Investigation and implementation of data transmission look-ahead D flip-flops

Yuan Yongyi
LITH-isy-EX-ET-0289-2004
Linköping, 2004.5.18
Investigation and implementation of data transmission look-ahead D flip-flops

Bachelor thesis
Division of Electronic Systems
Department of Electrical Engineering
Linköping Institute of Technology

Yuan Yongyi

LITH-ISY-EX-ET-0289-2004

Supervisor: Weidong Li
Examiner: Lars Wanhammar

Linköping, 2004.5.18
Språk
Language
Svenska/Swedish
X Engelska/English

Rapporttyp
Report category
Licentiatavhandling
X Examensarbete
C-uppsats
D-uppsats
Övrig rapport

ISBN

IRSN
LITH-ISY-EX-ET-0289-2004

Serietitel och serienummer
Title of series, numbering

ISSN

Title
Investigation and implementation of data transmission look-ahead D flip-flops

Author
Yuan Yongyi

Sammanfattning
Abstract
This thesis investigates four D flip-flops with data transmission look-ahead circuits. Based on logical effort and power-delay products to resize all the transistor widths along the critical path in 0.18µm CMOS technology. The main goal is to verify and proof this kind of circuits can be used when the input data have low switching probabilities. From comparing the average energy consumption between the normal D flip-flops and D flip-flops with look-ahead circuits, D flip-flops with look-ahead circuits consume less power when the data switching activities are low.

Nyckelord
Keyword
power consumption, switch activity, flip-flop, logical effort, power-delay product
Abstract

This thesis investigates four D flip-flops with data transmission look-ahead circuits. Based on logical effort and power-delay products to resize all the transistor widths along the critical path in 0.18\(\mu\)m CMOS technology. The main goal is to verify and proof this kind of circuits can be used when the input data have low switching probabilities. From comparing the average energy consumption between the normal D flip-flops and D flip-flops with look-ahead circuits, D flip-flops with look-ahead circuits consume less power when the data switching activities are low.
I would like to take this opportunity to thank some people who made it possible for me to write this bachelor’s thesis. First I would like to thank my examiner professor Lars Wanharmmar and supervisor Weidong Li for their great support and advices during this project.

I did my thesis work at Electronic Systems division that is part of Electrical Engineering department in Linköping University. Electronic Systems division is focused on design and implementation of signal processing and communication systems, including like methodology for efficient design and implementation of digital signal processing systems with low power consumption, covering computational properties of DSP algorithms, synthesis of application/algorithm-specific architectures, arithmetic, analog and digital filters as well as digital and analog circuits and systems as well as mixed analog/digital circuits. It offered me the necessary facilities as well as a great environment to write my thesis. Hereby I would like to express my gratitude to all members of Electronic System division for giving all the conveniences and help during my thesis work.

Last but not least I give my thanks to my friend Ping An who chose to do the opposition of this thesis.
Contents

Abstract........................................................................................................................... i

Acknowledgements ......................................................................................................... ii

Chapter 1 .......................................................................................................................... 1
1. Introduction .................................................................................................................. 1
2. Sequential logic circuits .............................................................................................. 2
3. Flip-flops ....................................................................................................................... 2
4. Classification of flip-flops ............................................................................................ 2
5. Latches and D flip-flops .............................................................................................. 4
6. Synchronous interconnection ...................................................................................... 5
   6.1 Skew ...................................................................................................................... 5
   6.2 Jitter ...................................................................................................................... 6
       6.2.1 Absolute jitter ............................................................................................... 6
       6.2.2 Cycle-to-cycle jitter .................................................................................... 7
   6.3 Combined Impact of Skew and Jitter: ................................................................. 7
7. Timing metrics for flip-flops ...................................................................................... 7
   7.1 Propagation delay ................................................................................................. 8
   7.2 Setup time ............................................................................................................ 9
   7.3 Hold time ............................................................................................................. 10
8. Energy metrics for flip-flops .................................................................................... 11
   8.1 Power .................................................................................................................. 11
   8.2 Power-delay product (PDP) ............................................................................... 11
   8.3 Energy per transmission .................................................................................... 11
9. Transistor sizing ....................................................................................................... 12
10. Logical effort .............................................................................................................. 12

Chapter 2 .......................................................................................................................... 15
1. Circuit topology ........................................................................................................... 15
   1.1 Transmission gate ............................................................................................... 15
   1.2 Data-transition look-ahead D flip-flops ............................................................... 15
   1.3 Clock-on-demand flip-flop (COD-FF) ............................................................... 16
   1.4 Conditional-capture flip-flop (CCFF) ................................................................. 17
   1.5 Gated TGFF ...................................................................................................... 18
2. Simulation setup ........................................................................................................ 19
   2.1 Testbench .......................................................................................................... 20
   2.2 Testbench for D flip-flops ................................................................................ 21
3. Design procedure for data-transmission look-ahead D flip-Flop............................ 21
4. Design procedure for Conditional-Capture flip-flop (CCFF) ........ 24
5. Design procedure for Clock-on-demand (COD-FF) .................. 24
6. Design procedure for Gated TGFF .................................... 25
7. Simulation results for Gated TGFF ..................................... 26

Chapter 3 ................................................................................. 37
1. Comparison of power consumption for Gated TGFF ............... 37
  1.1 Simulation results for Gated TGFF ................................... 38
  1.2 Comparison .................................................................. 39
2. Comparison of power consumption for DL-DFF .................... 40
  2.1 Simulation result for DL-DFF ......................................... 40
  2.2 Comparison .................................................................. 40
3. Comparison of power consumption for COD-DFF ................. 42
  3.1 Simulation result for COD-DFF ....................................... 42
  3.2 Comparison .................................................................. 42
4. Conclusion ......................................................................... 43
5. Improvement ...................................................................... 44

Reference ................................................................................ 45

Appendix A ............................................................................. 46
1. Simulation results ............................................................... 46
  1.1 Simulation results for LD-DFF ........................................ 46
  1.2 Simulation results for COD-DFF ..................................... 55
  1.3 Simulation results for CCFF ........................................... 63
List of tables and figures

Figure 1.1 Symbol of D flip-flop ................................................................. 3
Figure 1.2 Waveform of latch ................................................................. 4
Figure 1.3 Waveform of flip-flop ........................................................... 5
Figure 1.4 Impact of skew on performance ........................................... 6
Figure 1.5 Impact of jitter on performance ............................................. 7
Figure 1.6 Definition of times ............................................................... 8
Figure 1.7 Timing parameters ............................................................... 9
Figure 1.8 Relationship among timing parameters ......................... 10
Table 1.1 Logical effort [3][4] ............................................................... 13
Figure 2.1 Data –transmission look-ahead D flip-flop .............. 16
Figure 2.2 Clock-on-demand flip-flop .................................................. 17
Figure 2.3 Conditional-capture flip-flop ............................................. 18
Figure 2.4 Gated TGFF .................................................................. 19
Table 2.1 Simulation setup ............................................................... 20
Figure 2.5 Testbench ..................................................................... 20
Figure 2.6 Testbench for D flip-flop .................................................. 21
Table 2.2 Widths based on logical effort ....................................... 23
Table 2.3 Widths for CCFF based on logical effort ................. 25
Table 2.4 Widths for TGFF based on logical effort ................. 26
Table 2.5 PDP for Wn7 ................................................................. 27
Figure 2.7 Delays for Wn7 ............................................................... 27
Figure 2.8 PDP for Wn7 ................................................................. 28
Table 2.6 Power-delay products for Wn8 ............................... 28
Figure 2.9 Delays for Wn8 ............................................................... 29
Figure 2.10 PDP for Wn8 ............................................................... 29
Table 2.7 Power-delay products for Wn1 ............................... 30
Figure 2.11 Delays for Wn1 ............................................................... 30
Figure 2.12 PDP for Wn1 ............................................................... 31
Table 2.8 Power-delay products for Wn2 ............................... 31
Figure 2.13 Delays for Wn2 ............................................................... 32
Figure 2.14 PDP for Wn2 ............................................................... 32
Table 2.9 Power-delay products for Wn3 ............................... 33
Figure 2.15 Delays for Wn3 ............................................................... 33
Figure 2.16 PDP for Wn3 ............................................................... 34
Table 2.10 Power-delay products for Wn4 ...................................... 34
Figure 2.17 Delays for Wn4 ............................................................... 35
Figure 2.18 PDP for Wn4 ............................................................... 35
Table 3.1 Energy and data activity rate ........................................... 37
Figure 3.1 Comparisons ................................................................. 38
Table 3.2 Energy consumption for normal D flip-flop ................ 38
Table 3.3 Energy consumption for GTFF ................................................................. 39
Figure 3.2 Comparisons ....................................................................................... 39
Table 3.4 Energy consumption for normal D flip-flop ........................................... 40
Table 3.5 Energy consumption for DL-DFF ............................................................ 40
Figure 3.3 Comparisons ....................................................................................... 41
Table 3.6 Energy consumption for normal D flip-flop ........................................... 42
Table 3.7 Energy consumption for CODDFF .......................................................... 42
Figure 3.4 Comparisons ....................................................................................... 43
Table A.1 Power-delay products for Wn1 .............................................................. 46
Figure A.1 Delays for Wn1 ..................................................................................... 47
Figure A.2 PDP for Wn1 ....................................................................................... 47
Table A.2 Power-delay products for Wn2 .............................................................. 48
Figure A.3 Delays for Wn1 ..................................................................................... 48
Figure A.4 PDP for Wn2 ....................................................................................... 49
Table A.3 Power-delay products for Wn3 .............................................................. 49
Figure A.5 Delays for Wn3 ..................................................................................... 50
Figure A.6 PDP for Wn3 ....................................................................................... 50
Table A.4 Power-delay products for Wn4 .............................................................. 51
Figure A.7 Delays for Wn4 ..................................................................................... 51
Figure A.8 PDP for Wn4 ....................................................................................... 52
Table A.5 Power-delay products for Wn5 .............................................................. 52
Figure A.9 Delays for Wn5 ..................................................................................... 53
Figure A.10 PDP for Wn5 ..................................................................................... 53
Table A.2 Power-delay products for Wn8 .............................................................. 54
Figure A.11 Delays for Wn8 .................................................................................... 54
Figure A.12 PDP for Wn8 ..................................................................................... 55
Table A.7 Power-delay products for Wn1 .............................................................. 55
Figure A.13 Delays for Wn1 .................................................................................... 56
Figure A.14 PDP for Wn1 ..................................................................................... 56
Table A.8 Power-delay products for Wn2 .............................................................. 57
Figure A.9 Delays for Wn2 .................................................................................... 57
Figure A.10 PDP for Wn2 ..................................................................................... 58
Table A.9 Power-delay products for Wn8 .............................................................. 58
Figure A.11 Delays for Wn8 ................................................................................... 59
Figure A.12 PDP for Wn8 ..................................................................................... 59
Table A.10 Power-delay products for Wn3 ............................................................. 60
Figure A.13 Delays for Wn3 ................................................................................... 60
Figure A.14 PDP for Wn3 ..................................................................................... 61
Table A.11 Power-delay products for Wn6 ............................................................. 61
Figure A.15 Delays for Wn6 ................................................................................... 61
Figure A.16 PDP for Wn6 ..................................................................................... 62
Table A.12 Power-delay products for Wn2 ............................................................. 63
Figure A.17 Delays for Wn2 ................................................................................... 64
Figure A.18 PDP for Wn2 ..................................................................................... 64
Table A.13 Power-delay products for Wn1 ................................................................. 65
Figure A.19 Delays for Wn1 ..................................................................................... 65
Figure A.20 PDP for Wn1 ....................................................................................... 66
Table A.14 Power-delay products for Wn7 ............................................................... 66
Figure A.21 Delays for Wn7 ..................................................................................... 67
Figure A.22 PDP for Wn7 ....................................................................................... 68
Chapter 1

1. Introduction

Integration density and performance of integrated circuits had gone through a big revolution, because of the development of CMOS technology. The circuit becomes smaller and smaller, and the higher clock frequency can also be achieved, so far CMOS has been the dominant technology for VLSI implementations. The trend of development of CMOS technology will continuously increase. However as the performance and clock frequency get better and higher, the low power consumption becomes an important issue for the modern VLSI designers. The high power consumption limits the circuit performance and the size of the circuit as well. There are two kinds of power consumption mentioned in digital integrated circuits, static power consumption and dynamic power consumption.

- Static power consumption:

\[ P_{\text{static}} = I_{\text{Leakage}} \cdot V_{DD} \]

Where \( V_{DD} \) is the supply voltage, \( I_{\text{Leakage}} \) is the leakage current, which is the subthreshold current of the transistors. The CMOS transistor can experience a drain-source current, even when \( V_{GS} \) is smaller than the threshold voltage, which means the transistor doesn’t work. The smaller the threshold voltage, the larger the leakage current, and then the static power consumption is high because of the high leakage current. The leakage current cannot be eliminated totally, so the only way to reduce the static power consumption is to reduce the supply voltage. However reducing power supply voltage while keeping the threshold voltage constant results bad performance, especially when \( V_{DD} \) approaches \( 2V_T \) [2], so choosing the right value of the threshold voltage and supply voltage is a tradeoff between performance and static power consumption.

- Dynamic power consumption:

\[ P_{\text{dynamic}} = C_L \cdot V_{DD}^2 \cdot f \]

Where \( C_L \) is the load capacitor, it gets charged or discharged through the closer transistor, its voltage changes between 0 to supply voltage. \( V_{DD} \) is the supply
voltage and $f$ is the switching activity or frequency of transitions, in my thesis work, they can be from stage $0\rightarrow0$, $0\rightarrow1$, $1\rightarrow0$ and $1\rightarrow1$. As mentioned above, reducing supply voltage too much is not a good idea to reduce the power consumption; the only interesting parameter from the formula to reduce dynamic power consumption is the switching activity $f$. If the input signals remain unchanged, which means no switching happens, the dynamic power consumption is zero, because $f$ is zero. The other way round, if the input data switch states all the time, the dynamic power consumption will get large. Having a switch component in a digital system with lower switching frequency will be highly appreciated. D flip-flop is the basic switch component and mostly used gate in the digital system. In a high performance processor, the number of flip-flops is about 60,000 and the clock circuits that driver the flip-flops consume about 40% of the total power. In my thesis work, I verify a kind of look-ahead D flip-flops, which have extra look-ahead circuit parts with internal clock signals and proof this kind of circuits can be used for input data with low switching activities. Size four types of this kind of circuits by using logical effort [4], and get the final conclusion by comparing the total power consumption between the normal D flip-flops and the D flip-flops with look-ahead part circuits.

2. Sequential logic circuits

All the circuits I used in my thesis are sequential circuits that are the output of the circuits depends not only on the current values of the inputs, but also on preceding input values. In other words, a sequential circuit remembers some of the past history of the system — it has the memory. [3] There are two kinds of memory elements. One type is latch; another type is flip-flop.

3. Flip-flops

The basic building blocks of the sequential digital circuit are flip-flops. Another name for flip-flops is bistable circuit, which means it has two stable states "0" and "1". When any triggering comes, it makes the circuit to be in one state, and any change of the triggering will cause the flip-flop to change its state, so a trigger pulse must be used into the circuit in order to change the state. Besides that, flip-flops use clock signals as control inputs. Each flip-flop stores one bit.

4. Classification of flip-flops

- SR flip-flop (set-reset flip-flop)
• JK flip-flop
• T flip-flop or toggle
• Master-Slave and Edge-triggered flip-flops
• Static and Dynamic flip-flops
• Single clock phase flip-flops
• Multi-clock phase flip-flops
• Single-edge-triggered flip-flops
• Double-edge-triggered flip-flops
• Single ended flip-flops
• Differential flip-flops
• Flip-flops with internal clock gating
• D Flip-flops

The flip-flops used most in my thesis are the flip-flops with internal clock and D flip-flops. The main feature of the flip-flops with internal clocks is that they turn off the internal clock signals when input and output data are equal, so they are a kind of data dependent flip-flops. D flip-flop has two inputs normally, the data input and clock input. When the data comes, the output of the D flip-flop is changed according to the clock input. In my thesis work both kinds of flip-flops are positive edge-triggered flip-flops, which means the flip-flops change output values as long as the clock is at its positive edge.

![Figure 1.1 Symbol of D flip-flop](image-url)
The symbol of D flip-flops looks like the figure above.

**5. Latches and D flip-flops**

The difference between these two types is the dependence of timing of the system. Latch: is transparent when the internal memory is set from the data input and the changes of the input can be transmitted to the output. For example, see the figure below, the output Q changes according to the input data when the internal clock is “high” and keep the same value of the input data when the clock is “low”. The problem with latches is that glitch at the input will be generated at the output as well.

![Waveform of latch](image)

D-Flip-flop: is not transparent. The states of the output change depending on the edges of the clock signal, so here we use the flip-flops as edge triggered flip-flops. The output changes according to the input data when the rising edges of the clock come.
In my thesis work, I concentrate on comparing the D flip-flops and flip-flops with extra look-ahead circuits. Each circuit has only one external clock signal, by timing classification of the digital systems, all the signals are synchronous signals, which have the exact same frequency as the local clock and maintain a known fixed phase offset to that clock. However in reality, the clock is not ideal. The phase of the same clock signal at different points in the system is not the same. The different clock events turn out to be neither perfectly periodic nor perfectly simultaneous. The clock signals have both spatial and temporal variations, which degrade the system performance. [2]

6. Synchronous interconnection

The spatial variation in the arrival time of a clock transition on an integrated circuit is commonly referred to as clock skew. It should be constant from cycle to cycle. Skew is happened in sequential circuits. Flip-flops clocked by the “same” clock signal observe different transition times depending on their distance from the clock source [3]. The clock skew can be positive or negative depending on the direction and position of the clock source.

![Figure 1.3 Waveform of flip-flop](image-url)
The temporal variation of the clock period of a given point is referred to as clock jitter, which is the clock period can reduce or expand on a cycle-by-cycle basis. Jitter is a zero mean random variable. There are two kinds of the jitter:

6.2 Jitter

The temporal variation of the clock period of a given point is referred to as clock jitter, which is the clock period can reduce or expand on a cycle-by-cycle basis. Jitter is a zero mean random variable. There are two kinds of the jitter:

6.2.1 Absolute jitter

Absolute jitter refers to the worst-case variation of a clock edge at a given location with respect time-varying deviations of a single clock period relative to an ideal
reference clock. [3]

6.2.2 Cycle-to-cycle jitter

Cycle-to-cycle jitter refers to the time-varying deviations of a single clock period relative to an ideal reference clock. Under the worst case, the magnitude of the cycle-to-cycle jitter equals twice the absolute jitter. [3]

![Figure 1.5 Impact of jitter on performance](image)

\[ T > t_{\text{Clk-to-Output}} + t_{\text{logic}} + t_{\text{Setup}} + 2t_{\text{Jitter}} \] [3]

6.3 Combined Impact of Skew and Jitter:

\[ T > t'_{\text{Clk-to-Output}} + t_{\text{logic}} + t_{\text{Setup}} - t_{\text{Skew}} + 2t_{\text{Jitter}} \]

\[ t_{\text{Skew}} < t'_{\text{Clk-to-Output,cl}} + t_{\text{Logic,cl}} - t_{\text{hold}} - 2t_{\text{Jitter}} \]

Where \( t'_{\text{Clk-to-Output,cl}} \) and \( t_{\text{Logic,cl}} \) are contamination delay. [3]

7. Timing metrics for flip-flops

There are three important timing parameters and delays used in sequential circuits: setup time; hold time, and propagation delay time. Delay is a function of the slopes of the input and output signals of the gates. In order to quantify them, rise and fall times are applied to signal waveforms instead of the gates. They express how fast a signal transits between different levels. The measurement of rise and fall time is very important for characterizing the circuit performance, especially
the power consumption, so we use 10%, 50% and 90% to measure the waveform slopes. For example, propagation delay time is measured between the 50% transition points of the input and output waveforms. The rise time and fall time are measured between 10% and 90% of the slopes.

![Diagram of waveform with labels:](image)

Figure 1.6. Definition of times

### 7.1 Propagation delay

The propagation delay of a gate is a way to describe how quickly it responds to input. For inverter, it is measured between the 50% transition points of the input and output waveforms. Because the gate displays different response times for rising or falling input waveforms, so there are two other definitions related to the propagation delay, $t_{p_{ua}}$ and $t_{p_{ua}}$. The overall propagation delay is the average of $t_{p_{ua}}$ and $t_{p_{ua}}$, where $t_{p_{ua}}$ defines the time of the output from low to high and $t_{p_{ua}}$ defines the time of the output from high to low:

$$t_p = \frac{t_{p_{ua}} + t_{p_{ua}}}{2}$$
For sequential circuits, propagation delay time is the time between the clock signal and the output signal with the corresponding wave edges, so it is called clock-to-output as well. The measurements of low to high transitions and high to low transitions are between 50% of the transition points of the waveforms. But they are different, so we choose the maximum value between them, which means:

\[
t_{\text{Clock-to-Output}} = \text{Max}(t_{\text{Clock-to-Output}_{\text{ih}}}, t_{\text{Clock-to-Output}_{\text{ihi}}})
\]

7.2 Setup time

When the data comes in, there should be a time interval for the data to be stable, the time between the edges of the input data and corresponding clock edge is called setup time. The edge–triggered flip-flop with right Setup time can function correctly. Also the measurements of low to high transitions and high to low transitions are different, so it is chosen by the maximum value between them,
which means:

\[ t_{\text{Smp}} = \text{Max}(t_{\text{Smp_{in}}}, t_{\text{Smp_{out}}}) \]

The distance between the data input signal and the clock signal would affect the output performance too much, which means setup time, hold time and propagation delay are not independent from each other. If the data transition happens long before the clock signal, the clock-to-output time could be the same. But narrowing the time interval between data and clock signal will degrade the performance gradually, instead of instantaneous failure. For example the transition data inputs are getting closer to the clock, the clock-to-output time will getting higher. As long as the data transitions get too close to the clock edge, the flip-flops cannot work properly.

![Graph](image_url)

**Figure 1.8 Relationship among timing parameters**

### 7.3 Hold time

It is also required that the state of the input to be held for some time after the clock edge. The time interval after the time that the input data is stable is called Hold time. The measurements of low to high transitions and high to low transitions are different as well, so it is chosen by the maximum value between them, which means:
There are three power dissipations of a flip-flop mentioned in this paper:
1. Internal Power dissipation: it concludes the power consumed by the flip-flop and the powered used by driving the output load.
2. Local data Power dissipation: concludes the power consumed by driving the data input of the flip-flop.
3. Local clock Power dissipation: concludes the power consumed by driving the clock input of the flip-flop.

The total power consumption is got from adding these three power dissipations.

The propagation delay time is related to the power consumption, since it is always used to measure the speed of the response from the changes of inputs to the outputs. The gate capacities can store the energy when the change happens. The faster the state changes, the higher the power consumed, so power-delay product can be used to measure the devices with switching properties.

There are four transmission states 0→0, 0→1, 1→0 and 1→1, so if we simulate these four transition stages and measure the energies for $E_{00}$, $E_{01}$, $E_{10}$ and $E_{11}$. Besides that, if the transition probability of the input data is given as well, we can get the average energy consumption of a flip-flop by:

$$E_{\text{avg, energy}} = E_{00} \cdot \alpha_{00} + E_{01} \cdot \alpha_{01} + E_{10} \cdot \alpha_{10} + E_{11} \cdot \alpha_{11}$$

In addition, we can use this formula to compare the power consumptions between the normal flip-flops and the flip-flops with look-ahead circuits.
9. Transistor sizing

A tradeoff between speed and power consumption is often determined by the application. The transistors with small sizes often consume less power, but with slower speed. However, the transistors with larger sizes are often faster but consume more power. The delay and power are the functions of transistor widths, so sizing the transistor widths is the main work for D flip-flop designs. The critical path in each circuit is the priori to be sized and regarding to minimize the power consumption, I just try to size the transistors in the critical path and keep other transistors with the minimum sizes. The common method of sizing the transistors is done like the following steps.

1. All the $N_{mos}$ or $P_{mos}$ transistors are set to the minimum size first. In my thesis, I use 0.18 $\mu$m CMOS technology, which means the minimum width of the transistor is 0.28 $\mu$m and the $L = 0.18 \mu m$, where $L$ is dimension of the transistor.

2. Run the simulation and make sure the transistors can work properly. Record the values of power-delay products.

3. Use the values of the transistor widths along the critical path obtained from using logical effort [4]. Run the simulation again and compute the power-delay products.

4. Change the widths of the transistors in the critical path. Run the simulation again, and calculate the power-delay products.

5. Repeat performing the step 4 until the minimum power-delay product is reached.

6. Record the sizes of the transistors corresponding to the minimum power-delay product.

This method is based on the logical effort method, so it is a relatively optimized and better solution for sizing all the transistors.

10. Logical effort

Power-delay product (PDP) is the mostly measured power consumption in my simulation; it is the product of power and propagation delay time, so it is efficient to reduce the delay in order to minimize the total power consumption. Logical effort is a useful concept for concerning delay in the circuit. It examines the schematic of the circuit and determines an optimal size of the transistors in order to maximize the speed, so it is easy to select gate sizes for minimum delay by using that. There are some definitions about the logical effort as well:
• Electrical effort $H$: the ratio of output capacitance to input capacitance:

$$H = \frac{C_L}{C_{in}}$$

• Electrical effort for each stage $h$:

$$h = \frac{C_{\text{next-stage}}}{C_{\text{previous-stage}}}$$

• Logical effort $g$: tells how much more input capacitance a gate presents to deliver the same output current as an inverter. The logical efforts of some common gates are given as follows:

<table>
<thead>
<tr>
<th>Gate Type</th>
<th>Number of inputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inverter</td>
<td>1</td>
</tr>
<tr>
<td>NAND</td>
<td>4/3, 5/3, 6/3, 7/3, $\frac{n+2}{3}$</td>
</tr>
<tr>
<td>NOR</td>
<td>5/3, 7/3, 9/3, 11/3, $\frac{2n+1}{3}$</td>
</tr>
<tr>
<td>Multiplexer</td>
<td>2, 2, 2, 2, 2</td>
</tr>
<tr>
<td>XOR, XNOR</td>
<td>4, 12, 32</td>
</tr>
</tbody>
</table>

• Path logical effort $G$: is got by multiplying all the logical efforts of the gates along the path.

$$G = \prod_{i=1}^{N} g_i$$

• Branching effort $b$:

$$b = \frac{C_{\text{on-path}} + C_{\text{off-path}}}{C_{\text{on-path}}}$$

• Path branching effort: is the product of the branching efforts at each stage along the path.

$$B = \prod_{i=1}^{N} b_i$$
• Total path effort $F$:

$$F = G \cdot B \cdot H$$

• Gate effort $f$: is used to minimize the path delay.

$$f = \frac{k}{\sqrt[4]{F}}$$

The procedure of using the logical effort is below:

1) Determine the logical effort $g$ for each stage, and then calculate $G$ by multiplying all the values of $g$ along the path.

2) Determine the electric effort $H$.

3) Determine the branching effort $b$ for each stage, and then calculate $B$ by multiplying all the values of $b$.

4) Calculate the total effort: $F = G \cdot B \cdot H$.

5) Compute the stage effort: $f = F^{1/N}$

6) Parasitic delay is approximately to be 1.

7) Each $h$ for each stage can be calculated by $f = g \cdot b \cdot h$. As long as we know the values of $b$, $g$ and $f$ for each stage, we can get the values of $h$ easily.

8) Calculate the multiply that is more or less the same as calculating the capacitance for each stage, for example, we use the electrical effort in last stage being divided by output capacitance, then the result is the capacitance for the previous stage.

$$multiply = \frac{C_{out}}{C_{previous-stage}} = C_{previous-stage}$$

9) Use the multiply timing the old transistor sizes in order to get the new transistor sizes.
Chapter 2

1. Circuit topology

1.1 Transmission gate

All the circuits I used in my thesis work contain the transmission gate. Transmission gate is composed of one Nmos and one Pmos arranged in parallel. It acts as a bi-directional switch controlled by the gates, which are a signal with its complementary signal. The advantage of using that is that: it can make either Nmos or Pmos turn off easily without reducing noise margin and increasing static power consumption, so transmission gate can be efficiently used in some complex gate circuits. But the drawback is that gate signal with its corresponding complement must be available.

1.2 Data-transition look-ahead D flip-flops

This circuit consists of three very important parts: pulse generator; clock control and data-transition look-ahead circuit based on the ordinal D flip-flops. The data-transition look-ahead part plays a XNOR logical function between input data signal D and output signal Q. Before the data input goes through the first transmission gate, if Q = D, the XNOR is “0”, no internal clock signals will be generated. The output will just keep its previous value or state, so only part of the whole circuit works, the total power consumption is less than normal D flip-flops. However if Q is not equal to D, then the clock control part will work and generate the internal clock signals. The pulse generator part generates a pulse on every clock rising edge of the external clock signal. [2]
This circuit has data-transition look-ahead, pulse generator parts as DL-DFF does. The data-transition look-ahead part plays the XNOR logic function between input data signal D and output signal Q as that of DL-DFF. When D is equal to Q, the result of logic function XNOR is “0” so no internal signals will be generated. However the pulse generator generates internal signals by external clock signal when D is not equal to Q.

Unlike DL-DFF, this one has its local pulse generator, which helps avoid distortion problem with the pulse in the clock distribution and power penalty of the pulse clock generator. The clock control function is integrated in the internal pulse generator of the COD-FF as opposed to the DL-DFF, which reduces the area and has better energy efficiency than DL-DFF. [2]
This is a positive edge triggered differential input and differential output flip-flop. When the external clock signal is “low”, the flip-flop is precharged, and then the SR latch is disabled. When the external clock is “high”, and the new input data is not the same as the previous output data, then one of output of the NOR gates will be high, and pull down the internal signals. During this short transparency period, the SR latches at the output latch a new data. [2]
The comparator performs the XNOR logic function between input data signal D and output signal Q. When D is not equal to Q, the comparator will enable the external clock to generate the internal clocks.

“The comparator is implemented in complementary pass-transistor logic (CPL) technique taking advantage of freely available true and complementary signals. This reduces transistor count of the clock gating circuitry. The pull-up side of the input clock inverter is chosen to be gated because the CPL realization of an XNOR has better pull-down allowing for the generation of the internal clocks faster than if pull-down side of the input inverter was gated.” [2]
2. Simulation setup

I use the 0.18 µm CMOS technology to simulate all the circuits. The temperature is at room temperature 25°C. The supply voltage is 1.8V. The summation of all the settings for the simulation is given below:
Table 2.1 Simulation setup

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clock period</td>
<td>8 ns</td>
</tr>
<tr>
<td>Temperature</td>
<td>25°C</td>
</tr>
<tr>
<td>Capacitance load</td>
<td>24 fF</td>
</tr>
<tr>
<td>Supply voltage</td>
<td>1.8V</td>
</tr>
<tr>
<td>Transistor type</td>
<td>Pmos4, Nmos4</td>
</tr>
</tbody>
</table>

2.1 Testbench

The testbench contains four output capacitances as the normal load for D flip-flops. This load is 24 fF for each. There are two “inverters” set in series to each input of the flip-flop in the testbench. The second inverter, or in other words, the one that is closer to the flip-flop, is a buffer that is used to provide realistic clock or data inputs, so when we calculate the power consumption of the local data and clock, we can measure the currents through these buffers.

![Figure 2.5 Testbench](image-url)
2.2 Testbench for D flip-flops

There is another testbench used to measure the setup time, hold time and clock width or whatever.

*Figure 2.6 Testbench for D flip-flop*

3. Design procedure for data-transmission look-ahead D flip-Flop

1. Draw the schematic of the circuit, set all the widths of the transistors to be the minimum sizes first.
2. Because we have to size the values of the transistors which are either \( Nmos \) or \( Pmos \). It is complicated to change it all the time in the interface of the “transistor property box” in Cadence, so a better and easier way to do this is setting all the widths of the \( Nmos \) and \( Pmos \) to be parameters, like \( Wn \) and \( Wp \).
Here we can assume that all the \( Nmos \) are with the same size, and the same as all the \( Pmos \).

3. We also have to consider about the relationship between the sizes of \( Nmos \) and those of \( Pmos \). In the digital integrated circuit, we can size the \( Nmos \) and \( Pmos \) by using the pull-down and pull-up network, where we can find some relationship between them. Normally we set one width value, \( Wn \), to all \( Nmos \) transistors and one value of width \( Wp \) for all \( Pmos \) transistors; then \( Wn \) and \( Wp \) have some ratio between them, so here we can set a ratio, \( \text{ratio} \), between the \( Nmos \) and \( Pmos \) as a parameter as well. In this case we not only simply the simulation but also reduce workload of sizing. In my thesis work, I set the ratio to be 1.7 between the widths of \( Nmos \) and \( Pmos \) at first, which means:

\[
Wn = \text{ratio} \cdot Wp
\]

4. Put the parameters of \( Nmos \) and \( Pmos \), \( Wn \) and \( Wp \) into the boxes of widths in “Parameter property” instead of the real values. For the simplicity and efficiency, we can just resize the values of the transistors along the critical path and keep others’ transistor sizes with minimum values.

5. Generate the symbol of the schematic of D flip-flops with look-ahead circuits; add the variables in the symbol interface with the same number of variables as that of being set in the schematic figure.

6. Construct the testbench as mentioned above.

7. Set “Setup time”, “Hold time”, “Clock width”, and “data width” and so on as variables in the Cadence.

8. Since these circuits are D flip-flop with internal clocks or look-ahead circuits, we can extract the D flip-flop circuits without look-ahead part from them and then set all the transistors with minimum sizes.

9. With the minimum sizes of the D flip-flop, I simulate it with the “testbench for D flip-flop”. We can find the minimum setup time, hold time, clock width and data width while keeping it work properly. First we give enough values for the setup time, data width and others to let the D flip-flop work properly. Then I set each parameter as a variable, sweep the value of that in the simulation. Since the setup time, hold time and propagation delay are related to each other in the real circuit, as long as the changed value is not suitable, the output will get expanded or have some other strange output results.

10. We have to size the whole circuits and at the same time we must make sure the D flip-flop can work properly as well. The way to size and find optimized sizes of the transistors is to use the logical effort. From the circuit above, we can see that this circuit is 6 stages. And the way to judge how many stages we have is to check how many inverters and transmission gates along the critical path. Each inverter or transmission gate corresponds to one stage. The logical effort \( g_i \) for transmission gate and inverter are approximately 1, so we have \( G \):

\[
G = \prod_{j=1}^{N} g_i = 1
\]
Secondly the electrical effort $H$:

$$H = \frac{C_L}{C_{in}} = 4$$

From the circuit figure, we can see that not all the branching effort for each stages are “1” here, because in stage 1 and stage 5, there are some branches.

$$b_1 = 3, b_2 = 2, \text{ so}$$

$$B = \prod_{1}^{N} b_i = 6$$

Based on the values of $G$, $B$ and $H$, I can get the total path effort $F$:

$$F = G \cdot B \cdot H = 24$$

And then the value of the gate effort $f$ is ready:

$$f = F^{1/N} = 1.7$$

But we must be aware that $f = g_i \cdot b_i \cdot h_i$ must be a constant in each stage.
So the values of $h_i$ can be different according to the values of $b_i$ and $g_i$.
The figure below shows the values calculated for each stage:

<table>
<thead>
<tr>
<th>$g_i$</th>
<th>$b_i$</th>
<th>$h_i$</th>
<th>$Cg/Cin$</th>
<th>$Wp/Wn$</th>
<th>$Wn (\mu m)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>3</td>
<td>0.57</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1.7</td>
<td>0.57</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1.7</td>
<td>0.95</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td>1.7</td>
<td>1.61</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>2</td>
<td>0.86</td>
<td>2.74</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td>1.7</td>
<td>2.35</td>
<td>2</td>
</tr>
</tbody>
</table>

The values of $Cg/Cin$ are calculated by $4 / 1.7$ first, since the output capacitance is 4. And then use the value of division in previous step to divide $h_i$ where is from the previous stage and so on until the value is equal or approximately to 1.
The value of \[ \frac{W_p}{W_n} \]

is the ratio from the sizing of the pull-down and pull-up network. For inverter and transmission gate, it should be 2. Now we put the new values of \( W_n \) calculated from the logical effort into the simulation table and keep other transistors with minimum sizes. We run the simulation with the testbench mentioned above. Based on working properly, we change the sizes of transistors on the critical path one by one. For example, we can change the value of one transistor, and keep others as before. We run the simulation, measure the currents of the clock, data and internal block and then calculate the power-delay product. We keep changing the value of this transistor, calculating PDP until we find the minimum value, then we can keep and record this value, and change the second transistor, then third one and so on until all the transistors along the critical path can be resized, each with corresponding minimum power-delay product.

This is the whole procedure to size the transistors for DL - DFF.

4. Design procedure for Conditional-Capture flip-flop (CCFF)

The simulation procedures are more or less the same procedure as for Data-transmission look-ahead circuit, but for step 10, there is a little different from the first one. We don’t need to find the optimal values of the transistors by using the logical effort. It is a differential input with differential output flip-flop. So we can skip step 9 and do the step 10 directly instead. Fortunately there are not too many variables along the critical path to be changed in order to get the minimum power-delay product. We can change the first transistor, resize it, and calculate the currents and power-delay product until the changed value gets the minimum power-delay product, so we record it and keep going with the second, third transistor and so forth.

5. Design procedure for Clock-on-demand (COD-FF)

The same design procedure as before until the step 10:
Here the critical path of COD-FF has 3 stages, because there are just three inverters and one transmission gate. Logical effort \( g_e = 1 \) so
\[ G = \prod_{i=1}^{N} g_i = 1 \]

Electrical effort \( H \):
\[ H = \frac{C_L}{C_{in}} = 4 \]

Branching effort for each stage, \( b_1, b_2, b_3 \) are equal to 2, because of two branches, so
\[ B = \prod_{i=1}^{N} b_i = 8 \]

Then
\[ F = G \cdot B \cdot H = 32. \]

Gate effort can be got by:
\[ f = F^{1/3} = 3.15 \]

From them, we can obtain all the values from the logical effort shown below:

<table>
<thead>
<tr>
<th>( N )</th>
<th>( g_i )</th>
<th>( b_i )</th>
<th>( h_i )</th>
<th>( \frac{C_g}{C_{in}} )</th>
<th>( \frac{W_p}{W_n} )</th>
<th>( W_n (\mu m) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1.58</td>
<td>1</td>
<td>1</td>
<td>1*0.28</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1.58</td>
<td>1.60</td>
<td>2</td>
<td>1.60*0.28</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>2</td>
<td>1.58</td>
<td>2.53</td>
<td>2</td>
<td>2.53*0.28</td>
</tr>
</tbody>
</table>

All the values are calculated by the same way as mentioned before.

6. **Design procedure for Gated TGFF**

The different step of the design procedure is at step 10 as well. There are 6 stages without branching; \( g_i \) and \( b_i \) for each stage are 1, so:
\[ G = \prod_{i=1}^{N} g_i = 1 \]
\[ B = \prod_{i} b_i = 1 \]

\[ H = \frac{C_L}{C_{in}} = 4 \]

So

\[ F = G \cdot B \cdot H = 4 \]

Then

\[ f = f^{MN} = 1.26 \]

Based on the values above, a table of all the corresponding values can be generated:

<table>
<thead>
<tr>
<th>N</th>
<th>( g_i )</th>
<th>( b_i )</th>
<th>( h_i )</th>
<th>( \frac{C_g}{C_{in}} )</th>
<th>( \frac{W_p}{W_n} )</th>
<th>( W_n ) (( \mu m ))</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1.26</td>
<td>1</td>
<td>2</td>
<td>0.28*1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1.26</td>
<td>1.26</td>
<td>2</td>
<td>0.28*1.26</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1.26</td>
<td>1.59</td>
<td>2</td>
<td>0.28*1.59</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td>1.26</td>
<td>2.00</td>
<td>2</td>
<td>0.28*2.00</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>1</td>
<td>1.26</td>
<td>2.52</td>
<td>2</td>
<td>0.28*2.52</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td>1.26</td>
<td>3.18</td>
<td>2</td>
<td>0.28*3.18</td>
</tr>
</tbody>
</table>

The way to calculate all the values are the same as mentioned before.

### 7. Simulation results for Gated TGFF

Based on the logical effort, we get the sizes of the transistors on the critical path: \( W_{n7} = 0.3 \, \mu m, W_{n8} = 0.353 \, \mu m, W_{n1} = 0.445 \, \mu m, W_{n2} = 0.56 \, \mu m, W_{n3} = 0.71 \, \mu m, W_{n4} = 0.89 \, \mu m \).

- Change the size of \( W_{n7} \), which is the first transistor from the input, and keep the sizes of others as before:
Table 2.5 PDP for Wn7

<table>
<thead>
<tr>
<th>Width (µm)</th>
<th>$I_{data}$ (nA)</th>
<th>$I_{clock}$ (nA)</th>
<th>$I_{internal}$ (µA)</th>
<th>Delay (ps)</th>
<th>PDP ($10^{-20}$ J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wn7 = 0.28</td>
<td>361.7</td>
<td>983</td>
<td>9.271</td>
<td>210</td>
<td>401.27</td>
</tr>
<tr>
<td>Wn7 = 0.3</td>
<td>377.7</td>
<td>983.4</td>
<td>9.282</td>
<td>196.8</td>
<td>377.02</td>
</tr>
<tr>
<td>Wn7 = 0.4</td>
<td>377.7</td>
<td>983.4</td>
<td>9.382</td>
<td>198</td>
<td>383.53</td>
</tr>
<tr>
<td>Wn7 = 0.5</td>
<td>548</td>
<td>983.2</td>
<td>9.496</td>
<td>196.9</td>
<td>390.83</td>
</tr>
<tr>
<td>Wn7 = 0.8</td>
<td>816.1</td>
<td>983.7</td>
<td>9.992</td>
<td>197</td>
<td>418.14</td>
</tr>
<tr>
<td>Wn7 = 1.0</td>
<td>997.5</td>
<td>983.9</td>
<td>10.29</td>
<td>197</td>
<td>435.14</td>
</tr>
<tr>
<td>Wn7 = 1.1</td>
<td>1086</td>
<td>974.9</td>
<td>12.45</td>
<td>210.2</td>
<td>549.03</td>
</tr>
<tr>
<td>Wn7 = 1.2</td>
<td>1180</td>
<td>983.7</td>
<td>10.81</td>
<td>210.2</td>
<td>490.87</td>
</tr>
</tbody>
</table>

Figure 2.7 Delays for Wn7
From the figure above, we can find that the value of width corresponding to minimum power-delay product is $Wn7 = 0.3 \mu m$.

- From the logical effort, the width of $Wn8$ is 0.343$\mu m$, so I change the size of $Wn8$ from 0.4$\mu m$, and keep the sizes of others saved as before:

<table>
<thead>
<tr>
<th>Width ($\mu m$)</th>
<th>$I_{f}$ (nA)</th>
<th>$I_{clock}$ (mA)</th>
<th>$I_{inorm}$ (mA)</th>
<th>Delay (ps)</th>
<th>PDP ($10^{-20}$ J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Wn8 = 0.4$</td>
<td>377.7</td>
<td>983.4</td>
<td>9.403</td>
<td>198</td>
<td>389.69</td>
</tr>
<tr>
<td>$Wn8 = 0.5$</td>
<td>377.5</td>
<td>983.2</td>
<td>9.684</td>
<td>200.7</td>
<td>399.04</td>
</tr>
<tr>
<td>$Wn8 = 0.6$</td>
<td>377.5</td>
<td>983.5</td>
<td>9.953</td>
<td>203.2</td>
<td>413.82</td>
</tr>
<tr>
<td>$Wn8 = 0.7$</td>
<td>377.6</td>
<td>983.5</td>
<td>10.23</td>
<td>205.9</td>
<td>429.59</td>
</tr>
<tr>
<td>$Wn8 = 0.8$</td>
<td>377.6</td>
<td>983.5</td>
<td>10.51</td>
<td>208.4</td>
<td>445.31</td>
</tr>
<tr>
<td>$Wn8 = 0.9$</td>
<td>377.6</td>
<td>983.4</td>
<td>10.78</td>
<td>210.8</td>
<td>460.68</td>
</tr>
<tr>
<td>$Wn8 = 1.0$</td>
<td>377.4</td>
<td>984.1</td>
<td>11.06</td>
<td>221.3</td>
<td>494.80</td>
</tr>
<tr>
<td>$Wn8 = 1.1$</td>
<td>377.6</td>
<td>984.3</td>
<td>11.33</td>
<td>222.7</td>
<td>508.77</td>
</tr>
</tbody>
</table>

Figure 2.8 PDP for $Wn7$
Figure 2.9 Delays for Wn8

Figure 2.10 PDP for Wn8
The power-delay product changes linearly as the width of $Wn8$ increases, so use the value obtained from the logical effort then we can get the minimum power.

- Change the size of $Wn1$ from 0.6 $\mu$m which is got from logical effort, and keep the sizes of other transistors as before:

**Table 2.7 Power-delay products for Wn1**

<table>
<thead>
<tr>
<th>Width ($\mu$m)</th>
<th>$I_{\text{Max}}$ (nA)</th>
<th>$I_{\text{Clock}}$ (nA)</th>
<th>$I_{\text{Internal}}$ (µA)</th>
<th>Delay (ps)</th>
<th>PDP ($10^{-20}$ J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Wn1 = 0.6$</td>
<td>381.7</td>
<td>965.6</td>
<td>9.875</td>
<td>193.7</td>
<td>391.27</td>
</tr>
<tr>
<td>$Wn1 = 0.7$</td>
<td>381.6</td>
<td>965.3</td>
<td>10.22</td>
<td>192.4</td>
<td>400.58</td>
</tr>
<tr>
<td>$Wn1 = 0.8$</td>
<td>381.5</td>
<td>965.4</td>
<td>10.59</td>
<td>191.4</td>
<td>411.25</td>
</tr>
<tr>
<td>$Wn1 = 0.9$</td>
<td>381.5</td>
<td>964.9</td>
<td>11</td>
<td>190.8</td>
<td>424.02</td>
</tr>
<tr>
<td>$Wn1 = 1.0$</td>
<td>381.7</td>
<td>965.3</td>
<td>11.43</td>
<td>191.5</td>
<td>440.42</td>
</tr>
<tr>
<td>$Wn1 = 1.1$</td>
<td>381.6</td>
<td>965.2</td>
<td>11.89</td>
<td>190</td>
<td>452.70</td>
</tr>
<tr>
<td>$Wn1 = 1.2$</td>
<td>381.6</td>
<td>964.7</td>
<td>12.37</td>
<td>188.7</td>
<td>465.89</td>
</tr>
<tr>
<td>$Wn1 = 1.3$</td>
<td>381.4</td>
<td>964.8</td>
<td>12.88</td>
<td>187.7</td>
<td>480.65</td>
</tr>
</tbody>
</table>

**Figure 2.11 Delays for Wn1**
The same situation happens as the previous one, so we keep the origin value of $Wn1$.

- Change the size of $Wn2$ from 0.6µm and keep the sizes of other transistors as before:

$$
\begin{array}{cccccccc}
<table>
<thead>
<tr>
<th>Width (\mu m)</th>
<th>I_{Data} (nA)</th>
<th>I_{Cstatic} (nA)</th>
<th>I_{trans} (\mu A)</th>
<th>Delay (ps)</th>
<th>PDP (10^{-20} J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wn2 = 0.6</td>
<td>377.4</td>
<td>983.5</td>
<td>9.359</td>
<td>197.1</td>
<td>380.32</td>
</tr>
<tr>
<td>Wn2 = 0.7</td>
<td>377.7</td>
<td>982.8</td>
<td>9.56</td>
<td>198.4</td>
<td>389.99</td>
</tr>
<tr>
<td>Wn2 = 0.8</td>
<td>377.7</td>
<td>982.6</td>
<td>9.766</td>
<td>199.8</td>
<td>400.15</td>
</tr>
<tr>
<td>Wn2 = 0.9</td>
<td>377.5</td>
<td>982.6</td>
<td>9.977</td>
<td>211.6</td>
<td>431.81</td>
</tr>
<tr>
<td>Wn2 = 1.0</td>
<td>377.7</td>
<td>982.6</td>
<td>10.2</td>
<td>212.8</td>
<td>442.81</td>
</tr>
<tr>
<td>Wn2 = 1.1</td>
<td>377.7</td>
<td>982.3</td>
<td>10.4</td>
<td>214.2</td>
<td>453.42</td>
</tr>
<tr>
<td>Wn2 = 1.2</td>
<td>377.6</td>
<td>982.2</td>
<td>10.62</td>
<td>215.8</td>
<td>465.34</td>
</tr>
<tr>
<td>Wn2 = 1.3</td>
<td>377.5</td>
<td>982.2</td>
<td>10.84</td>
<td>217.4</td>
<td>477.40</td>
</tr>
</tbody>
</table>
\end{array}
$$

Figure 2.12 PDP for Wn1
Figure 2.13 Delays for Wn2

Figure 2.14 PDP for Wn2
Studying from the figure above, I keep the origin value of $Wn_2$, and resize next transistor.

- The optimized value obtained from logical effort for the size of $Wn_3$ is 0.71$\mu$m, so begin sizing from 0.8$\mu$m and keep the sizes of other transistors as before:

<table>
<thead>
<tr>
<th>Width (µm)</th>
<th>$I_{\text{max}}$ (nA)</th>
<th>$I_{\text{Clock}}$ (nA)</th>
<th>$I_{\text{in}}$ (µA)</th>
<th>Delay (ps)</th>
<th>PDP ($10^{-20}$ J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Wn_3 = 0.8$</td>
<td>377.6</td>
<td>983.7</td>
<td>9.493</td>
<td>196.1</td>
<td>383.14</td>
</tr>
<tr>
<td>$Wn_3 = 0.9$</td>
<td>377.3</td>
<td>984</td>
<td>9.74</td>
<td>195.9</td>
<td>391.45</td>
</tr>
<tr>
<td>$Wn_3 = 1.0$</td>
<td>377.6</td>
<td>983.6</td>
<td>10</td>
<td>214.6</td>
<td>438.86</td>
</tr>
<tr>
<td>$Wn_3 = 1.1$</td>
<td>377.3</td>
<td>983.9</td>
<td>10.28</td>
<td>216.6</td>
<td>453.87</td>
</tr>
<tr>
<td>$Wn_3 = 1.2$</td>
<td>377.2</td>
<td>984.3</td>
<td>10.57</td>
<td>218.7</td>
<td>469.70</td>
</tr>
<tr>
<td>$Wn_3 = 1.3$</td>
<td>377.3</td>
<td>984.3</td>
<td>10.88</td>
<td>221</td>
<td>486.97</td>
</tr>
<tr>
<td>$Wn_3 = 1.4$</td>
<td>377.1</td>
<td>984.5</td>
<td>11.19</td>
<td>223.3</td>
<td>504.50</td>
</tr>
</tbody>
</table>

Figure 2.15 Delays for $Wn_3$
The results of power-delay products are obvious, record the transistor size generating the minimum one and resize others.

- Change the size of $W_n4$ and keep the sizes of others as before:

Table 2.10 Power-delay products for $W_n4$

<table>
<thead>
<tr>
<th>Width ($\mu$m)</th>
<th>$I_{Data}$ (nA)</th>
<th>$I_{Clock}$ (nA)</th>
<th>$I_{Internal}$ (µA)</th>
<th>Delay (ps)</th>
<th>PDP ($10^{-20}$ J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$W_n4 = 0.9$</td>
<td>377.6</td>
<td>983.4</td>
<td>9.320</td>
<td>196.9</td>
<td>378.55</td>
</tr>
<tr>
<td>$W_n4 = 1.0$</td>
<td>377.4</td>
<td>983.7</td>
<td>9.488</td>
<td>210.4</td>
<td>410.88</td>
</tr>
<tr>
<td>$W_n4 = 1.1$</td>
<td>377.6</td>
<td>983.3</td>
<td>9.677</td>
<td>210.9</td>
<td>419.02</td>
</tr>
<tr>
<td>$W_n4 = 1.2$</td>
<td>377.4</td>
<td>983.2</td>
<td>9.875</td>
<td>211.6</td>
<td>427.94</td>
</tr>
<tr>
<td>$W_n4 = 1.3$</td>
<td>377.6</td>
<td>983.7</td>
<td>10.07</td>
<td>212.4</td>
<td>461.13</td>
</tr>
<tr>
<td>$W_n4 = 1.4$</td>
<td>377.5</td>
<td>983.4</td>
<td>10.28</td>
<td>213.3</td>
<td>446.94</td>
</tr>
<tr>
<td>$W_n4 = 1.5$</td>
<td>377.8</td>
<td>983.8</td>
<td>10.49</td>
<td>214.3</td>
<td>457.16</td>
</tr>
</tbody>
</table>
Figure 2.17 Delays for Wn4

Figure 2.18 PDP for Wn4
Linearly changes of the power-delay product as the width increases, keep the size of $W_{n4}$ obtained from the logical effort.
This is the whole procedure to size all the transistor widths along the critical path based on the logical effort. The power-delay product generated by using these transistor widths could be the minimum and optimized.

- Simulation results for others circuits are in Appendix.
Chapter 3

1. Comparison of power consumption for Gated TGFF

There are 4 transition stages, 0-0, 0-1, 1-1, 1-0. And according to the formula, the total average power consumption is calculated by:

\[ E_{\text{avg}} = E_{0-0} \cdot \alpha_{0-0} + E_{0-1} \cdot \alpha_{0-1} + E_{1-0} \cdot \alpha_{1-0} + E_{1-1} \cdot \alpha_{1-1} \]

\( \alpha \) is the data activity or switch activity. As long as the circuits are sized optimistically, they are simulated at different data activity rates, 0, 0.5 and 1 in my simulation.

<table>
<thead>
<tr>
<th>( \alpha )</th>
<th>( E_{0-0} )</th>
<th>( E_{0-1} )</th>
<th>( E_{1-0} )</th>
<th>( E_{1-1} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>*</td>
<td>0</td>
<td>0</td>
<td>*</td>
</tr>
<tr>
<td>0.5</td>
<td>1/4</td>
<td>1/4</td>
<td>1/4</td>
<td>1/4</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1/2</td>
<td>1/2</td>
<td>0</td>
</tr>
</tbody>
</table>

* means either \( E_{0-0} \) or \( E_{1-1} \).

Circuits should be simulated with different data activity rates; the simulation results can decide whether the circuit can be used in a particular data activity. Regarding the circuits I used in my thesis work, the results should be like below:
When the data activity is low, the power consumption of normal D Flip-flop is higher than that of the D flip-flop with look-ahead circuit part. Because when the data is either one or zero, there are not too many transitions, the probability for that input data D equals to output D is high, the look-ahead transition circuit part turns off the internal clocks by comparing the output and input data signals, so only a part of the circuit works, it should consume less than normal D Flip-flop. But when the data activity is high, which means the transition happens all the time, the look-ahead circuit part consumes more with its overhead look-ahead transition circuit part.

1.1 Simulation results for Gated TGFF

Normal DFF

<table>
<thead>
<tr>
<th>Stage</th>
<th>Clip(n)</th>
<th>$I_{\text{Data}}$ (nA)</th>
<th>$I_{\text{Clock}}$ (nA)</th>
<th>$I_{\text{Normal}}$ (nA)</th>
<th>Period (nA)</th>
<th>$E_{\text{average}}$ ($10^{-18}$ J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 – 0</td>
<td>5 – 13</td>
<td>67.56</td>
<td>524.1</td>
<td>1010</td>
<td>8</td>
<td>23063.90</td>
</tr>
<tr>
<td>0 – 1</td>
<td>20 – 28</td>
<td>497</td>
<td>420</td>
<td>4584</td>
<td>8</td>
<td>79214.4</td>
</tr>
<tr>
<td>1 – 1</td>
<td>30 - 38</td>
<td>1.579</td>
<td>580.3</td>
<td>75.56</td>
<td>8</td>
<td>7290.99</td>
</tr>
<tr>
<td>1 – 0</td>
<td>45 - 53</td>
<td>5.433</td>
<td>601.9</td>
<td>2904</td>
<td>8</td>
<td>50563.20</td>
</tr>
</tbody>
</table>
GTFF

Table 3.3 Energy consumption for GTFF

<table>
<thead>
<tr>
<th>Stage</th>
<th>Clip(n)</th>
<th>$I_{\text{Data}}$ (nA)</th>
<th>$I_{\text{Clock}}$ (nA)</th>
<th>$I_{\text{norma}}$ (nA)</th>
<th>Period (nA)</th>
<th>$E_{\text{average}}$ ($10^{-18}$ J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 – 0</td>
<td>5 – 13</td>
<td>0.2697</td>
<td>564.3</td>
<td>8.902</td>
<td>8</td>
<td>8257.99</td>
</tr>
<tr>
<td>0 – 1</td>
<td>20 – 28</td>
<td>517.7</td>
<td>626.2</td>
<td>6628</td>
<td>8</td>
<td>111915.36</td>
</tr>
<tr>
<td>1 – 1</td>
<td>30 - 38</td>
<td>1.701</td>
<td>562.9</td>
<td>-10.39</td>
<td>8</td>
<td>7980.64</td>
</tr>
<tr>
<td>1 – 0</td>
<td>45 - 53</td>
<td>2.718</td>
<td>625.9</td>
<td>5025</td>
<td>8</td>
<td>81412.10</td>
</tr>
</tbody>
</table>

1.2 Comparison

![Graph showing energy consumption comparison](image)

Figure 3.2 Comparisons

The same conclusion can be generated: the D flip-flop with data transition look-ahead circuit can consume less power than normal D Flip-flop when the data switching activity is low.
2. Comparison of power consumption for DL-DFF

2.1 Simulation result for DL-DFF

Normal DFF

Table 3.4 Energy consumption for normal D flip-flop

<table>
<thead>
<tr>
<th>Stage</th>
<th>Clip(n)</th>
<th>$I_{Data}$ (nA)</th>
<th>$I_{Clock}$ (nA)</th>
<th>$I_{I_{norma}}$ (nA)</th>
<th>Period (nA)</th>
<th>$E_{average}$ ($10^{-18}$ J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 – 0</td>
<td>5 – 13</td>
<td>0.2644</td>
<td>463.2</td>
<td>1416</td>
<td>8</td>
<td>27064.29</td>
</tr>
<tr>
<td>0 – 1</td>
<td>20 – 28</td>
<td>488.8</td>
<td>462.8</td>
<td>6142</td>
<td>8</td>
<td>102147.84</td>
</tr>
<tr>
<td>1 – 1</td>
<td>30 - 38</td>
<td>1.6</td>
<td>456.7</td>
<td>1330</td>
<td>8</td>
<td>25751.52</td>
</tr>
<tr>
<td>1 – 0</td>
<td>45 - 53</td>
<td>5.643</td>
<td>462.1</td>
<td>4617</td>
<td>8</td>
<td>73220.30</td>
</tr>
</tbody>
</table>

DFF with Look-Ahead

Table 3.5 Energy consumption for DL-DFF

<table>
<thead>
<tr>
<th>Stage</th>
<th>Clip(n)</th>
<th>$I_{data}$ (nA)</th>
<th>$I_{i}$ (nA)</th>
<th>$I_{I_{norma}}$ (nA)</th>
<th>Period (nA)</th>
<th>$E_{average}$ ($10^{-18}$ J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 – 0</td>
<td>5 – 13</td>
<td>0.2691</td>
<td>486.4</td>
<td>3009</td>
<td>8</td>
<td>50337.64</td>
</tr>
<tr>
<td>0 – 1</td>
<td>20 – 28</td>
<td>491.5</td>
<td>488</td>
<td>11980</td>
<td>8</td>
<td>186616.8</td>
</tr>
<tr>
<td>1 – 1</td>
<td>30 - 38</td>
<td>1.611</td>
<td>493.4</td>
<td>3032</td>
<td>8</td>
<td>50788.96</td>
</tr>
<tr>
<td>1 – 0</td>
<td>45 - 53</td>
<td>6.222</td>
<td>488</td>
<td>10520</td>
<td>8</td>
<td>158604.80</td>
</tr>
</tbody>
</table>

2.2 Comparison
Unfortunately the figure used for comparisons is not the same as we expected when the data activity is low. The D flip-flop with data transition look-ahead circuit part compares the input data D and output Q turning off the internal clock inputs, so it should consume less energy. When the switch activity is high, the D flip-flop with data transition look-ahead circuit should consume more, because of its overhead functional circuit. However the result shows that when the switching rate is low, the D flip-flop with data transition look-ahead part consumes more than normal D Flip-flop, the look-ahead circuit part wastes power instead of saving it. There could be many reasons for that, but the most important one is that I use edge-triggered D flip-flop to compare, maybe I need to use level-triggered flip-flop instead. Another reason is that maybe the transistor sizes are not optimized enough, some transistors set in minimum sizes but it cannot work efficiently for saving power for the whole system. In some cases, the individual transistor should be sized larger consuming more power than that when it is in minimum size, but the larger size of that transistor saves the power of the whole circuit, so we should find a better-optimized solution to balance them. Then the whole circuit consumes less than normal one, when the data activity is low.
3. Comparison of power consumption for COD-DFF

3.1 Simulation result for COD-DFF

**Normal DFF**

Table 3.6 Energy consumption for normal D flip-flop

<table>
<thead>
<tr>
<th>Stage</th>
<th>Clip(n)</th>
<th>$I_{Data}$ (nA)</th>
<th>$I_{Clock}$ (nA)</th>
<th>$I_{Invert}$ (nA)</th>
<th>Period (nA)</th>
<th>$E_{average}$ ($10^{-18}$ J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 – 0</td>
<td>5 – 13</td>
<td>108.7</td>
<td>479.7</td>
<td>809.9</td>
<td>8</td>
<td>20135.52</td>
</tr>
<tr>
<td>0 – 1</td>
<td>20 – 28</td>
<td>815</td>
<td>416.3</td>
<td>3483</td>
<td>8</td>
<td>67885.92</td>
</tr>
<tr>
<td>1 – 1</td>
<td>30 – 38</td>
<td>1.608</td>
<td>418</td>
<td>4.223</td>
<td>8</td>
<td>6103.17</td>
</tr>
<tr>
<td>1 – 0</td>
<td>45 - 53</td>
<td>-4.148</td>
<td>469.6</td>
<td>2563</td>
<td>8</td>
<td>43609.71</td>
</tr>
</tbody>
</table>

**CODDFF**

Table 3.7 Energy consumption for CODDFF

<table>
<thead>
<tr>
<th>Stage</th>
<th>Clip(n)</th>
<th>$I_{Data}$ (nA)</th>
<th>$I_{1}$ (nA)</th>
<th>$I_{Invert}$ (nA)</th>
<th>Period (nA)</th>
<th>$E_{average}$ ($10^{-18}$ J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 – 0</td>
<td>5 – 13</td>
<td>0.2697</td>
<td>910.6</td>
<td>28.99</td>
<td>8</td>
<td>13533.98</td>
</tr>
<tr>
<td>0 – 1</td>
<td>20 – 28</td>
<td>998.5</td>
<td>988.4</td>
<td>2230</td>
<td>8</td>
<td>60723.36</td>
</tr>
<tr>
<td>1 – 1</td>
<td>30 – 38</td>
<td>1.608</td>
<td>920.7</td>
<td>-8.403</td>
<td>8</td>
<td>13160.23</td>
</tr>
<tr>
<td>1 – 0</td>
<td>45 - 53</td>
<td>-7.519</td>
<td>974.3</td>
<td>7797</td>
<td>8</td>
<td>126198.45</td>
</tr>
</tbody>
</table>

3.2 Comparison
The result shows that when the switching rate is low, the D flip-flop with data transition look-ahead circuit part consumes less than normal D Flip-flop.

4. Conclusion

In my thesis work, we have tested four types of D flip-flops with look-ahead circuits. All the flip-flops have been simulated in 0.18µm CMOS technology at 1.8V. Based on the simulation results, the performances of Clock-on-demand flip-flop (CODFF), Conditional-capture flip-flop (CCFF) and Gated TGFF have been shown to outperform the normal D flip-flops, when the transition probabilities in the input data are low.

However the Data-transmission look-ahead (DL-DFF) got the unexpected comparison result, this is may caused by the bad comparative object. I used all the edge-triggered D flip-flops as the comparative objects to compare with the D flip-flops with look-ahead circuits, maybe I should use level-triggered D flip-flop as the comparator to proof that D flip-flop with look-ahead circuit can consume less power when the input data have low switching activities.
5. Improvement

All the simulation I have done is followed the *ratio* at 1.7. But the *ratio* can be some other values, it could be a parameter to be resized as well, so that the transistor seizes can be more optimized and the performance can be better. The way to size the *ratio* is more or less the same as sizing all the transistors. We set the *ratio* as a parameter, and each time we set it a value and run all the simulations with it, the simulation procedure is the same as mentioned before, and finally compare all the power consumption to choose the value that can produce minimum power-delay product.
Reference

Appendix A

1. Simulation results

1.1 Simulation results for LD-DFF

The method is the same as mentioned before. I choose some transistor widths to simulate and from the simulation results I can find out that the power consumption can be changed as long as the width increases, so these chosen points can express the changing of the whole procedure. We can also find the minimum power consumption from these points.

Based on the logical effort, we obtain the sizes of the transistors on the critical path: \( Wn1 = 0.28 \mu m, Wn2 = 0.28 \mu m, Wn3 = 0.28 \mu m, Wn4 = 0.45 \mu m, Wn5 = 0.77 \mu m, Wn8 = 0.658 \mu m \).

- Change the size of \( Wn1 \) and keep the sizes of others as before:

<table>
<thead>
<tr>
<th>Width (( \mu m ))</th>
<th>( I_{\text{data}} ) (nA)</th>
<th>( I_{\text{clock}} ) (nA)</th>
<th>( I_{\text{intra}} ) (( \mu A ))</th>
<th>Delay (ps)</th>
<th>PDP (( 10^{-20} J ))</th>
</tr>
</thead>
<tbody>
<tr>
<td>( Wn1 = 0.28 )</td>
<td>353.1</td>
<td>788.6</td>
<td>19.71</td>
<td>306.8</td>
<td>115.116</td>
</tr>
<tr>
<td>( Wn1 = 0.3 )</td>
<td>370.2</td>
<td>788.8</td>
<td>19.71</td>
<td>306.8</td>
<td>115.247</td>
</tr>
<tr>
<td>( Wn1 = 0.4 )</td>
<td>458.6</td>
<td>789</td>
<td>19.75</td>
<td>306.8</td>
<td>115.957</td>
</tr>
<tr>
<td>( Wn1 = 0.5 )</td>
<td>550.8</td>
<td>788.8</td>
<td>19.84</td>
<td>306.9</td>
<td>117.000</td>
</tr>
<tr>
<td>( Wn1 = 0.6 )</td>
<td>644.7</td>
<td>788.7</td>
<td>19.95</td>
<td>306.9</td>
<td>118.126</td>
</tr>
<tr>
<td>( Wn1 = 0.7 )</td>
<td>739.9</td>
<td>788.3</td>
<td>20.09</td>
<td>306.9</td>
<td>119.423</td>
</tr>
<tr>
<td>( Wn1 = 0.8 )</td>
<td>835.9</td>
<td>788.1</td>
<td>20.25</td>
<td>307</td>
<td>120.876</td>
</tr>
<tr>
<td>( Wn1 = 0.9 )</td>
<td>932.1</td>
<td>789.1</td>
<td>20.43</td>
<td>306.9</td>
<td>122.368</td>
</tr>
<tr>
<td>( Wn1 = 1.0 )</td>
<td>1029</td>
<td>788.2</td>
<td>20.63</td>
<td>307</td>
<td>124.043</td>
</tr>
</tbody>
</table>
Figure A.1 Delays for Wn1

Figure A.2 PDP for Wn1
From the figure above, we can see the lowest power consumption is produced when the width of $Wn1$ is 0.28µm, so record this value and change the widths of other transistors.

- Change the size of $Wn2$ and keep the sizes of others as before:

Table A.2 Power-delay products for $Wn2$

<table>
<thead>
<tr>
<th>Width (µm)</th>
<th>$I_{dio}$ (nA)</th>
<th>$I_i$ (nA)</th>
<th>$I_{incom}$ (µA)</th>
<th>Delay (ps)</th>
<th>PDP ($10^{-20} J$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Wn2 = 0.3$</td>
<td>353.1</td>
<td>788</td>
<td>19.77</td>
<td>307.5</td>
<td>115.743</td>
</tr>
<tr>
<td>$Wn2 = 0.4$</td>
<td>353.2</td>
<td>789.1</td>
<td>20.07</td>
<td>311.1</td>
<td>118.785</td>
</tr>
<tr>
<td>$Wn2 = 0.5$</td>
<td>353.1</td>
<td>788.7</td>
<td>20.4</td>
<td>314.7</td>
<td>122.026</td>
</tr>
<tr>
<td>$Wn2 = 0.6$</td>
<td>353.2</td>
<td>788.8</td>
<td>20.74</td>
<td>318.3</td>
<td>125.371</td>
</tr>
<tr>
<td>$Wn2 = 0.7$</td>
<td>353.2</td>
<td>788.3</td>
<td>21.07</td>
<td>321.8</td>
<td>128.658</td>
</tr>
<tr>
<td>$Wn2 = 0.8$</td>
<td>353.3</td>
<td>789.5</td>
<td>21.42</td>
<td>325.2</td>
<td>132.074</td>
</tr>
<tr>
<td>$Wn2 = 0.9$</td>
<td>353.2</td>
<td>788.3</td>
<td>21.78</td>
<td>328.6</td>
<td>135.576</td>
</tr>
<tr>
<td>$Wn2 = 1.0$</td>
<td>353.3</td>
<td>788.2</td>
<td>22.14</td>
<td>332</td>
<td>139.130</td>
</tr>
</tbody>
</table>

Figure A.3 Delays for $Wn1$
From the figure above, the lowest power consumption is obvious, so we keep that width value and continue.

- Change the size of $Wn3$, and keep the sizes of others as before:

Table A.3 Power-delay products for $Wn3$

<table>
<thead>
<tr>
<th>Width ($\mu$m)</th>
<th>$I_{Data}$ (nA)</th>
<th>$I_{Clock}$ (nA)</th>
<th>$I_{internal}$ ($\mu$A)</th>
<th>Delay (ps)</th>
<th>PDP ($10^{-20}$ J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Wn3 = 0.3$</td>
<td>353.2</td>
<td>789.2</td>
<td>19.72</td>
<td>305.1</td>
<td>114.572</td>
</tr>
<tr>
<td>$Wn3 = 0.4$</td>
<td>353.2</td>
<td>789</td>
<td>18.52</td>
<td>299.4</td>
<td>105.963</td>
</tr>
<tr>
<td>$Wn3 = 0.5$</td>
<td>353.1</td>
<td>788.4</td>
<td>18.73</td>
<td>296.1</td>
<td>105.912</td>
</tr>
<tr>
<td>$Wn3 = 0.6$</td>
<td>353.2</td>
<td>788.7</td>
<td>18.97</td>
<td>293.9</td>
<td>106.391</td>
</tr>
<tr>
<td>$Wn3 = 0.7$</td>
<td>353.2</td>
<td>788.7</td>
<td>19.25</td>
<td>292.3</td>
<td>107.289</td>
</tr>
<tr>
<td>$Wn3 = 0.8$</td>
<td>353.1</td>
<td>788</td>
<td>19.55</td>
<td>291.1</td>
<td>108.417</td>
</tr>
</tbody>
</table>
Figure A.5 Delays for Wn3

Figure A.6 PDP for Wn3
The lowest power consumption is generated when the \( Wn3 = 0.5 \mu m \), so we record it and keep going.

- Change the size of \( Wn4 \), because the width for this transistor is 0.45\( \mu m \) obtained from logical effort, I begin sizing from 0.5\( \mu m \) and keep the sizes of others as before:

<table>
<thead>
<tr>
<th>Width (( \mu m ))</th>
<th>( I_{Data} (nA) )</th>
<th>( I_{Clock} (nA) )</th>
<th>( I_{I/frequency} (\mu A) )</th>
<th>Delay (ps)</th>
<th>PDP (10^{-20} J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( Wn4 = 0.5 )</td>
<td>353.2</td>
<td>788.2</td>
<td>18.82</td>
<td>295.6</td>
<td>106.221</td>
</tr>
<tr>
<td>( Wn4 = 0.6 )</td>
<td>353.1</td>
<td>788.6</td>
<td>19.03</td>
<td>295.4</td>
<td>107.257</td>
</tr>
<tr>
<td>( Wn4 = 0.7 )</td>
<td>353.1</td>
<td>787.9</td>
<td>19.24</td>
<td>295.6</td>
<td>108.444</td>
</tr>
<tr>
<td>( Wn4 = 0.8 )</td>
<td>353.2</td>
<td>788.3</td>
<td>19.45</td>
<td>296</td>
<td>109.712</td>
</tr>
<tr>
<td>( Wn4 = 0.9 )</td>
<td>353.2</td>
<td>788</td>
<td>19.67</td>
<td>296.4</td>
<td>111.032</td>
</tr>
<tr>
<td>( Wn4 = 1.0 )</td>
<td>353.1</td>
<td>788.8</td>
<td>19.88</td>
<td>296.9</td>
<td>112.345</td>
</tr>
<tr>
<td>( Wn4 = 1.1 )</td>
<td>353.1</td>
<td>788.4</td>
<td>20.1</td>
<td>297.5</td>
<td>113.748</td>
</tr>
</tbody>
</table>

Figure A.7 Delays for \( Wn4 \)
The figures above are apparent for us to judge which one corresponding to the minimum power consumption. We record the value of the width that generates the minimum power consumption and keep simulating the widths of transistors along the critical path.

- Change the size of $Wn5$ from 0.8µm because of logical effort, and keep the sizes of other transistors as before:

### Table A.5 Power-delay products for $Wn5$

<table>
<thead>
<tr>
<th>Width (µm)</th>
<th>$I_{dd}$ (nA)</th>
<th>$I_i$ (nA)</th>
<th>$I_{incond}$ (µA)</th>
<th>Delay (ps)</th>
<th>PDP ($10^{-20}$ J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Wn5 = 0.8$</td>
<td>353.2</td>
<td>788</td>
<td>18.79</td>
<td>296.5</td>
<td>106.373</td>
</tr>
<tr>
<td>$Wn5 = 0.9$</td>
<td>353.1</td>
<td>788.4</td>
<td>19.03</td>
<td>298</td>
<td>108.199</td>
</tr>
<tr>
<td>$Wn5 = 1.0$</td>
<td>353.2</td>
<td>789.2</td>
<td>19.3</td>
<td>299.8</td>
<td>110.315</td>
</tr>
<tr>
<td>$Wn5 = 1.1$</td>
<td>353.3</td>
<td>788.9</td>
<td>19.6</td>
<td>301.8</td>
<td>112.679</td>
</tr>
<tr>
<td>$Wn5 = 1.2$</td>
<td>353.4</td>
<td>788.4</td>
<td>19.94</td>
<td>303.8</td>
<td>115.284</td>
</tr>
<tr>
<td>$Wn5 = 1.3$</td>
<td>353.3</td>
<td>788.5</td>
<td>20.3</td>
<td>305.9</td>
<td>118.063</td>
</tr>
<tr>
<td>$Wn5 = 1.4$</td>
<td>353.4</td>
<td>789.2</td>
<td>20.69</td>
<td>308.2</td>
<td>121.119</td>
</tr>
</tbody>
</table>
Figure A.9 Delays for Wn5

Figure A.10 PDP for Wn5
Because the power-delay product gets higher when the width of $Wn5$ increases, we still have to keep the width of $Wn5$ obtained from the logical effort so that the power-delay product generated by this width is the minimum one as the width changes.

- Change the size of $Wn8$ and keep the sizes of others as before:

<table>
<thead>
<tr>
<th>Width ($\mu m$)</th>
<th>$I_{\text{drain}}$ (nA)</th>
<th>$I_i$ (nA)</th>
<th>$I_{\text{inout}}$ (µA)</th>
<th>$\text{Delay (ps)}$</th>
<th>PDP ($10^{-20}$ J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Wn8 = 0.7$</td>
<td>353.1</td>
<td>787.9</td>
<td>18.81</td>
<td>295.5</td>
<td>106.119</td>
</tr>
<tr>
<td>$Wn8 = 0.8$</td>
<td>353</td>
<td>788.8</td>
<td>19</td>
<td>294.6</td>
<td>106.808</td>
</tr>
<tr>
<td>$Wn8 = 0.9$</td>
<td>353.3</td>
<td>787.8</td>
<td>19.19</td>
<td>294.4</td>
<td>107.739</td>
</tr>
<tr>
<td>$Wn8 = 1.0$</td>
<td>353.1</td>
<td>789</td>
<td>19.39</td>
<td>294.6</td>
<td>108.876</td>
</tr>
<tr>
<td>$Wn8 = 1.1$</td>
<td>353.2</td>
<td>788.8</td>
<td>19.6</td>
<td>295</td>
<td>110.140</td>
</tr>
<tr>
<td>$Wn8 = 1.2$</td>
<td>353.2</td>
<td>788.2</td>
<td>19.81</td>
<td>295.7</td>
<td>111.52</td>
</tr>
<tr>
<td>$Wn8 = 1.3$</td>
<td>353.2</td>
<td>788.8</td>
<td>20.03</td>
<td>296.5</td>
<td>112.995</td>
</tr>
<tr>
<td>$Wn8 = 1.4$</td>
<td>353.1</td>
<td>788.8</td>
<td>20.25</td>
<td>297.4</td>
<td>114.515</td>
</tr>
</tbody>
</table>

Figure A.11 Delays for $Wn8$
Keep the value of the width corresponding to the minimum power-delay product and then the whole procedure for resizing LD-DFF is done.

1.2 Simulation results for COD-DFF

Based on the logical effort, we have the sizes of the transistors on the critical path: $W_{n1} = 0.28\mu m$, $W_{n2} = 0.45\mu m$, $W_{n8} = 0.71\mu m$.

- Change the size of $W_{n1}$ and keep the sizes of other transistors as before:

<table>
<thead>
<tr>
<th>Width (µm)</th>
<th>$I_{data}$ (nA)</th>
<th>$I_{clock}$ (nA)</th>
<th>$I_{internal}$ (µA)</th>
<th>Delay (ps)</th>
<th>PDP (10^{-20} J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$W_{n1} = 0.28$</td>
<td>491.1</td>
<td>1.568</td>
<td>13.37</td>
<td>171.6</td>
<td>476.57</td>
</tr>
<tr>
<td>$W_{n1} = 0.3$</td>
<td>506.9</td>
<td>1.568</td>
<td>13.31</td>
<td>147</td>
<td>465.52</td>
</tr>
<tr>
<td>$W_{n1} = 0.4$</td>
<td>591</td>
<td>1.568</td>
<td>13.17</td>
<td>142.2</td>
<td>431.82</td>
</tr>
<tr>
<td>$W_{n1} = 0.5$</td>
<td>677.9</td>
<td>1.568</td>
<td>13.16</td>
<td>138.6</td>
<td>415.41</td>
</tr>
<tr>
<td>$W_{n1} = 0.6$</td>
<td>766.5</td>
<td>1.567</td>
<td>13.23</td>
<td>136.1</td>
<td>407.33</td>
</tr>
<tr>
<td>$W_{n1} = 0.7$</td>
<td>822.8</td>
<td>1.569</td>
<td>15.06</td>
<td>134.2</td>
<td>446.69</td>
</tr>
<tr>
<td>$W_{n1} = 0.8$</td>
<td>913.1</td>
<td>1.568</td>
<td>15.22</td>
<td>132.8</td>
<td>445.75</td>
</tr>
<tr>
<td>$W_{n1} = 0.9$</td>
<td>1004</td>
<td>1.568</td>
<td>15.41</td>
<td>131.8</td>
<td>446.99</td>
</tr>
<tr>
<td>$W_{n1} = 1.0$</td>
<td>1095</td>
<td>1.568</td>
<td>15.64</td>
<td>131.1</td>
<td>431.91</td>
</tr>
</tbody>
</table>
Figure A.13 Delays for Wn1

Figure A.14 PDP for Wn1
Here we find the lowest power consumption happens at the width of $Wn1$ is 0.6µm, so record the width value and keep finding other sizes of transistors on the critical path.

- Change the size of $Wn2$ and keep the sizes of other transistors as before:

<table>
<thead>
<tr>
<th>Width (µm)</th>
<th>$I_{dio}$ (nA)</th>
<th>$I_{i}$ (nA)</th>
<th>$I_{incurr}$ (µA)</th>
<th>Delay (ps)</th>
<th>PDP ($10^{-20}$J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Wn2 = 0.5$</td>
<td>733.2</td>
<td>1.569</td>
<td>15.04</td>
<td>135.6</td>
<td>423.29</td>
</tr>
<tr>
<td>$Wn2 = 0.6$</td>
<td>733.2</td>
<td>1.569</td>
<td>15.25</td>
<td>135.6</td>
<td>428.41</td>
</tr>
<tr>
<td>$Wn2 = 0.7$</td>
<td>733.3</td>
<td>1.568</td>
<td>15.49</td>
<td>136.3</td>
<td>436.49</td>
</tr>
<tr>
<td>$Wn2 = 0.8$</td>
<td>733.2</td>
<td>1.568</td>
<td>15.74</td>
<td>137.4</td>
<td>446.19</td>
</tr>
<tr>
<td>$Wn2 = 0.9$</td>
<td>733.2</td>
<td>1.569</td>
<td>16.0</td>
<td>138.8</td>
<td>457.26</td>
</tr>
<tr>
<td>$Wn2 = 1.0$</td>
<td>733.2</td>
<td>1.568</td>
<td>16.26</td>
<td>140.4</td>
<td>469.08</td>
</tr>
<tr>
<td>$Wn2 = 1.1$</td>
<td>733.2</td>
<td>1.568</td>
<td>16.53</td>
<td>142.2</td>
<td>482.00</td>
</tr>
<tr>
<td>$Wn2 = 1.2$</td>
<td>733.2</td>
<td>1.568</td>
<td>16.81</td>
<td>144.0</td>
<td>495.36</td>
</tr>
</tbody>
</table>

Figure A.9 Delays for Wn2
As before, we set the width corresponding to the minimum power consumption in the simulation tool, record it and then change the values of widths of others.

- Change the size of $W_{n8}$, its value obtained from logical effort is 0.71µm, so I begin resizing it from that value and keep the sizes of other transistors as before:

<table>
<thead>
<tr>
<th>Width (µm)</th>
<th>$I_{\text{data}}$ (nA)</th>
<th>$I_{\text{clock}}$ (nA)</th>
<th>$I_{\text{internal}}$ (µA)</th>
<th>Delay (ps)</th>
<th>PDP ($10^{-20}$ J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$W_{n8} = 0.71$</td>
<td>766.5</td>
<td>1.567</td>
<td>13.23</td>
<td>136.1</td>
<td>381.275</td>
</tr>
<tr>
<td>$W_{n8} = 0.8$</td>
<td>766.6</td>
<td>1.567</td>
<td>13.46</td>
<td>135.3</td>
<td>384.637</td>
</tr>
<tr>
<td>$W_{n8} = 0.9$</td>
<td>766.7</td>
<td>1.567</td>
<td>13.73</td>
<td>135.1</td>
<td>390.637</td>
</tr>
<tr>
<td>$W_{n8} = 1.0$</td>
<td>766.5</td>
<td>1.567</td>
<td>14</td>
<td>135.2</td>
<td>397.49</td>
</tr>
<tr>
<td>$W_{n8} = 1.1$</td>
<td>766.5</td>
<td>1.568</td>
<td>14.29</td>
<td>135.8</td>
<td>406.37</td>
</tr>
<tr>
<td>$W_{n8} = 1.2$</td>
<td>766.5</td>
<td>1.567</td>
<td>14.59</td>
<td>136.6</td>
<td>416.12</td>
</tr>
<tr>
<td>$W_{n8} = 1.3$</td>
<td>766.6</td>
<td>1.568</td>
<td>14.9</td>
<td>137.7</td>
<td>427.18</td>
</tr>
<tr>
<td>$W_{n8} = 1.4$</td>
<td>766.5</td>
<td>1.568</td>
<td>15.22</td>
<td>138.8</td>
<td>438.58</td>
</tr>
</tbody>
</table>
Figure A.11 Delays for Wn8

Figure A.12 PDP for Wn8
The width of the transistor corresponding to the minimum power-delay product is obvious from the figure above.

- Change the size of $Wn3$ and keep other transistors’ sizes:

<table>
<thead>
<tr>
<th>Width ($\mu$m)</th>
<th>$I_{in}$ (nA)</th>
<th>$I_{t}$ (nA)</th>
<th>$I_{in FLT}$ (µA)</th>
<th>Delay (ps)</th>
<th>$PDP \ (10^{-20}\ J)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Wn3 = 0.3$</td>
<td>766.6</td>
<td>1.567</td>
<td>13.26</td>
<td>136.5</td>
<td>409.52</td>
</tr>
<tr>
<td>$Wn3 = 0.4$</td>
<td>766.6</td>
<td>1.568</td>
<td>13.45</td>
<td>138.8</td>
<td>422.77</td>
</tr>
<tr>
<td>$Wn3 = 0.5$</td>
<td>766.6</td>
<td>1.567</td>
<td>13.7</td>
<td>141.1</td>
<td>438.39</td>
</tr>
<tr>
<td>$Wn3 = 0.6$</td>
<td>766.6</td>
<td>1.567</td>
<td>13.97</td>
<td>143.6</td>
<td>454.87</td>
</tr>
<tr>
<td>$Wn3 = 0.7$</td>
<td>1018</td>
<td>1.567</td>
<td>14.26</td>
<td>146.2</td>
<td>440.21</td>
</tr>
<tr>
<td>$Wn3 = 0.8$</td>
<td>1078</td>
<td>1.567</td>
<td>14.57</td>
<td>148.7</td>
<td>460.78</td>
</tr>
<tr>
<td>$Wn3 = 0.9$</td>
<td>1138</td>
<td>1.568</td>
<td>17.91</td>
<td>151.4</td>
<td>561.83</td>
</tr>
<tr>
<td>$Wn3 = 1.0$</td>
<td>1198</td>
<td>1.568</td>
<td>18.45</td>
<td>154</td>
<td>588.11</td>
</tr>
</tbody>
</table>

Figure A.13 Delays for $Wn3$
Using the same logic to find the width corresponding to minimum power-delay product.

- Change the size of \( W_{n6} \) and keep the sizes of other transistors as before:

Table A.11 Power-delay products for \( W_{n6} \)

<table>
<thead>
<tr>
<th>Width (( \mu m ))</th>
<th>( I_{oa} ) (nA)</th>
<th>( I_i ) (nA)</th>
<th>( I_{inorm} ) (( \mu A ))</th>
<th>Delay (ps)</th>
<th>( PDP ) ((10^{-20} J))</th>
</tr>
</thead>
<tbody>
<tr>
<td>( W_{n6} = 0.3 )</td>
<td>778.5</td>
<td>1.567</td>
<td>13.29</td>
<td>136.4</td>
<td>411.46</td>
</tr>
<tr>
<td>( W_{n6} = 0.4 )</td>
<td>838.4</td>
<td>1.568</td>
<td>13.59</td>
<td>138</td>
<td>432.48</td>
</tr>
<tr>
<td>( W_{n6} = 0.5 )</td>
<td>898.4</td>
<td>1.567</td>
<td>13.9</td>
<td>139.6</td>
<td>454.24</td>
</tr>
<tr>
<td>( W_{n6} = 0.6 )</td>
<td>958</td>
<td>1.568</td>
<td>14.22</td>
<td>141.2</td>
<td>476.56</td>
</tr>
<tr>
<td>( W_{n6} = 0.7 )</td>
<td>1018</td>
<td>1.568</td>
<td>14.54</td>
<td>142.8</td>
<td>440.21</td>
</tr>
<tr>
<td>( W_{n6} = 0.8 )</td>
<td>1078</td>
<td>1.568</td>
<td>14.87</td>
<td>144.3</td>
<td>454.96</td>
</tr>
<tr>
<td>( W_{n6} = 0.9 )</td>
<td>1138</td>
<td>1.569</td>
<td>15.19</td>
<td>145.9</td>
<td>470.01</td>
</tr>
<tr>
<td>( W_{n6} = 1.0 )</td>
<td>1198</td>
<td>1.569</td>
<td>15.52</td>
<td>147.5</td>
<td>485.52</td>
</tr>
</tbody>
</table>
Figure A.15 Delays for Wn6

Figure A.16 PDP for Wn6
Record all the values of transistor widths, so that we can get the minimum power-delay product.

### 1.3 Simulation results for CCFF

Based on the logical effort, we get the sizes of the transistors on the critical path: $W_{n1} = 0.28 \mu m$, $W_{n2} = 0.45 \mu m$, $W_{n7} = \mu m$. Besides changing the width of the transistors as I mentioned above, we still have to consider the sequence of sizing, for example, some transistors work dependent on each other, so sizing this one will affect another one. In some cases we have to reorder the sequence to size the transistors in the critical path. For this circuit, sizing $W_{n7}$ first will affect the power-delay product dramatically, so we size $W_{n2}$ first instead.

- $t_{	ext{setup}} = 80 \text{ps}$, $t_{	ext{skew}} = 75 \text{ps}$, $t_{	ext{walk}} = 600 \text{ps}$.

- Change the size of $W_{n2}$ and keep the sizes of others as before:

<table>
<thead>
<tr>
<th>Width ($\mu m$)</th>
<th>$I_{\text{data}}$ (nA)</th>
<th>$I_{\text{clock}}$ (nA)</th>
<th>$I_{\text{inverse}}$ (mA)</th>
<th>Delay (ps)</th>
<th>PDP ($10^{-20} J$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$W_{n2} = 0.3$</td>
<td>394.8</td>
<td>1.634</td>
<td>82.33</td>
<td>191.1</td>
<td>2901.77</td>
</tr>
<tr>
<td>$W_{n2} = 0.4$</td>
<td>477.2</td>
<td>2.085</td>
<td>17.62</td>
<td>175.2</td>
<td>636.47</td>
</tr>
<tr>
<td>$W_{n2} = 0.5$</td>
<td>567.5</td>
<td>2.474</td>
<td>16.97</td>
<td>158.9</td>
<td>572.37</td>
</tr>
<tr>
<td>$W_{n2} = 0.6$</td>
<td>657.7</td>
<td>2.863</td>
<td>17.51</td>
<td>151.3</td>
<td>572.75</td>
</tr>
<tr>
<td>$W_{n2} = 0.7$</td>
<td>748.3</td>
<td>3.252</td>
<td>18.36</td>
<td>145.5</td>
<td>585.62</td>
</tr>
<tr>
<td>$W_{n2} = 0.8$</td>
<td>839.4</td>
<td>3.642</td>
<td>19.35</td>
<td>140.6</td>
<td>603.13</td>
</tr>
<tr>
<td>$W_{n2} = 0.9$</td>
<td>920.5</td>
<td>4.082</td>
<td>21.7</td>
<td>135.8</td>
<td>652.72</td>
</tr>
<tr>
<td>$W_{n2} = 1.0$</td>
<td>1009</td>
<td>4.477</td>
<td>23.18</td>
<td>130.8</td>
<td>674.91</td>
</tr>
</tbody>
</table>
Figure A.17 Delays for Wn2

Figure A.18 PDP for Wn2
The minimum power consumption happens when the width of Wn2 is 0.5µm. Simulate the circuit again to get the width of Wn7 that can generate the lowest power-delay product.

- Change the size of Wn1 and keep the sizes of other transistors as before:

### Table A.13 Power-delay products for Wn1

<table>
<thead>
<tr>
<th>Width (µm)</th>
<th>$I_{dr}$ (nA)</th>
<th>$I_i$ (nA)</th>
<th>$I_{in_c}(µA)$</th>
<th>Delay (ps)</th>
<th>PDP (10$^{-20}$J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wn1 = 0.3</td>
<td>567.1</td>
<td>2.472</td>
<td>18.35</td>
<td>112.1</td>
<td>431.58</td>
</tr>
<tr>
<td>Wn1 = 0.4</td>
<td>567.2</td>
<td>2.472</td>
<td>18.78</td>
<td>112</td>
<td>439.88</td>
</tr>
<tr>
<td>Wn1 = 0.5</td>
<td>567.3</td>
<td>2.473</td>
<td>19.24</td>
<td>112.3</td>
<td>450.37</td>
</tr>
<tr>
<td>Wn1 = 0.6</td>
<td>567.1</td>
<td>2.473</td>
<td>19.73</td>
<td>113.3</td>
<td>464.37</td>
</tr>
<tr>
<td>Wn1 = 0.7</td>
<td>567.6</td>
<td>2.474</td>
<td>20.03</td>
<td>114.5</td>
<td>475.51</td>
</tr>
<tr>
<td>Wn1 = 0.8</td>
<td>567.5</td>
<td>2.474</td>
<td>20.26</td>
<td>115.9</td>
<td>486.12</td>
</tr>
<tr>
<td>Wn1 = 0.9</td>
<td>567.4</td>
<td>2.474</td>
<td>20.58</td>
<td>117.8</td>
<td>500.87</td>
</tr>
<tr>
<td>Wn1 = 1.0</td>
<td>567.8</td>
<td>2.474</td>
<td>20.9</td>
<td>119.9</td>
<td>516.71</td>
</tr>
</tbody>
</table>

![Figure A.19 Delays for Wn1](image-url)
The figure above can tell us which width can get minimum power-delay product, so we can write down the value of that one.

- Change the size of Wn7 and keep the sizes of other transistors as before:

Table A.14 Power-delay products for Wn7

<table>
<thead>
<tr>
<th>Width (µm)</th>
<th>$I_{av}$ (nA)</th>
<th>$I_{i}$ (nA)</th>
<th>$I_{incom}$ (µA)</th>
<th>Delay (ps)</th>
<th>PDP ($10^{-20}$ J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wn7 = 0.3</td>
<td>567.5</td>
<td>2.474</td>
<td>17.06</td>
<td>155.9</td>
<td>564.09</td>
</tr>
<tr>
<td>Wn7 = 0.4</td>
<td>567.1</td>
<td>2.473</td>
<td>17.48</td>
<td>143.4</td>
<td>529.66</td>
</tr>
<tr>
<td>Wn7 = 0.5</td>
<td>567</td>
<td>2.473</td>
<td>17.89</td>
<td>134.4</td>
<td>506.34</td>
</tr>
<tr>
<td>Wn7 = 0.6</td>
<td>567.2</td>
<td>2.473</td>
<td>18.33</td>
<td>127.5</td>
<td>490.49</td>
</tr>
<tr>
<td>Wn7 = 0.7</td>
<td>567.3</td>
<td>2.473</td>
<td>18.76</td>
<td>122.3</td>
<td>479.91</td>
</tr>
<tr>
<td>Wn7 = 0.8</td>
<td>567.1</td>
<td>2.472</td>
<td>19.14</td>
<td>118</td>
<td>471.08</td>
</tr>
<tr>
<td>Wn7 = 0.9</td>
<td>567</td>
<td>2.472</td>
<td>19.57</td>
<td>114.8</td>
<td>467.19</td>
</tr>
<tr>
<td>Wn7 = 1.0</td>
<td>567.1</td>
<td>2.471</td>
<td>20</td>
<td>112.2</td>
<td>465.28</td>
</tr>
<tr>
<td>Wn7 = 1.1</td>
<td>567.1</td>
<td>2.471</td>
<td>20.43</td>
<td>110.2</td>
<td>465.51</td>
</tr>
<tr>
<td>Wn7 = 1.2</td>
<td>567.2</td>
<td>2.47</td>
<td>20.87</td>
<td>108.6</td>
<td>467.34</td>
</tr>
<tr>
<td>Wn7 = 1.3</td>
<td>566.9</td>
<td>2.47</td>
<td>21.3</td>
<td>107.5</td>
<td>470.92</td>
</tr>
<tr>
<td>Wn7 = 1.4</td>
<td>567</td>
<td>2.47</td>
<td>21.73</td>
<td>106.5</td>
<td>474.78</td>
</tr>
<tr>
<td>Wn7 = 1.5</td>
<td>567</td>
<td>2.469</td>
<td>22.18</td>
<td>105.7</td>
<td>479.76</td>
</tr>
<tr>
<td>Wn7 = 1.6</td>
<td>567</td>
<td>2.47</td>
<td>22.64</td>
<td>105.4</td>
<td>487.14</td>
</tr>
<tr>
<td>Wn7 = 1.7</td>
<td>566.9</td>
<td>2.47</td>
<td>23.12</td>
<td>105</td>
<td>494.37</td>
</tr>
<tr>
<td>Wn7 = 1.8</td>
<td>566.7</td>
<td>2.47</td>
<td>23.63</td>
<td>104.9</td>
<td>503.52</td>
</tr>
<tr>
<td>Wn7 = 1.9</td>
<td>567</td>
<td>2.47</td>
<td>24.16</td>
<td>105.1</td>
<td>514.53</td>
</tr>
</tbody>
</table>
Figure A.21 Delays for Wn7
The minimum value of PDP corresponds to the width of $W_{n7}$ at 1\(\mu\)m. With all the transistor widths recorded, we can get the minimum power-delay product for CCFF.
Upphovsrätt

Detta dokument hålls tillgängligt på Internet – eller dess framtida ersättare – under 25 år från publiceringsdatum under förutsättning att inga extraordinära omständigheter uppstår.

Tillgång till dokumentet innebär tillstånd för var och en att läsa, ladda ner, skriva ut enstaka kopior för enskilt bruk och att använda det oförändrat för icke-kommersiell forskning och för undervisning. Överföring av upphovsrätten vid en senare tidpunkt kan inte upphäva detta tillstånd. All annan användning av dokumentet kräver upphovsmannens medgivande. För att garantera äktheten, säkerheten och tillgängligheten finns lösningar av teknisk och administrativ art.

Upphovsmannens ideella rätt innefattar rätt att bli nämnd som upphovsman i den omfattning som god sed kräver vid användning av dokumentet på ovan beskrivna sätt samt skydd mot att dokumentet ändras eller presenteras i sådan form eller i sådant sammanhang som är kränkande för upphovsmannens litterära eller konstnärliga anseende eller egenart.

För ytterligare information om Linköping University Electronic Press se förlagets hemsida http://www.ep.liu.se/

Copyright

The publishers will keep this document online on the Internet – or its possible replacement – for a period of 25 years from the date of publication barring exceptional circumstances.

The online availability of the document implies a permanent permission for anyone to read, to download, to print out single copies for your own use and to use it unchanged for any non-commercial research and educational purpose. Subsequent transfers of copyright cannot revoke this permission. All other uses of the document are conditional on the consent of the copyright owner. The publisher has taken technical and administrative measures to assure authenticity, security and accessibility.

According to intellectual property law the author has the right to be mentioned when his/her work is accessed as described above and to be protected against infringement.

For additional information about the Linköping University Electronic Press and its procedures for publication and for assurance of document integrity, please refer to its www home page: http://www.ep.liu.se/

© Yuan Yongyi