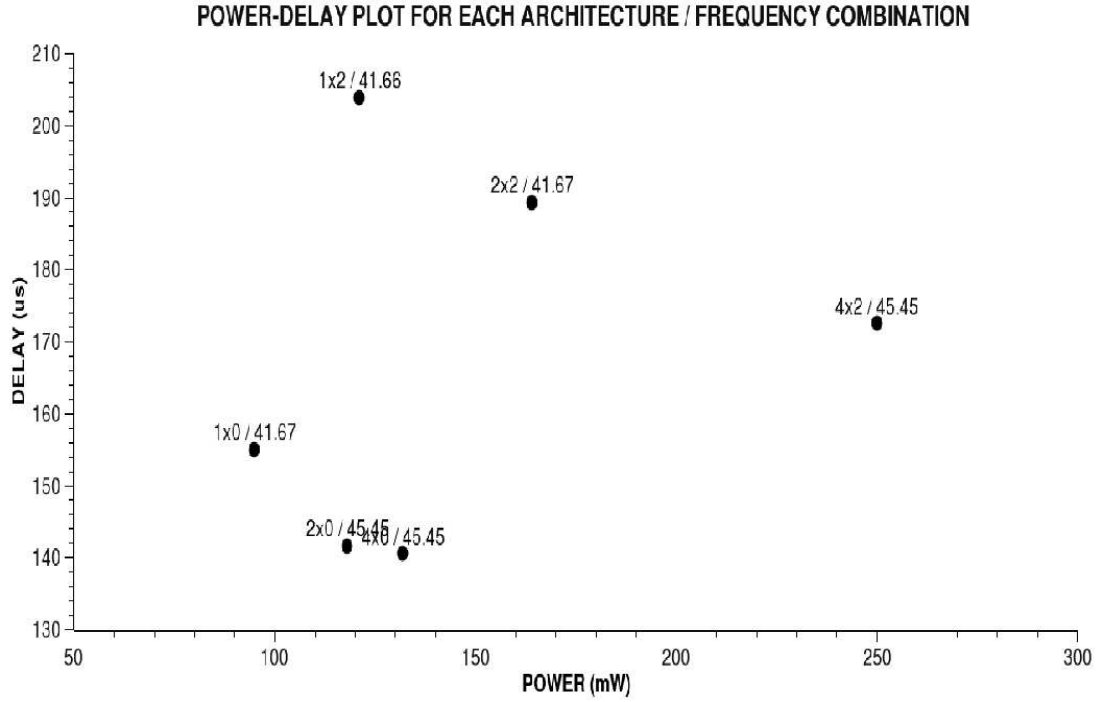


Fig. 8. Power-Delay plot for FFT

TABLE VII  
ENERGY / DELAY / AREA FOR EACH ARCHITECTURE/FREQUENCY COMBINATION

AES	1x2	1x0	2x0	2x2	4x2	4x0
Energy (uJ)	310.97 and 386.03	147.38 and 144.664	560.07	157.58	846.28	177.5
Delay (ms)	1.21 and 0.769	0.67 and 0.43	0.762	0.418	0.759	0.655
Area (mm <sup>2</sup> )	7.67 and 9.03	6.63 and 6.5	13	7.5	20.25	9.4

to the non-pipelined memory. We can observe that there's no significant improvement in delay as we increase the number of banks. Also another trend that we observe is that circuits are more energy efficient at higher frequencies.

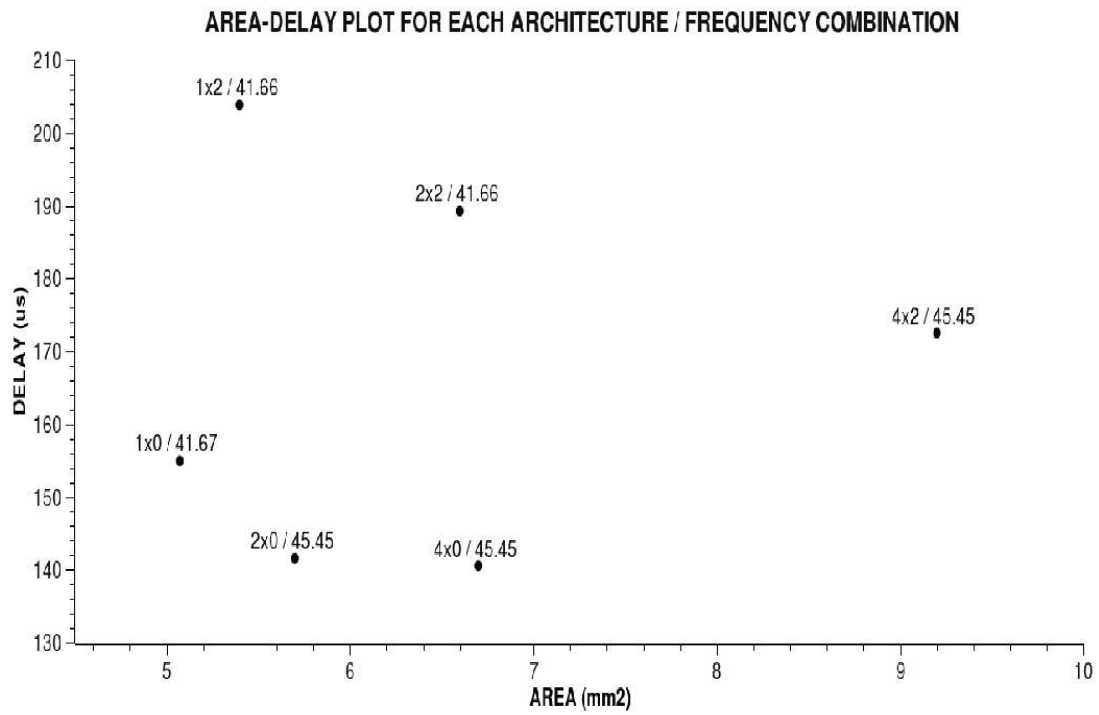


Fig. 9. Area-Delay plot for FFT

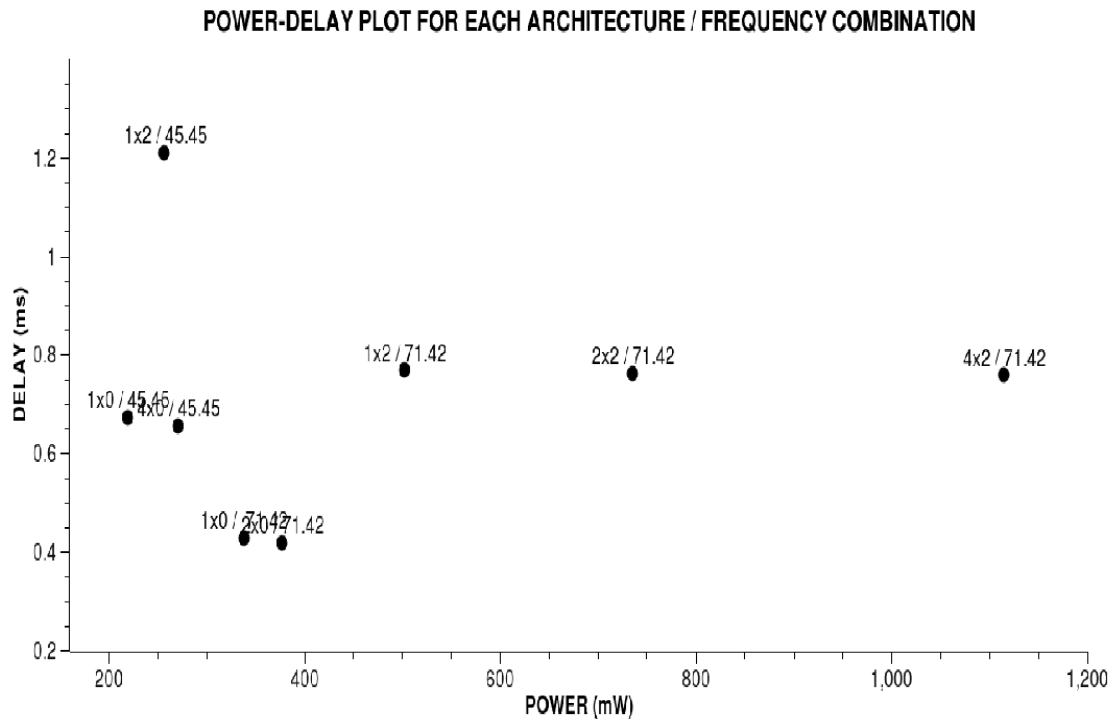


Fig. 10. Power-Delay plot for AES



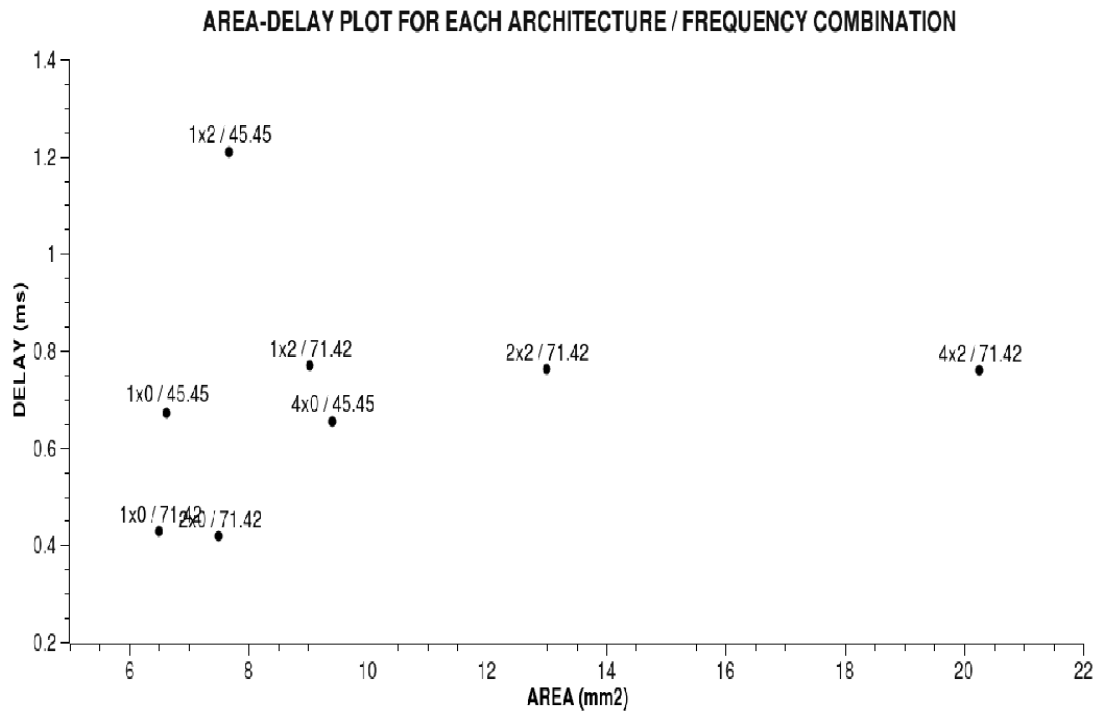


Fig. 11. Area-Delay plot for AES

#### REFERENCES

- [1] Premkishore Shivakumar, Norman P. Jouppi. Cacti 3.0 : An Integrated Cache Timing, Power, and Area model, **Western Research Laboratory**, 250 University Avenue Palo Alto, California 94301 USA.
- [2] Banit Agrawal, Timothy Sherwood. Guiding Architectural SRAM models, **Published in International Conference of Computer Design (ICCD) 2006**, Department of Computer Science, University of California, Santa Barbara.