ASIC LIBRARY DESIGN

3

Key concepts: Tau, logical effort, and the prediction of delay • Sizes of cells, and their drive strengths • Cell importance • The difference between gate-array macros, standard cells, and datapath cells

ASIC design uses predefined and precharacterized cells from a library—so we need to design or buy a cell library. A knowledge of ASIC library design is not necessary but makes it easier to use library cells effectively.

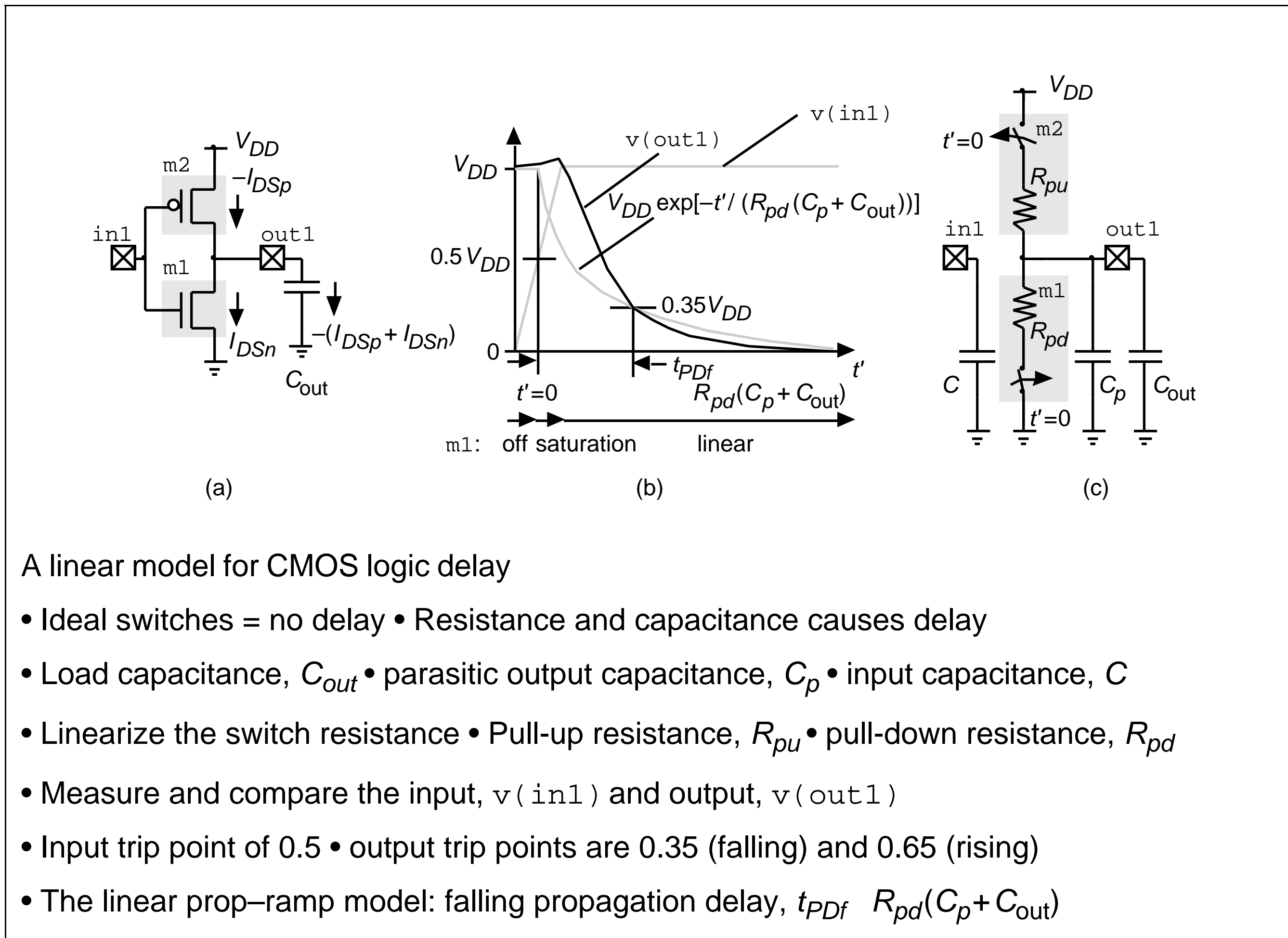
3.1 Transistors as Resistors

$$0.35 V_{DD} = V_{DD} \exp \frac{-t_{PDf}}{R_{pd}(C_{out} + C_p)}$$

An output trip point of 0.35 is convenient because $\ln(1/0.35) = 1.04 \quad 1$ and thus

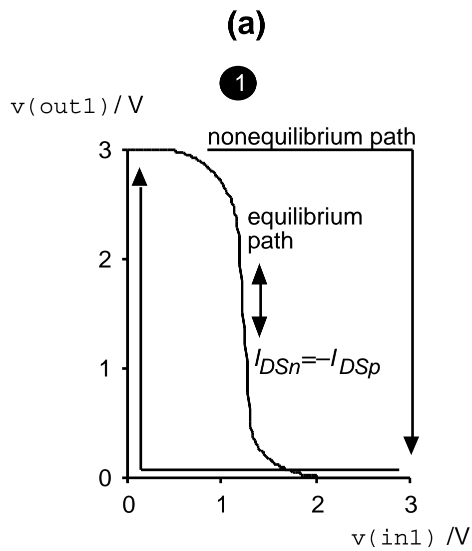
$t_{PDf} = R_{pd}(C_{out} + C_p) \ln(1/0.35) \quad R_{pd}(C_{out} + C_p)$

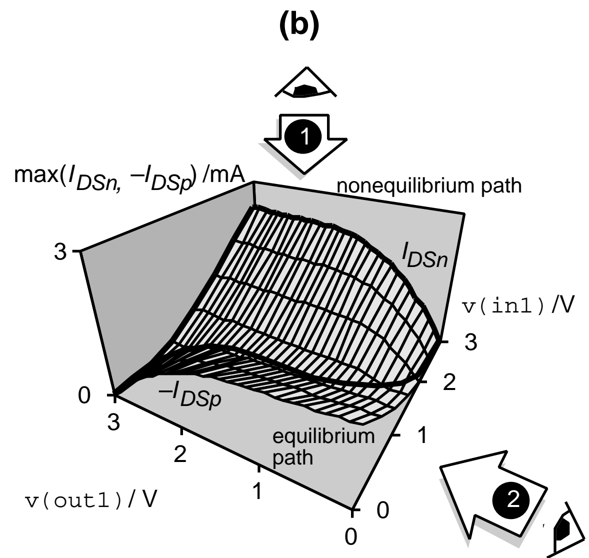
For output trip points of 0.1/0.9 we multiply by $-\ln(0.1) = 2.3$, because $\exp(-2.3) = 0.100$

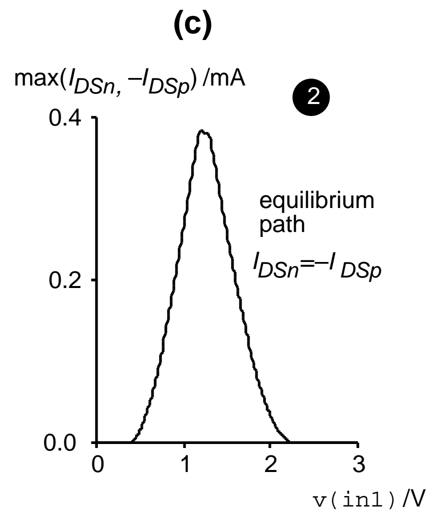


A linear model for CMOS logic delay

- Ideal switches = no delay • Resistance and capacitance causes delay
- Load capacitance, C_{out} • parasitic output capacitance, C_p • input capacitance, C
- Linearize the switch resistance • Pull-up resistance, R_{pu} • pull-down resistance, R_{pd}
- Measure and compare the input, `v(in1)` and output, `v(out1)`
- Input trip point of 0.5 • output trip points are 0.35 (falling) and 0.65 (rising)
- The linear prop–ramp model: falling propagation delay, t_{PDf} $R_{pd}(C_p + C_{out})$



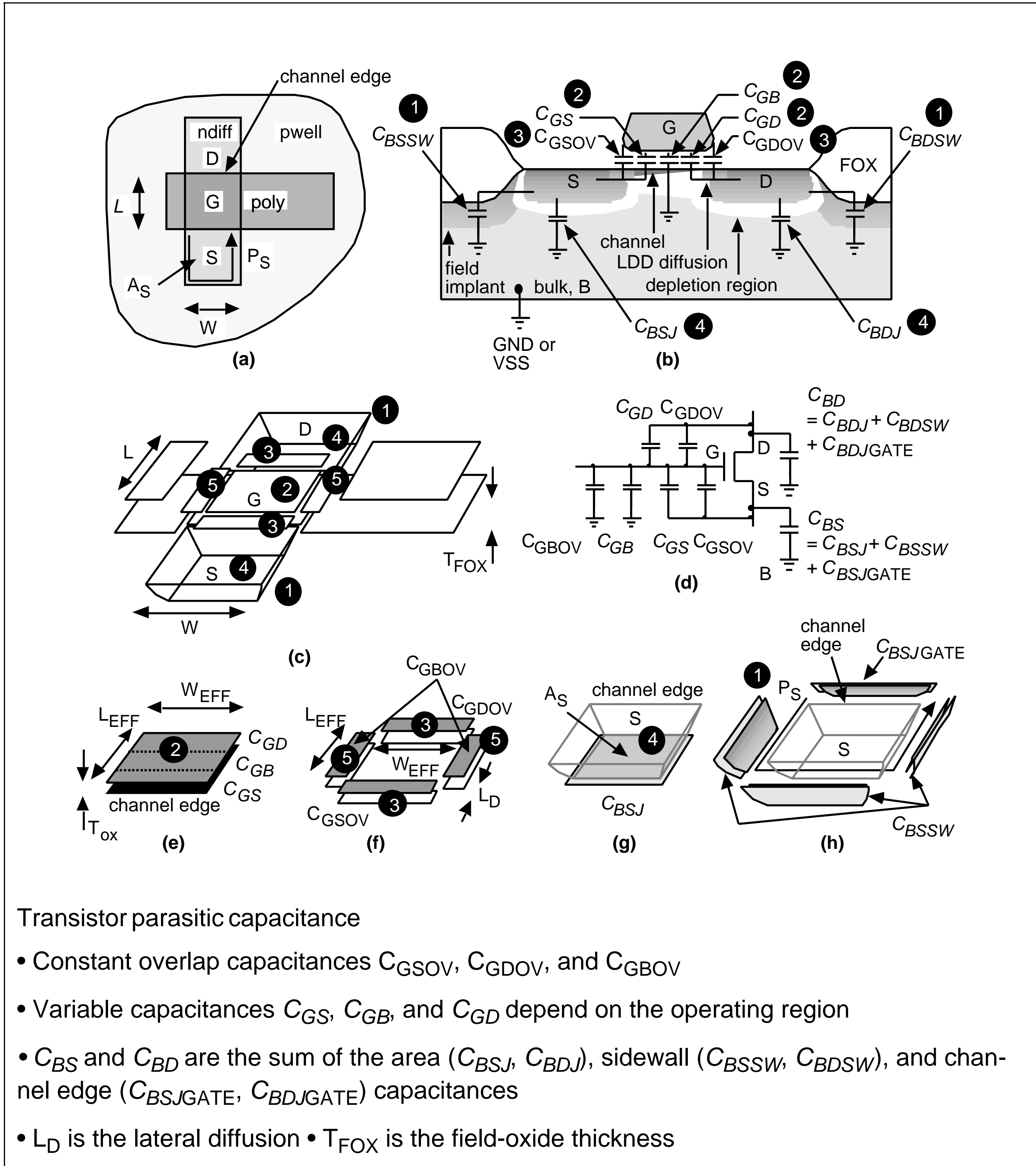




CMOS inverter characteristics

- Equilibrium switching
- Non-equilibrium switching
- Nonlinear switching resistance
- Switching current

3.2 Transistor Parasitic Capacitance



Transistor parasitic capacitance

- Constant overlap capacitances C_{GSOV}, C_{GDOV}, and C_{GBOV}
- Variable capacitances C_{GS}, C_{GB}, and C_{GD} depend on the operating region
- C_{BS} and C_{BD} are the sum of the area (C_{BSJ}, C_{BDJ}), sidewall (C_{BSSW}, C_{BDSW}), and channel edge ($C_{BSJ\mathrm{GATE}}$, $C_{BDJ\mathrm{GATE}}$) capacitances
- L_D is the lateral diffusion • T_{FOX} is the field-oxide thickness

NAME	m1	m2
MODEL	CMOSN	CMOSP
ID	7.49E-11	-7.49E-11
VGS	0.00E+00	-3.00E+00
VDS	3.00E+00	-4.40E-08
VBS	0.00E+00	0.00E+00
VTH	4.14E-01	-8.96E-01
VDSAT	3.51E-02	-1.78E+00
GM	1.75E-09	2.52E-11
GDS	1.24E-10	1.72E-03
GMB	6.02E-10	7.02E-12
CBD	2.06E-15	1.71E-14
CBS	4.45E-15	1.71E-14
CGSOV	1.80E-15	2.88E-15
CGDOV	1.80E-15	2.88E-15
CGBOV	2.00E-16	2.01E-16
CGS	0.00E+00	1.10E-14
CGD	0.00E+00	1.10E-14
CGB	3.88E-15	0.00E+00

- `ID` (I_{DS}), `VGS`, `VDS`, `VBS`, `VTH` (V_t), and `VDSAT` ($V_{DS(sat)}$) are DC parameters
- `GM`, `GDS`, and `GMB` are small-signal conductances (corresponding to I_{DS}/ V_{GS}, I_{DS}/ V_{DS}, and I_{DS}/ V_{BS}, respectively)

Calculations of parasitic capacitances for an n-channel MOS transistor.		
PSpice	**Equation**	**Values[1] for V_{GS}=0V, V_{DS}=3V, V_{SB}=0V**
CBD	$C_{BD} = C_{BDJ} + C_{BDSW}$ $C_{BDJ} + A_D\ C_J\ (1 + V_{DB}/\ _B)^{-mJ}$ ($_B$ = PB) $C_{BDSW} = P_D\ C_{JSW}\ (1 + V_{DB}/\ _B)^{-mJSW}$ (P_D may or may not include channel edge)	$C_{BD} = 1.855 \times 10^{-13} + 2.04 \times 10^{-16} = 2.06 \times 10^{-13}$ F $C_{BDJ} = (4.032 \times 10^{-15})(1 + (3/1))^{-0.56} = 1.86 \times 10^{-15}$ F $C_{BDSW} = (4.2 \times 10^{-16})(1 + (3/1))^{-0.5} = 2.04 \times 10^{-16}$ F
CBS	$C_{BS} = C_{BSJ} + C_{BSSW}$ $C_{BSJ} + A_S\ C_J\ (1 + V_{SB}/\ _B)^{-mJ}$ $C_{BSSW} = P_S\ C_{JSW}\ (1 + V_{SB}/\ _B)^{-mJSW}$	$C_{BS} = 4.032 \times 10^{-15} + 4.2 \times 10^{-16} = 4.45 \times 10^{-15}$ F $A_S\ C_J = (7.2 \times 10^{-15})(5.6 \times 10^{-4}) = 4.03 \times 10^{-15}$ F $P_S\ C_{JSW} = (8.4 \times 10^{-6})(5 \times 10^{-11}) = 4.2 \times 10^{-16}$ F
CGSOV	$C_{GSOV}=W_{EFF}C_{GSO}$; $W_{EFF}=W-2WD$	$C_{GSOV} = (6 \times 10^{-6})(3 \times 10^{-10}) = 1.8 \times 10^{-16}$ F
CGDOV	$C_{GDOV}=W_{EFF}C_{GSO}$	$C_{GDOV} = (6 \times 10^{-6})(3 \times 10^{-10}) = 1.8 \times 10^{-15}$ F
CGBOV	$C_{GBOV}=L_{EFF}C_{GBO}$; $L_{EFF}=L-2L_D$	$C_{GDOV} = (0.5 \times 10^{-6})(4 \times 10^{-10}) = 2 \times 10^{-16}$ F
CGS	C_{GS}/C_O = 0 (off), 0.5 (lin.), 0.66 (sat.) C_O (oxide capacitance) = $W_{EF}\ L_{EFF}\ \ _{ox} / T_{ox}$	$C_O = (6 \times 10^{-6})(0.5 \times 10^{-6})(0.00345) = 1.03 \times 10^{-14}$ F C_{GS} = 0.0 F
CGD	C_{GD}/C_O = 0 (off), 0.5 (lin.), 0 (sat.)	C_{GD} = 0.0 F
CGB	C_{GB} = 0 (on), = C_O in series with C_{GS} (off)	$C_{GB} = 3.88 \times 10^{-15}$ F, C_S=depletion capacitance
[1]Input	.MODEL CMOSN NMOS LEVEL=3 PHI=0.7 TOX=10E-09 XJ=0.2U TPG=1 VTO=0.65 DELTA=0.7 + LD=5E-08 KP=2E-04 UO=550 THETA=0.27 RSH=2 GAMMA=0.6 NSUB=1.4E+17 NFS=6E+11 + VMAX=2E+05 ETA=3.7E-02 KAPPA=2.9E-02 CGDO=3.0E-10 CGSO=3.0E-10 CGBO=4.0E-10 + CJ=5.6E-04 MJ=0.56 CJSW=5E-11 MJSW=0.52 PB=1 m1 out1 in1 0 0 cmosn W=6U L=0.6U AS=7.2P AD=7.2P PS=8.4U PD=8.4U	

3.2.1 Junction Capacitance

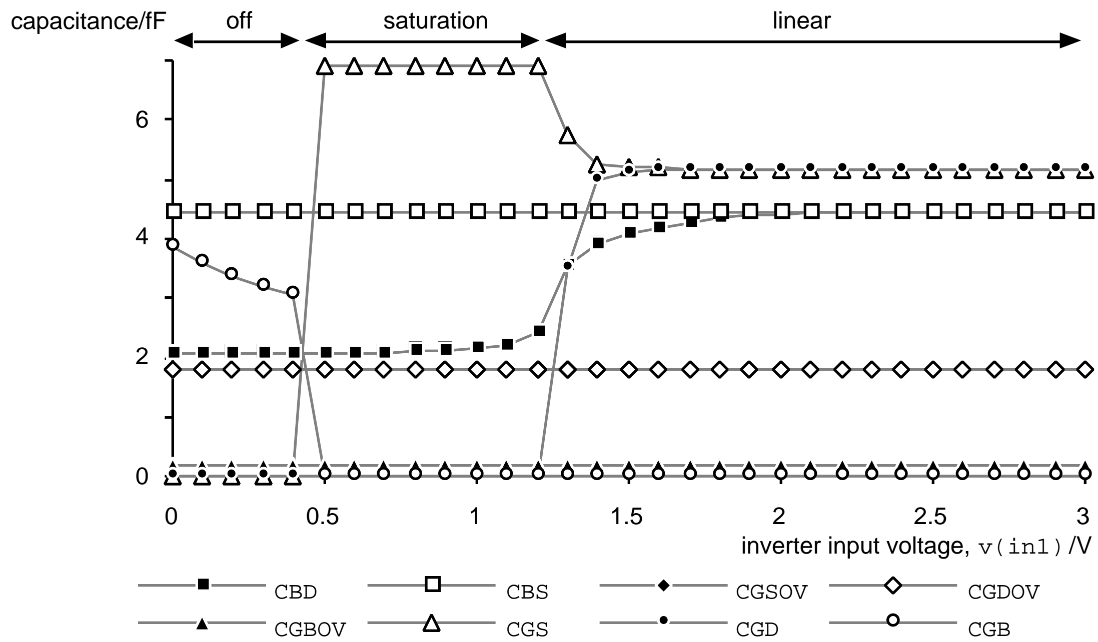
- Junction capacitances, C_{BD} and C_{BS}, consist of two parts: junction area and sidewall
- Both C_{BD} and C_{BS} have different physical characteristics with parameters: `CJ` and `MJ` for the junction, `CJSW` and `MJSW` for the sidewall, and `PB` is common
- C_{BD} and C_{BS} depend on the voltage across the junction (V_{DB} and V_{SB})
- The sidewalls facing the channel ($C_{BSJ\mathrm{GATE}}$ and $C_{BDJ\mathrm{GATE}}$) are different from the sidewalls that face the field
- It is a mistake to exclude the gate edge assuming it is in the rest of the model—it is not
- In HSPICE there is a separate mechanism to account for the channel edge capacitance (using parameters `ACM` and `CJGATE`)

3.2.2 Overlap Capacitance

- The overlap capacitance calculations for $\mathrm{C_{GSOV}}$ and $\mathrm{C_{GDOV}}$ account for lateral diffusion
- SPICE parameter `LD=5E-08` or $\mathrm{L_D}$=0.05 μm
- Not all SPICE versions use the equivalent parameter for width reduction, `WD`, in calculating $\mathrm{C_{GDOV}}$
- Not all SPICE versions subtract $\mathrm{W_D}$ to form $\mathrm{W_{EFF}}$

3.2.3 Gate Capacitance

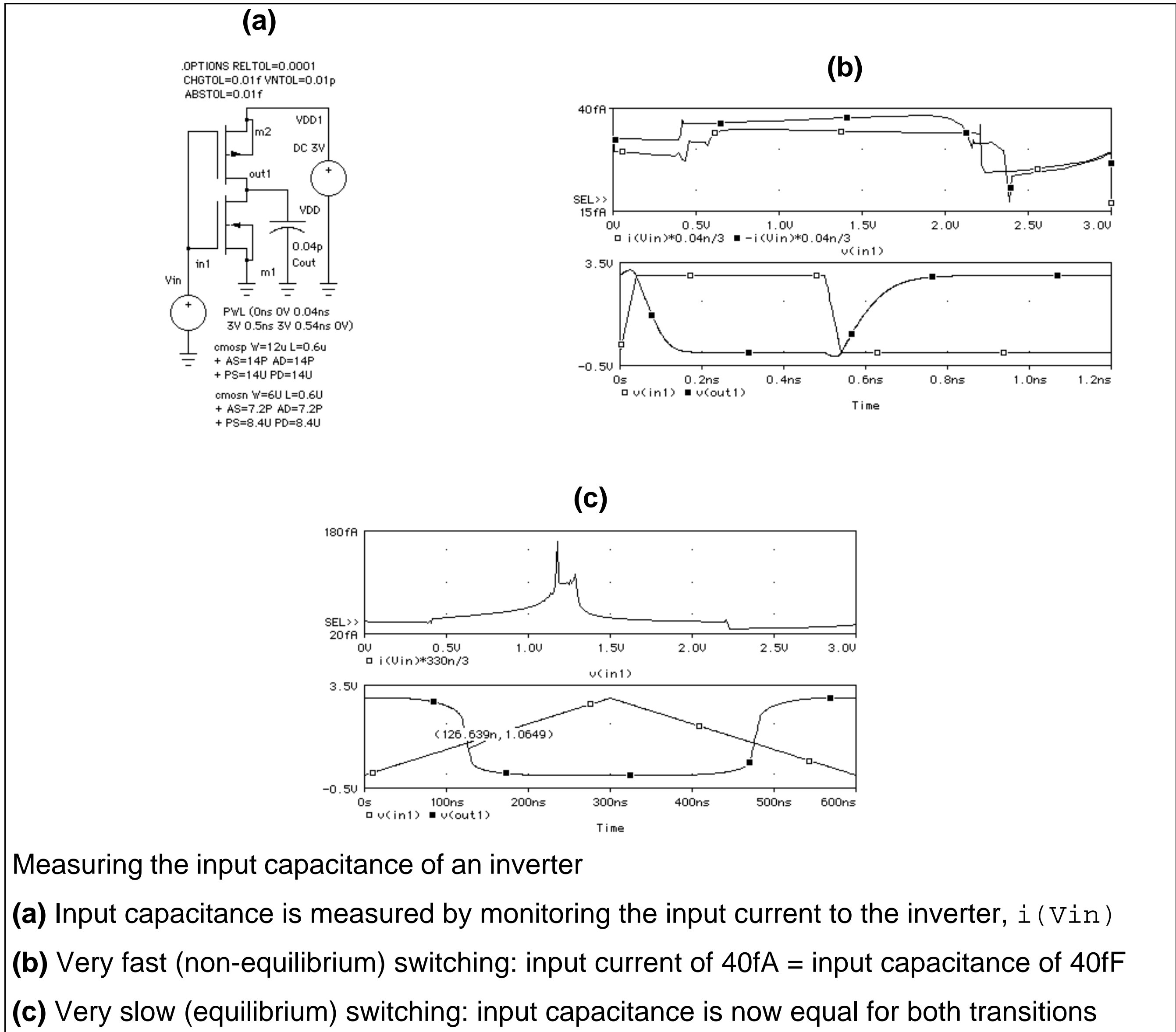
- The gate capacitance depends on the operating region
- The gate–source capacitance C_{GS} varies from zero (off) to 0.5$\mathrm{C_O}$ in the linear region to (2/3)$\mathrm{C_O}$ in the saturation region
- The gate–drain capacitance C_{GD} varies from zero (off) to 0.5$\mathrm{C_O}$ (linear region) and back to zero (saturation region)
- The gate–bulk capacitance C_{GB} is two capacitors in series: the fixed gate-oxide capacitance, $\mathrm{C_O}$, and the variable depletion capacitance, C_S
- As the transistor turns on the channel shields the bulk from the gate—and C_{GB} falls to zero
- Even with V_{GS}=0V, the depletion width under the gate is finite and thus C_{GB} is less than $\mathrm{C_O}$



The variation of n-channel transistor parasitic capacitance

- PSpice v5.4 (LEVEL=3)
- Created by varying the input voltage, v(in1), of an inverter
- Data points are joined by straight lines
- Note that CGSOV=CGDOV

3.2.4 Input Slew Rate

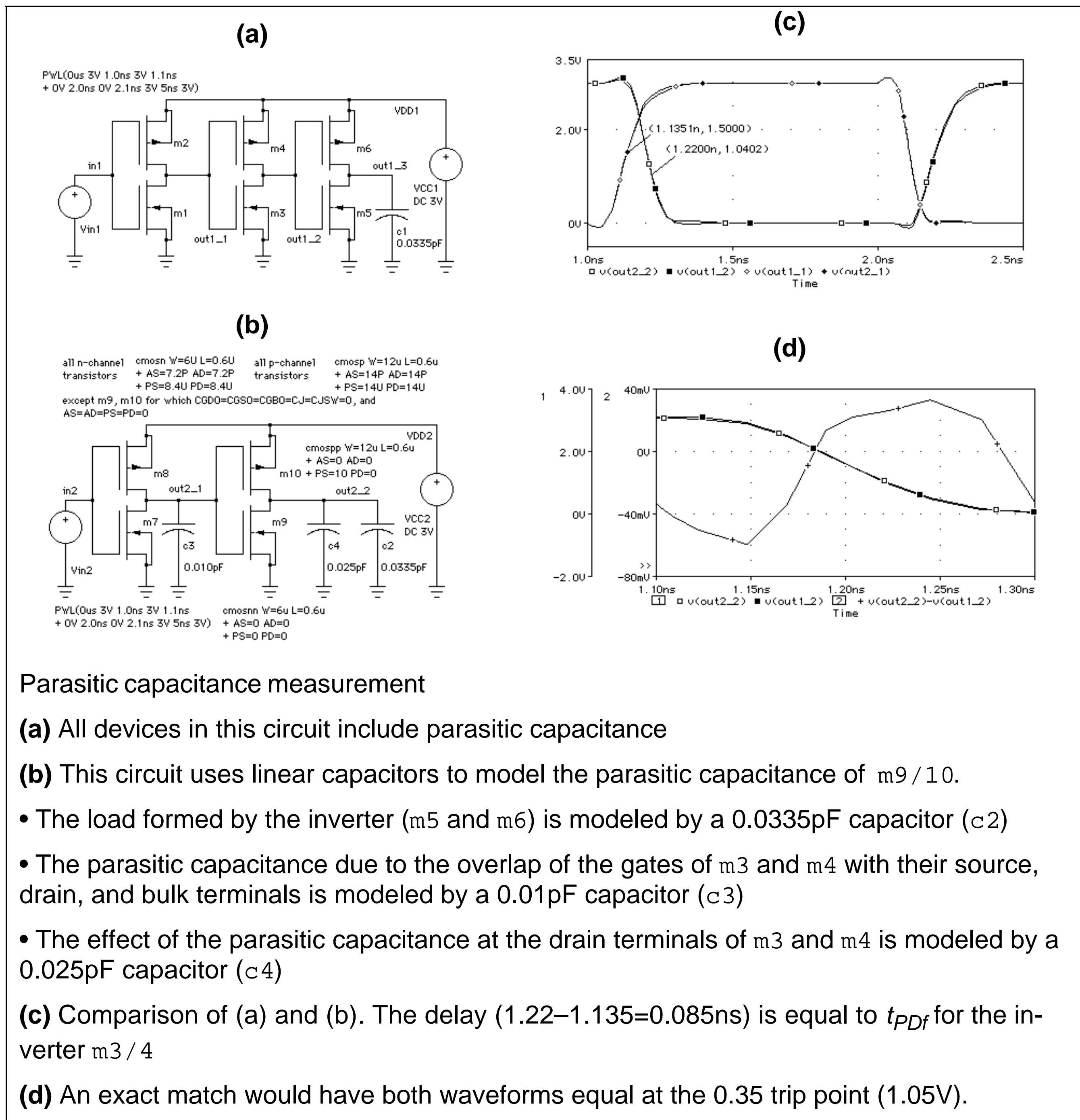


Measuring the input capacitance of an inverter

(a) Input capacitance is measured by monitoring the input current to the inverter, `i(Vin)`

(b) Very fast (non-equilibrium) switching: input current of 40fA = input capacitance of 40fF

(c) Very slow (equilibrium) switching: input capacitance is now equal for both transitions



Parasitic capacitance measurement

(a) All devices in this circuit include parasitic capacitance

(b) This circuit uses linear capacitors to model the parasitic capacitance of `m9/10`.

• The load formed by the inverter (`m5` and `m6`) is modeled by a 0.0335pF capacitor (`c2`)

• The parasitic capacitance due to the overlap of the gates of `m3` and `m4` with their source, drain, and bulk terminals is modeled by a 0.01pF capacitor (`c3`)

• The effect of the parasitic capacitance at the drain terminals of `m3` and `m4` is modeled by a 0.025pF capacitor (`c4`)

(c) Comparison of (a) and (b). The delay (1.22–1.135=0.085ns) is equal to t_{PDf} for the inverter `m3/4`

(d) An exact match would have both waveforms equal at the 0.35 trip point (1.05V).

3.3 Logical Effort

We extend the prop–ramp model with a "catch all" term, t_q, that includes:

- delay due to internal parasitic capacitance
- the time for the input to reach the switching threshold of the cell
- the dependence of the delay on the slew rate of the input waveform

$t_{PD} = R(C_{out} + C_p) + t_q$

We can **scale** any logic cell by a scaling factor s: $t_{PD} = (R/s)\cdot(C_{out} + sC_p) + st_q$

$$t_{PD} = RC \frac{C_{out}}{C_{in}} + RC_p + st_q$$

Normalizing the delay: $d = \dfrac{(RC)\,(C_{out} / C_{in}) + RC_p + st_q}{\quad} = f + p + q$

The time constant **tau**, $= R_{inv}\, C_{inv}$, is a basic property of any CMOS technology

The delay equation is the sum of three terms, $d = f + p + q$ or
delay = **effort delay** + **parasitic delay** + **nonideal delay**

The effort delay f is the product of **logical effor**t, g, and **electrical effort**, h: $f = gh$

Thus, delay = logical effort × electrical effort + parasitic delay + nonideal delay

- R and C will change as we scale a logic cell, but the RC product stays the same
- Logical effort is independent of the size of a logic cell
- We can find logical effort by scaling a logic cell to have the same drive as a 1X minimum-size inverter
- Then the logical effort, g, is the ratio of the input capacitance, C_{in}, of the 1X logic cell to C_{inv}

2 units of gate capacitance

1 unit

C_{inv}

1. Measure the input capacitance of a minimum-size inverter.

C_{inv} =2+1=3

2. Make the cell have the same drive strength as a minimum-size inverter.

3. Measure ratio of cell input capacitance to that of a minimum-size inverter.

C_{in} =2+2=4

$g = C_{in}/C_{inv}$= 4/3

(a) (b) (c)

Logical effort • For a two-input NAND cell, the logical effort, g=4/3

(a) Find the input capacitance, C_{inv}, looking into the input of a minimum-size inverter in terms of the gate capacitance of a minimum-size device

(b) Size a logic cell to have the same drive strength as a minimum-size inverter (assuming a logic ratio of 2). The input capacitance looking into one of the logic-cell terminals is then C_{in}

(c) The logical effort of a cell is C_{in}/ C_{inv}

The h depends only on the load capacitance C_{out} connected to the output of the logic cell and the input capacitance of the logic cell, C_{in}; thus

electrical effort $h = C_{out} /C_{in}$

parasitic delay $p = RC_p/$ (the parasitic delay of a minimum-size inverter is: $p_{inv} = C_p/ C_{inv}$)

nonideal delay $q = st_q /$

Cell effort, parasitic delay, and nonideal delay (in units of) for single-stage CMOS cells				
Cell	**Cell effort (logic ratio=2)**	**Cell effort (logic ratio=r)**	**Parasitic delay/**	**Nonideal delay/**
inverter	1 (by definition)	1 (by definition)	p_{inv} (by definition)	q_{inv} (by definition)
n-input NAND	$(n+2)/3$	$(n+r)/(r+1)$	np_{inv}	nq_{inv}
n-input NOR	$(2n+1)/3$	$(nr+1)/(r+1)$	np_{inv}	nq_{inv}

3.3.1 Predicting Delay

- Example: predict the delay of a three-input NOR logic cell
- 2X drive
- driving a net with a fanout of four
- 0.3pF total load capacitance (input capacitance of cells we are driving plus the interconnect)
- $p=3p_{\mathrm{inv}}$ and $q=3q_{\mathrm{inv}}$ for this cell
- the input gate capacitance of a 1X drive, three-input NOR logic cell is equal to gC_{inv}
- for a 2X logic cell, $C_{\mathrm{in}} = 2gC_{\mathrm{inv}}$

$$gh = g \frac{C_{\mathrm{out}}}{C_{\mathrm{in}}} = \frac{g\cdot(0.3\ \mathrm{pF})}{2gC_{\mathrm{inv}}} = \frac{(0.3\ \mathrm{pF})}{(2)\cdot(0.036\ \mathrm{pF})}$$ (Notice g cancels out in this equation)

The delay of the NOR logic cell, in units of , is thus

$$d = gh + p + q = \frac{0.3 \times 10^{-12}}{(2)\cdot(0.036 \times 10^{-12})} + (3)\cdot(1) + (3)\cdot(1.7)$$

$= 4.1666667 + 3 + 5.1$

$= 12.266667$ equivalent to an absolute delay, t_{PD} $12.3 \times 0.06\mathrm{ns} = 0.74\mathrm{ns}$

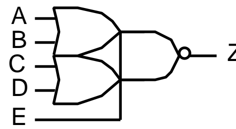
The delay for a 2X drive, three-input NOR logic cell is $t_{PD} = (0.03 + 0.72C_{\mathrm{out}} + 0.60)$ ns

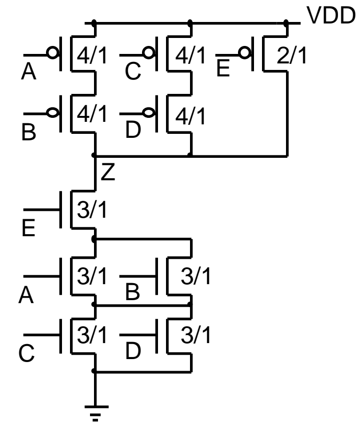
With $C_{\mathrm{out}}=0.3\mathrm{pF}$, $t_{PD} = 0.03 + (0.72)\cdot(0.3) + 0.60 = 0.846$ ns compared to our prediction of 0.74ns

3.3.2 Logical Area and Logical Efficiency

An OAI221 logic cell

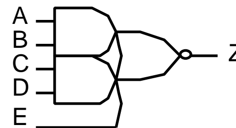
- **Logical-effort vector g**=(7/3, 7/3, 5/3)
- The **logical area** is 33 **logical squares**

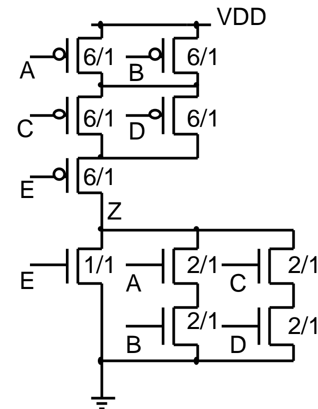




An AOI221 logic cell

- **g**=(8/3, 8/3, 7/3)
- Logical area is 39 logical squares
- Less **logically efficient** than OAI221





3.3.3 Logical Paths

$$\textbf{path delay } D = \sum_{i \in \text{path}} g_i h_i + \sum_{i \in \text{path}} (p_i + q_i)$$

3.3.4 Multistage Cells

(a)

$g_0 = 1$, $p_0 = 1$, $q_0 = 1.7$; $g_2 = 1.4$, $p_2 = 2$, $q_2 = 3.4$; $g_3 = 1.4$, $p_3 = 2$, $q_3 = 3.4$; $g_1 = (2, 1.6)$, $p_1 = 3$, $q_1 = 5.4$; $g_4 = 1$, $p_4 = 1$, $q_4 = 1.7$

AOI221, AOI21, A1, A2, B1, B2, C, ZN

delay d1; $h_0 = 1.4$; $h_2 = 1.0$; $h_3 = 0.7$; $h_4 = C_L$

$$d1 = (g_0 h_0 + p_0 + q_0) + (g_2 h_2 + p_2 + q_2) + (g_3 h_3 + p_3 + q_3) + (g_4 h_4 + p_4 + q_4)$$

$$= (1 \times 1.4 + 1 + 1.7) + (1.4 \times 1 + 2 + 3.4) + (1.4 \times 0.7 + 2 + 3.4) + (1 \times C_L + 1 + 1.7) = 20 + C_L$$

(b)

$g_0 = 1$, $p_0 = 1$, $q_0 = 1.7$; $g_1 = (2.6, 2.6, 2.2)$, $p_1 = 5$, $q_1 = 8.5$

AOI221, delay d1

$$d1 = (1 \times 2.6 + 1 + 1.7) + (1 \times C_L + 5 + 8.5) = 18.8 + C_L$$

(b) is slightly faster than (a)

Logical paths • Comparison of multistage and single-stage implementations

(a) An AOI221 logic cell constructed as a multistage cell, $d1 = 20 + C_L$

(b) A single-stage AOI221 logic cell, $d1 = 18.8 + C_L$

3.3.5 Optimum Delay

path logical effort $G = \quad g_i$

$\qquad\qquad i \quad \text{path}$

path electrical effort $H = \quad h_i \quad \dfrac{C_{out}}{C_{in}}$

$\qquad\qquad i \quad \text{path}$

C_{out} is the load and C_{in} is the first input capacitance on the path

path effort $F = GH$

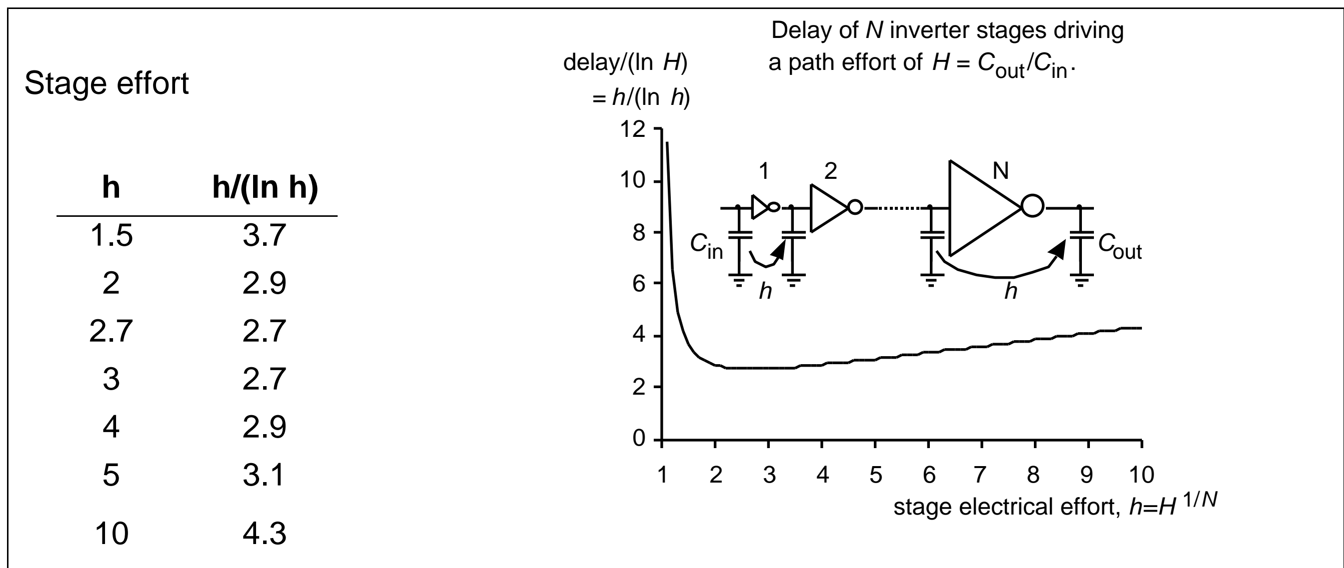
optimum effort delay $\hat{f}_i = g_i h_i \quad = F^{1/N}$

optimum path delay $\hat{D} = NF^{1/N} \quad = N(GH)^{1/N} + P + Q$

$P + Q = \quad p_i + h_i$

$\qquad\qquad i \quad \text{path}$

3.3.6 Optimum Number of Stages


Stage effort

h	h/(ln h)
1.5	3.7
2	2.9
2.7	2.7
3	2.7
4	2.9
5	3.1
10	4.3

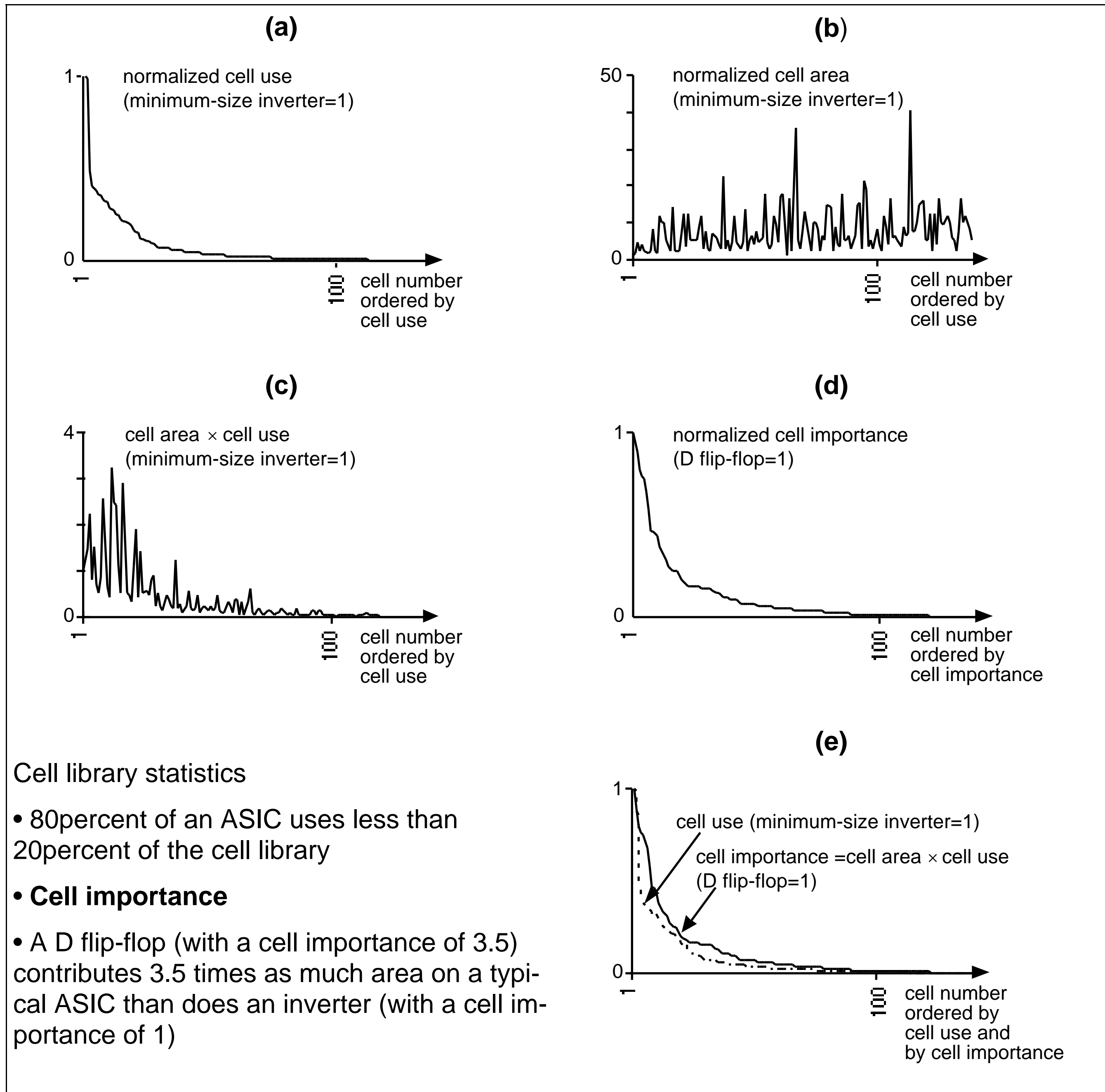
- Chain of N inverters each with equal stage effort, $f=gh$
- Total path delay is $Nf=Ngh=Nh$, since $g=1$ for an inverter

- To drive a path electrical effort H, $h^N = H$, or $N \ln h = \ln H$
- Delay, $Nh = h \ln H / \ln h$
- Since $\ln H$ is fixed, we can only vary $h/\ln(h)$
- $h/\ln(h)$ is a shallow function with a minimum at $h = e$ 2.718
- Total delay is $Ne = e \ln H$

3.4 Library-Cell Design

- A big problem in library design is dealing with design rules
- Sometimes we can **waive** design rules
- **Symbolic layout**, **sticks** or **logs** can decrease the library design time (9 months for Virtual Silicon–currently the most sophisticated standard-cell library)
- Mapping symbolic layout uses 10–20 percent more area (5–10 percent with compaction)
- Allowing 45° layout decreases silicon area (some companies do not allow 45° layout)

3.5 Library Architecture

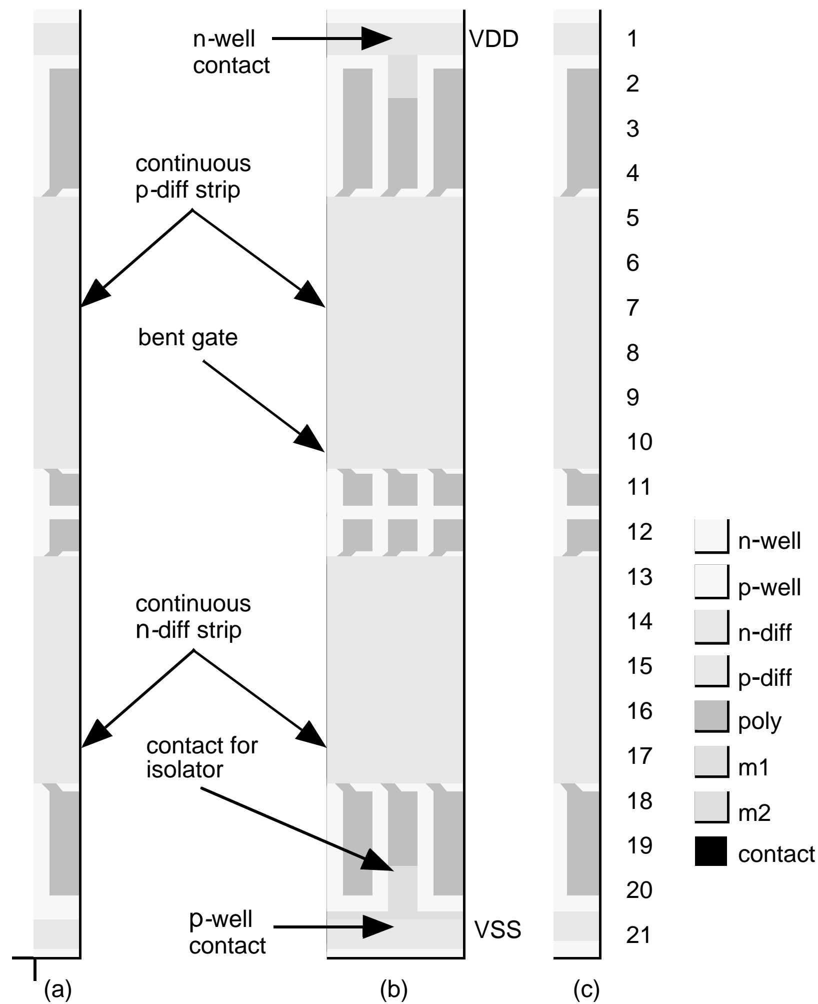


Cell library statistics

- 80percent of an ASIC uses less than 20percent of the cell library
- **Cell importance**
- A D flip-flop (with a cell importance of 3.5) contributes 3.5 times as much area on a typical ASIC than does an inverter (with a cell importance of 1)

3.6 Gate-Array Design

Key words: gate-array base cell (or base cell) • gate-array base (or base) • horizontal tracks • vertical track • gate isolation • isolator transistor • oxide isolation • oxide-isolated gate array

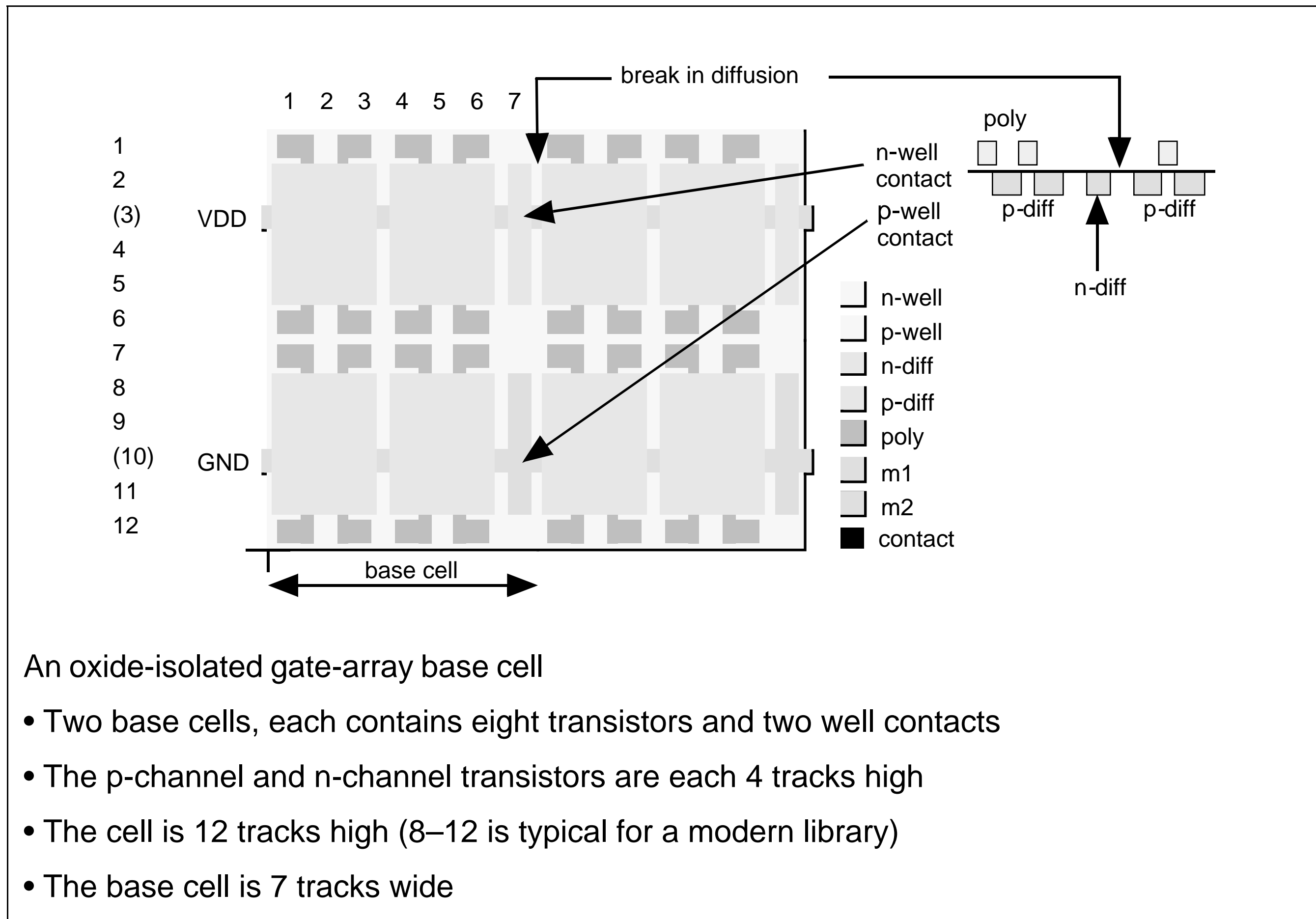


The construction of a gate-isolated gate array

(a) The one-track-wide base cell containing one p-channel and one n-channel transistor

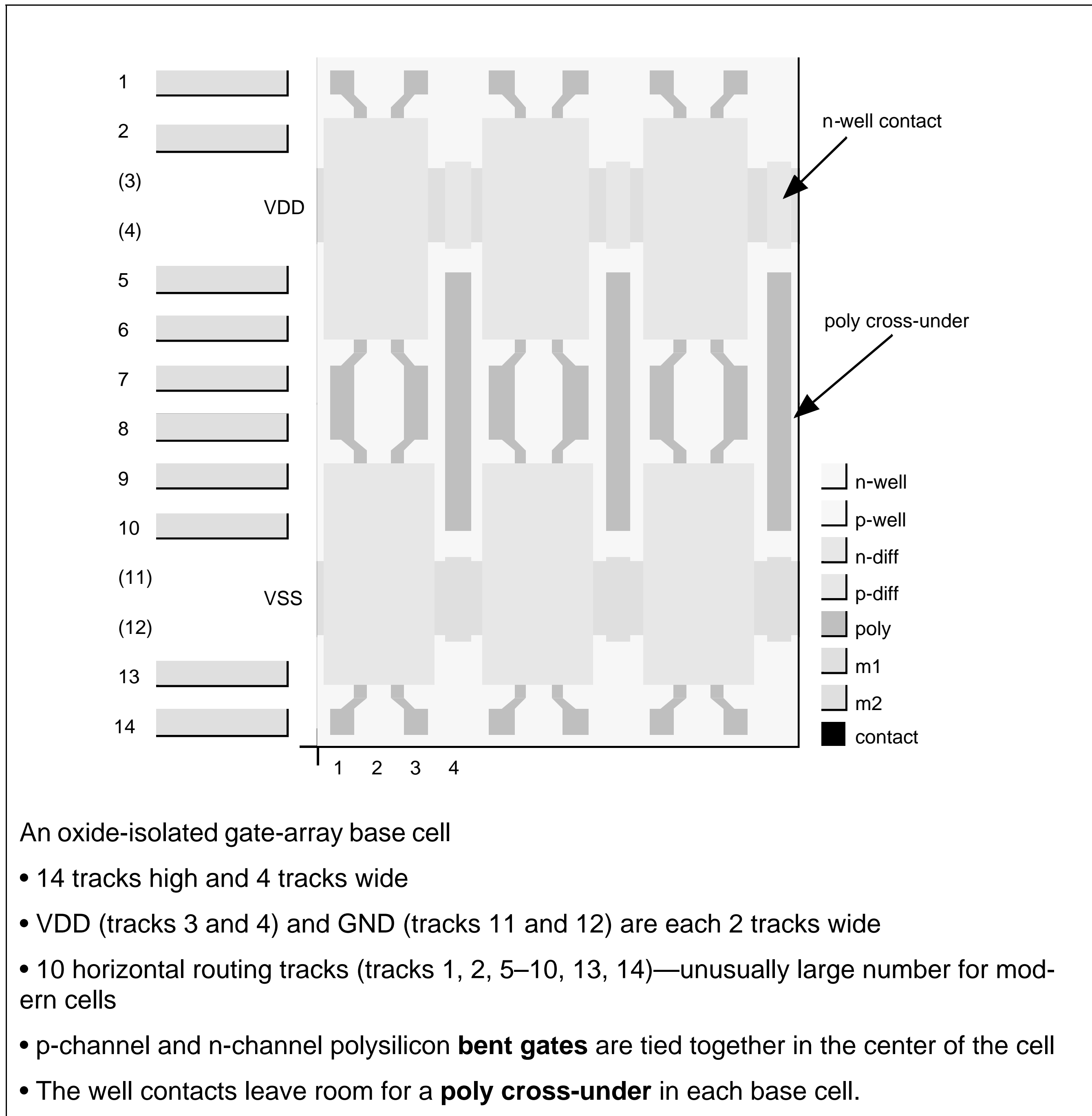
(b) The center base cell is isolating the base cells on either side from each other

(c) The base cell is 21 tracks high (high for a modern cell library)



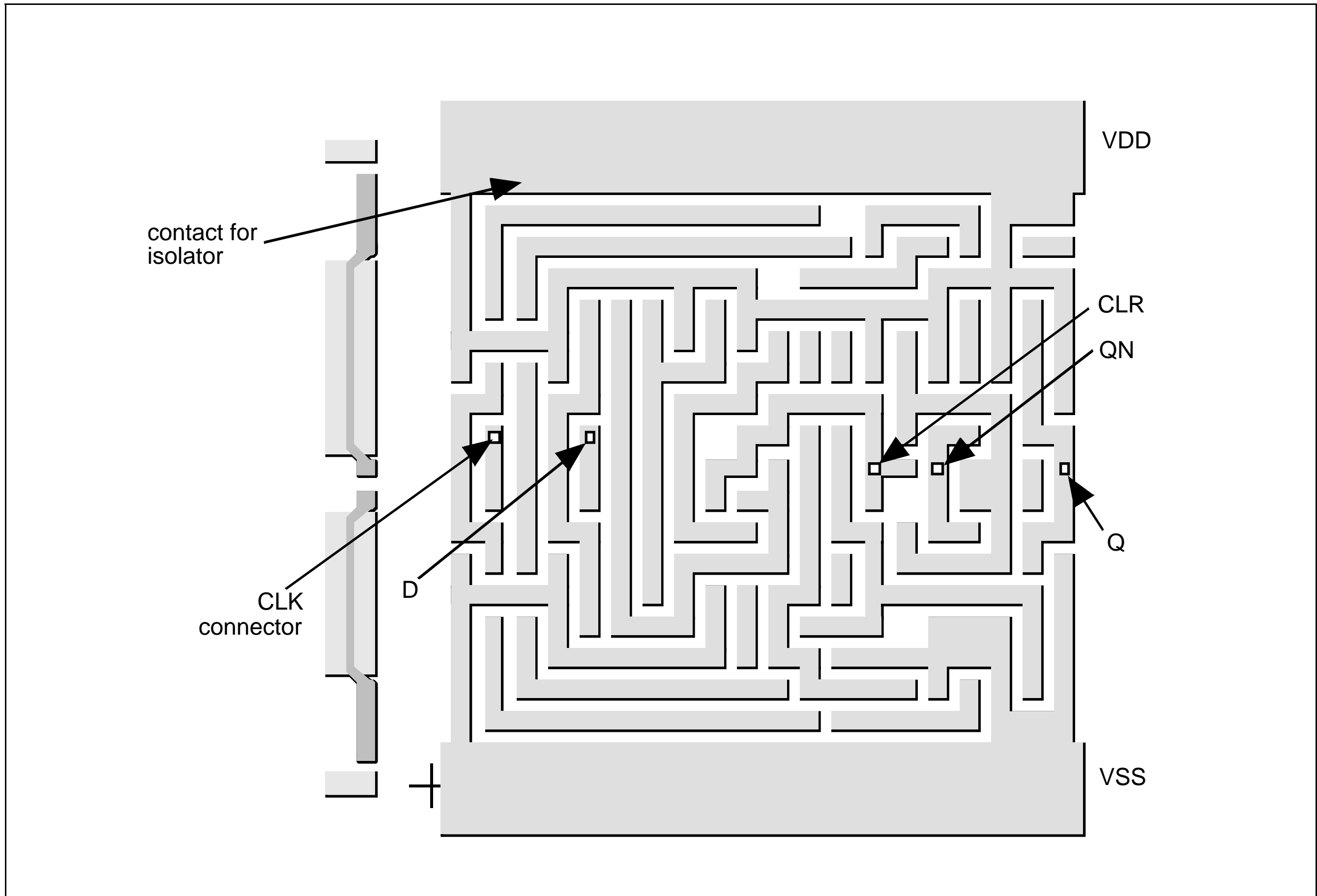
An oxide-isolated gate-array base cell

- Two base cells, each contains eight transistors and two well contacts
- The p-channel and n-channel transistors are each 4 tracks high
- The cell is 12 tracks high (8–12 is typical for a modern library)
- The base cell is 7 tracks wide



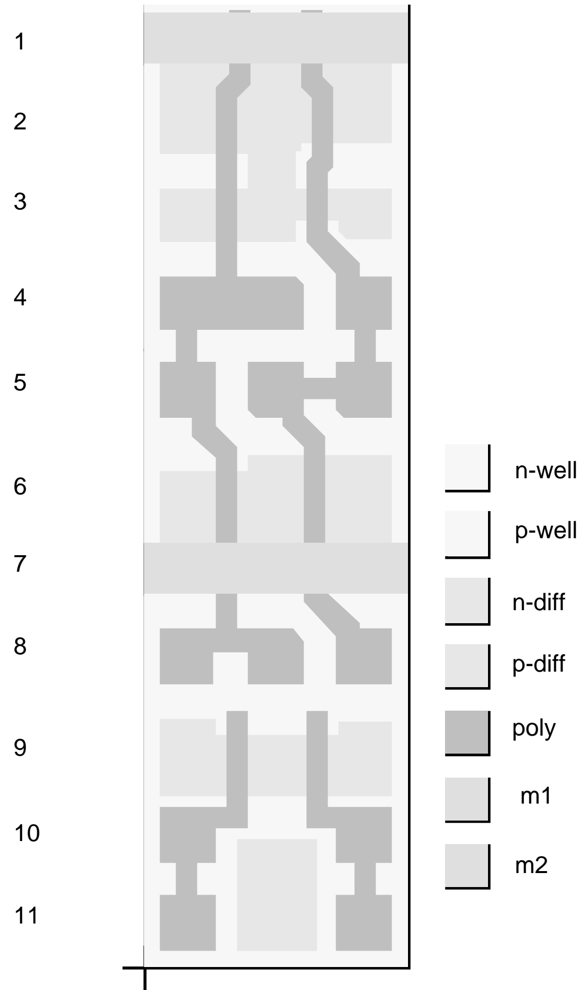
An oxide-isolated gate-array base cell

- 14 tracks high and 4 tracks wide
- VDD (tracks 3 and 4) and GND (tracks 11 and 12) are each 2 tracks wide
- 10 horizontal routing tracks (tracks 1, 2, 5–10, 13, 14)—unusually large number for modern cells
- p-channel and n-channel polysilicon **bent gates** are tied together in the center of the cell
- The well contacts leave room for a **poly cross-under** in each base cell.



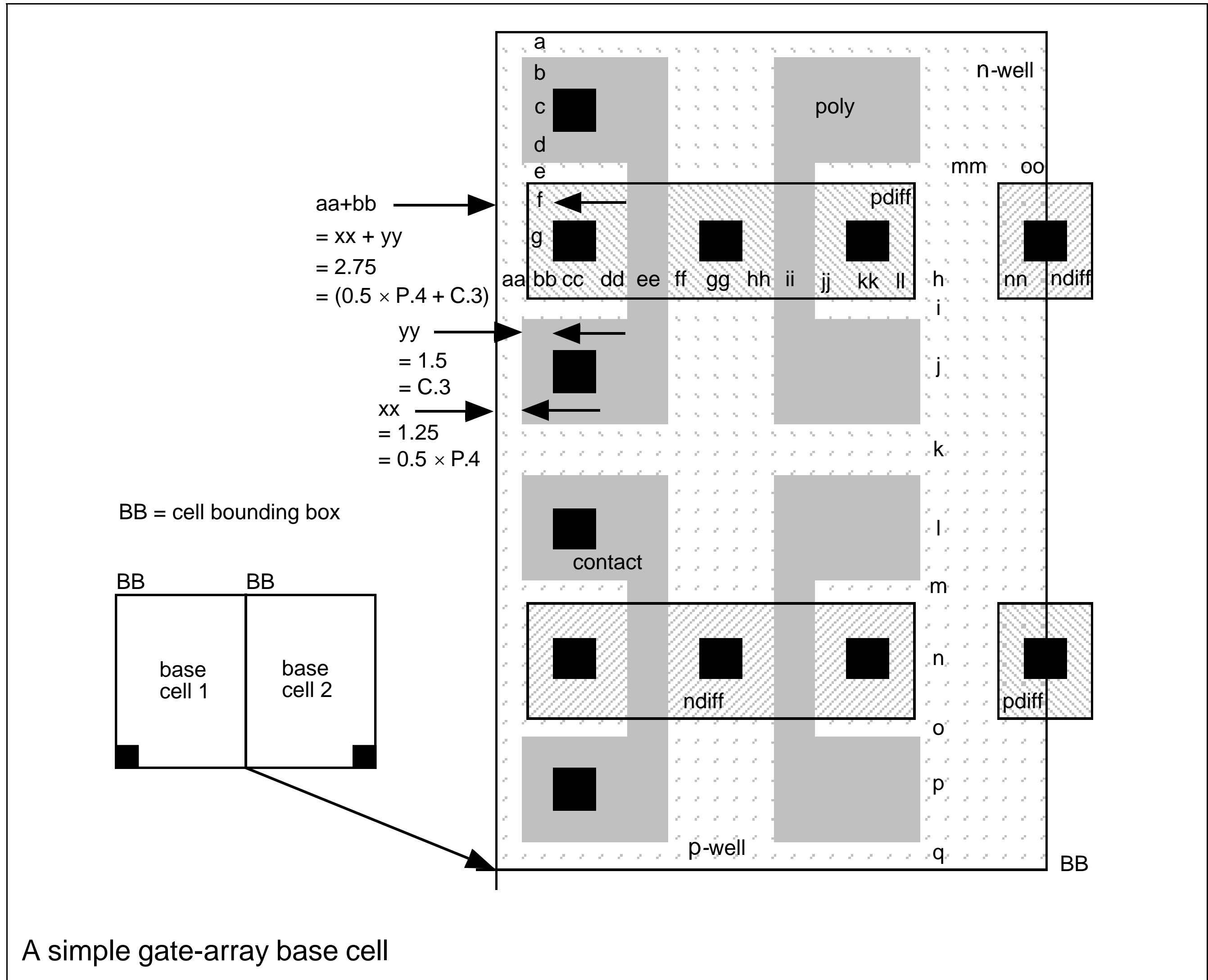
Flip-flop macro in a gate-isolated gate-array library

- Only the first-level metallization and contact pattern, the **personalization**, is shown, but this is enough information to derive the schematic
- This is an older topology for 2LM (cells for 3LM are shorter in height)



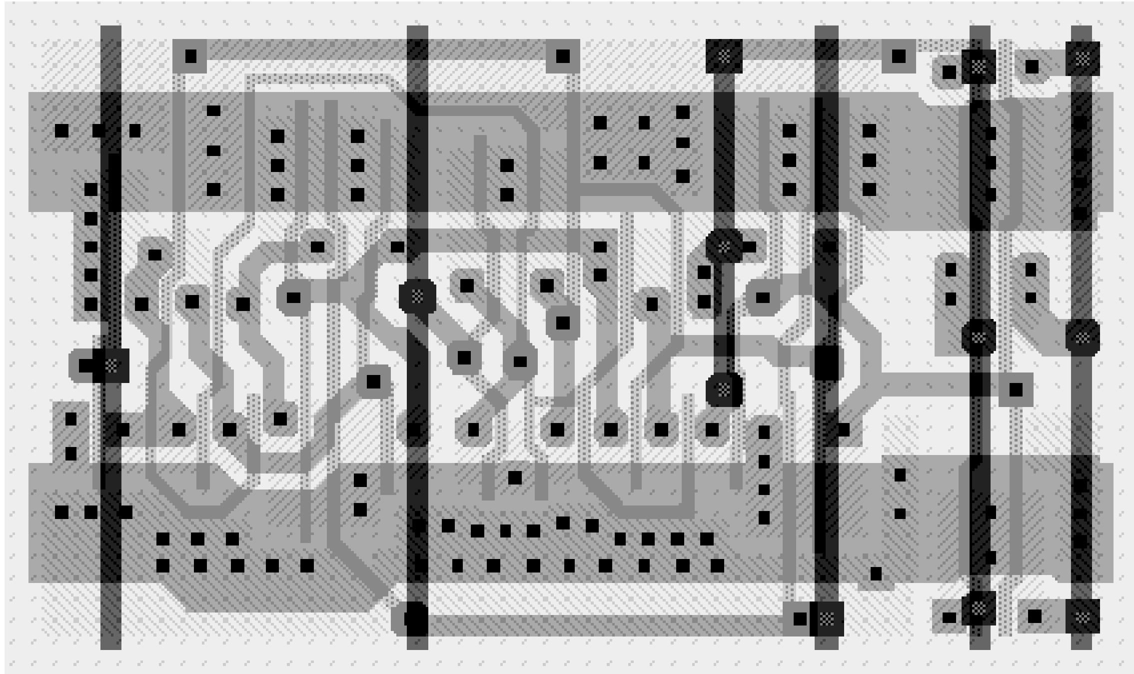
The SiARC/Synopsys cell-based array (CBA) basic cell

- This is CBA I for 2LM (CBA II is intended for 3LM and salicide proceses)



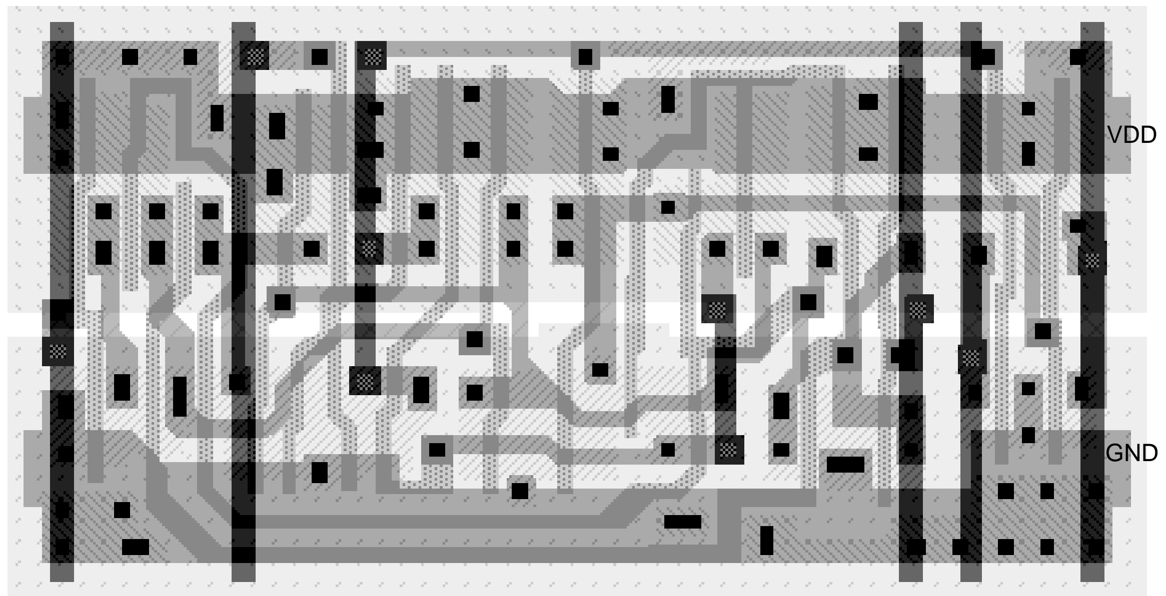
A simple gate-array base cell

3.7 Standard-Cell Design

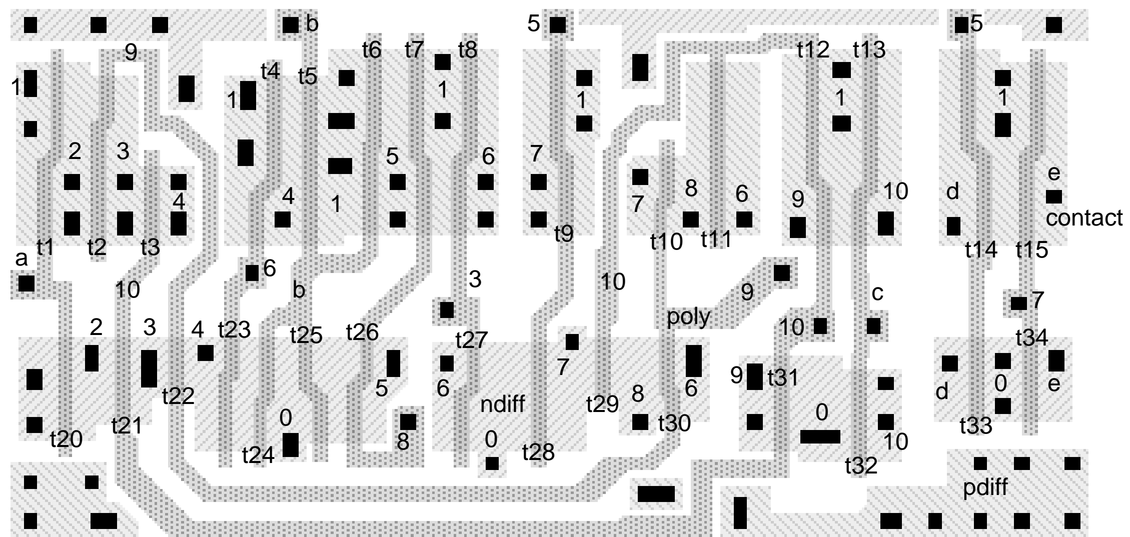


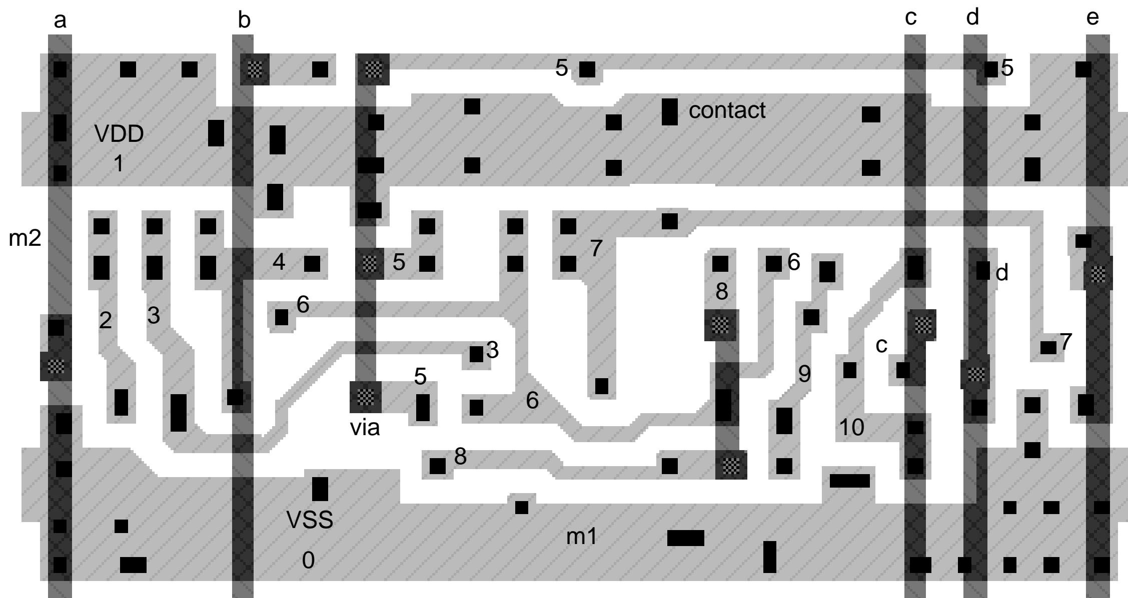
A D flip-flop standard cell

- **Performance-optimized library** • **Area-optimized library**
- Wide **power buses** and transistors for a performance-optimized cell
- **Double-entry cell** intended for a 2LM process and channel routing
- Five **connectors** run vertically through the cell on m2
- The extra short vertical metal line is an internal **crossover**
- **bounding box** (BB) • **abutment box** (AB) • **physical connector** • **abut**



A D flip-flop from a 1.0µm standard-cell library



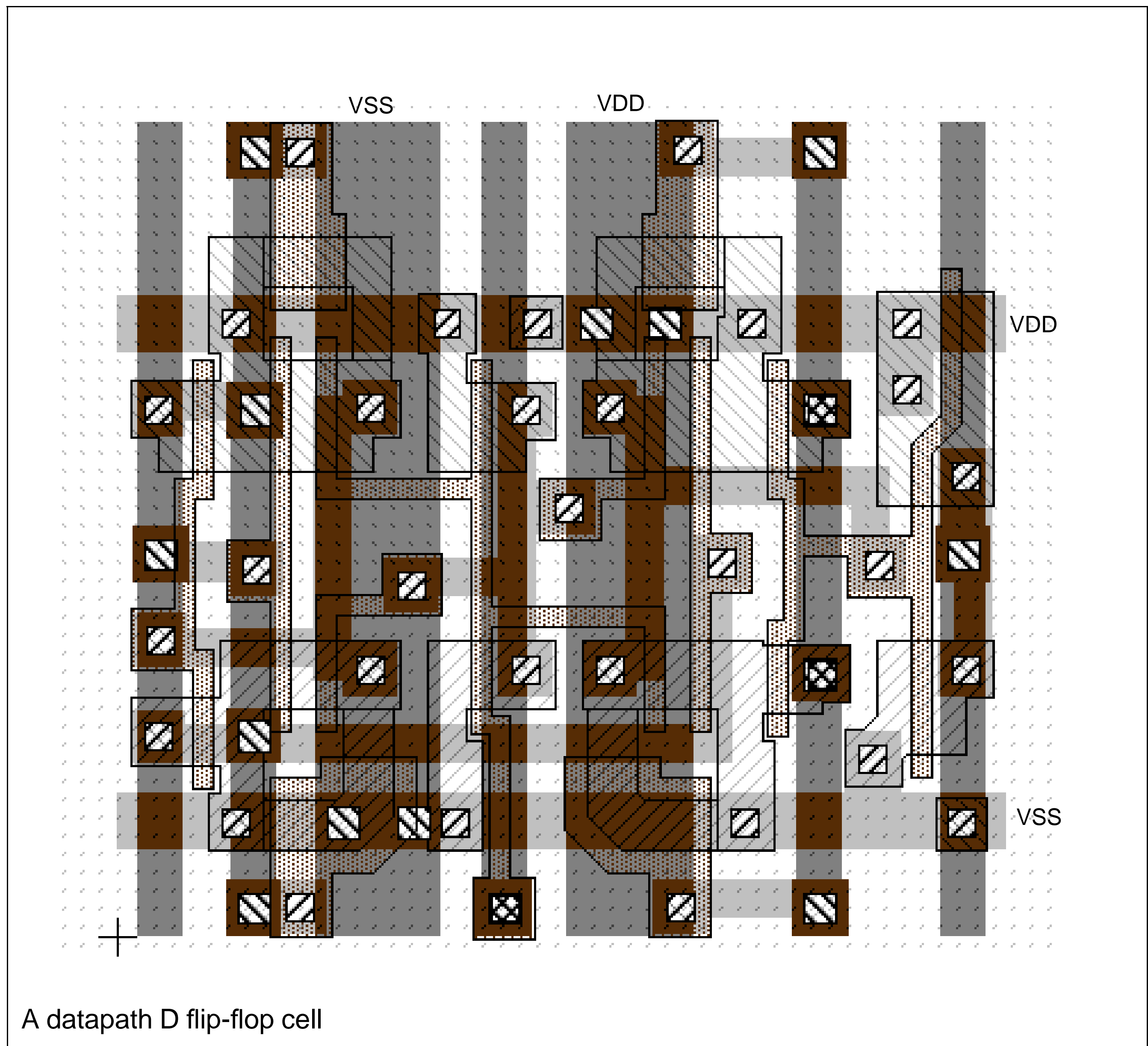


D flip-flop

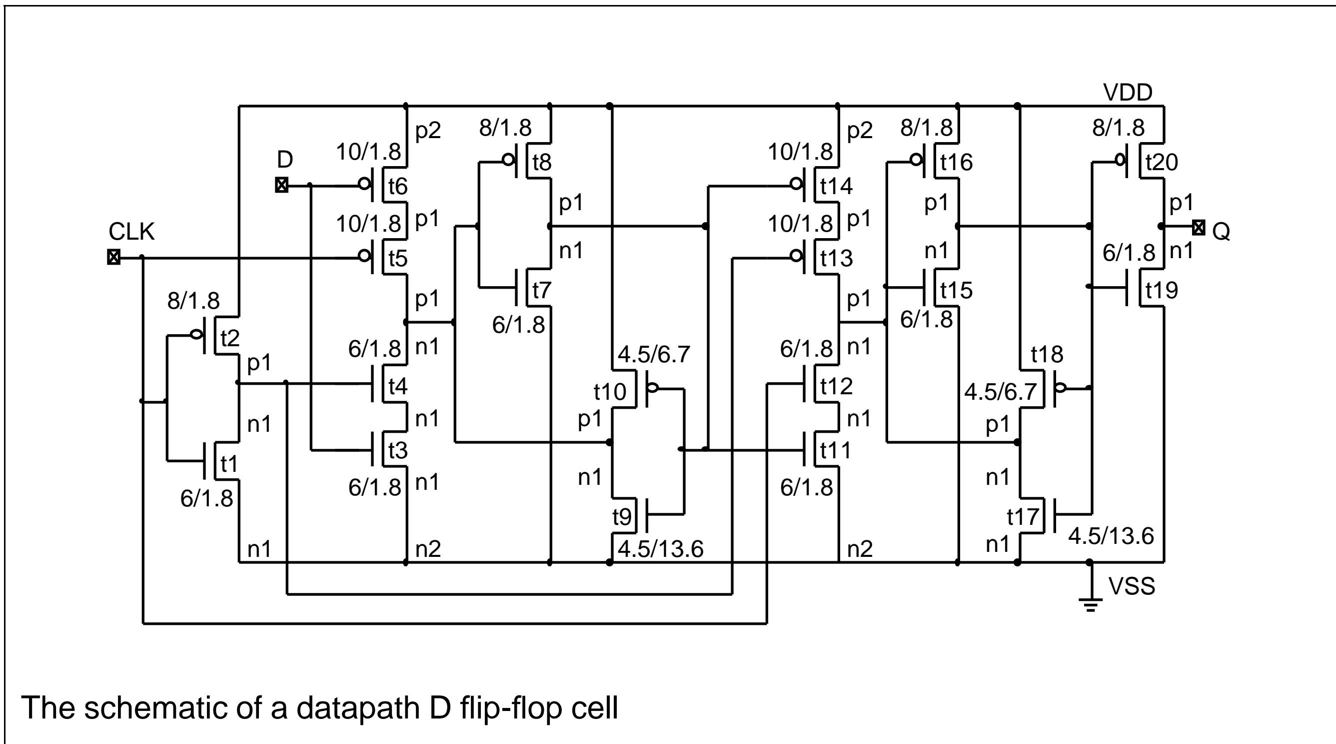
(Top) n-diffusion, p-diffusion, poly, contact (n-well and p-well are not shown)

(Bottom) m1, contact, m2, and via layers

3.8 Datapath-Cell Design



A datapath D flip-flop cell



The schematic of a datapath D flip-flop cell

(a)

(b)

A narrow datapath

(a) Implemented in a two-level metal process

(b) Implemented in a three-level metal process

3.9 Summary

Key concepts:

- Tau, logical effort, and the prediction of delay
- Sizes of cells, and their drive strengths
- Cell importance
- The difference between gate-array macros, standard cells, and datapath cells