國　立　交　通　大　學

電子工程學系　電子研究所碩士班

碩士論文

90 奈米混合臨界電壓標準元件庫設計及特性化

90nm Mixed-Threshold Voltage Standard Cell Library Design and Characterization

指導教授：周 世 傑　博士

研 究 生：林 俊 誼

中 華 民 國　九 十 六　年　十　月

90 奈米混合臨界電壓標準元件庫設計及特性化

90nm Mixed-Threshold Voltage Standard Cell Library Design and Characterization

研究生：林俊誼　　Student: Jyun-Yi Lin

指導教授：周世傑　　Advisor: Dr. Shyh-Jye Jou

國　立　交　通　大　學

電子工程學系　電子研究所碩士班

碩士論文

A Thesis

Submitted to Department of Electronics Engineering & Institute of Electronics

College of Electrical and Computer Engineering

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of Master

In

Electronics Engineering

October 2007

Hsinchu, Taiwan, Republic of China

中華民國　九十六　年　十　月

90 奈米混合臨界電壓標準元件庫設計及特性化

學生: 林俊誼　　指導教授: 周世傑

國立交通大學

電子工程學系　電子研究所碩士班

摘要

隨著製程的進步以及各種攜帶型電子產品需求的增加，功率消耗對於這些產品變得相當的重要，例如：使用太陽能電池的助聽器 、新型手機…等。在本篇論文中，我們提出首先介紹關於深次微米 CMOS 標準元件庫的時脈效能和能量特性化流程的概論。接著我們提出一種在電路中使用混合臨界電壓電晶體的方式來取代單一臨界電壓電晶體的方式，使電路能夠在不犧牲速度的情況下，達到低功率的效果。我們找出在拉升及拉降結構中延遲時間的關鍵路徑，以及關鍵路徑中最長延遲時間的關鍵電晶體。接著我們將關鍵電晶體置換成較低臨界電壓的電晶體，並且重新調整較低臨界電壓的電晶體尺寸，使新電路的速度能與原本的電路接近。利用這種方式，我們不需要增加額外的電路，也不需要更改電路架構，即可達到低功率的需求。此外大部分電晶體路徑中的漏電流將會被阻擋住。

我們利用這種混合臨界電壓的方式來建立 90 奈米低功率標準元件庫。接著利用這個低功率標準元件庫來合成電路並且和高臨界電壓標準元件庫的效能比較。我們的混合臨界電壓標準元件庫在動態功率消耗上可以節省 5%到 30%，在延遲時間功率乘積上可以節省 20%到 55%，而在面積方面因為佈局規則的限制，增加了 0%到 40%。

90nm Mixed-Threshold Voltage Standard Cell Library Design and Characterization

Student: Jyun-Yi Lin Advisor: Dr. Shyh-Jye Jou

Department of Electronics Engineering

Institute of Electronics

National Chiao Tung University

Abstract

With the advance of process technology and the increasing requirement of portable electric products, the power consumption of these products becomes very important. In this thesis, we first make the overview about the advanced characterization flow of timing and power in deep submicron CMOS standard cell library. Then, we propose a methodology using mixed-threshold voltage transistors in a circuit instead of single normal-threshold voltage transistors to reduce power consumption with the same timing performance. We find out the critical path and the critical transistors on the critical path that result in the longest delay time in the pull-up and pull-down networks, respectively. Then we replace the critical transistors with lower threshold voltage transistors and do resizing to meet the time performance of original circuits. Using this technique, we do not have to use additional transistors and do not change the structure of circuits to obtain the requirement of low power. Moreover, the leakage current is also blocked in most of the transistor paths.

We apply this mixed-threshold voltage methodology to establish our 90nm low power standard cell library. Then we use many design examples to compare the performance with the high-Vt standard cell library and make the conclusion that we can have around 5% to 30% dynamic power saving, 20% to 55% delay-power product saving and the area is 0% to 40% larger than the standard cells with single high-Vt transistors.

誌謝

本論文的完成，首先謝周世傑教授用心指導及修改，經由一次又一次與老師的討論中，獲得許多寶貴的經驗。藉由經驗的累積，讓我在研究的過程愈來愈順利。感謝劉建男老師、黃俊達老師、陳繼展經理在口試時給我的指導與建議，讓我的論文能夠更加完備，口試委員們詳盡的意見，補足我思慮不周之處。

除了老師之外，同學及朋友們在精神上給予我很大的鼓勵，周 group 的學長姐們，林志憲學長、林育群學長、陳筱筠學姐、王儷蓉學姐、胡嘉琳學姐、魏庭楨學長、朱昌敏學長、莊誌華學長、劉瑋昌學長、嚴紹維學長，在我研究遇到問題時，大力提供幫助。SPICE、阿樸、俊男、國光、晉欽、建君、小 VAN、篤雄、JUJU、秀逗、怡秀以及各位同學及朋友，有你們的陪伴幫助，讓我在兩年研究生活過得很充實，不論將來各自如何發展，我們的友誼長存。

最後，我想將完成碩士學位的光榮獻給我的家人，感謝他們的教誨及信任，無悔的付出讓我能夠沒有後顧之憂，盡全力發揮。

林俊誼

謹誌於　新竹

2007 十月

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction of Standard Cell Library

With the novel process technology is going to deep-submicron generation, the high integrated and high complex system on chip (SOC) design methodology becomes practical and popular. When circuit designers would like to use the cell-based design flow to design a digital chip, they have to ensure its specifications such as the timing performance, power consumption, area…etc of the chip could meet the requests of the circuit. A typical cell-based design flow is shown in Fig. 1.1. For timing specification, most designers and the synthesis tools use the method of static timing analysis (STA) to verify their timing performance. If the timing performance doesn't meet the specification, circuit designers or synthesis tools will replace some cells in the circuit or change the architecture of the circuit to improve the timing performance. Then, they would reiterate above steps until the timing performance meets the specification. The methods to meet the design performance of power and area are like that of timing tuning process. After verifying the characteristics of the chip meets the specification, designers have to use an automatic placement and routing tool to draw the circuit layout in order to tape out the chip.

The standard cell library plays a very important role in the cell-based design flow

from circuit synthesis step to automatic place and route step. The synthesis tool can implement a large circuit with the cells supplied by the cell library. The standard cell library also supplies the timing, power, and area information of cells to let the synthesis tool optimize the circuit design to meet the specifications.

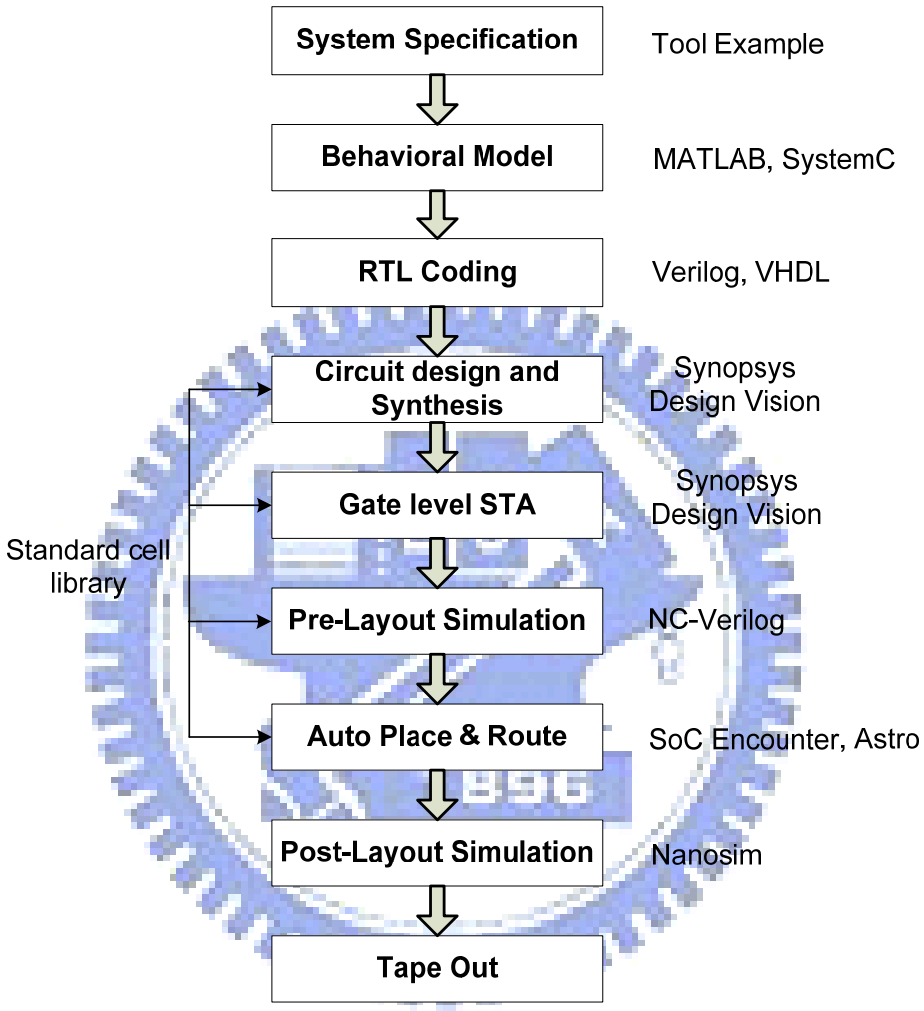


Fig. 1.1 Cell-based design flow

1.2 Deep-submicron Circuit Design Issues

As the novel technology process progress, the total power dissipation is not only determined by the switching power and the internal power dissipation but also the leakage power dissipation. When the transistor size scales down, the supply voltage has to be scaled down at the same time in order to save power and considers the

problem of device reliability. With the decreasing of the supply voltage, the threshold voltage has to be scaled down to meet the performance requirements. However, low threshold voltage will increase the sub-threshold leakage current and it will dominate the total leakage power at 90nm and below deep-submicron technology processes. In 90nm process, the threshold voltage between 0.1V and 0.2V causes l0nA-order sub-threshold leakage current per logic gate in a standby mode, which leads to l0mA standby current for 1M-gate VLSIs [9]. Therefore, we have to reduce the leakage power very carefully when we design low power circuits at the deep submicron process.

1.3 Motivation and Goals

Many commercial standard cell libraries have been proposed. In the cell base design flow, we can synthesize our design by using the standard cell libraries supported by the foundry factory. However, we can not modify the data in the commercial cell libraries to improve the circuit performance for special design requirement. Therefore, we would like to build a procedure to create a low power standard cell library. With this procedure, we can add some properties that we need in the library to meet our research requirement. For example, if we design a new D-type flip flop or Latch, we can add these new cells in our standard cell library for others to use at any time.

Furthermore, with the growing use of solar batteries, portable and wireless electronic systems, the designer has to reduce the power consumption in the novel VLSI circuit and system designs [1]. Thus, we would like to use low power design technique in our standard cells in order to create a low power standard cell library. There are many methodologies and architectures for low power circuit design. We

choose the low power design methodology on circuit and logic level in our low power cell library. In order to reduce the loading of the designer to design a low power circuit such as finding the critical path of the circuit、designing additional circuits used in the standby mode, etc, we want to create low power cells without changing the schematics of cells. Therefore, we would use one of low power design methodologies of multiple threshold voltage (Vt), mixed-Vt, in a cell by replacing the MOSs on the critical path and resizing them to establish our low power standard cell library. Besides, in order to avoid the over design causing the waste of area and power, we characterize this mixed-Vt low power standard cell library by the advanced characterization tool, Parex. Finally, we use our mixed-Vt low power standard cell library to synthesize several design examples and compare the timing and power performance with the single high-Vt standard cell library to demonstrate that our mixed-Vt method is very effective.

1.4 Thesis Organization

In this thesis, a design flow and methodology of Mixed-Vt 90nm CMOS standard cell library is presented. Design and implementation results are demonstrated to show the performance of the proposed mixed-Vt cell library. The thesis organization is described as follows:

Chapter 2 introduces power dissipation of CMOS circuits and the basics of a standard cell library. We will also overview the improvement of present standard cell library in advanced technology.

The characterization flows of time and power in a standard cell library are demonstrated in Chapter 3.

Based on present commercial standard cell library, we will show the proposed mixed-Vt low power standard cell library. We will also propose the methodology of low power cell design and establish the flow of setting up the low power standard cell library.

Finally, we use several design examples to demonstrate our low power standard cell library in Chapter 5 and a conclusion is made in Chapter 6.

Chapter 2

Background Overview

2.1 Power Dissipation in CMOS circuits

We know that the power consumption of CMOS circuits is composed of two components, the dynamic power and the static power. The power dissipation can be expressed as:

$$P_{total} = P_{dynamic} + P_{static} \quad (2.1)$$

where P_{total} is the total power dissipation, $P_{dynamic}$ is the dynamic power and P_{static} is the static power dissipation.

2.1.1 Dynamic Power Dissipation

Dynamic power consumption occurs when the input signal transition results in the signal state transient in the output. At this time, the power that the circuit exhausts is called dynamic power consumption. The following equation shows the power dissipation components of dynamic power.

$$P_{dynamic} = P_{switching} + P_{internal} + P_{short-circuit} \quad (2.2)$$

where $P_{dynamic}$ is the dynamic power, $P_{switching}$ is the switching power, $P_{internal}$ is the internal power and $P_{short\text{-}circuit}$ is the short circuit power dissipation.

For dynamic power consumption, there are three components. One is the power in using to charge or discharge the output ($P_{switching}$) and the parasitic capacitance ($P_{internal}$)(see Fig. 2.1). The other is short circuit power (see Fig. 2.1) due to the non-zero rise and fall time of input waveforms. This situation will cause the N/PMOS conducting simultaneously when input signal transits and the current will flow from VDD into ground. In the high speed circuit, the amount of short-circuit power can be ignored due to the fast transition time. The last one is the internal power component. It results from the power supply charges/discharges the internal parasitic capacitance.

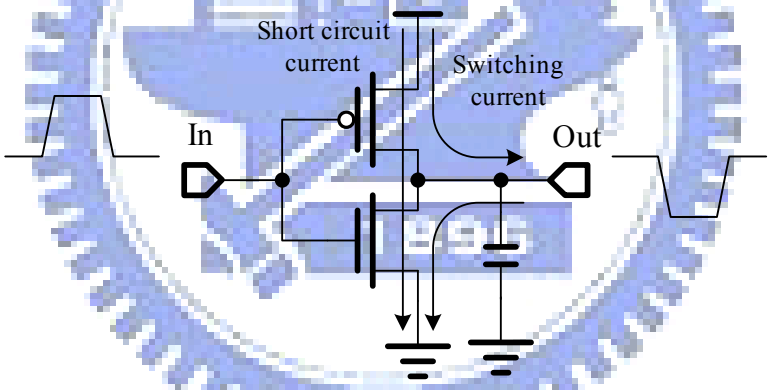


Fig. 2.1 Switching power example

2.1.2 Static Power Dissipation

The static power of a CMOS circuit is determined by the leakage current through each transistor. It can be expressed as:

$$P_{static} = I_{static} \cdot V_{DD} \tag{2.3}$$

Let us take the static CMOS inverter shown in Fig. 2. for instance. When the input signal is at the static state, a continuous high or low voltage level, one MOS of

the inverter will be always turn on and the other one will be always turn off. So we can realize that there will be no switching current at this time in the ideal situation. But the secondary effects of leakage current we ignored before becomes more and more significant in the deep submicron process. The leakage sources for the static CMOS circuits are illustrated in Fig. 2.2. These secondary effects including the sub-threshold leakage, the PN reverse bias junction leakage, the Gate Induced Drain Leakage (GIDL) and the punch-through gate oxide tunneling cause small static current flowing through the turned off transistor. These leakage currents will produce power consumption called leakage power. These static currents were ignored in the 0.18μm or the earlier process but they will occupy more portions of the total power consumption in the 90nm and below process.

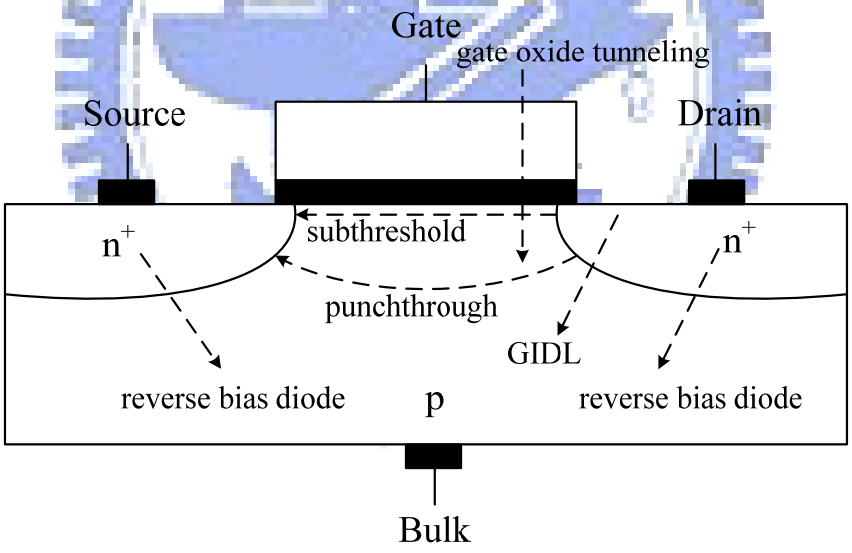


Fig. 2.2 Static CMOS leakage sources

Designers have to consider many low power design methods to diminish leakage power for low power circuits.

Before introducing the low power design methods, we have to know that what the leakage sources of the MOS are and the reason cause these leakage currents. The leakage sources of the static CMOS circuits in deep submicron process are illustrated

in Fig. 2. and we will introduce them respectively.

– PN reverse bias diode junction leakage

It is due to the minority carrier drift near the edge of the depletion region and the electron-hole pair generation in the depletion region. It is very small and can be ignored. When the electric field across a reverse-biased p-n junction is continuously high, significant current flow can occur due to band-to-band tunneling. The PN reverse bias diode junction leakage current is about 0.1nA with 1 V reverse bias voltage and 75℃ [19].

– Gate Induced Drain Leakage (GIDL)

It occurs at negative VG and high VD. Where VG is the voltage applied to the gate of a transistor and VD is the voltage applied to the drain of a transistor. It is due to the high electric field under the gate and drain overlap region, which results in the band-to-band tunneling [1-2]. The gate induced drain leakage (GIDL) is about 1nA with VG= -1 V, VD=1.5 V, physical gate width is 1μm and gate length is 100 nm [18].

– Sub-threshold leakage

It is the weak inversion current between source and drain of MOS transistor when the gate voltage is less than the threshold voltage. It increases exponentially with the reduction of the threshold voltage. So it is the critical for low voltage low power (LVLP) CMOS circuit design. The short channel effect (SCE), such as Vth roll-off and Drain-Induced-Barrier-Lowering (DIBL), make the sub-threshold leakage even worse. The sub-threshold leakage current is about 5nA with VGS= 0.5 V, gate width = 1μm and gate length= 90nm in 90nm process [17]. Where VGS is the gate related to source voltage of a transistor.

– Punch-through

It occurs when the drain voltage is high and the drain and source depletion

regions approach each other. In the punch-through condition, the gate totally loses the control of the channel current and the sub-threshold slope starts to degrade.

– Gate Oxide Tunneling

It is due to the high electric filed in the gate oxide, includes Fowler-Nordheim tunneling through the oxide band and the direct tunneling through the gate. Fowler-Nordheim tunneling is negligible for the normal device operations, but the direct tunneling is important when the oxide thickness is less than 23nm [1-2]. The gate oxide tunneling current is about 1nA with VGS= 1 V, gate width = 10μm and gate length= 10um in 90nm process [17]. Where VGS is the gate related to source voltage of a transistor.

2.2 Common Formats of Standard Cell Library

Currently, there are two different kinds of standard cell library. The first type is called Advanced Library Format (ALF) [3], and the second type is called Liberty (.lib) [4]. ALF is an IEEE standard 1603-2003. It is a modeling language for library elements used in IC technology. The content of ALF are electrical, functional, and physical models of technology-specific libraries for cell-based and block-based design in a formal language suitable for electronic design automation (EDA) application tools targeted for design and analysis of an IC [3].

Liberty is proposed by Synopsys Corporation. It is the most popular library format in the cell-based design flow now. The main difference between Liberty and ALF is the ranges that they can model. ALF can describe many kinds of characterizations from functional model to physical model. But Liberty is only focusing on the model of timing, power, and signal integrity. We will introduce Liberty more detailed in the following sections.

2.3 Brief Introduction to Liberty File

There are some basic attributes declared in the beginning of the .lib file. These basic attributes define many specific characteristics. In every cell, there are attributes of timing, power, area, capacitance, and footprint in the characteristic descriptions. We will introduce these characteristic descriptions in accordance with each cell in the following paragraph.

- Footprint: Each cell in the standard cell library has its own name. The name is composed of function and driving ability. We take the NAND2X4 for instance. NAND2X4 can separate into two parts. One is NAND2 and the other one is X4. NAND2 stands for its function and X4 means its driving ability. Although NAND2X2 and NAND2X4 have different cell name, we know they have some relations. From the above explanation, we know that NAND2X2 and NAND2X4 have the same function and the numbers of input/output pin, but the synthesis tools cannot know this characteristic. So we have to use the footprint attribute to let synthesis tools know that these two cells have the same function and the same numbers of input/output pin in the .lib file even if they have different names and driving abilities. In the cell-based design flow, the synthesis tool can replace the cell with the same footprint which has the more proper driving ability.
- Area: This attribute can help the designer to estimate the area of the circuit roughly in the synthesis stage. Users can determine if the chip area could be accepted or they have to change some gates in the circuit.
- Power: This attribute declares the power consumption including the active and static power dissipation. Designers can use this attribute to Users can use this attribute to estimate the power consumption of the circuit and see if the power dissipation can meet the specification.

– Capacitance: This attribute records the equivalent capacitances of input and output pins. It can help designer to analyze the circuit and adjust the circuit to meet the specification.

– Timing: This attribute records the transition time and propagation delay time with different output loading when the cell is transiting. It can offer the delay information for synthesis tool and static timing analysis (STA). After the complete analysis of STA, users can determine if their circuits can operate normally at the specified clock.

We take a section of our standard cell for instance (see Fig. 2.3). By this example, we can see the above attributes in a practical .lib file.

```
cell (NAND2X1) {                          //cell name
   cell_footprint : nand2;                //footprint attribute
   area : 3.92;                           //area attribute
   pin(A) {                               //input pin group
      direction : input;                  //direction of pin
      capacitance : 0.00118294;           //input capacitance (pF)
   }
   pin(B) {                               //input pin group
      direction : input;                  //direction of pin
      capacitance : 0.00109301;           //input capacitance (pF)
   }
   pin(Y) {                               //output pin group
      direction : output;                 //direction of pin
      capacitance : 0.0;                  //output capacitance (pF)
      function : "(!(A B))";              //logic function related to output pin
      internal_power() {                  //power attribute
         related_pin : "A";               //output transition related to input pin
         rise_power(ptable2){             //rise power lookup table*
            values("0.0009,0.0010",\      //values of internal power (pJ)
                   "0.0007,0.0007");}
         fall_power(ptable2){             //fall power lookup table*
            values("0.0005,0.0007",\      //values of internal power (pJ)
                   "0.0006,0.0006");}
      }
      timing() {                          //timing attribute
         related_pin : "A";               //output transition related to input pin
         timing_sense : negative_unate;   //timing sense**
         cell_rise(table2){               //rise propagation delay time lookup table*
            values ("0.026925,0.037688",\ //values of delay time (ns)
                    "1.2158,1.219");}
         cell_fall(table2){               //fall propagation delay time lookup table*
            values ("0.0226,0.033009",\   //values of delay time (ns)
                    "0.95619,0.96855");}
         rise_transition(table2){         //rise transition time lookup table*
            values ("0.0175,0.020095",\   //values of delay time (ns)
                    "1.1532,1.1588");}
```

```
        fall_transition(table2){                    //fall transition time lookup table*
          values ("0.013014,0.013843",\             //values of delay time (ns)
                  "0.85331,0.87062");}
    }
    internal_power() {                              //power attribute
      related_pin : "B";                            //output transition related to input pin
      rise_power(ptable2){                          //rise power lookup table*
        values("0.0008,0.0011",\                    //values of internal power (pJ)
               "0.0011,0.0009");}
      fall_power(ptable2){                          //fall power lookup table*
        values("0.0006,0.0006",\                    //values of internal power (pJ)
               "0.0009,0.0007");}
    }
    timing() {                                      //timing attribute
      related_pin : "B";                            //output transition related to input pin
      timing_sense : negative_unate;                //timing sense**
      cell_rise(table2){                            //rise propagation delay time lookup table*
        values ("0.031405,0.042728",\               //values of delay time (ns)
                "1.2167,1.2238");}
      cell_fall(table2){                            //fall propagation delay time lookup table*
        values ("0.024612,0.027471",\               //values of delay time (ns)
                "0.96379,0.96674");}
      rise_transition(table2){                      //rise transition time lookup table*
        values ("0.020571,0.021102",\               //values of delay time (ns)
                "1.1477,1.1341");}
      fall_transition(table2){                      //fall transition time lookup table*
        values ("0.013076,0.014476",\               //values of delay time (ns)
                "0.85792,0.85046");}
    }
    max_capacitance : 0.0314066;                    //maximum output load capacitance (pF)
  }
  cell_leakage_power : 180.38;                      //leakage power (pW)
  leakage_power () {
  when   : "A   B" ;
  value : 83.255;
  }
  leakage_power () {
  when   : "A   !B" ;
  value : 1291.2;
  }
  leakage_power () {
  when   : "!A   B" ;
  value : 108.62;
  }
  leakage_power () {
  when   : "!A   !B" ;
  value : 180.38;
  }
}
```

* Lookup table will be introduced in section 2.3.3.

**A function is said to be *unate* if a rising (or falling) change on a positive (or negative) unate input variable causes the output function variable to rise (or fall) or not change.

Fig. 2.3 Example of .lib file

2.3.1 Classification of Power [4] [5]

We know that the total power is composed of the dynamic power and the static power. The dynamic power consists of switching power and internal power. We classify these components respectively in the following paragraph.

(1) Switching power: The switching power can be expressed as :

$$P_{Switching} = \frac{V_{DD}^{\ 2}}{2} \sum_{\forall Nets(i)} \left(C_{Load_i} \cdot ToggleRate_i \right) \quad (2.4)$$

The logic transitions at each net will charge/discharge the load capacitance connected to it. At this time, circuits have to consume the power and we call this kind of power consumption as switching power consumption. As shown in Eqn. (2.4), we know that the switching power is related to supply voltage, load capacitance, and toggle rate. First, the operating condition of the cell library with Synopsys model will record the supply voltage. Second, we can derive the load capacitance from the input capacitance of the device model and the load capacitance of the net. Because we can derive the information of the toggle rate of each node by the vectors simulation and statistics implemented, we don't derive the switching power from the power model of the device but from the calculation by the synthesis tool. By the above description, if we have the information of correct load capacitance and the toggle rate, the synthesis tool can calculate the switching power automatically.

So we just have to focus on the power models of the internal power and the leakage power. The switching power will be calculated by the synthesis tool.

(2) Internal power:

The definition of the internal power defined by Synopsys consists of short-circuit power and the power consumed by charging/discharging internal parasitic capacitance

of the cell. The internal parasitic capacitances include the interconnection of the MOSs connected by metal lines and the source/drain to substrate equivalent capacitance. The input transition will charge/discharge these internal parasitic capacitances. This kind of power is called internal power consumption. When we calculate the internal power consumption, we have to sum up the power of each pin including the input and output pin. The synthesis tool will sum up a power value from the internal power model of each pin when the pins transit. The unit of the internal power value that the synthesis tool calculates is energy not power as shown in Eqn. (2.5).

$$P_{InternalTotal} = \sum_{\forall CellPins(i)} \left(E_i \cdot ActivityFactor_i \right) \quad (2.5)$$

where the E_i is the energy of each pin including the input and output pin and the $ActivityFactor_i$ is the toggle rate of each pin.

Because we have to sum up the power of each pin including the input and output pin, we need to avoid double calculating the internal power consumption in each pin group when we establish the model. Therefore, the internal power model is just related to the input transition time.

When we calculate the power consumption which is produced by the output pin transition, we can realize that the power of the output pin is related to input transition time and output load capacitance. Even if there is no output transiting, there is still power consumption with the transient of the input pin. We can view it as this power consumption is contributed by the input pin and record this value as the power of the input pin. The two indices of output pin power consumption and the two indices of the pin- to-pin delay are the same with each other, so we can characterize the output pin power and pin-to-pin delay at the same time. We demonstrate the steps to calculate the power of the output pin.

First we record the value of power consumption during characterization procedure. Then we have to subtract the power of the input pin and the switching power of the output load capacitance from this value. Thus we can get the power of the output pin and avoid double calculating the power consumption by the subtraction step.

We take a common master-slave DFF without set and reset pin for instance and we set the following situation. The input pin D remains constantly and CLK remains the transient state. So the output pin Q will not change its state. But we know that the transistors in this cell will still turn on and off even if the output pin doesn't transit due to the conduction path inside the cell. So we realize that there is also power consumption at this time. In this case, we suppose that the output pin does not transit its state if we want to measure the input pin power. We know that the power of the input pin is only related to the input transition time but not related to the output load capacitance.

On the other hand, the transient of the output pin is related to the input pin CLK but not related to the input pin D in this case. So the power consumption of the output pin that we will measure would contain the power consumption produced by the output pin transition and the input CLK transition at the same time. If we only want to get the power of the output pin, we have to dismiss the power produced by the input CLK in order to avoid double calculating. Through these steps, the synthesis tool can calculate the power consumption produced by each pin respectively. By the above description, we know that if a input pin transit its state but the output pin doesn't, the power consumption is viewed as contributed by input pin and if the input pin and output pin transit their states simultaneously, the synthesis tool will calculate the power consumption by summing up the values from the lookup table of the input and output pins.

(3) Leakage power:

The leakage power can be expressed by the following equation:

$$P_{LeakageTotal} = \sum_{\forall Cells(i)} P_{CellLeakage} \tag{2.6}$$

The leakage power has three main segments: The reverse junction current of a MOS between drain (or source) to substrate, the sub-threshold current and the DC current. The method that is used to measure the leakage power is assigning the static input vectors to the cell. As the process scales down to deep submicron, the leakage power occupies larger portion to total power consumption and the leakage paths in the cell becomes more and more. Actually the leakage power of a cell is quite different when we feed the cell with the different input vectors. But it only records the worst case of the leakage power in the present commercial standard cell library. In our standard cell library, we would like to record the leakage power of all combinations of static input vectors for a more accurate leakage model.

We take the 2-input NOR for instance (see Fig. 2.4). The probable combinations of the input signals A and B are 00, 01, 10, 11, respectively, and the leakage power will vary with the different input signals. This phenomenon will be more obvious in complicated cells.

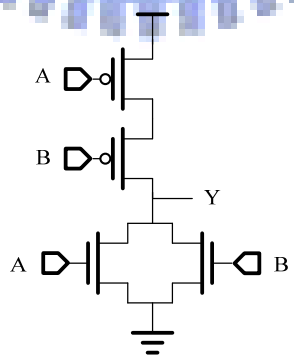


Fig. 2.4 2-Input NOR gate

It only records the maximum leakage power in the present commercial standard cell library. But we realize that the leakage power becomes more dominant to total

power consumption and how to estimate the leakage power accurately is very important. If there is still only maximum leakage power stored in the novel standard cell library, it will estimate the total power consumption excessively. So we would like to establish our .lib file with the leakage power information that is input dependent.

2.3.2 Classification of Time

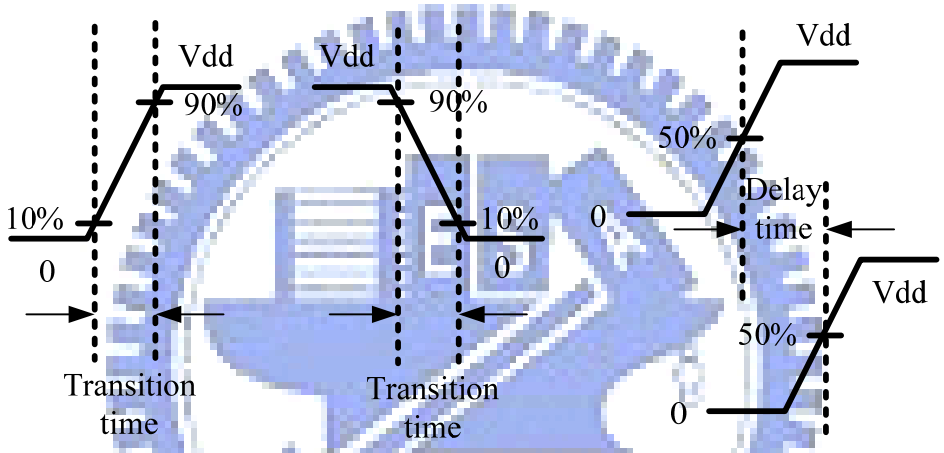


Fig. 2.5 Transition time and propagation delay time

There are two main kinds of timing performance. One is the transition time and the other one is the propagation delay time. The definition of the transition time is different in the library supported by different corporations. It may be the time for the signal level changes from 10% to 90%, 20% to 80% or 30% to 70 % of the power supply. But there is only one definition of the propagation delay time. It is defined by the time difference between 50% input signal to 50% output signal as shown in Fig. 2.5.

We take the combinational circuit for instance to explain how to express the propagation delay time. We can realize that the propagation delay time is related to the input transition time and output load capacitance and we express this relation in the Eqn. (2.8). This equation is a linear model where the unit of K_{load} is ns/pF, and the

unit of Cload is pF.

$$\text{Cell delay} = \text{pin to pin intrinsic delay} + K_{load}*\text{Cload} \quad (2.6)$$

There is another non-linear delay model (NLDM) using the input transition time and output load capacitance as two indices to establish the timing table. When we synthesize a circuit, the STA tool can find out the correct delay time from the NLDM table automatically. When the process advances to the deep submicron generation, NLDM is more accurate than the linear delay model. So NLDM is the most popular timing model in the present commercial standard cell libraries. Fig. 2.6 shows NLDM model. Designers can determine the approximate propagation delay by solving the A, B, C, and D coefficients of Eqn. (2.9). Then we insert the coefficient values into Eqn. (2.7) to determine z which is related to the fall propagation delay. We will demonstrate the flow about finding out the wanted fall propagation delay, Z.

$$Z = A + B \cdot x + C \cdot y + D \cdot x \cdot y \quad (2.7)$$

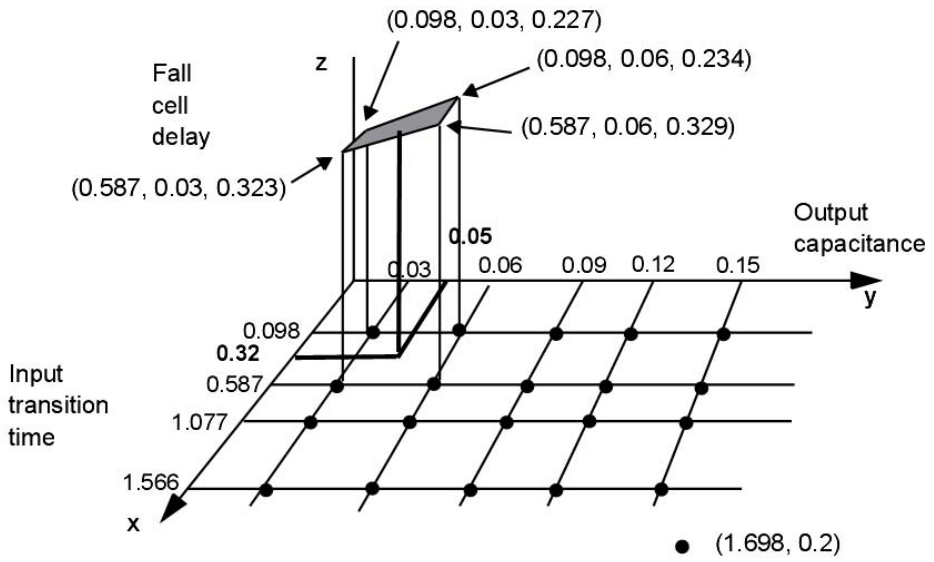


Fig. 2.6 Non-Linear Delay Model (NLDM) example

We can derive the coefficients A, B, C, and D first by the Gaussian elimination. Then we use these coefficients A, B, C, and D to find out the wanted fall cell delay, z. For example, we use following equations to find out the coefficients of Eqn. (2.7)

first.

$$0.227 = A + B * 0.098 + C * 0.03 + D * 0.098 * 0.03$$

$$0.234 = A + B * 0.098 + C * 0.06 + D * 0.098 * 0.06$$

$$0.323 = A + B * 0.587 + C * 0.03 + D * 0.587 * 0.03$$

$$0.329 = A + B * 0.587 + C * 0.06 + D * 0.587 * 0.06$$

$$\rightarrow A = 0.2006,\ B = 0.1983,\ C = 0.2399,\ D = 0.0677$$

Then we want to find out the fall cell delay, z, with input transition time, x, is 0.32 and output capacitance, y, is 0.05. So we take the above coefficients and x=0.32 and y=0.05 into Eqn. (2.7) to solve the fall cell delay, z, is 0.2771.

2.3.3 Create Look-up Table

NLDM is used to establish the timing model and power model in .lib file. It uses the lookup table to record timing and power information. The two indices of the lookup table are input transition time and output loading capacitance. We take a section of lookup table for instance (see Fig. 2.7). The title, cell rise, means that it is a look up table to record the propagation delay time of the cell when the output signal transits from low to high. The “tmg_ntin_oload _7x7” means that this look up table uses the tmg_ntin_oload _7x7 to be its template and we can know what the two indices, index_1 and index_2, stand for from that table. The “7x7” means that the table is a 7 by 7 matrix. The “index_1” represents that the input transition time and the unit is ns. The “index_2” represents the output load capacitance and the unit is pF. The values represent the propagation delay time of the cell with the combination of the different input transition time and different output loading capacitance and the unit of these values is ns. The internal power is also recorded in this kind of the lookup table. The values in the internal power table are energy and the unit is pJ.

```
cell_rise(tmg_ntin_oload_7x7) {
        index_1("0.011174, 0.051191, 0.091174, 0.142020, 0.263899,
                0.434953, 0.660005");//input transition time (ns)
        index_2("0.000000, 0.001516, 0.006839, 0.017004, 0.032841, \
                0.055062, 0.084301"); //output loading cap. (pF)
        values("0.008674, 0.014361, 0.032649, 0.066906, 0.119974, 0.194454, 0.292436",\
                "0.013862, 0.022335, 0.044519, 0.079757, 0.132954, 0.207426, 0.305412",\
                "0.017241, 0.027600, 0.053516, 0.092211, 0.146563, 0.221198, 0.319142",\
                "0.020991, 0.032762, 0.062407, 0.105392, 0.162707, 0.238360, 0.336490",\
                "0.028080, 0.042691, 0.078579, 0.129826, 0.195174, 0.276307, 0.376968",\
                "0.036270, 0.053742, 0.096277, 0.155833, 0.230843, 0.321114, 0.428598",\
                "0.045501, 0.065889, 0.115369, 0.183423, 0.268260, 0.369300, 0.486860");
    }//delay time (ns)
```

Fig. 2.7 Example of lookup table of the propagation delay time

We explain the flow that we use a lookup table to find out the characteristic values by Fig. 2. and use Table 2.1 to find out the propagation delay time of I_2 in Fig. 2.8. We know that the input transition time of I_2 can be derived from the output transition time of I_1 and the input capacitance of I_3 and the equivalent capacitance of the connecting net are summed to get the total equivalent output load capacitance of I_2. Then the tool can look up the propagation delay time of I_2 from its lookup table by these two known indices, input transition time and output load capacitance. For example, the input transition time of I_2 is 0.0092ns and the output load capacitance of I_2 is 0.0015pF, then, the propagation delay of I_2 is 0.0269ns.

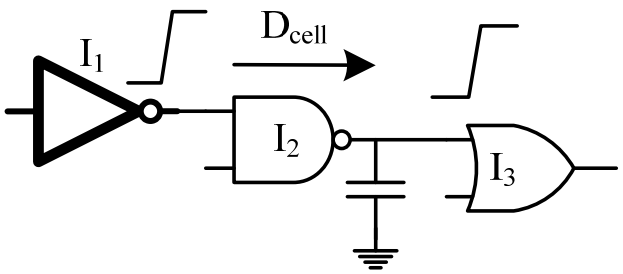


Fig. 2.8 Example of using lookup table

Table 2.1 The lookup table of 2-input NAND

Delay time (ns)		Input transition time (ns)		
		0.0092	0.0491	0.0892
output load capacitance（pF）	0.0015	0.0269	0.0376	0.0474
	0.0067	0.0696	0.0713	0.0818

2.4 Design of Standard Cell Library

We have introduced the overviews of the standard cell library, and the content of the liberty File. Then we will begin to design our standard cell library. We have to decide which kinds of cell shall be included in our standard cell library at the first step. Second we have to define the timing, power and area specification of each cell. We run the simulation and tune the size of each cell to meet the specification at the third step. Then we draw the layout of each cell according to the tuned size. After we draw the layout, we have to use tool to characterize the performances of the cell with layout parasitic. We create the data sheet, synthesis models by the characterized results and the HDL models for the synthesis and automatic placement and route tools at the sixth step. Finally, we have to verify if the standard cell library can function normally in the cell base design flow finally.

We use Fig. 2.9 to explain the flow of characterization. At the first step, we have to prepare the LPE netlist, the SPICE models, the environment setup file, the signal generation file, the header file and the load file. At the second step, we characterize the cells by using the characterization tool, PAREX, which is established by Industrial Technology Research Institute (ITRI). The characterization tool, PAREX, will produce the SPICE simulation file and produce simulation results automatically. Then we put Synopsys templates, verilog templates, and the layout data into PAREX translator and it will translate the results to Synopsys models, verilog models, and the manual for users.

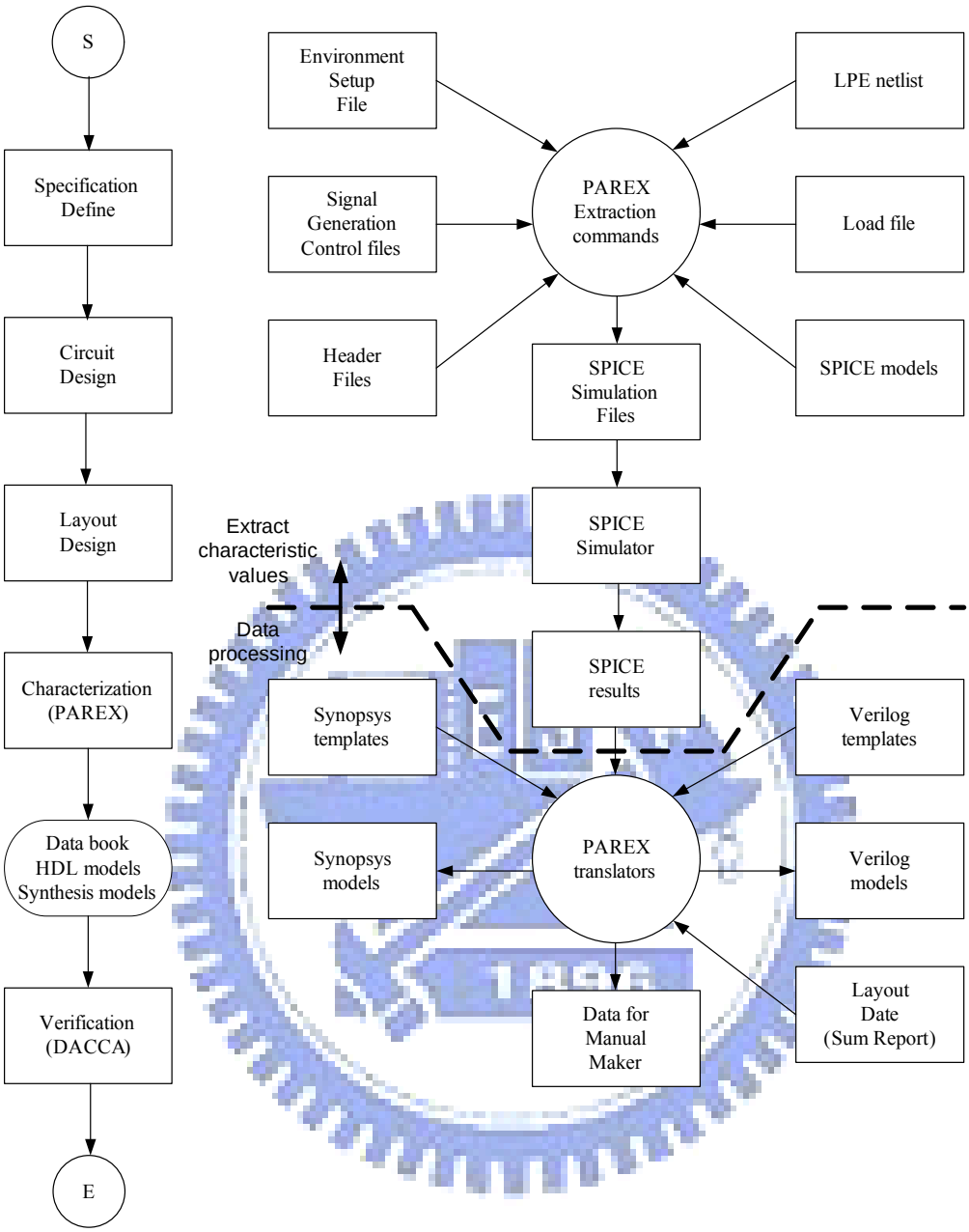


Fig. 2.9 The design flow of creating a standard cell library [5]

2.5 Summary

We have introduced different kinds of power dissipation in the CMOS circuit, the common format of standard cell library and the content of the liberty file. Then we explain the flow that we design and characterize the standard cell. We also introduce the NLDM lookup table of timing and power in the liberty file in this chapter.

Chapter 3

Timing and Power Model Characterization Flow

3.1 Timing Characterization Flow

3.1.1 Transition Time and Propagation Delay time

In this section, we will introduce the flow that we characterize the propagation delay time and transition time. The propagation delay time and transition time actually can be characterized at one simulation. The procedures that we characterize the propagation delay time and transition time are described as following:

(1) The first step: Determining the size of lookup table and choosing the ranges of indices. The way of selecting the ranges of indices can refer to the following rule.

(i) The minimum index of input transition time:

We use the cell with the largest driving ability to drive the cell with the smallest driving ability in the standard cell library. Then we can obtain the output transition time of the largest driving ability cell. It is defined as the minimum index of input transition time because this is the best timing case that one gate can drive the loading cell. Taking the inverter gate for instance,

we use the largest driving ability inverter to drive the smallest driving ability one. Then the output transition time of that largest driving ability inverter is defined as the minimum index of input transition time.

(ii) The maximum index of input transition time:

We can see that the curves of the input transition time vs. the propagation delay time in Fig. 3.1 are close to be a straight line on the larger segments of the input transition time. As long as the maximum index of transition time is large enough, we can calculate the output propagation delay time that is out of the maximum index by linear extrapolation.

(iii)The maximum index of output loading capacitance:

The rule that we define the maximum index of output loading capacitance is the same with the rule of defining the maximum index of input transition time. The curve of the output loading capacitance vs. the propagation delay time is also close to be a straight line on the larger segments of the output loading capacitance. So we define three times of the largest driving ability inverter input capacitance as the maximum index of output loading capacitance. We also can calculate the output propagation delay time that is out of the maximum index by linear extrapolation.

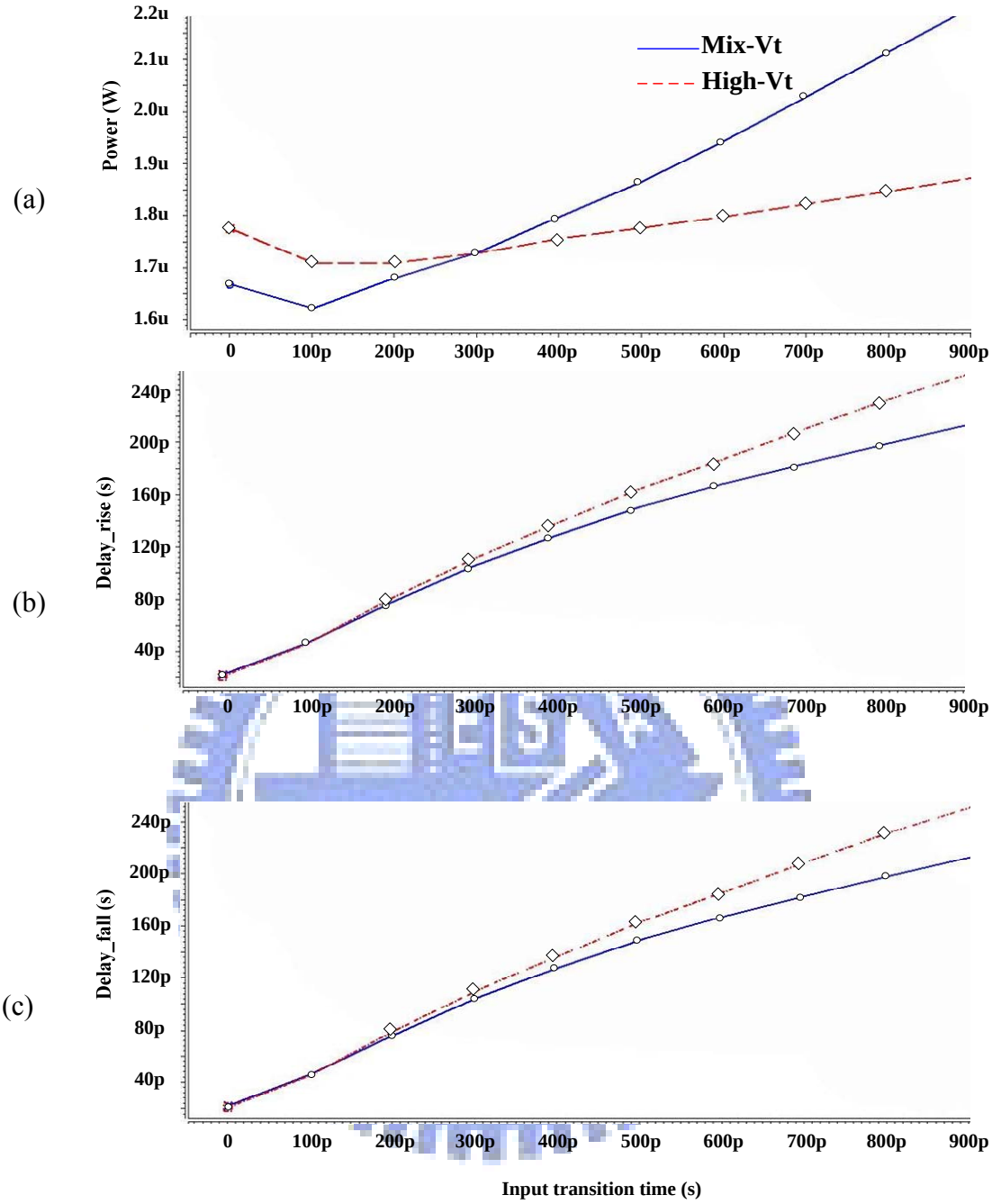


Fig. 3.1 (a) Power vs. input transition time (b) Delay_rise vs. input transition time and (c) Delay_fall vs. input transition time with fixed output load capacitance of an inverter

The size of the lookup table can be determined by the following observations. From Fig. 3.2, we can realize that the curves of the input transition time vs. the propagation delay time or the output loading capacitance vs. the propagation delay time are non-linear on the smaller index and linear on the larger index. So we use the tactic that we choose finer and more indices on the smaller index

region and fewer indices on the larger index region to establish the lookup table after determining the minimum and maximum values of index. By this way, we can describe the curve more accurately.

(2) The second step: Determining and importing the input pattern according to the functions of different kinds of cells

– We use a 3-input NAND gate as shown in the Fig. 3. to explain this step. Table 3.1 is the truth table of a 3-input NAND gate. We transit the specific input pin that we want to measure its transition time or propagation delay time and set the other pins on the high level. Through this way, we can obtain the transition time or propagation delay time of the specific input pin. If we want to characterize the timing performances related to input pin, in1, we have to import an input pattern which Y changes with in1 transition and keep other input at high level. Then we can measure the timing performances.

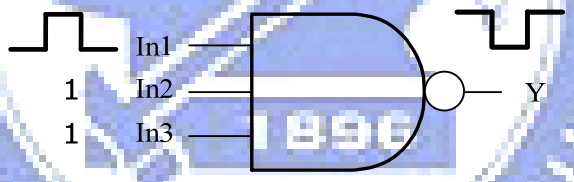


Fig. 3.2 3-input NAND schematic

Table 3.1 Truth table of 3-input NAND

In1	In2	In3	Y
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

We can find out the combinations of the input pattern that can fit our above requirement are the fourth pattern and the eighth pattern in the Table 3.1.

(3) The third step: Run the SPICE simulation and get the results.

Before we run the SPICE simulation, we have to know the following definitions about timing performance.

- Transition time: The definition of transition time is the time difference between 10% VDD and 90% VDD of the output signal. (It also can be from 20% to 80% VDD or from 30% to 70% VDD.)
- Delay time: The definition of delay time is the time difference between 50% VDD of the input signal and 50% VDD of the output signal.

Thus we follow the above three steps and definitions to measure the timing performance.

3.1.2 Input Capacitance

In this section, we will explain the flow that we measure the input capacitances of the cells

(1) Create a lookup table of output load capacitance vs. delay time:

In the first step, we will create a look up table of output load capacitance vs. delay time. We use an arbitrary inverter to drive many different values of capacitance and record the propagation delay time of every simulation. Then we can create the look up table that we want. Fig. 3. and Table 3.2 show what we have to do in this step.

(2) We use the same inverter that we use at the first step to drive the specific input pin of the cell under measurement and then record the delay time of every simulation. Fig. 3.4 shows the action to execute this step.

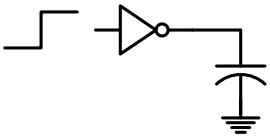


Fig. 3.3 Circuit diagram of creating capacitance vs. delay time look up table

Table 3.2 Capacitance vs. Delay table

Capacitance	1fF	2fF	3fF	4fF
Delay time	1ps	2ps	3ps	4ps

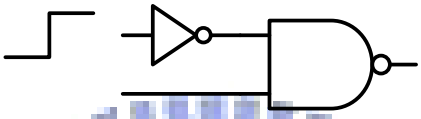


Fig. 3.4 Use the inverter in step 1 to drive circuit under measurement

(3) In this step, we have to compare the delay time that we have measured in the step 2 with the lookup table created in the step 1. Then, we can find the input capacitance of the cell under measurement from the corresponding delay time in the lookup table. If the delay time of the cell is not exact in the look up table, we can use the interpolation to get the input capacitance.

3.2 Power Characterization Flow

3.2.1 Internal Power

The flow that we characterize the internal power is the same with the transition time and propagation delay time. So we can characterize them at the same time. There is one thing that we have to notice. The values stored in the internal power table are energy and the unit of these values is joule. It can be expressed by the following equation.

Energy = Power * Time = V(VDD) * I(VDD) * Time

When the characterization tool uses SPICE to measure the power, it will record

the switching power and the energy of the VDD and VSS and at the same time. So it will subtract the switching power from the total power to get the internal power and avoid double calculating the internal power consumption in each pin group. At this time, the tool will also record the energy of VDD and VSS as the output transits, and sum up these two values to obtain the energy consumption in this transition.

3.2.2 Leakage Power

We know that the leakage power of a cell is quite different when we feed the cell with the different input vectors. So we would like to record the leakage power of all combinations of static input vectors. The flow that we characterize the leakage power is described in the following steps.

(1) We import all combinations of static input vectors and measure the power consumption.

(2) We record all power consumptions of the corresponding combinations of input vectors.

Table 3.3 is the leakage power of a 2-input NOR gate and Fig. 3.5 shows an example of input dependent leakage power in the .lib file.

Table 3.3 Leakage power of a 2-input NOR gate

Input AB	00	01	10	11
Leakage power (pW)	353.42	231.18	2795.10	90.46

```
cell_leakage_power : 2795.1;        //maximum leakage power of cell
   leakage_power () {               //input dependent leakage power block
   when   : "A   B" ;               //leakage power when input is 11
   value : 80.462;
   }
   leakage_power () {
   when   : "A   !B" ;              //leakage power when input is 10
   value : 2795.1;
   }
   leakage_power () {
   when   : "!A   B" ;              //leakage power when input is 01
   value : 231.18;
   }
   leakage_power () {
   when   : "!A   !B" ;             //leakage power when input is 00
   value : 353.42;
   }
```

Fig. 3.5 Example of input dependent leakage power format

3.3 Summary

We have introduced our characterization flow of timing and power model in this chapter. The major difference in our characterization flow is that we create the input dependent leakage power model. It can help the designers to estimate the leakage power more accurately especially in the 90nm or below process. We cooperate with ITRI-STC and use the automatic characterization tool – PAREX established by ITRI-STC to finish our characterization flow.

Chapter 4

Low Power Standard Cell Library

4.1 Overview of Low Power Standard Cell Design Methodology

There are many design methodologies to design low power circuits. It can be assorted as circuit/logic level, technology level, system level, algorithm level, and architecture level. In this thesis, we focus on circuit and logic level to design a low power standard cell library. We know that total power consumption includes both dynamic power and static power from Eqn. (2.1). Therefore, the total power consumption will be reduced as a result of diminishing either dynamic power consumption or static power consumption. The dynamic power consumption can be expressed by the follow equation:

$$P_{dynamic} = \alpha \cdot C_L \cdot V_{DD}^{\ 2} \cdot f \qquad (4.1)$$

Where α is toggle rate, C_L is output loading capacitance, V_{DD} is supply voltage and f is operating frequency.

From this equation, we know that we can curtail the switching activity of the nets, output load capacitance, supply voltage, and operating frequency to reduce dynamic power consumption. The main methods that we can do to achieve this target in the

standard cell library is shrinking the widths of devices or using low supply voltage cells. Shrinking the widths of devices can also reduce the parasitic capacitance and gate capacitance of the device. It is equivalent to reduce the output load capacitance C_L of the cell. The other method, using low supply voltage, would reduce the performance of cells and the dynamic power consumption at the same time. So, designers can use multiple supply voltages in a chip. The final target is to reduce the total power consumption while meet the required performance. Of course we can also reduce the power consumption by diminishing the leakage power. Eqn. (2.3) provides the information that the leakage power is related to leakage current directly. Thus, we can decrease leakage power by reducing leakage current. We will introduce several techniques to reduce leakage current in the following sections.

4.1.1 Multiple Threshold Voltage Circuit

Multiple-threshold CMOS circuit means that there are at least two different kinds of threshold transistors in a chip. Transistors with different threshold voltages have distinct characterizations. High threshold transistors are used to suppress sub-threshold leakage current, but it will degrade the performance seriously. The utility of low threshold transistors is to achieve high performance, but its sub-threshold leakage current is much greater than the high threshold transistors. The effect of standard threshold transistors is between low and high threshold transistors. According to the above description about multiple threshold technology, there have been several proposed multiple thresholds CMOS design techniques.

The first type is Multi-threshold-Voltage CMOS (MTCOMS) circuit which was proposed by inserting high threshold devices in series to low-Vth circuitry [7]. Fig. 4.1(a) shows the schematic of a MTCMOS circuit.

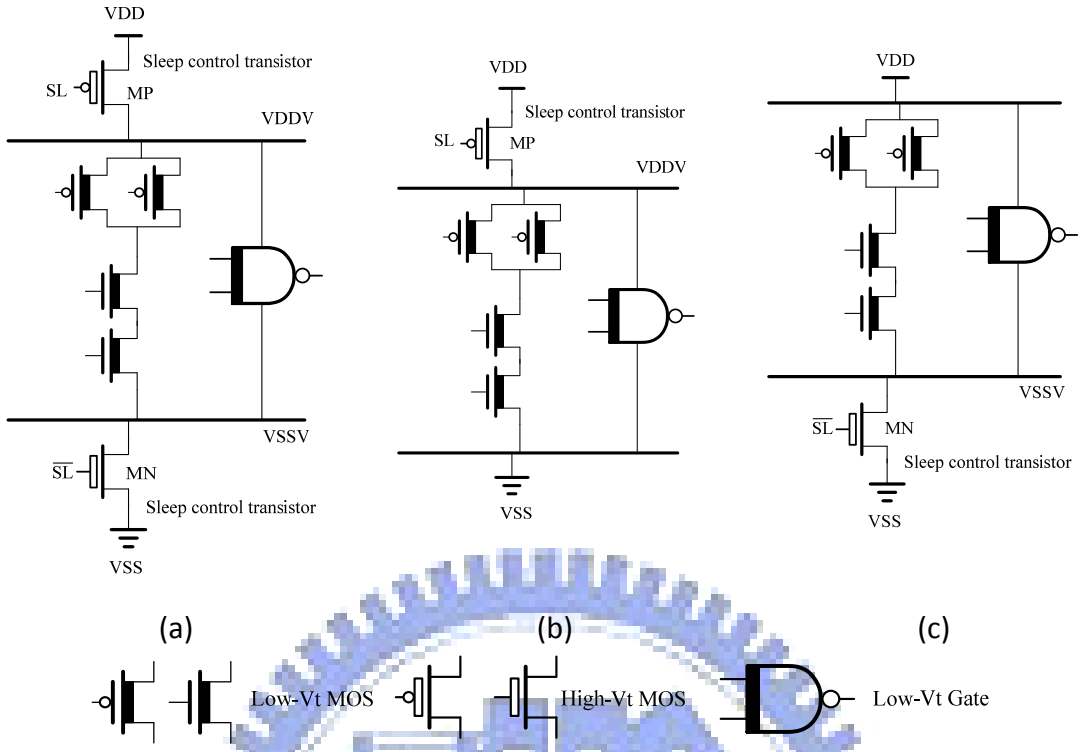


Fig. 4.1 Schematic of MTCMOS circuits (a) Original MTCMOS (b) PMOS insertion MTCMOS and (c) NMOS insertion MTCMOS

The utility of the sleep control transistor is to do efficient power management. When circuit is in the active mode, the pin SL is applied to low and the sleep control transistors (MP and MN) with high-Vt are turned on. Because the on-resistances of sleep control transistors are very small, the virtual supply voltages (VDDV and VSSV) are quite close to real ones. When the circuit is turned into the standby mode, the pin SL is set to high, MP and MN are turned off and they can cut the leakage current efficiently. Actually, in the practical design, it needs only one type of high-Vt transistor for leakage control. Fig. 4.1(b) and (c) show the PMOS insertion and NMOS insertion schemes, respectively. Most designers prefer the NMOS insertion due to the on-resistance of NMOS is quite smaller than PMOS with the same size. So designers can use the smaller size NMOS to be the sleep control transistor. MTCMOS can be easily implemented based on existing circuits. However, the main drawback of MTCMOS is it can only deal with the standby leakage power. The other problem is the large inserted MOSFETs will increase the area and delay significantly. Besides, if

the data retention is required in standby mode, it needs an additional high-Vt memory circuits to maintain the data [8].

The second type of multiple-threshold CMOS circuit is super cut-off CMOS (SCCMOS). The schematic of PMOS and NMOS insertion SCCMOS circuits are shown in Fig. 4.2(a) and (b), respectively. SCCMOS uses rather low-Vth transistors with an inserted gate bias generator than high-Vt sleep control transistors used in MTCMOS [9].

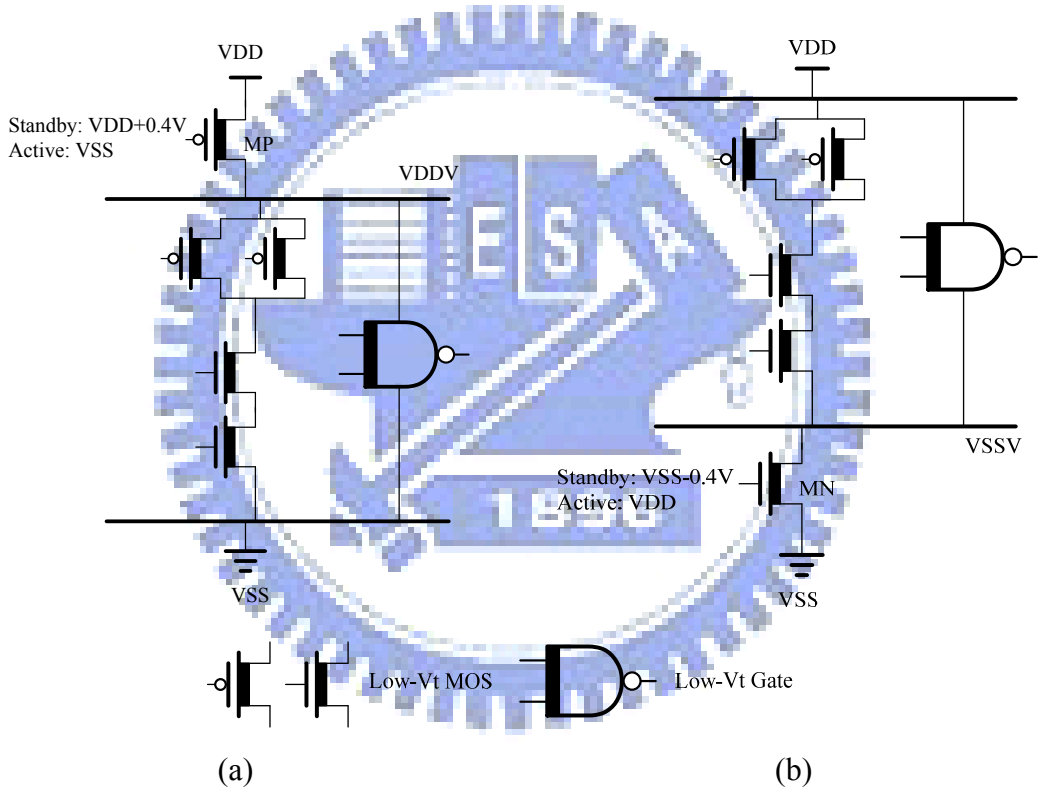


Fig. 4.2 Schematic of SCCMOS circuits (a) PMOS insertion SCCMOS and (b) NMOS insertion SCCMOS

For the PMOS insertion SCCMOS, the gate is applied to VSS and the low-Vt PMOS is turned on in the active mode. At this time, the virtual supply voltage (VDDV) is very close to real power supply voltage. When the circuit is turned into the standby mode, the gate is set to VDD+0.4V to fully turn off the low-Vt PMOS. Because the reverse bias is applied to the gate of PMOS, SCCMOS can fully cut off

the leakage current. On the other hand, the operation of NMOS insertion is the same as PMOS one. The gate of NMOS is set to VDD in the active mode and VSS-0.4V to fully cut off the leakage current in the standby mode, respectively. With the same reason as MTCMOS, it needs only one type of insertion SCCMOS for leakage control in the practical design.

The third type is Dual Threshold CMOS. We know that high threshold transistors are used to suppress sub-threshold leakage current, but it will degrade the performance seriously. For a logic circuit, high threshold transistors can be assigned in non-critical paths to reduce the leakage current, while the low threshold transistors in the critical paths can maintain the performance. By this method, both high performance and low power can be achieved simultaneously and it doesn't need any additional transistors. Dual Threshold CMOS circuit is shown in Fig. 4.3. This dual threshold technique can diminish the leakage power during both standby and active mode very well. But the main difficulty of using this method is not all the transistors in non-critical paths can be replaced by high threshold voltage transistors due to the complexity of a circuit or the critical path of the circuit may change, thereby increasing the critical delay [1]. So it is hard for the tools to synthesize circuits with the consideration of this method.

Due to the above reason, designers have to use this technique carefully to avoid changing the critical path of the circuit. Note that this algorithm only deals with the circuits at the gate level. Thus, the transistors in a gate will have the same threshold voltage.

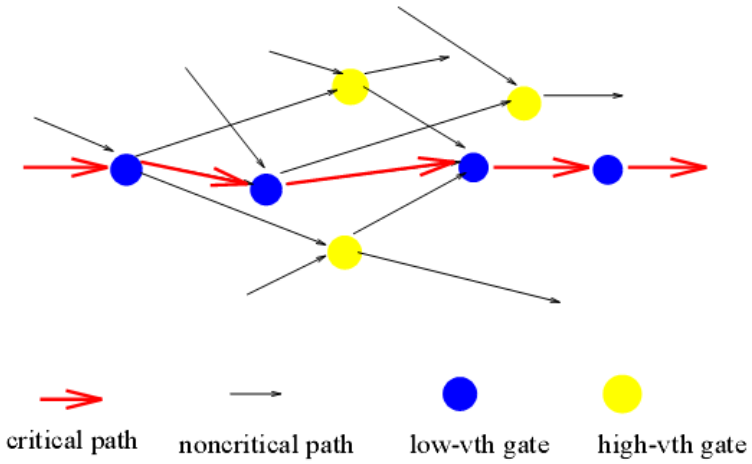


Fig. 4.3 Dual-threshold CMOS circuit [1]

The next type is mixed-Vth CMOS circuit scheme. [10] introduced two types of mixed-Vth CMOS circuits. Mixed-Vth schemes can have different threshold voltages within a gate. For type I scheme (MVT1), it is not allowed different threshold transistors in p pull-up or n pull-down networks. In the first step, designers have to find out the MOSs on the critical path. If the MOSs on the critical paths are in p pull-up or n pull-down networks, designers need to replace all of the MOSs in p pull-up or n pull-down networks with the same low threshold voltage MOSs to improve the performance. For example, the MOS transistors in the square (see Fig. 4.4(a)) are on the critical path. In the NOR gate, both p pull-up and n pull-down networks have the MOSs on the critical paths. So we change all the PMOSs and NMOSs for low threshold MOSs. In the inverter gate, we can see that only NMOS is on the critical path. So we just replace the NMOS with low threshold MOS and keep the high threshold MOS in the p pull-up network.

In another scheme of mixed-Vth CMOS circuit (MVT2), it allows different threshold transistors anywhere except for the series connected transistors. The transistors on the series connected networks must be the same threshold MOSs. When using the MVT2 technique, designers have to find out the MOSs on the critical paths, first. This step is the same as MVT1. Then designers have to change all the MOSs on

the critical paths on the series networks for low threshold transistors. The main difference between MVT1 and MVT2 is that MVT2 will just replace the MOSs on the critical path on the parallel networks with low threshold MOSs and keep other MOSs with high threshold transistors. For example, the NOR gate in Fig. 4.4(b), both the p pull-up series structure and the n pull-down parallel structure networks have MOSs on the critical path , respectively. With the above description, MVT2 replace all the PMOSs in the series structure at the critical path with low threshold MOSs and replace NMOS in the critical to low threshold transistors. MVT2 keeps other NMOSs on the parallel structure networks with high threshold transistors. The situation of inverter gate is the same as MVT1.

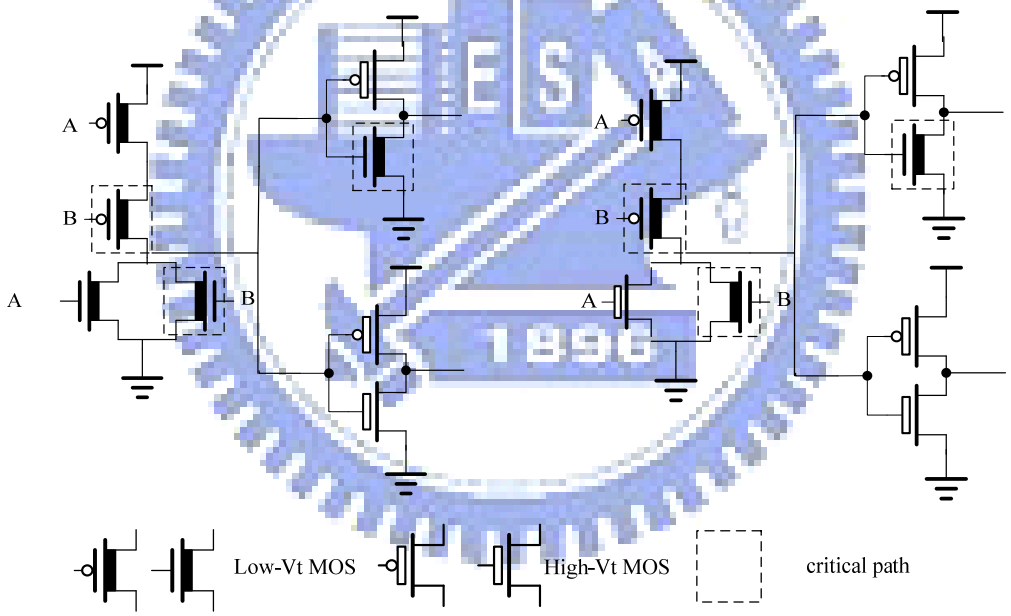


Fig. 4.4 MVT schemes of [10] (a) MVT1 scheme and (b) MVT2 scheme

A new Mixed-Vth (MVT) CMOS design technique is proposed to reduce the static power dissipation on gate-level in [12]. The goal of MVT-Gates is to reduce the leakage within a gate without varying the performance. This will be achieved by replacing normal-Vth transistors with high-Vth and low-Vth transistors. Optimization of a gate should not increase the worst case delay.

In a logic cell, stacked transistors usually form the critical path, and the MOSs on it must be low-Vth transistors. We can use different threshold transistors in such a stack to reduce leakage and keep the performance. In MLVT-gates scheme (Fig. 4.5(b)), all the transistors on the critical path are low-Vth transistors and the transistors on the non-critical path are still high-Vth transistors. Another scheme is called MVT-gates. The scheme of MVT-gates is the same as MLVT-gates on the non-critical path, but the MVT-gates use different threshold transistors on the critical path at the same time (Fig. 4.5(a)). Because we use the high-Vt device to block the leakage current and use the low –Vt device to keep the timing performance.

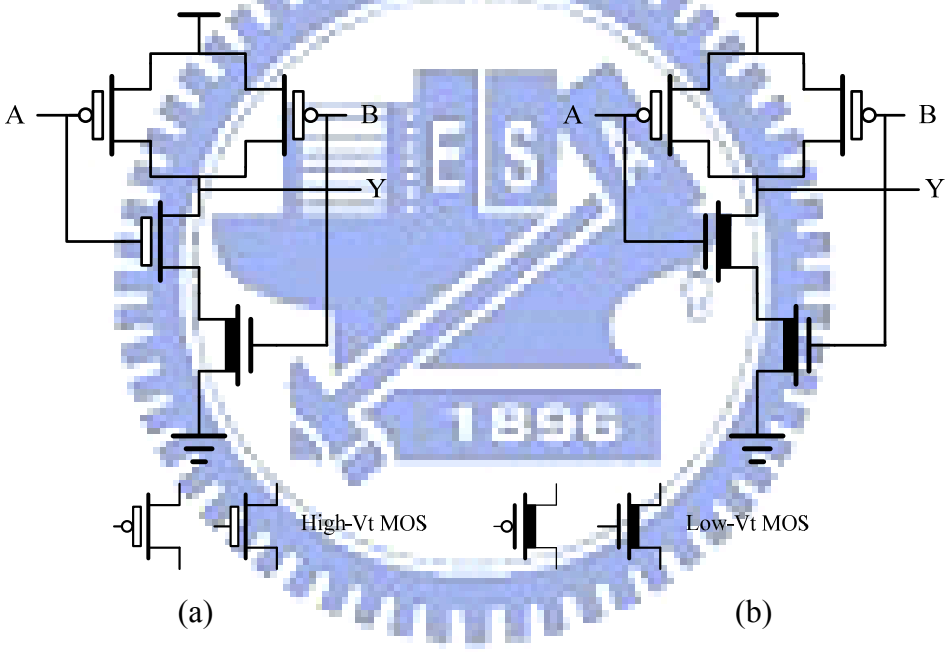


Fig. 4.5 MVT schemes of [12] (a) MVT-NAND2 and (b) MLVT-NAND2

There has been another method proposed called dynamic threshold CMOS (DTMOS) [1]. The threshold voltage can be altered dynamically to suit the operating state of the circuit in this architecture. When circuit is in the standby mode, a high threshold voltage is given to diminish the leakage current. While a low threshold voltage is allowed for higher current drives in the active mode of operation. Designers can establish the DTMOS by tying the gate and body together [11]. Fig. 4.6 shows the schematic of a DTMOS inverter. The supply voltage of DTMOS is limited by the

diode built-in potential. The pn diode between source and body should be reverse biased. So this technique is only suitable for ultra-low voltage.

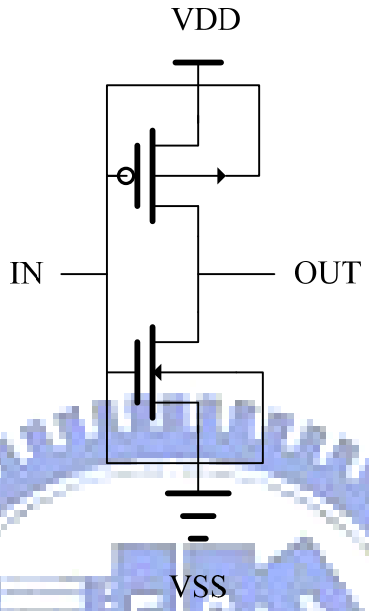


Fig. 4.6 Schematic of DTMOS inverter

4.1.2 Multiple Supply Voltage

We know that the dynamic power consumption is proportional to the square of the supply voltage. Therefore, designers can save a significant amount of dynamic power by lowing the supply voltage. In practical circuits, designers can separate a system into two blocks. One is timing-critical block, and the other one is timing-non-critical block. In the timing-critical blocks, designers have to use normal supply voltage of the process technology to meet the performance requirement. On the other hand, designers can use lower supply voltage in the timing-non-critical block to save the dynamic power. By this method, we can meet the specification and save the dynamic power in the system at the same time. In the multiple supply voltage circuit, it needs the level converter to be the interface between the VDDH block and VDDL block. Fig. 4.7 shows an example of level converter and Fig. 4.8 shows the

block diagram of multiple supply voltage.

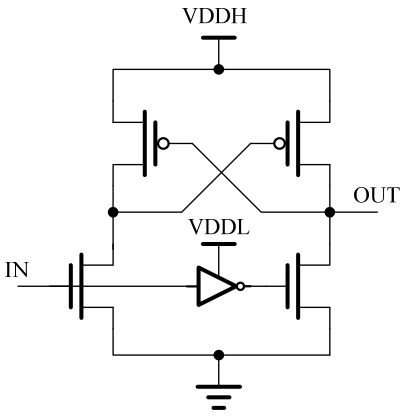


Fig. 4.7 Example of level converter

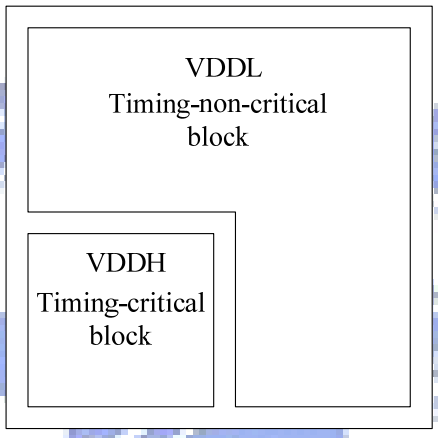


Fig. 4.8 Block diagram of multiple supply voltage

This method is called clustered voltage scaling (CVS) technique. It reduces the power consumption with two supply voltages [13]. The block diagram of CVS is shown in Fig. 4.9.

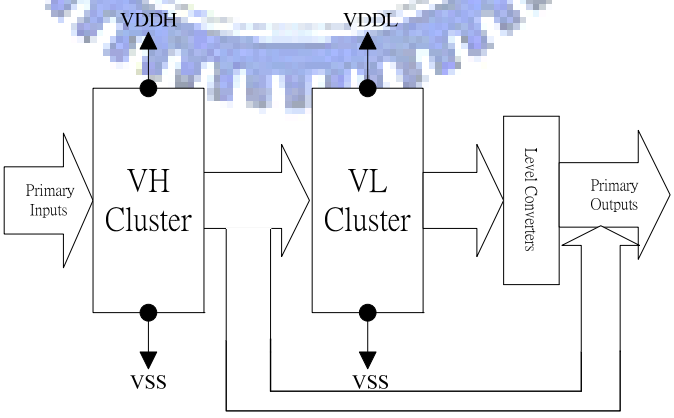


Fig. 4.9 The block diagram of CVS

A new type of multiple supply voltage circuit has been proposed. This method is called converter-free multiple-voltage (CFMV) structure [14]. The CFMV structures make use of multiple supply voltages and do not require level converters. On the other

hand, the CVS structure works employing multiple supply voltages need level converters to prevent static currents, which may result in large power consumption. The block diagram of CFMV is shown in Fig. 4.10.

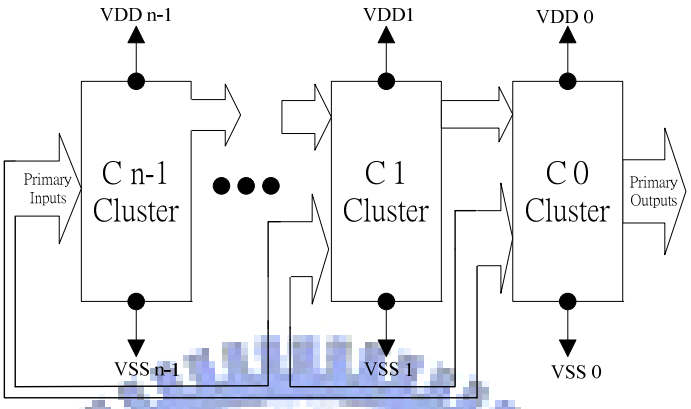


Fig. 4.10 The block diagram of CFMV

4.2 Standard Cell Selection Rule

A simple transistor sizing technique called logical effort has been proposed in [6]. It is a convenient method for designers to choose the best topology and number of stages of logic for a function. Designers can roughly estimate the minimum possible delay for a given topology and choose proper gate sizes that achieve this delay. Logical effort of a gate is defined as the ratio of the input capacitance of the gate to the input capacitance of an inverter that can deliver the same output current. As the logical effort mentioned, the smaller logical effort indicates that the logic gate is simpler. For example, an inverter is the simplest logic gate, and its logical effort is 1. The logical efforts of frequently used gates is shown in Table 4.1. A gate with small logical effort implies that we can tune the size more easily. By the above description, we would rather modify the size of a logic gate with smaller logical effort than the large one. It means that in a complicated gate we would use a cascaded stages so that we can increase the size of an inverter or a buffer of the second stage rather than to increase the size of the complex gate at the first stage if the driving ability of a

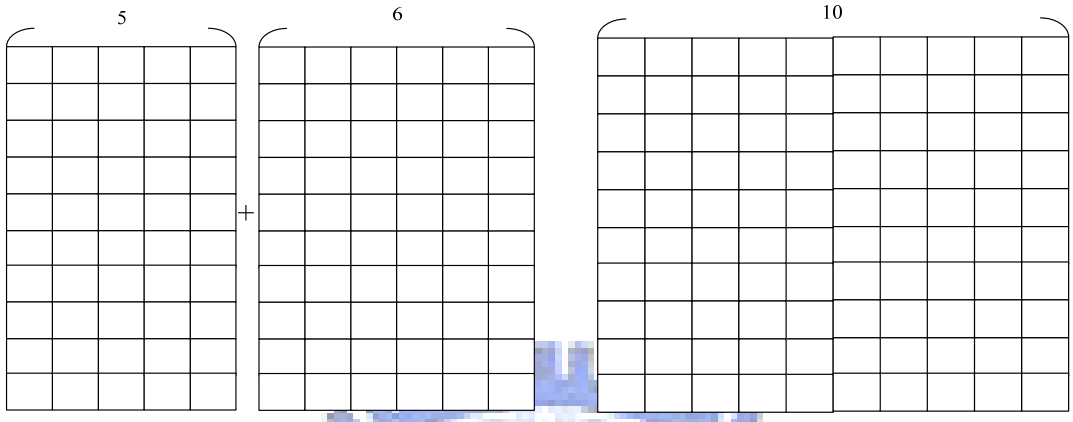
complicated circuit is not enough. We will pay more cost such as power or area if we try to tune the sizes of large logical effort circuit than simpler circuit, like inverters or buffers.

Table 4.1 Logical effort of frequently used gates [6]

Gate type	Number of inputs				
	1	2	3	4	n
INV	1				
NAND		4/3	5/3	6/3	(n+2)/3
NOR		5/3	7/3	9/3	(2n+1)/3
tristate, multiplexer	2	2	2	2	2
XOR, XNOR		4	6	8	2n

We know that inverters or buffers are used frequently in clock tree and the bus. We cannot choose the most suitable inverter or buffer for our circuit if we do not have finer driving ability of inverters or buffers. For instance, we need a buffer with driving ability among X4 and X6, however, there are only X4 and X6 buffers in the present commercial standard cell library. If we choose the X4 buffer, the driving ability is not enough. So we can just choose the X6 buffer for our design. The driving ability is indeed enough in this situation, but we have to waste some power and area. If we add an additional X5 buffer cells, we can choose X5 instead of X4 or X6. Thus we could get the suitable driving ability and reduce the power consumption and area at the same time. Base on this reason, we will include finer driving ability of inverter (INV) and buffer (BUF) in our standard cell library. By this method, we can choose the most suitable inverters or buffers to increase driving ability of the circuit without extra redundant power consumption. We can also save area if we have finer driving ability in INV or BUF. For example, we can obtain a BUF with driving ability X5 by consisting of BUFX2 and BUFX3. BUFX2 needs 5 pitches and BUFX3 needs 6 pitches in the layout. It needs total 11 pitches to compose a BUF with driving ability X5. But the BUFX5 in our standard cell library only needs 10 pitches, as shown in Fig.

4.11. With the above explain, we can save the power consumption and area at the same time through adding finer driving ability for INV and BUF in our standard cell library.



(a) (b)

Fig. 4.11 Numbers of pitch for BUF with driving ability X5
(a) BUFX2+BUFX3 (b) BUFX5

Besides INV and BUF, our standard cell library will contain NAND、NOR、AND、OR、XOR、XNOR、MX、AOI、OAI、ADD、DFF, etc. NAND, NOR, AND, OR, XOR, and XNOR are basic logic function gates. AOI and OAI are hybrid gates formed by combining series and parallel switch structures. They can be used in the circuit to implement the complex Boolean equations more easily by fewer gates. Designers can gain the smaller area by using AOI or OAI gates to implementing the same Boolean equation than by NAND, NOR, and inverter. We also include ADD 1-bit half and full adders for the arithmetic circuit. For the sequential circuit, we have tradition master-slave DFF and single edge trigger DFF (SETDFF) with positive triggered function. The types of DFF include without set or reset signals, with only set signal, and with only reset signal and with set and reset signals at the same time.

There are still other cells for clock signal, like CLKINV, CLKBUF, CLKAND, CLKXOR, and CLKMX. The most important characteristic of these cells is that the duty cycle must be close to 50%. Also, due to the clock loading, they have larger

driving ability than the other cells.

4.3 Low Power Standard Cell Library Design Flow

It uses only single threshold voltage (Vt) like standard-Vt、low-Vt and high -Vt transistors to design the cells in current commercial standard cell library. The low-Vt cell library is used for circuits attached importance to the performance. On the other hand, the high-Vt cell library is used for low power circuits. So we want to establish a set of standard cell library with the above two characteristics. In section 4.1, we have introduced several techniques for low power circuit design. The main target of the multiple threshold voltage techniques mentioned in section 4.1 is to reduce the leakage current by using multiple threshold technique.

Our main goal is to establish a set of mixed-Vt low power standard cell library with similar propagation delay time to single-Vt library and we want to reduce the leakage power of circuits synthesized by this mixed-Vt standard cell library. Because of the process we can access are high threshold (V_{th}) and normal (medium) threshold voltage (V_{tm}), we will use these two different kinds of devices to establish a mixed-Vt (MVT) low power standard cell library. Also, the performance comparisons are made with high-Vt cell library (HVT). In our library, the low power issue is the major consideration in our design flow, we try to make the delay roughly the same with the high-Vt cell library (HVT). So we replace the MOS on the critical path with the V_{tm} MOS, and do not change the size of the V_{tm} device in the first step. The charge/discharge current will increase due to the reduction of Vt at this time. Fig. 4.12 and Fig. 4.13 show this phenomenon for NMOS and PMOS transistors. Therefore, the rise/fall transition time and the propagation delay time on critical path are decreased according to the following equations.

$$t_{pd} \cong \frac{C_L V_{DD}}{(V_{DD} - V_t)^{\alpha}} \qquad (4.2)$$

where α models short channel effects. We can observe that the power and the speed of circuits are increased at this time. But the speed is not our main concerned target. So we have to adjust the speed like that of the original single-Vt cell and see if we can lower the power consumption. We can decrease the charge/discharge current on the critical path by shrinking the width of the lower threshold device. The rise/fall transition time will increase and we will stop shrinking the width until the propagation delay time is like the single-Vt cell. In this way, we could obtain lower total power consumption on a standard cell due to the decreasing of the charge/discharge current on the critical path and the drain/source parasitic capacitance of the V_{tm} device.

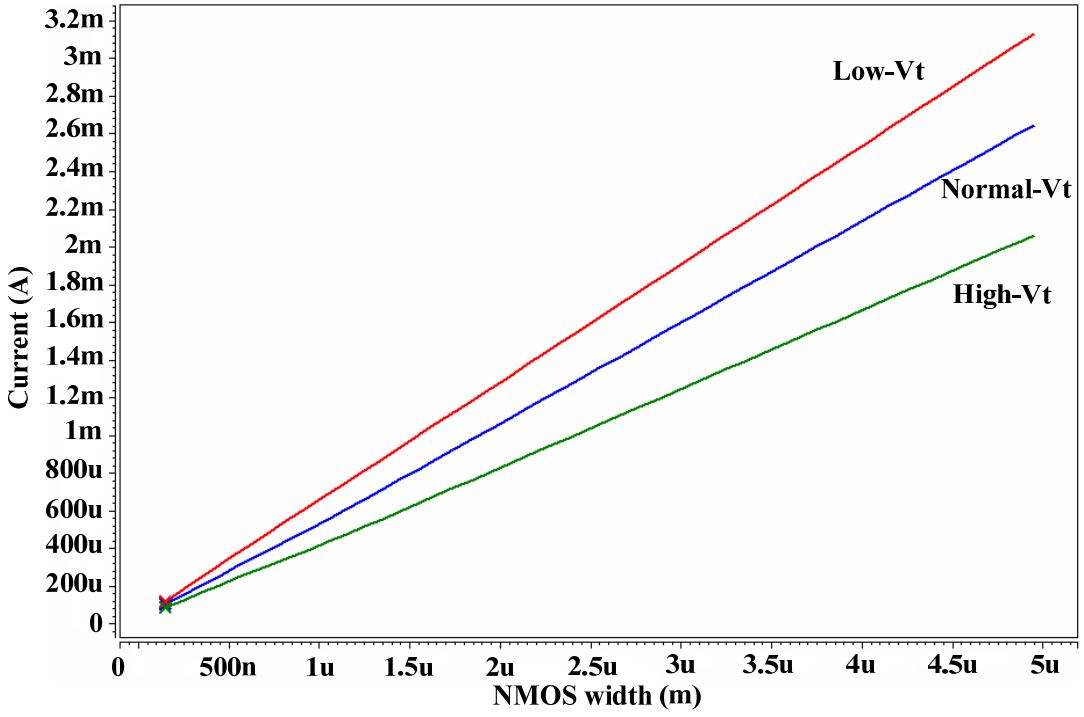


Fig. 4.12 NMOS saturation current with different threshold voltage and channel widths

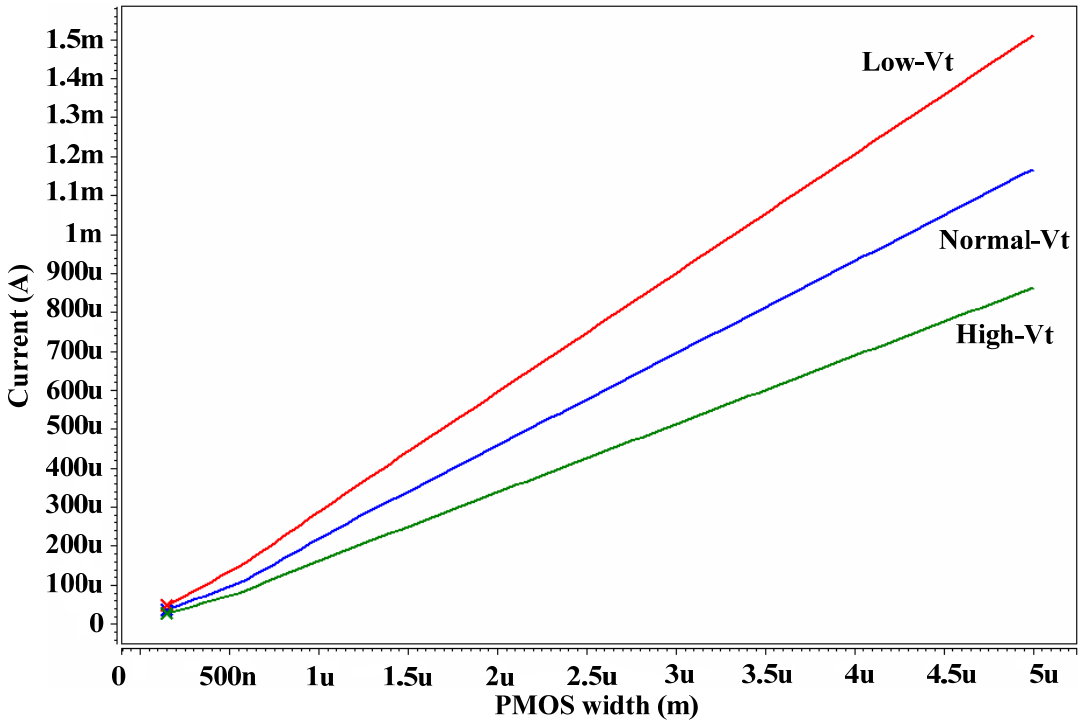


Fig. 4.13 PMOS saturation current with different threshold voltage and channel widths

Fig. 4.14 shows the simulated timing result of an inverter. Line 1 represents the output voltage waveform of the high-Vt (HVT) inverter. Line 2 represents the output voltage waveform of the MVT buffer with the width being the same as HVT one. Line 3 represents the output voltage waveform of the MVT inverter with the width of the normal threshold device being shrunk.

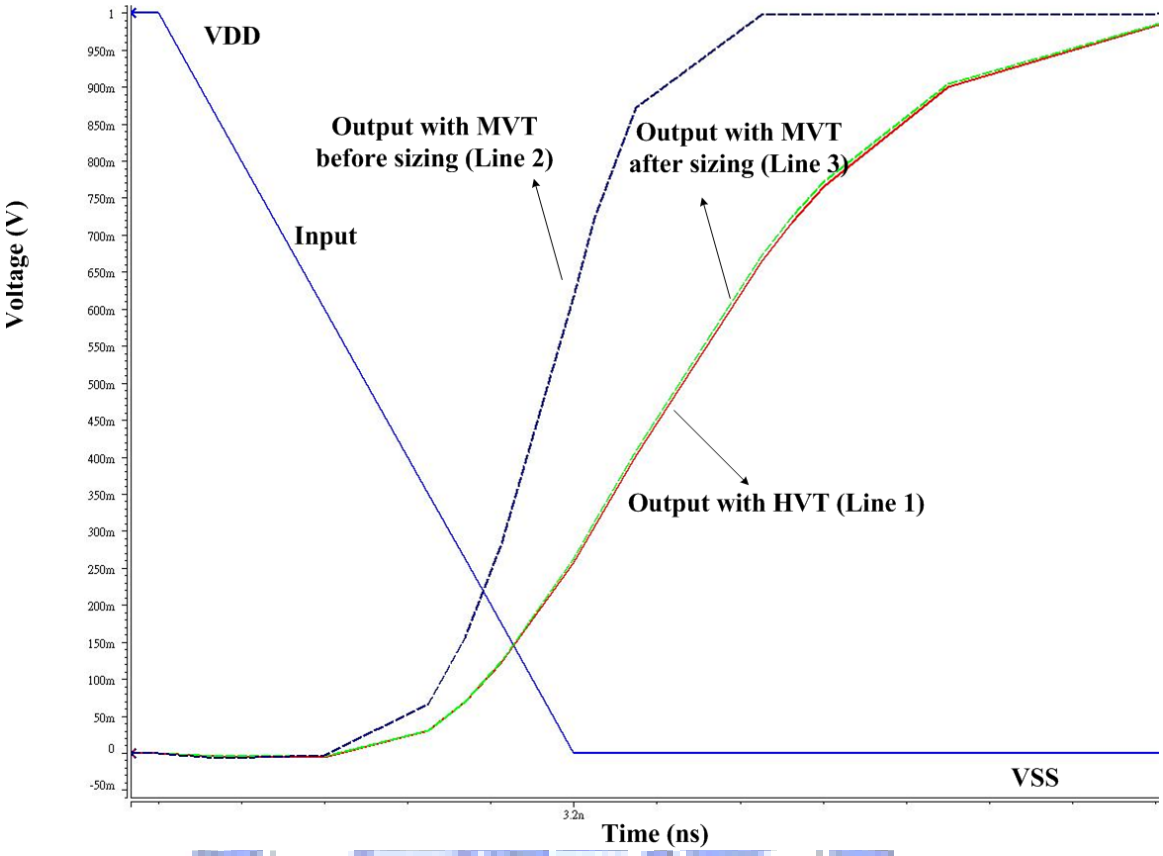


Fig. 4.14 The timing waveforms of the inverter gates

Due to the threshold voltage value of V_{tm} device in MVT cell is smaller than V_{th} device in HVT one, we can see that the rise/fall transition time and rise/fall propagation delay time of MVT cell are smaller than HVT one in Fig. 4.14. We know that the the charge/discharge current of a MOS will increase if the threshold voltage of a MOS is lower. Furthermore, smaller Vt will cause the MOS conduct faster because the V_{gs} of the MOS is greater than the Vt in an early time. We can see the above fact in Fig. 4.14. The red and green lines show the MOSs conduct earlier than the blue one. The rise/fall propagation delays of the red and green line cases are shorter than the blue line case because the V_{tm} MOS conducts earlier and the charge/discharge current of the V_{tm} MOS is larger V_{th} one. In order to keep the timing performance of MVT cell close to HVT one, we have to increase rise/fall transition time by shrinking the width of V_{tm} MOS to reduce the charge/discharge current. This method can also decrease the loading capacitance of previous stage circuits because

we shrink the size of V_{tm} MOS. We know that the dynamic power dissipation is also proportional with the output loading capacitance C_L from Eqn. (4.1). Therefore, we can replace the cell of previous stage by smaller driving ability cell because of the smaller loading capacitance of the laoding stage and the dynamic power dissipation will be reduced. According to the above concepts, we have proposed a mixed-Vt technique to save dynamic power.

The diffusion parasitic capacitance of the MOS depends on both the area AS and sidewall perimeter PS of the source/drain diffusion region. The geometry is illustrated in Fig. 4.15. The area and perimeter of the source and drain diffusion are AS = W · D and PS = 2 · W + 2 · D respectively [6]. The total parasitic capacitance of source and drain can be expressed as

$$C_{xb} = AS \cdot C_{jbx} + PS \cdot C_{jbxsw} \quad \text{(4.3)}$$

where x= s or d

where C_{jbx} is the area junction capacitance and has units of capacitance/area and C_{jbxsw} is the sidewall capacitance and has units of capacitance/length.

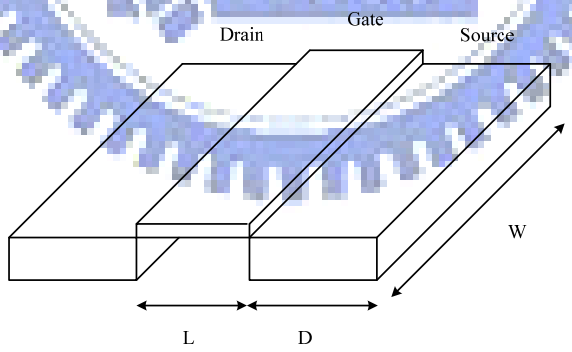


Fig. 4.15 Diffusion region geometry

So we can shrink the AS and PS of a MOS by reducing the width of it and the parasitic capacitance will be reduced through this approach.

The MOS gate is above the channel and may partially overlap the source and drain diffusion areas. So the gate capacitor has two components: the intrinsic capacitance (over the channel) and the overlap capacitance (to the source, drain, and

body). In fact, the gate capacitor can be viewed as a parallel plate capacitor with the gate on top and channel on bottom with the thin oxide dielectric between. The gate capacitance can be expressed as

$$C_g = C_{OX} \cdot W \cdot L \tag{4.4}$$

We can combine our MVT approach with other low power design methodologies, like MTCMOS, Dual-Vt, etc. Using our approach with other low power design methodologies can decrease dynamic power and leakage power at the same time.

Let us consider another approach. If we use all low-Vt devices in a cell, we can shrink the sizes of all low-Vt MOS to achieve the similar propagation delay time with NVT cells. Although it results in saving more dynamic power, the leakage current will increase very seriously due to the all low-Vt MOS. The leakage current (I_{leak}) of the MOS is exponentially proportion to Vt as shown in Eqn. (4.5)

$$I_{leak} \propto e^{-V_t} \tag{4.5}$$

Thus, reducing Vt slightly will increase the leakage current and power dramatically. The smaller the Vt is, the larger the leakage current and leakage power are. We can see this fact in Table 4.2. So our approach is to replace the MOS on critical path with V_{tm} device and the other paths use V_{th} devices. We want to block the leakage current when the cell is under the standby state by V_{th} MOS. Thus, it only increases the leakage current at certain input combinations instead of all input combinations by using our mixed-Vt approach.

Table 4.2 90nm Leakage current with normal-Vt and low-Vt

Leakage current	NMOS		PMOS	
	W=1um	increasing rate	W=1um	increasing rate
High-Vt	45.6pA	59.3pA/um	120.4pA	130.7pA/um
Standard-Vt	3.55nA	2.84nA/um	4.67nA	5.40nA/um

We compare the dynamic and leakage power by setting the propagation delay of mixed-Vt, high-Vt and standard-Vt inverter gates the same with each other. The result

is shown in the Table 4.3. Using standard-Vt gates to synthesize the circuits can save more dynamic power than mixed-Vt ones. But using only standard-Vt gates will raise the leakage power much greater than high-Vt for all input pattern. Mixed-Vt gate has less leakage current than standard-Vt one. We can draw the conclusion that designers would like to use standard-Vt gates to synthesize the circuit which is always in the active mode and they can gain the low power performance by reducing dynamic power consumption. But the main drawback of this kind of circuits is that there is large leakage power in the idle mode. On the other hand, if the circuits are usually in the idle mode but the timing performance is also a concern. Designers can use mixed-Vt gates to obtain the low power effect by reducing dynamic power and meet the appropriate performance. And it is important that the leakage power is increased a little compared with high-Vt.

Table 4.3 Inverter gate with high-Vt, mixed-Vt and normal-Vt

	Tplh (ps)	Tphl (ps)	Total Power (nW)	Leak_0 (pW)	Leak_1 (pW)
High-Vt	232.1	220.7	41.8	21.9	19.8
Mixed-Vt	193.4	144.4	35.9	15.3	336.7
Standard-Vt	103.7	63.1	23.51	309.8	336.7

Tplh: the propagation delay results from input transition and makes output transit from low to high
Tphl: the propagation delay results from input transition and makes output transit from high to low
Leak_0: the leakage power occurs when the input signal of an inverter is low (0)
Leak_1: the leakage power occurs when the input signal of an inverter is high (1)

We have introduced our mixed-Vt approach. Our main goal is to establish a set of mixed-Vt low power standard cell library. So when we use our mixed-Vt approach to design a low power standard cell, we have to find out the critical path of the cell first. As we know, there are rise and fall propagation delay of CMOS circuits. So the critical paths are separated into rise and fall paths and we have to find these two paths in the first step.

After we find out the critical devices on the critical path, we will replace them with normal-Vt devices. The critical device is defined as the MOS on the stacked

circuit that are the most far away from output. Because the MOS which is the most far away from output is on the longest charge/discharge path, the charge/discharge time of this device will dominant the charge/discharge time of the cell. We replace the critical device instead of all the devices on the critical path with standard-Vt device. This is because we can gain the better performance and shrink the width of the critical device on the critical path. By shrinking the width of the standard-Vt device, we can save the dynamic power consumption. Because we just change the transistor that is most far away from output with standard -Vt device on the stack, the leakage current will be blocked by other high-Vt devices on the stack.

For example, the NMOSs of the NOR gate are parallel connected. According to the above description, we will replace all the NMOSs by standard-Vt devices and shrink their widths. The parasitic capacitance of this cell output is dominant by the width of those parallel NMOSs because all the parallel NMOSs connect to the output directly. Through shrinking the width of the parallel NMOSs, we can reduce the parasitic capacitance of this cell output because of the smaller drain area of the MOSs. The parasitic capacitance of the cell output is one of the components of the output loading capacitance C_L and we can observe that smaller C_L will lower the dynamic power consumption from Eqn. (4.1). So diminishing the parasitic capacitance of the cell output can lower the dynamic power consumption.

4.4 Cell Design

The most important parameters of simulation environment of standard cells are input transition time and output load capacitance as shown in Fig. 4.16. In this section, we will define our simulation environment.

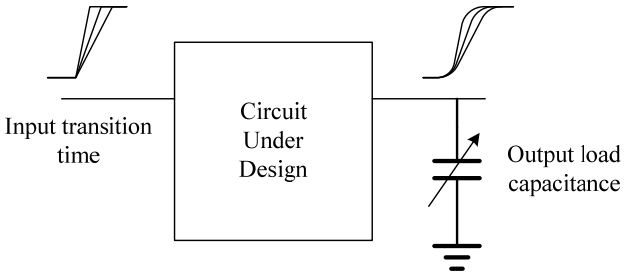


Fig. 4.16 Circuit simulation environment with capacitance load

We use the NAND2 gate to explain how do we define our simulation environment. First we sweep the input transition time of the high-Vt and mixed-Vt NAND2 gate with output load capacitance is fixed to 20fF. The result is shown in the Fig. 4.17. Fig. 4.17(a) shows that the power consumption of the mixed-Vt gate is smaller than the high-Vt one only when the input transition time is smaller than around 300ps. The reason for this result is that the short circuit current becomes significant when the input transition time becomes longer. Because Mixed-Vt gate turns on early, the short circuit current is larger than that is high-Vt gate. The rise and fall propagation delay waveforms of the NAND2 gate under different input transition time are shown in Fig. 4.17(b) and Fig. 4.17(c). We can see that the rise and fall propagation delay times of high-Vt gate are very close to the mixed-Vt ones when the input transition time is smaller than 200ps as shown in Fig. 4.17(b) and Fig. 4.17(c). If the input transition time is larger than 200ps, the rise propagation delay time of mixed-Vt gate will be smaller than the high-Vt one. The reason is also because the low-Vt device is turned on early during the transition. So we can conclude that we can achieve low power performance with similar propagation delay time of high-Vt cell, by setting the input transition time to 100ps.

(a)

(b)

(c)

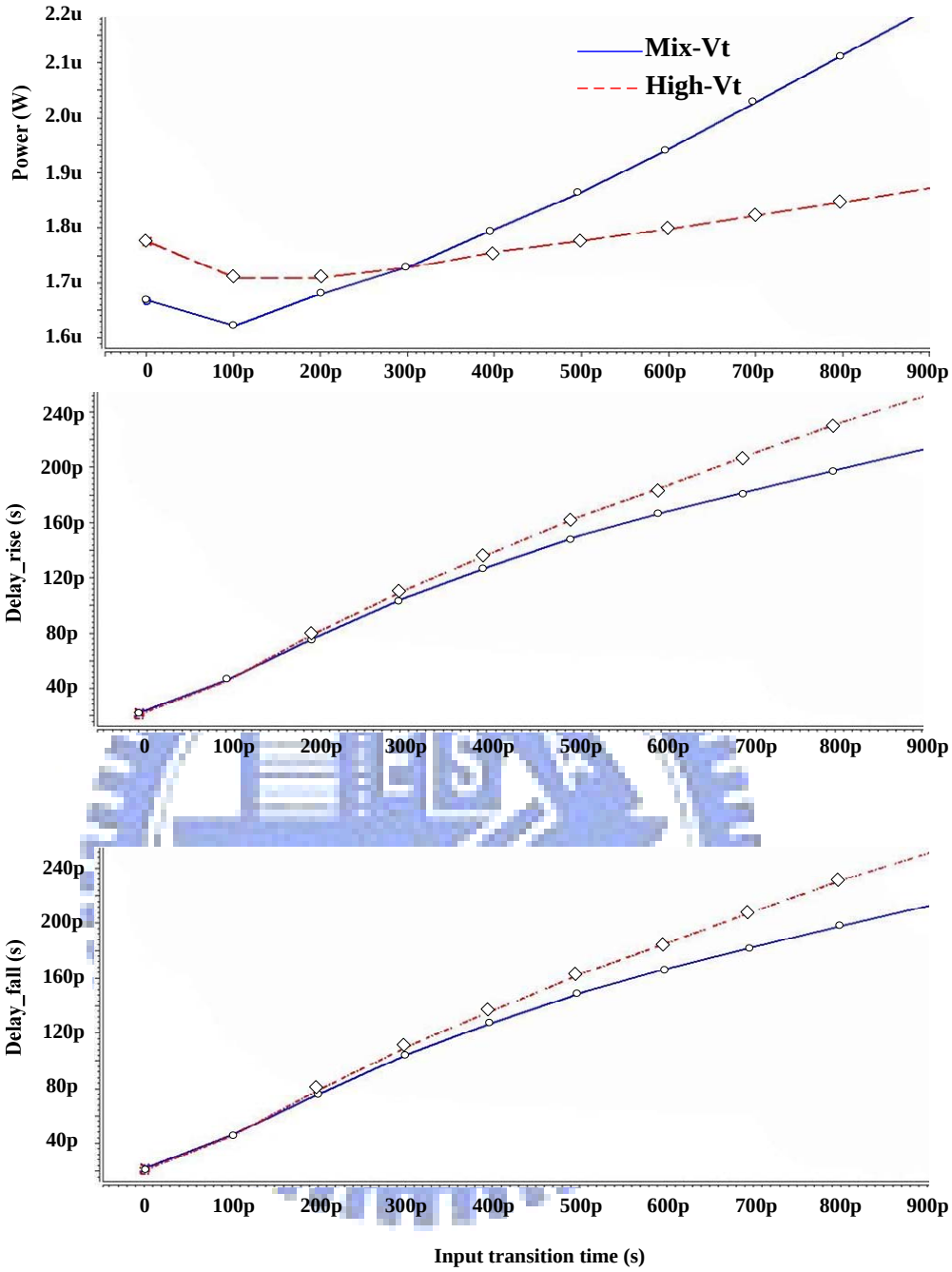


Fig. 4.17 Fixed output load capacitance and sweep input transition time of NAND2 (a) Power vs. input transition time (b) Delay_rise vs. input transition time and (c) Delay_fall vs. input transition time

According to the above conculsions, our input transition time of the simulation environment is 100ps. Our standard cell library will be used in a hearing aid project and the clock rate of the specification is 20 MHz. So our standard cell library is tuned for this kind of low power circuit.

Fig. 4.18 shows the power and delay waveforms of NAND2 gate with high-Vt

and mixed-Vt by sweeping output load capacitance and keeping input transition time constant. The results show that the power of mixed-Vt gate is always lower than high-Vt one in all ranges of output load capacitance in the Fig. 4.18 (a). From Fig. 4.18(b) and Fig. 4.18(c), the rise and fall propagation delay time of high-Vt gates and mixed-Vt ones are very close as the output load capacitance is around 10 fF. Through these three figures, we can conclude that the changing of output load capacitance does not change the characteristics of power and propagation delay very much. Our goal is to let the propagation delay time of mixed-Vt gates be similar to high-Vt ones to gain low power consumption. Form above observation, we decide to set our output load capacitance of simulation environment to be 10 fF.

By simulation, we obtain the equivalent input capacitance of FO4 2-input NAND with driving ability of X2 is around 10 fF. So we let the output load capacitance of simulation environment be the FO4 circuit under design. We use UMC 90nm process and 0.5v power supply to size and design our low power standard cell library. The simulation environment is shown in Fig. 4.19.

(a)

(b)

(c)

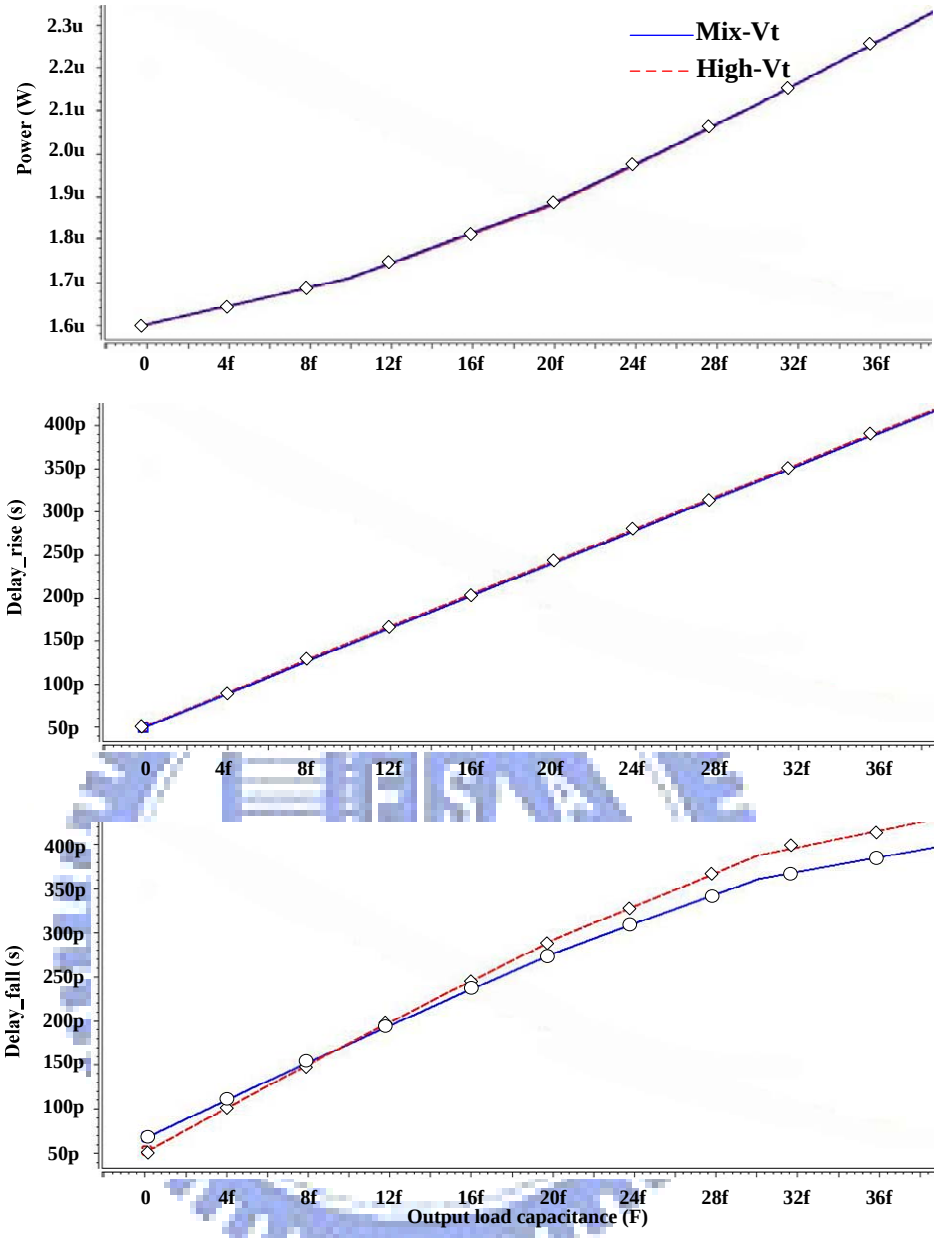


Fig. 4.18 Fixed input transition time and sweep output load capacitance (a) Power vs. output load capacitance(b) Delay_rise vs. output load capacitance and (c) Delay_fall vs. output load capacitance

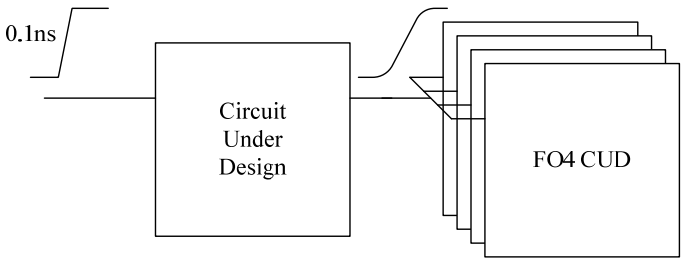


Fig. 4.19 Circuit simulation environment with FO4 load

4.4.1 INV

Through our analysis in above paragraph, we know that the critical path of inverter gate in Fig. 4.20(a) is on the PMOS. We can also demonstrate this conclusion from Table 4.4. The Tplh is related to the PMOS driving ability. So we change the high-Vt PMOS into normal-Vt PMOS. The mix-Vt inverter gate is shown in Fig. 4.20(b). Next we let the rise/fall propagation delay time of mixed-Vt inverter are similar to the high-Vt one by shrinking the normal-Vt PMOS. After these steps, we can gain 14.2% power saving in one inverter gate. However, the leakage power will be larger clearly only when the input is high and the PMOS is changed into normal-Vt PMOS. But it will be still smaller than the inverter with only normal-Vt. This reason which causes this result has been described in section 4.3.

(a) (b)

Fig. 4.20 Inverter gate (A) HVT and (B) MVT

Table 4.4 Time and power table of INV with VDD=0.5 v

	Tplh (ps)	Tphl (ps)	Power (W)	Leak_1 (W)	Leak_0 (W)
HVT	232.1	220.7	41.8n	21.9p	19.8p
MVT	193.4	144.4	35.9n	15.3p	336.7p

Tplh: the propagation delay results from input transition and makes output transit from low to high
Tphl: the propagation delay results from input transition and makes output transit from high to low
Leak_1: the leakage power occurs when the output signal of an inverter is high (1)
Leak_0: the leakage power occurs when the output signal of an inverter is low (0)

Then we use the above HVT and MVT inverters to cascade three stages tapped inverters to drive a 100fF capacitor. The result shows that we can obtain around 9%

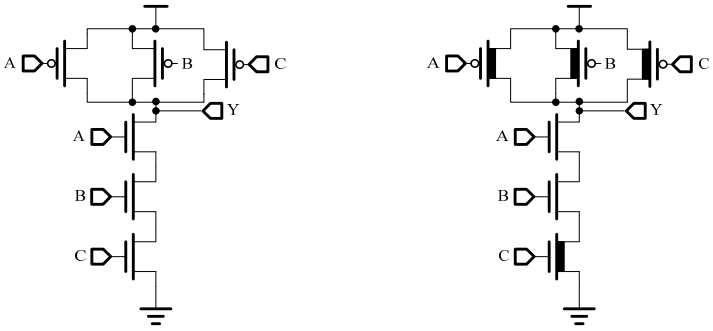
power saving and around 20% delay-power product less than HVT by using MVT inverters to cascade three stages when the . Another important phenomenon is that the propagation delay time of the MVT inverter is shorter than the HVT one. Because we shrink the PMOS size in the MVT gate, the gate capacitance of the MVT inverter is smaller than the HVT one. The smaller capacitance will shorten the propagation delay time. The result is shown in Table 4.5.

Table 4.5 Time and power of 3 statges inverter driving 100 fF with VDD=0.5 v

Tr_in=0.1ns Cload=100fF	Tplh (ns)	Tphl (ns)	Power (uW)	Delay-Power (ns · uW)
HVT	1.833	1.633	1.161	2.1
MVT	1.786	1.487	1.058	1.7

4.4.2 3-input NAND and AOI31

In this section, we will analyze the 3-input NAND gate and AOI31. We know that the critical path of 3-input NAND gate in Fig. 4.21(a) is on the input C of stacked NMOSs. So we change this NMOS into the normal-Vt NMOS. Also we replace all parallel PMOSs with normal-Vt PMOSs. The MVT 3-input NAND gate is shown in Fig. 4.21(b). We can also demonstrate this conclusion from Table 4.6. The Tplh and Tphl are longest at the input C. The leakage power of 3-input NAND gate with different input combinations is shown in Table 4.7. Thus, we can see that the MVT 3-input NAND gate can save the dynamic power and keep the leakage power only increasing at one state.



(a) HVT (b) MVT

Fig. 4.21 Schematic of 3-input NAND gate

Table 4.6 Time and power table of 3-input NAND with VDD=0.5 v

HVT	Tplh (ps)	Tphl (ps)	Power (nW)
In @ A	654.6	643.7	75.3
In @ B	721.1	726.9	85.1
In @ C	771.5	763.4	90.9
MVT	Tplh (ps)	Tphl (ps)	Power (nW)
In @ A	653.4	618.1	75.3
In @ B	665.3	664.1	83.7
In @ C	674.6	655.3	83.7

Table 4.7 Leakage power of 3-input NAND at all the input combinations, unit: pW

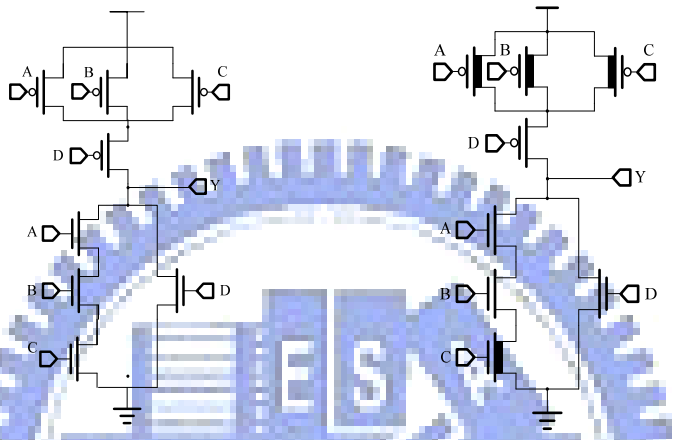
ABC	000	001	010	011	100	101	110	111
HVT	23.3	17.8	16.4	15.6	16.9	13.9	13.6	32.9
MVT	23.3	16.4	20.7	16.2	22.4	16.7	337.6	31.1

Due to the layout design rule restriction, when we put the high-Vt MOS in series the normal-Vt MOS, we have to keep a larger spacing from a high-Vt device to a normal–Vt device as compared with all high-Vt MOS case. Therefore, the PMOS parallel structure in Fig. 4.21(b) will not increase area overhead in the layout but the NMOS serial structure will increase area overhead in the layout due to the design rule.

The design rule of two neighbor transistors with the same threshold voltage can be very close. But we only replace one of the NMOSs in the stack, the design rule between high-Vt transistor and normal-Vt transistor are larger than two adjacent high-Vt or normal-Vt transistors. Although we have to pay some penalties for using

the MVT approach, we can gain the lower power consumption in the MVT gates.

For the AOI31 gate, the critical path of this gate in Fig. 4.22(a) is on the input C of stacked NMOSs and input A、B、C of parallel PMOSs. So we change the MOSs on the critical path into the normal-Vt transistors. The MVT AOI31 is shown in Fig. 4.22(b).



(a) NVT (b) MVT

Fig. 4.22 Schematic of AOI31 gate

Table 4.8 Power and leakage power of AOI31 with VDD=0.5 v

HVT	Power	Power_leak
In@A	212.70 nW	16.92 pW
In@B	197.41 nW	16.36 pW
In@C	190.70 nW	17.82 pW
In@D	174.92 nW	16.92 pW
MVT	Power	Power_leak
In@A	189.90 nW	22.42 pW
In@B	186.44 nW	20.70 pW
In@C	179.90 nW	16.36 pW
In@D	166.61 nW	21.23 pW

We can gain the 6.82% power consumption saving and just increase 18.6% leakage power than the HVT AOI31 gate in average. The power and leakage power of the AOI31 gate are shown in Table 4.8. The cells like AOI and OAI are very useful in the practical design. The circuit can be reduced a great deal of the area by using these

cells. So we design many kinds of AOI and OAI cells in our standard cell library.

4.4.3 1-bit Half Adder

We take another MVT gate, the 1-bit Half Adder, to be the example. We use the MVT approach to design our 1-bit half adder and to resize the transistors to let the propagation time be close to the 1-bit half adder using high-Vt gates, as shown in Fig. 4.23(a). The schematic of Mixed-Vt 1-bit Half Adder is shown in Fig. 4.23(b).

Fig. 4.23(a) Block diagram of 1-bit Half Adder

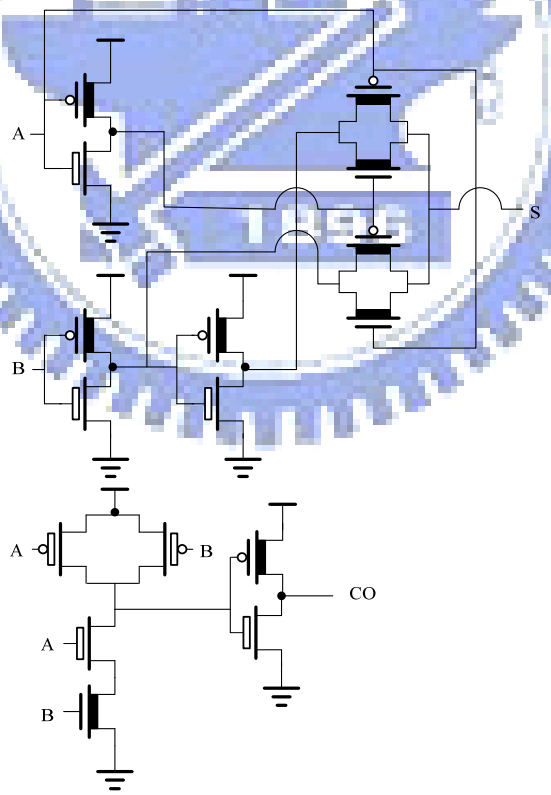


Fig. 4.24(b) Mixed-Vt 1-bit Half Adder schematic

The time and power table of our 1-bit half adder is shown in Table 4.9. We can see that the MVT 1-bit half adder gate can obtain around 13% power saving than the

HVT 1-bit half adder gate.

Table 4.9 Time and power table of 1-bit half adder with VDD=0.5 v

HVT	Tplh_CO (ps)	Tphl_CO (ps)	Tplh_S (ps)	Tphl_S (ps)	Power (nW)
In @ A	932.8	726.6	827.7	685.8	245.3
In @ B	877.7	679.7	646.2	436	198.4
MVT	Tplh_CO (ps)	Tphl_CO (ps)	Tplh_S (ps)	Tphl_S (ps)	Power (nW)
In @ A	851.1	717.5	796.4	669.1	203.4
In @ B	847.6	667.5	620.9	412.5	178.6

Tplh_CO: the propagation delay of CO from low to high
Tphl_CO: the propagation delay of CO from high to low
Tplh_S: the propagation delay of S from low to high
Tphl_S: the propagation delay of S from high to low

Table 4.10 shows all kinds of input combinations of leakage power of 1-bit half adder. The leakage power of our MVT 1-bit half adder increases around 4 times to HVT in average.

Table 4.10 Leakage power table of 1-bit half adder

unit: pW	00	01	10	11
HVT	251.6	235.9	217.2	178.4
MVT	872.5	703.5	693.8	689.7

4.4.4 DFF and SETDFF

The traditional master-slave D type flip-flop (DFF) is shown in Fig. 4.25 The critical path is marked by the dash line. We also use the MVT approach to design this gate and the power and leakage power of this traditional master-slave DFF is shown in the Table 4.11. By examining the Fig. 4.25, we can realize that the master-slave DFF is composed by many inverters and tri-state buffers. From Fig. 4.20(b), we can see that there is only one normal-Vt device without any high-Vt device in the pull up part. Because of the concerning about the timing performance, we don't use all high-Vt devices to compose the inverter gate and the buffer gate. This will cause that

there are many paths of the leakage current in the traditional master-slave DFF structure. We propose the single edge trigger flip-flop (SETDFF) in our standard cell library for designers to use it in sequential circuits. The SETDFF is shown in Fig. 4.26. The transistors on the critical path are mp1、mp4、mp6、mp7、mp8、mn8. We will change these devices into normal-Vt MOSs and other devices are still high-Vt MOSs. The power and leakage power of this SETDFF is shown in the Table 4.12. We can see that the leakage power of SETDFF is lower than the leakage power of traditional master-slave DFF. So the SETDFF in our standard cell library is effective to diminish the leakage in sequential circuits.

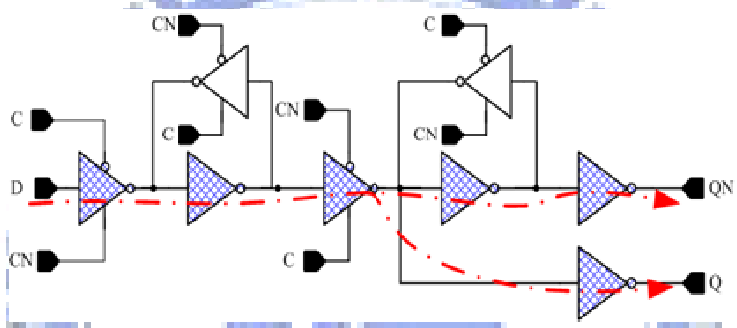


Fig. 4.25 Mixed-Vt DFF schematic

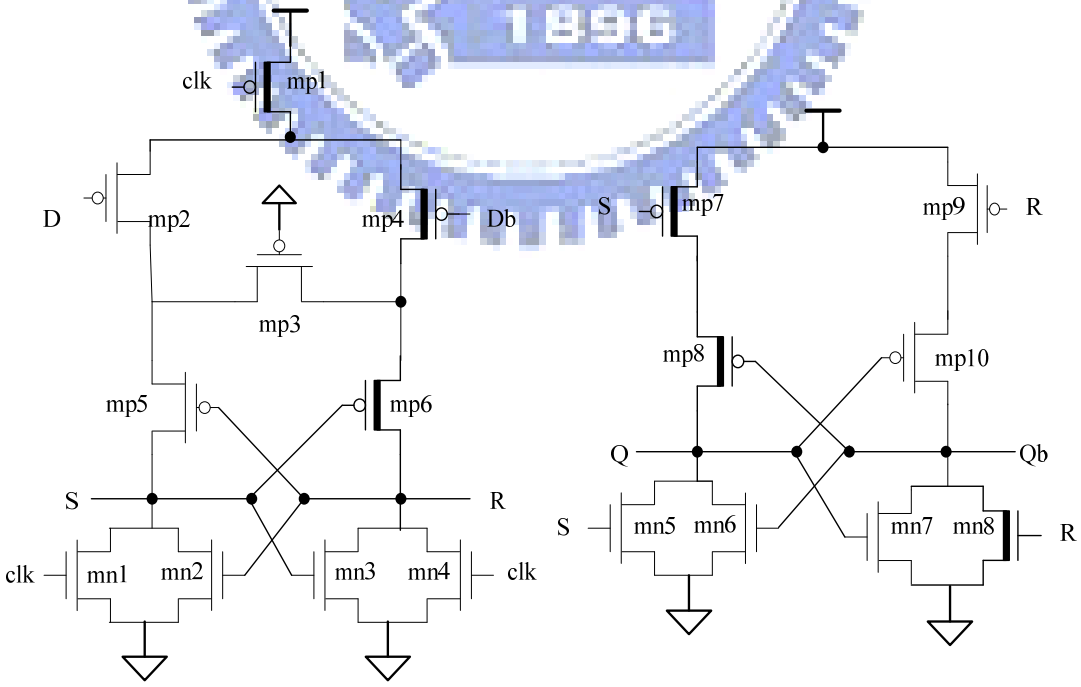


Fig. 4.26 Mixed-Vt SETDFF schematic

Table 4.11(a) Time and power table of Mixed-Vt DFF with VDD=0.5 v

HVT	Power (nW)	Leakage power (pW)	Tplh (ps)	Tphl (ps)
	90.5	247.2	1658.2	1321.3
MVT	Power (nW)	Leakage power (pW)	Tplh (ps)	Tphl (ps)
	62.2	2155.5	995.3	654.8

Table 4.121(b) Time and power table of Mixed-Vt DFF with VDD=1 v

HVT	Power (nW)	Leakage power (pW)	Tplh (ps)	Tphl (ps)
	391.9	510.3	267.3	248.2
MVT	Power (nW)	Leakage power (pW)	Tplh (ps)	Tphl (ps)
	308.9	4031.2	234.6	216.7

Table 4.132(a) Time and power table of Mixed-Vt SETDFF with VDD=0.5 v

HVT	Power (nW)	Leakage power (pW)	Tplh (ns)	Tphl (ns)
	72.2	164.7	2.476	1.914
MVT	Power (nW)	Leakage power (pW)	Tplh (ps)	Tphl (ps)
	60.3	789.8	2.072	1.416

Table 4.142(b) Time and power table of Mixed-Vt SETDFF with VDD=1 v

HVT	Power (nW)	Leakage power (pW)	Tplh (ps)	Tphl (ps)
	380.4	340.9	306.3	281.3
MVT	Power (nW)	Leakage power (pW)	Tplh (ps)	Tphl (ps)
	322.4	1368.6	286.7	250.9

4.5 Summary

We have introduced several low power circuit design methods and present a mixed-Vt method that can gain low dynamic and leakage power consumption. Then we use our mixed-Vt method to design a mixed-Vt low power standard cell library. In above sections, we explain the reasons why we choose the mixed-Vt method and describe the overall design flow and the simulation environment. We also explain the rule that is about the cell selecting criterion and the advantages we will obtain by using mixed-Vt method to establish a standard cell library. After that, we use some kinds of cells in our standard cell library for instance to demonstrate that our mixed-Vt method is very effective in performance.

From many experimental results, we can get around 5% to 30% dynamic power saving and the area is 0% to 30% larger than the standard cells with single high-Vt transistors. We size our standard cell library under 0.5 V supply voltage for the bio-electronics application. Then we characterization our standard cell library for 0.5 V、0.6 V and 1 V supply voltage. Because the threshold voltage of high-Vt transistor is about 0.3 V~0.4 V, we would use 0.6V instead of 0.5 V supply voltage to synthesize the design examples in Chapter 5.

Chapter 5

Design Examples by Low Power Standard Cell Library

5.1 C17

C17 is a benchmark of ISCAS85 as shown in Fig. 5.1 C17 Schematic is formed by six NAND gates.

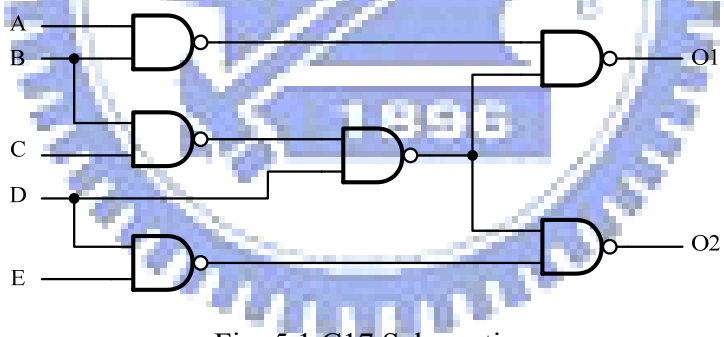


Fig. 5.1 C17 Schematic

We specify the input pin B、C、D to 1、1、1 and give active signals to input A and E to measure the time and power of C17 circuit. We use HVT and MVT NAND gates to establish this benchmark and the simulation result is shown in Table 5.1. We can see that the C17 circuit using the MVT approach can save around 9.4% power consumption and around 35.9% delay-power product less than the HVT one. Meanwhile, there is no area penalty due to the area of MVT NAND2 is equal to the HVT NAND2.

Table 5.1 Time and power table of C17 circuit with VDD=0.6 V

HVT	Tplh (ps)	Tphl (ps)	Power (nW)	Delay-Power (ns · nW)
In @ A	487.6	448.7	63.9	33.4
In @ E	577.6	578.6		
MVT	Tplh (ps)	Tphl (ps)	Power (nW)	Delay-Power (ns · nW)
In @ A	350.8	382.4	57.9	21.4
In @ E	370.8	371.8		

5.2 32-bit Ripple Adder

In this section, we will analyze the 32 bit ripple adder. We use our mix-Vt standard cell library and the high-Vt standard cell library of UMC to synthesize this circuit. We use the tool, prime power, to exercise 1M input pattern and the cycle time is 6 ns. The synthesis result is shown in the Table 5.2(a). We can see that the adder synthesized by our mix-Vt standard cell library can save around 12% dynamic power around 22% delay-power product less than the HVT one. The leakage power only increases around four times than the high-Vt one under the same clock rate. If we use the normal-Vt standard cell library to synthesize this 32 bit ripple adder, we can see that the leakage power will increase about eleven times than the high-Vt one. Although someone may think that the dynamic power is lower by using the normal-Vt standard cell library, designers will pay a very high cost for the leakage power in the low power circuit such as the portable products. We can see that the leakage power of the normal-Vt standard cell library case increases around eleven times than the high-Vt one under the same clock rate. We also synthesize this 32 bit ripple adder with 0.6 v supply voltage, and the result is shown in Table 5.2(b)

Table 5.2(a) Synthesis result of 32bit ripple adder with VDD=1 V

	Mixed-Vt@6ns	High-Vt@6ns	Normal-Vt@6ns
Internal power	48.1 uW	55.1 uW	44.2 uW
Switching power	4.4 uW	4.3 uW	4.8 uW
Total dynamic power	52.5 uW	59.4 uW	49.0uW
Leakage power	391.6 nW	100.1 nW	1100 .9nW
Total Power	52.9 uW	59.5 uW	50.1 uW
Total area	827.9	564.5	564.5
Delay	5.2 ns	5.9 ns	3.1 ns
Delay-Power	275.1 ns · uW	351.1 ns · uW	155.3 ns · uW

Table 5.2(b) Synthesis result of 32bit ripple adder with VDD=0.6 V

	Mixed-Vt@20ns
Internal power	4.7 uW
Switching power	749.1 nW
Total dynamic power	5.5 uW
Leakage power	145.4 nW
Total Power	5.6 uW
Total area	827.9
Delay	19.6 ns
Delay-Power	109.8 ns · uW

5.3 32-bit Wallace Tree Multiplier

We take another MVT circuit, the 32-bit wallace tree multiplier, to be the example. We also use the same input pattern with the same clock rate to measure the multipliers synthesized by our mix-Vt standard cell library, the high-Vt standard cell library and the normal-Vt standard cell library of UMC. The synthesis result is shown in the Table 5.3. We can see that the wallace tree multiplier synthesized by our mix-Vt

standard cell library can save around 9.4% dynamic power and around 29% delay-power product less than the HVT one. The leakage power increases around three times than the high-Vt one under the same clock rate. The normal-Vt one has the least dynamic power because the driving ability of the transistors in this library is better than that in the high-Vt and our mix-Vt standard cell library. But we can also observe that the leakage power is the major drawback of the normal-Vt standard cell library in this case. The leakage power will increase about nine times than the high-Vt one. We also synthesize this 32 bit wallace tree multiplier with 0.6 v supply voltage, and the result is shown in Table 5.3(b).

Table 5.3(a) Synthesis result of 32 bit wallace tree multiplier with VDD=1 V

	Mixed-Vt@6ns	High-Vt@6ns	Normal-Vt@6ns
Internal power	1.5 mW	1.8 mW	1.5 mW
Switching power	1.4 mW	1.4 mW	1.3 mW
Total dynamic power	2.9 mW	3.2 mW	2.8 mW
Leakage power	10.1 uW	3.1 uW	29.6 uW
Total Power	2.9 mW	3.2 mW	2.8 mW
Total area	25512.1	18975.9	18885.6
Delay	3.94 ns	5.05 ns	2.23 ns
Delay-Power	11.4 ns · mW	16.2 ns · mW	6.3 ns · mW

Table 5.3(b) Synthesis result of 32 bit wallace tree multiplier with VDD=0.6 V

	Mixed-Vt@20ns
Internal power	55.3 uW
Switching power	35.9 uW
Total dynamic power	91.2 uW
Leakage power	3.7 uW
Total Power	94.9 uW
Total area	25956.7
Delay	12.48 ns
Delay-Power	1184.4 ns · uW

5.4 32-bit Shift Register

We have introduced the circuits that composed of the combination logic in the above sections. In this sections, we take the 32bit shift register for instance to compare the sequential circuit synthesized by our mix-Vt standard cell library with the high-Vt and normal Vt standard cell library. The synthesis result is shown in the Table 5.. The shift register synthesized by our mix-Vt standard cell library can save around 14.2% dynamic power and around 54% delay-power product less than the HVT one. The leakage power increase around 11.29% than the high-Vt one under the same clock rate. We can see that the leakage power of the 32bit shift register with our mix-Vt standard cell library and the high-Vt one are very close. The leakage power of the normal-Vt standard cell library case is still larger than the high-Vt and our mix-Vt standard cell library cases. And we can see the area of our mix-Vt case increases around 22.8% than the high-Vt and the normal-Vt case. We also synthesize this 32 bit shift register with 0.6 v supply voltage, and the result is shown in Table 5.3(b).

Table 5.4(a) Synthesis result of 32 bit shift register with VDD=1 V

	Mixed-Vt@4ns	High-Vt@4ns	Normal-Vt@4ns
Internal power	98.7 uW	116.4 uW	97.1 uW
Switching power	17.2 uW	18.7 uW	18.4 uW
Total dynamic power	115.9 uW	135.1 uW	115.5 uW
Leakage power	239.4 nW	215.1 nW	741.9 nW
Total Power	116.1 uW	135.3 uW	117.2 uW
Total area	1010.5	822.7	828.4
Delay	0.46 ns	0.87 ns	0.29 ns
Delay-Power	53.4 ns · uW	117.7 ns · uW	33.9 ns · uW

Table 5.4(b) Synthesis result of 32 bit shift register with VDD=0.6 V

	Mixed-Vt@12ns
Internal power	17.3 uW
Switching power	4.6 uW
Total dynamic power	21.9 uW
Leakage power	98.1 nW
Total Power	22.0 uW
Total area	1183.1
Delay	3.62 ns
Delay-Power	79.6 ns · uW

Chapter 6

Conclusions

In this thesis, firstly, we have introduced the background of the standard cell library and its format. Then we discuss the timing and power models and their advantages and drawbacks in the present standard cell library. Because of the non-linear and non-ideal effects in the novel nanometer scale process, we can realize that it becomes difficult to model the timing ad power accurately using the conventional method. Therefore, we use the NLDM table to record the input dependent leakage power for all combinations of static input vectors and add the new timing/power model in the characterization tool to let the timing/power model of our standard cell library be more accurate in the 90nm or below process.

After realizing the background of the standard cell library and the flow of characterizing the cells, we would like to establish a low power standard cell library for the hearing aid using the solar battery or the portable electric products. So we have proposed a method using the mixed-Vt technique to reduce the power consumption for the low power circuit at first. This method is to replace the MOSs on the critical path and resizing them to achieve our low power requirements. Then we establish a 90nm low power standard cell library. Design examples are used to compare the performance and we can draw the conclusion that we can have around 5% to 30% dynamic power saving, 20% to 55% delay-power product saving and the area is 0% to 40% larger than the standard cells with single high-Vt transistors.

Reference

[1] L. Wei, K. Roy, and V. De, “Low Voltage Low Power CMOS Design Techniques for Deep Submicron ICs,” *VLSI Design, 2000. Thirteenth International Conference on* 3-7 Jan. 2000, pp: 24-29.

[2] K.S. Yeo and K. Roy, *Low-Voltage, Low-Power VLSI Subsystems*, 1st ed. New York: McGraw-Hill, 2005.

[3] IEEE Standard for an Advanced Library Format (ALF) Describing Integrated Circuit (IC) Technology, Cells, and Blocks, *IEEE Standard* 1603™-2003, 2004.

[4] Synopsys Corporation, *Library Compiler User Guide: Modeling Timing and Power*, 2004.

[5]郭建興, *標準元件庫特徵化程式使用者手冊 (User's Guide of Characterization Utilities)*, 2003.

[6] Neil H. E. Weste, and David Harris, CMOS VLSI DESIGN – A Circuit and System Perspective,” 3rd ed., Addison Wesley, 2004.

[7] S. Mutoh, et al., “1-V Power Supply High-Speed Digital Circuit Technology with Multithreshold-Voltage CMOS”, *IEEE Journal of Solid-State Circuits*, vol.30, no.8, pp. 847-854, 1995.

[8] S. Shigematsu, S. Mutoh, Y. Matsuya, Y. Tanabe, J. Yamada, “A 1-V high speed MTCMOS circuit scheme for power-down applications”, *IEEE J. Solid-State Circuits*, vol.32, pp. 861-869, Nov. 6, 1997.

[9] H. Kawaguchi, K. Nose and T. Sakurai, “A CMOS Scheme for 0.5V Supply Voltage with Pico-Ampere Standby Current”, *IEEE International Solid-State Circuits Conference*, 1998, pp. 192-193.

[10] L. Wei, Z. Chen, K. Roy, Y. Ye, V. De, “Mixed-Vth (MVT) CMOS Circuit Design Methodology for Low Power Applications” *Design Automation Conference, 1999. Proceedings. 36th*, Jun. 1999, pp. 430-435.

[11] F. Assaderaghi, et al., "A Dynamic Threshold Voltage MOSFET(DTMOS) for Ultra-Low Voltage Operation", *IEEE International Electron Devices Meeting*, 1994, pp. 809-812.

[12] F. Sill, F. Grassert, and D. Timmermann, “Low Power Gate-level Design with Mixed-Vth (MVT) Techniques”, *Integrated Circuits and Systems Design, 2004. SBCCI 2004. 17th Symposium*, Sep. 2004. pp. 278-282.

[13] K. Usami and M. Horowitz, “Clustered voltage scaling technique for low power design”, in Proc. ISLPD, Apr. 1995, pp. 3–8.

[14] Y. J. Yeh, S. Y. Kuo, and J. Y. Jou, “Converter-Free Multiple-Voltage Scaling Techniques for Low-Power CMOS Digital Design”, *IEEE Transactions on*

Computer-Aided Design of Integrated Circuits and Systems, Vol. 20, No. 1, January 2001.

[15] Standard Cell Characterization, Richard Sohnius, 16.12 2003.

[16] TSMC 0.13um (CL013G) Process 1.2-Volt SAGE-XTM Standard Cell Library Databook, Release 2.5, June 2002.

[17] UMC 90nm SP_RVT process 1.0V standard cell library databook.

[188] S. I. CHANG, J. H. LEE and H. C. SHIN, "Gate-Induced Drain Leakage Currents in Metal Oxide Semiconductor Field Effect Transistors with High-k Dielectric", *Japan J. Applied Physics*, Vol. 41 Part 1, No. 7A, 15 July 2002.

[19] W. T. Kang, J. S. Kim, K. Y. Lee, Y. C. Shin, T. H. Kim, Y. J. Park, and J. W. Park, "The Leakage Current Improvement in an Ultrashallow Junction NMOS with Co Silicided Source and Drain", *IEEE Electron Device Letters,* Vol. 41 Part 1, No. 7A, 15 July 2002.

作者簡歷

姓名：林俊誼

出生地：台灣省台中市

出生日期：1983 年 6 月 21 日

學歷：
1989.9 ~ 1995.6 台中市立成功國小
1995.9 ~ 1998.6 台中市立光明國中
1998.9 ~ 2001.6 國立台中一中
2001.9 ~ 2005.6 國立交通大學　電機與控制工程學系　學士
2005.9 ~ 2007.8 國立交通大學　電子研究所　系統組　碩士