

CHIPONTWERP

Faculteit Industriële Ingenieurswetenschappen Elektronica – ICT

Kris Myny (kris.myny@kuleuven.be)

Samenwerkingsverband tussen UHasselt en KU Leuven.
De opleiding industrieel ingenieur is een gezamenlijk
initiatief van UHasselt en KU Leuven.

universiteit
hasselt

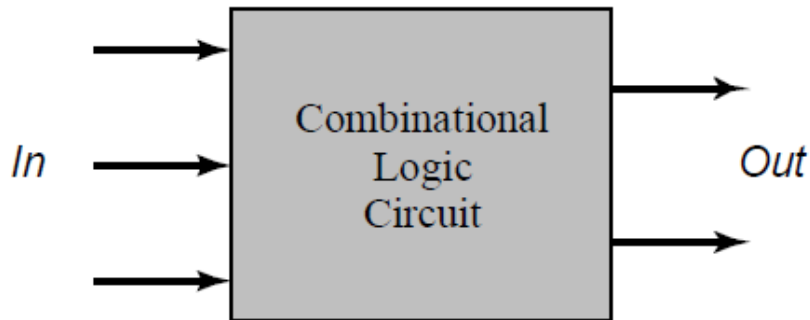
KU LEUVEN

Inhoud

- VLSI inleiding
- IC-ontwerp met standaardcellen
- Complexe cel lay-out met Euler-paden (*zie oefenzitting/labo 1*)
- Statische CMOS schakelingen
- **Combinatorische logica**
- Dynamische CMOS-schakelingen
- Verbindingen op IC
- BiCMOS-schakelingen
- Datapadoperatoren

Combinatorische logica

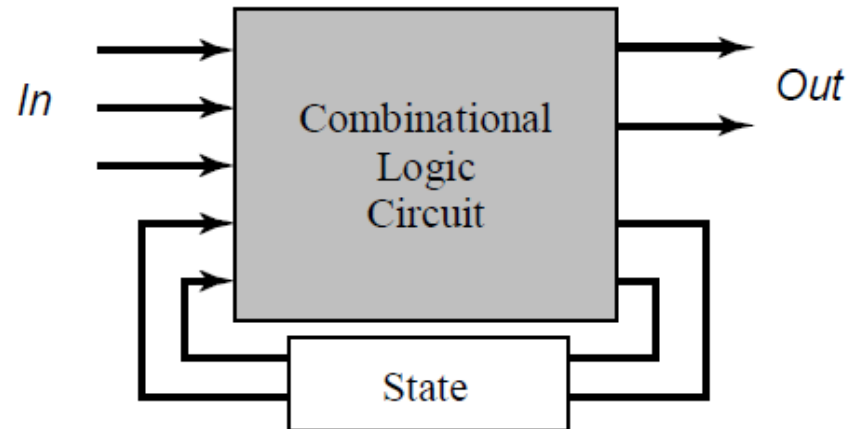
Combinatorische logica



Uitgang = $f(\text{In})$

Bevat enkel logische poorten

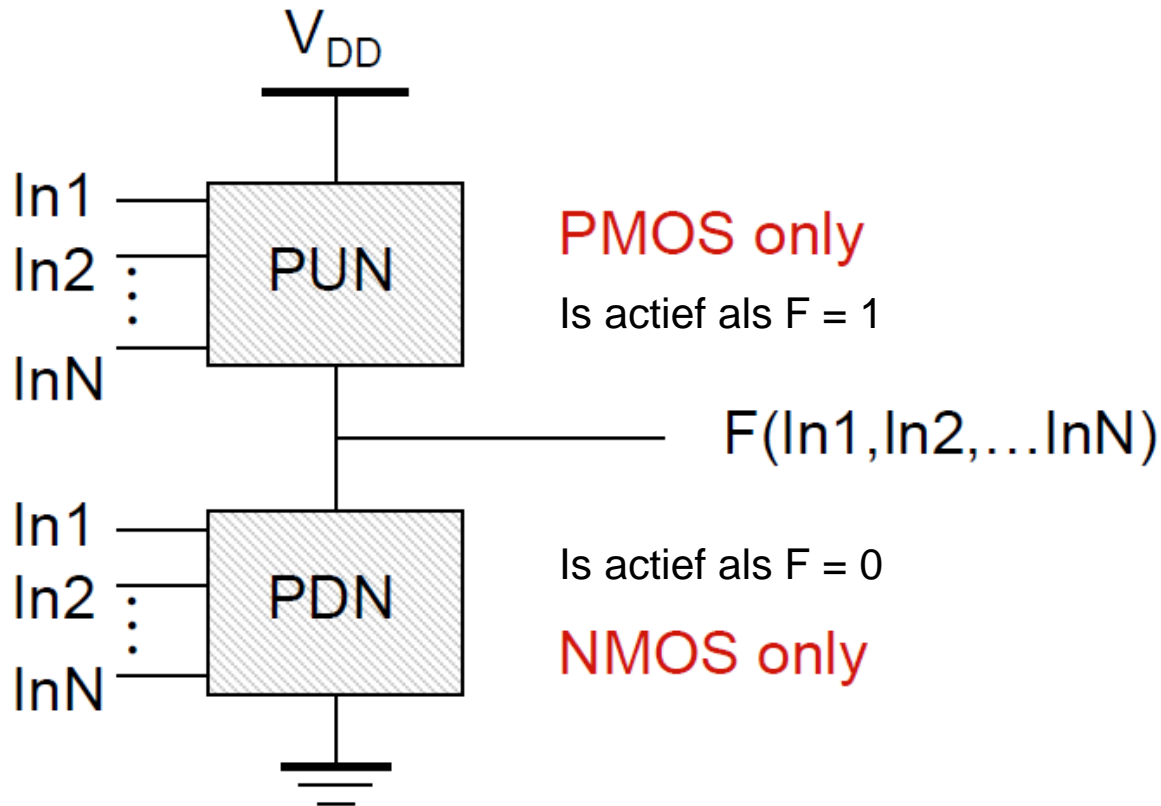
Sequentiële logica



Uitgang = $f(\text{In}, \text{vorige In})$

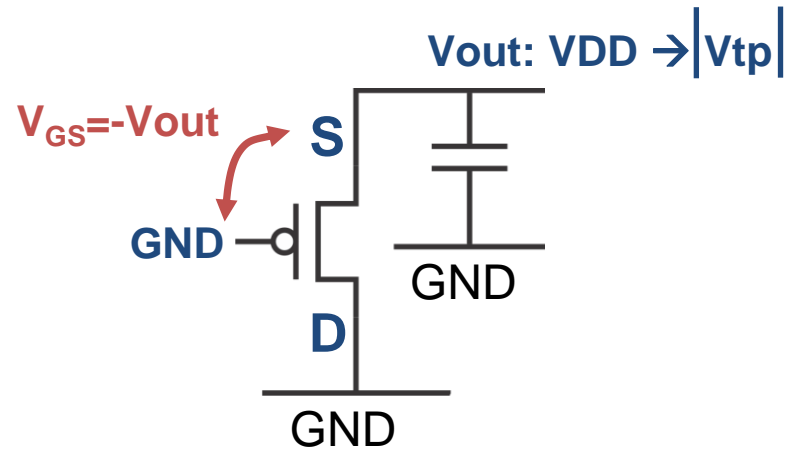
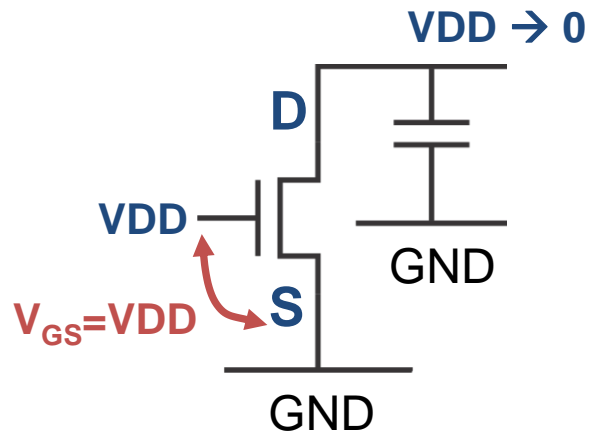
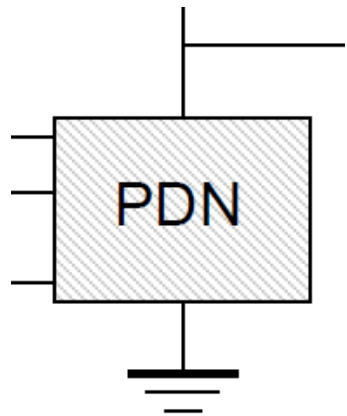
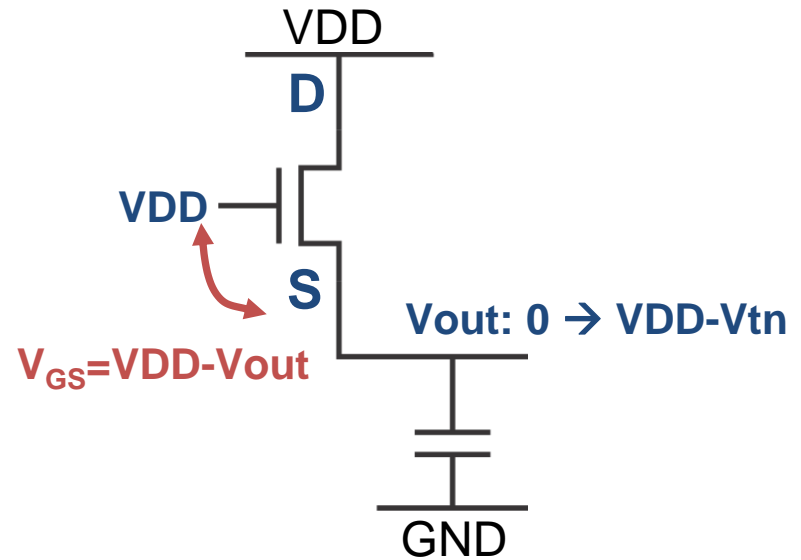
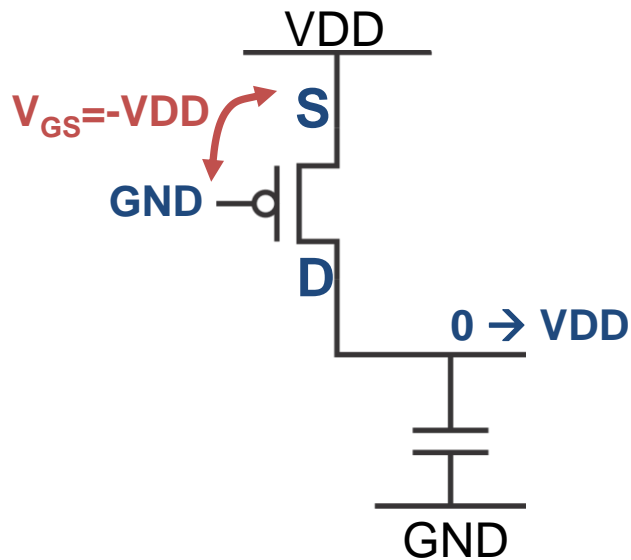
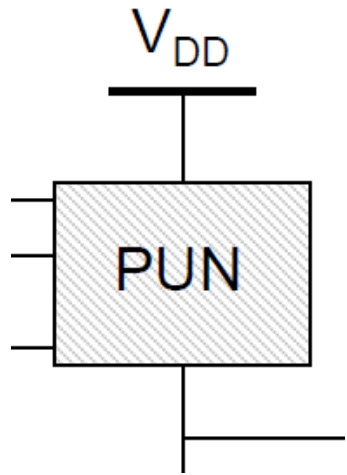
Heeft een geheugen voor de toestand

Statische complementaire CMOS logica



- PUN en PDN zorgen ervoor dat er nooit een pad is tussen GND en VDD!
- PUN en PDN zijn duale netwerken (EN wordt OF en 1 wordt 0)
- De poort is een inverterende poort!
- Het aantal transistoren om een N-input logische poort te maken is $2N$

pMOS of nMOS in PUN/PDN?

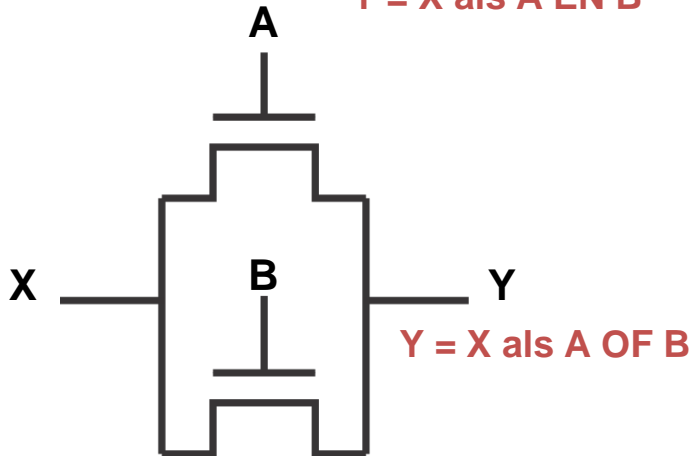
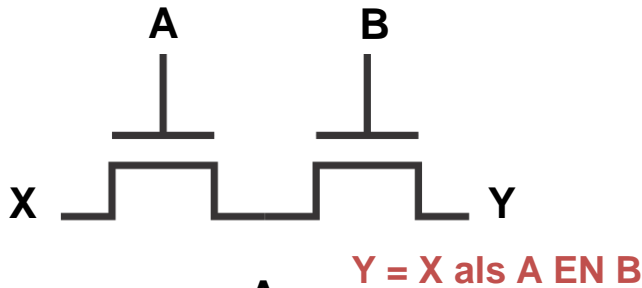


Sterk PU/PDN

Zwak PU/PDN

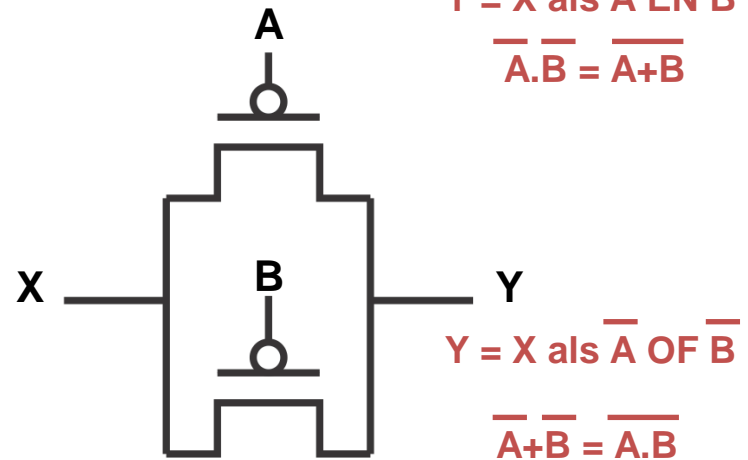
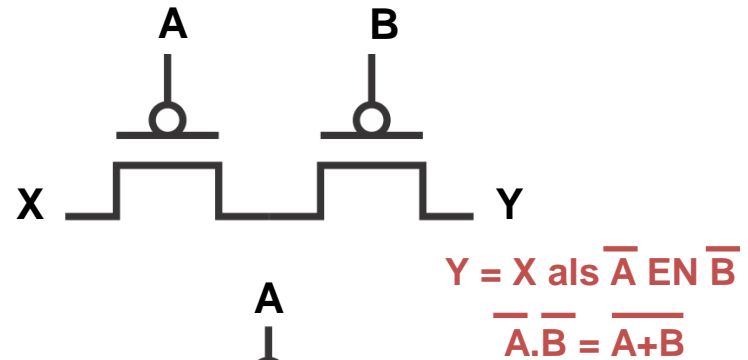
Serie/parallel schakeling van nMOS

nMOS



nMOS schakelaar
geleidt bij hoge V_{GS}

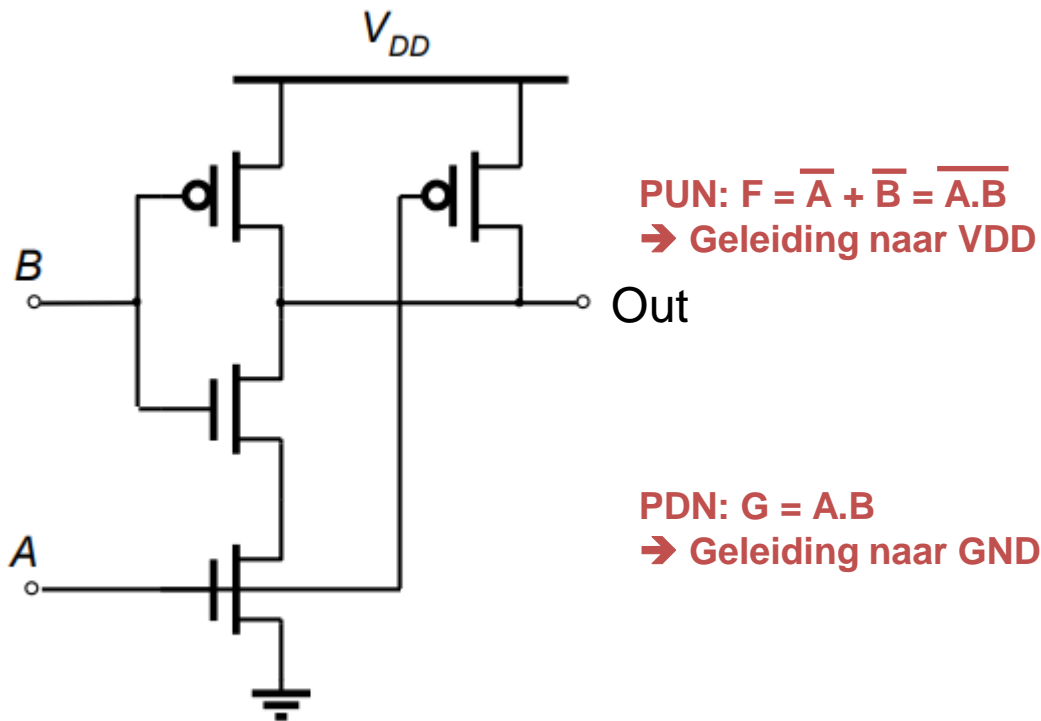
pMOS



pMOS schakelaar
geleidt bij lage V_{GS}

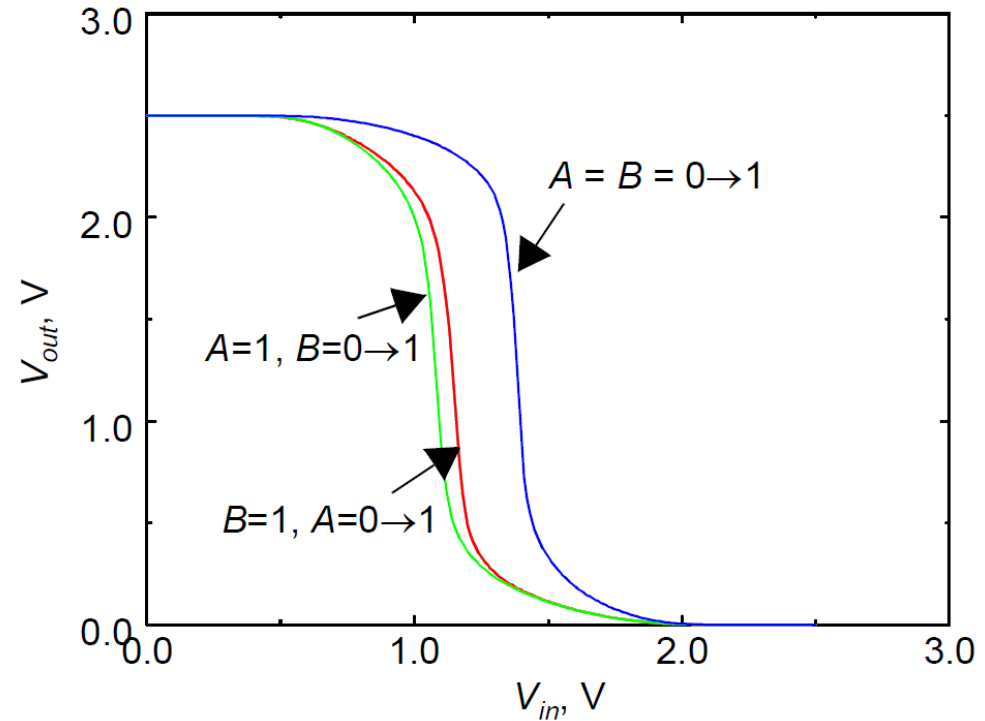
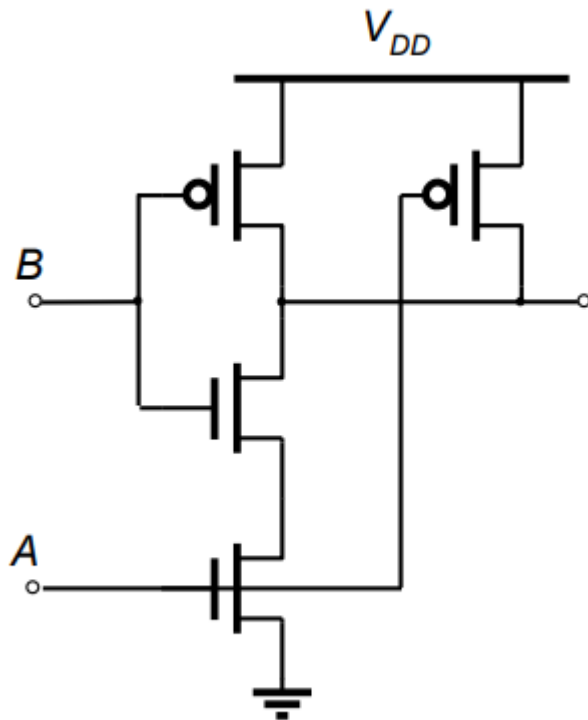
NAND

- $\text{Out} = \overline{A.B}$



A	B	Out
0	0	1
0	1	1
1	0	1
1	1	0

VTC van de NAND poort

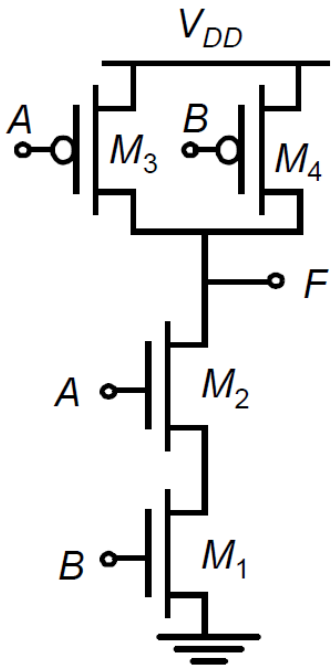


- Sterkte van het PUN hangt af van data aan ingangen A en B
- PDN transistoren hebben een andere V_T door het bulk effect

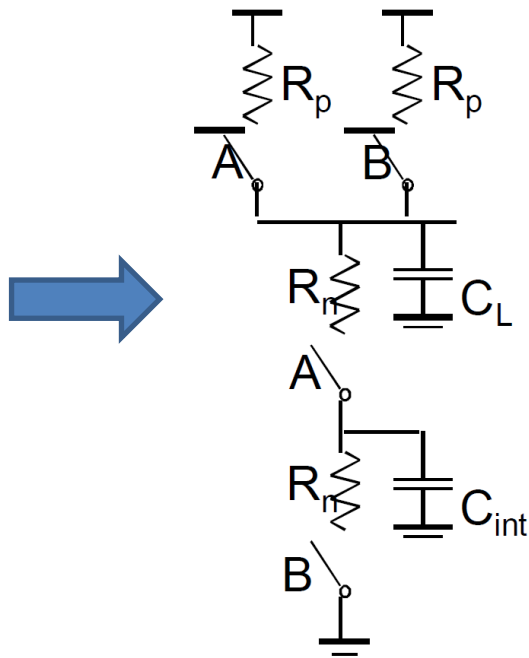
Switch delay model

- Elke transistor vervangen we door een schakelaar in serie met de equivalente weerstand

NAND2



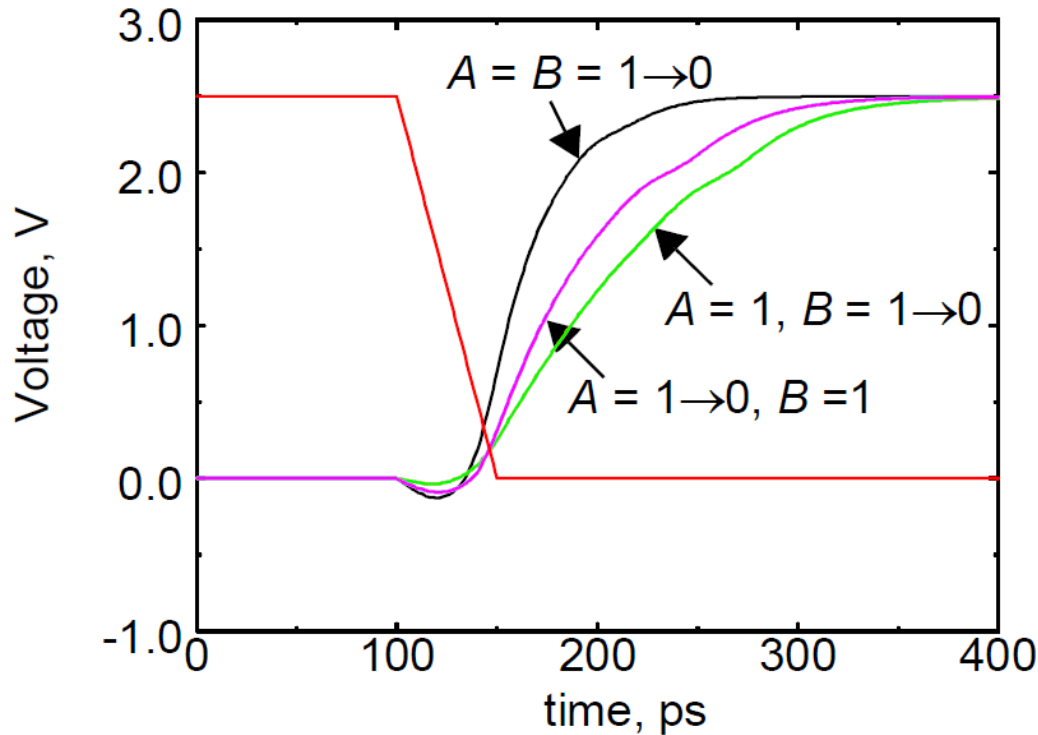
NAND2 - model



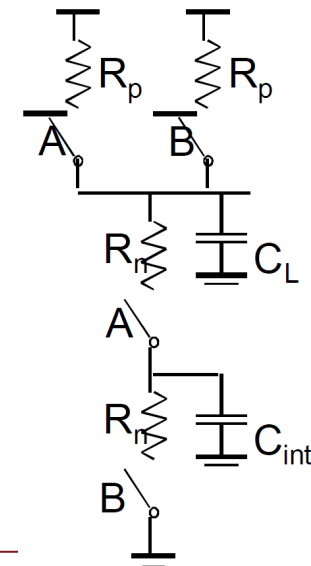
- Vertraging van de poort is afhankelijk van de ingang
- Laag-naar-hoog
 - $A = B = 0$
 - $t_p = 0.69 C_L R_p / 2$
 - $A = 0; B = 1$
 - $t_p = 0.69 C_L R_p$
- Hoog-naar-lag
 - $A = B = 1$
 - $t_p = 0.69 C_L 2R_p$

Vertraging van een NAND-poort

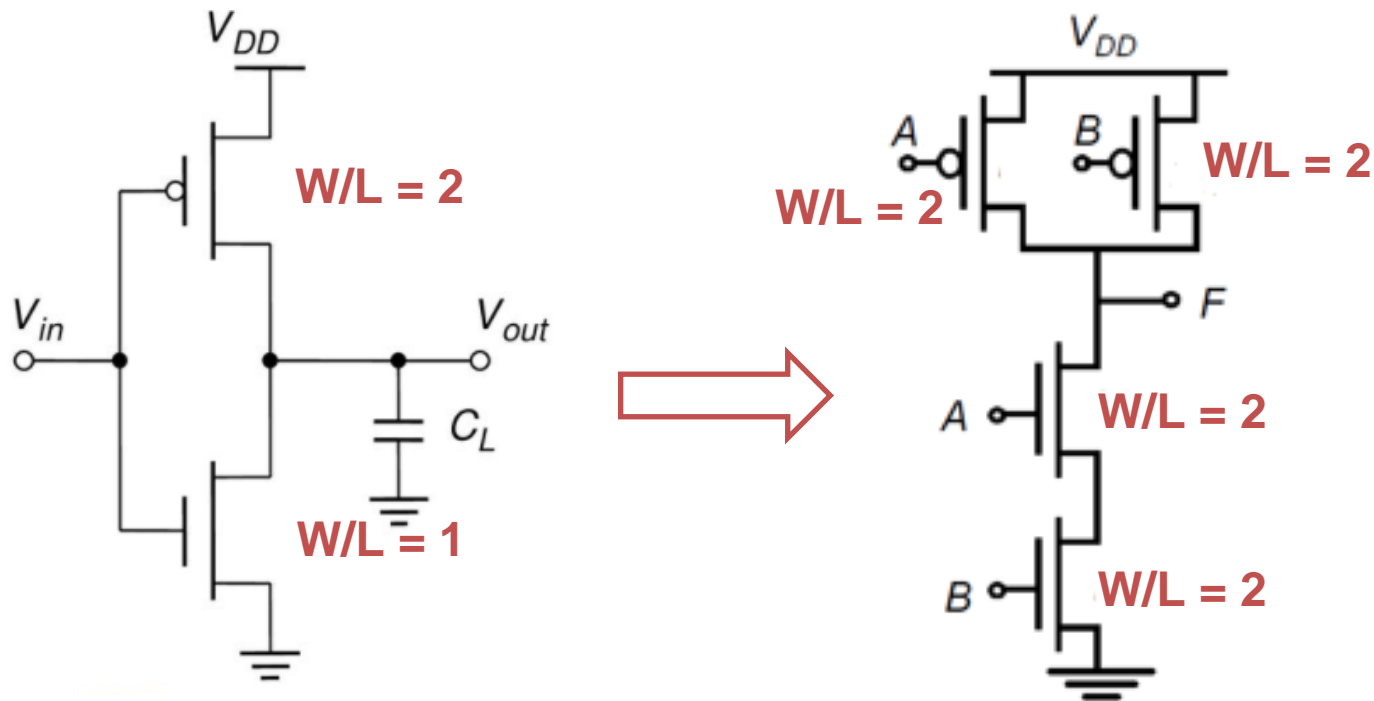
nMOS: 0.5/0.25 [$\mu\text{m}/\mu\text{m}$]
pMOS: 0.75/0.25 [$\mu\text{m}/\mu\text{m}$]
 $C_L = 100\text{fF}$



Ingangsdata	Vertraging [ps]
$A = B = 0 \rightarrow 1$	69
$A = 1, B = 0 \rightarrow 1$	62
$A = 0 \rightarrow 1, B = 1$	50
$A = B = 1 \rightarrow 0$	35
$A = 1, B = 1 \rightarrow 0$	76
$A = 1 \rightarrow 0; B = 1$	57



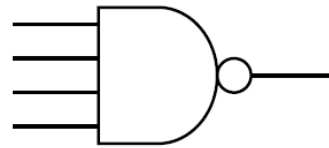
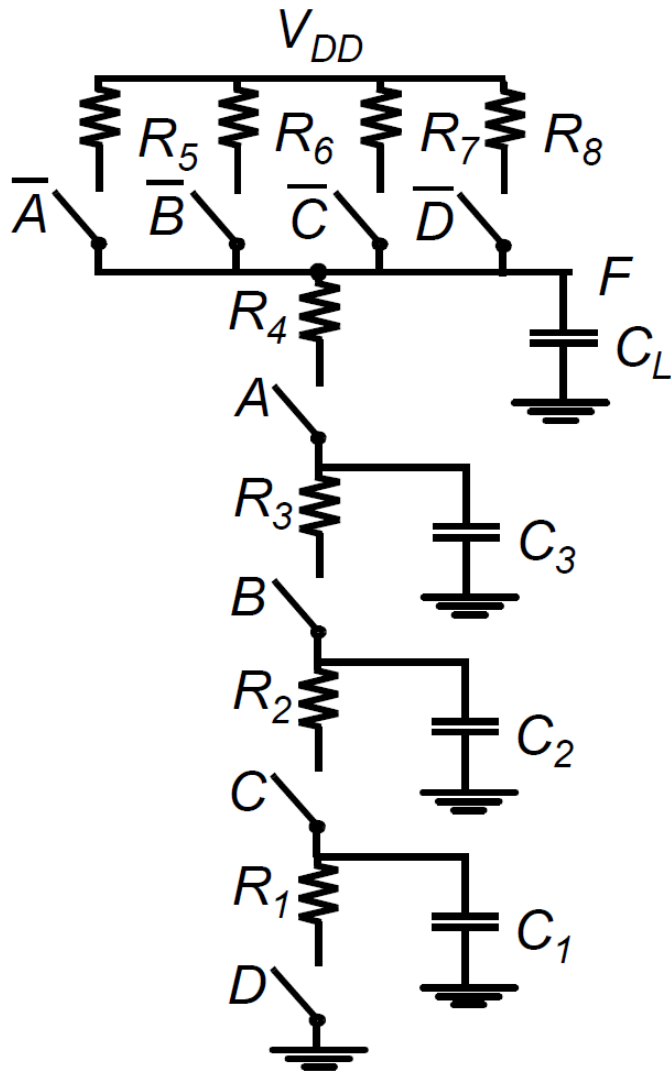
Transistor sizing



- nMOS is nu verdubbeld tov de invertor om
 - Dezelfde worst-case weerstand te hebben (R_{on})
 - Dezelfde worst-case vertraging
- Vertraging is nog steeds afhankelijk van de data aan de ingang

Sizing for “equal current drive” ($C_{internal}$ en fan-in worden genegeerd)

Vertraging van 4-input NAND



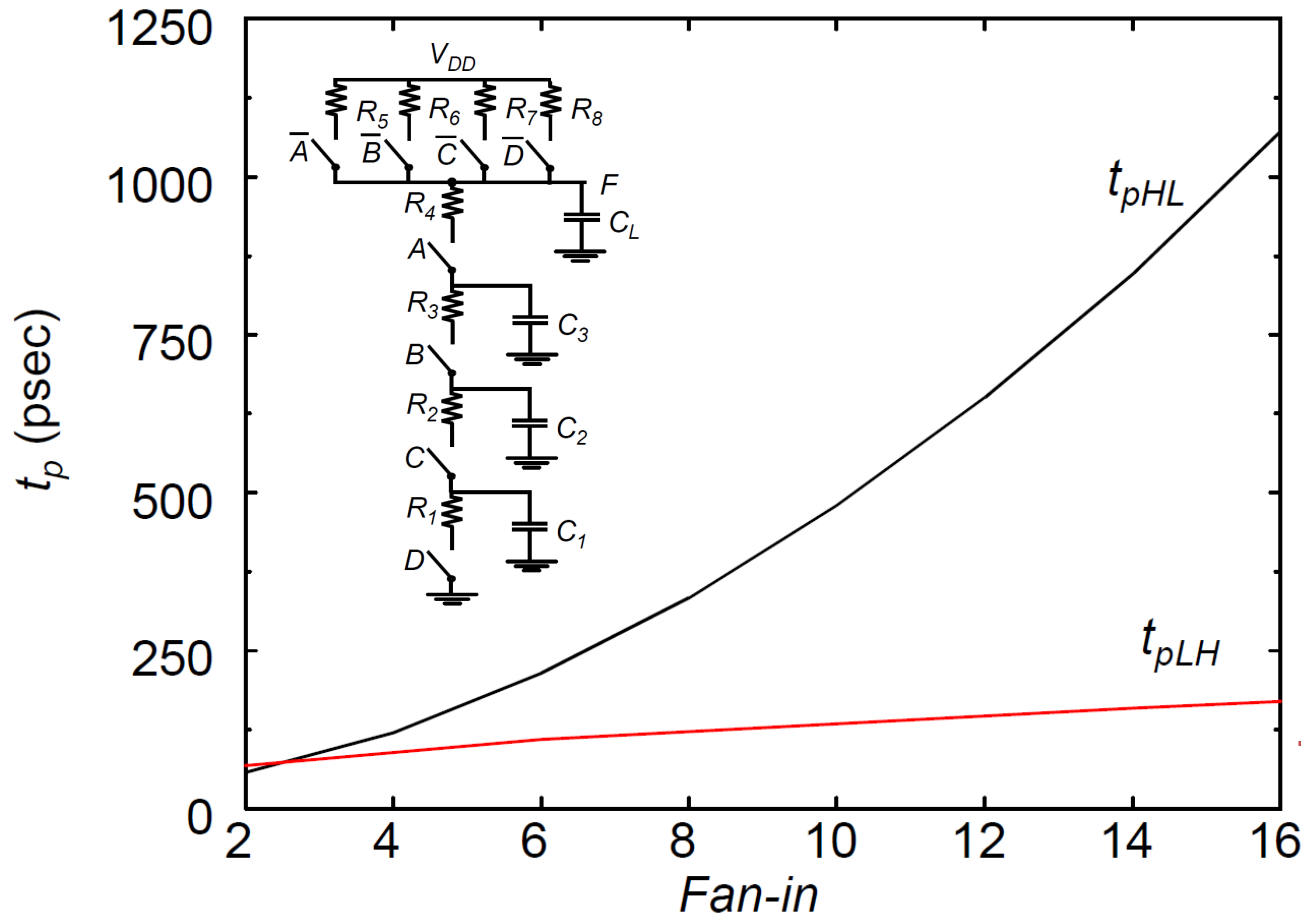
Distributed RC model (Elmore delay)

$$t_{pHL} = 0,69 \left(\begin{array}{c} R_1 C_1 \\ + (R_1 + R_2) C_2 \\ + (R_1 + R_2 + R_3) C_3 \\ + (R_1 + R_2 + R_3 + R_4) C_L \end{array} \right)$$

$$t_{pHL} = 0,69 R_n (C_1 + 2C_2 + 3C_3 + 4C_L)$$

- Weerstand R_1 van transistor M1 komt voor in elke term
→ Heel belangrijk voor delay minimalisatie
- Capaciteit kan bepaald worden door de layout
- **Delay is afhankelijk van de fan-in !!!**

Vertraging versus aantal ingangen NAND



N-input NAND
→ nMOS R vergroot linear
→ C_{int} vergroot linear
 t_{pHL} → **kwadratisch**

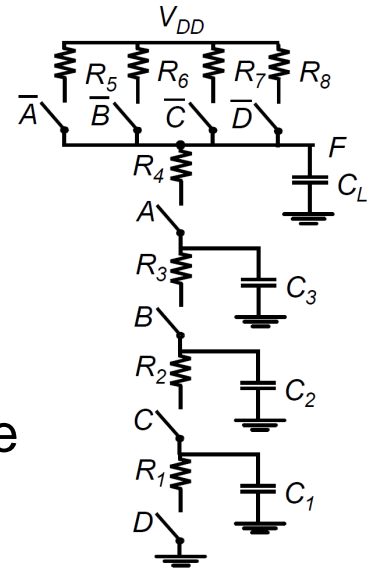
N-input NAND
→ #N transistoren in parallel
→ C_{int} vergroot linear
 t_{pLH} → **linear**

Het aantal ingangen > 4 wordt meestal vermeden!

Ontwerptechnieken voor snellere poorten

Techniek 1 – transistoren groter maken

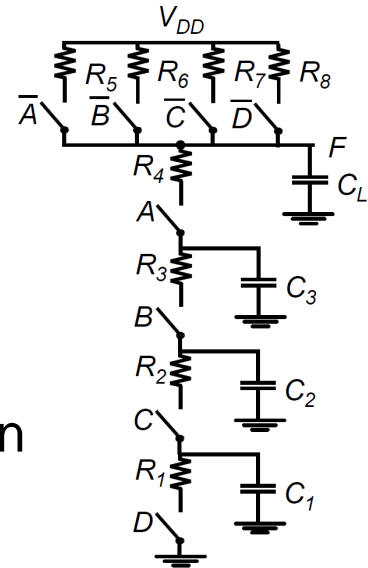
- Transistoren in serie groter maken (voor NAND)
- Serieweerstand wordt verlaagd → betere RC
- Let op:
 - Parasitaire cap vergroot → impact op RC, maar ook een grotere load aan de ingang voor de vorige trap
- Als de intrinsieke cap dominant is, “self-loading” en geen impact op vertraging



Ontwerptechnieken voor snellere poorten

Techniek 2 – transistoren progressief groter maken

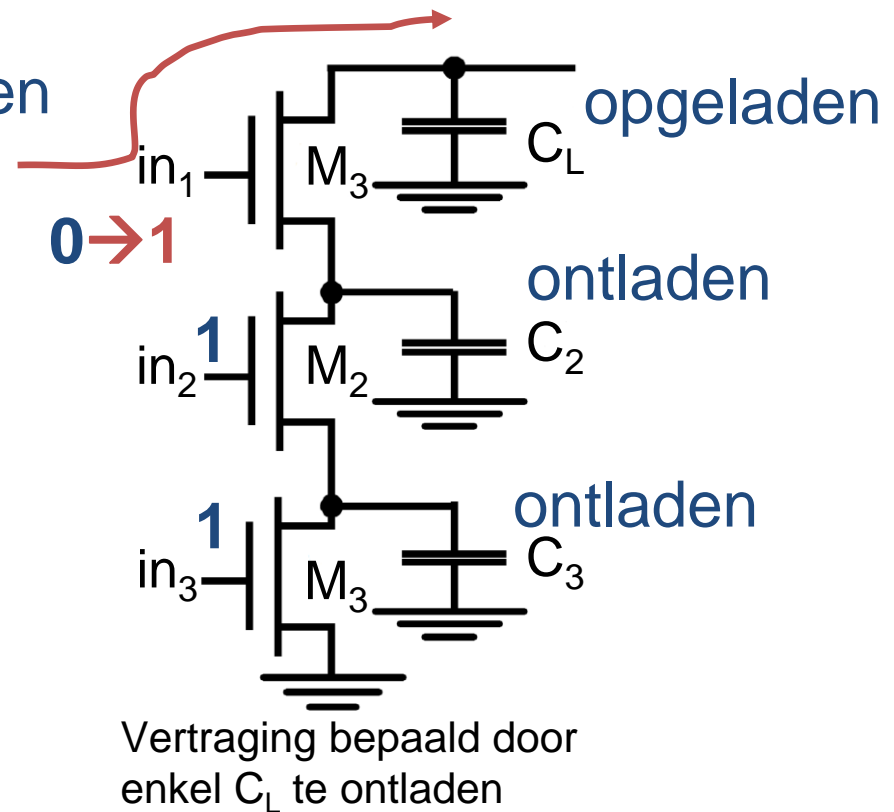
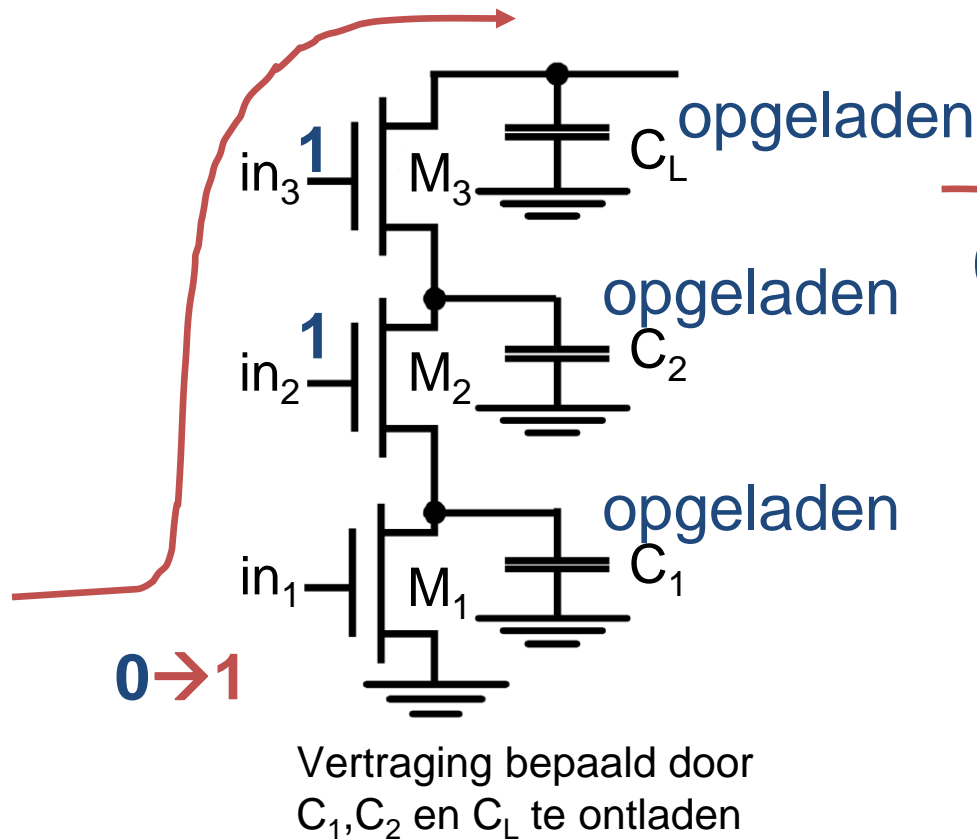
- Transistoren in serie groter maken (voor NAND)
- $R_1 < R_2 < R_3 < R_4$
→ of transistor 1 > transistor 2 > ...
- Nuttig zolang de fan-out capaciteit domineert:
 - Als de intrinsieke cap dominant is, “self-loading” en geen impact op vertraging
- Moeilijk voor de layout



Ontwerptechnieken voor snellere poorten

Techniek 3 – transistoren anders ordenen

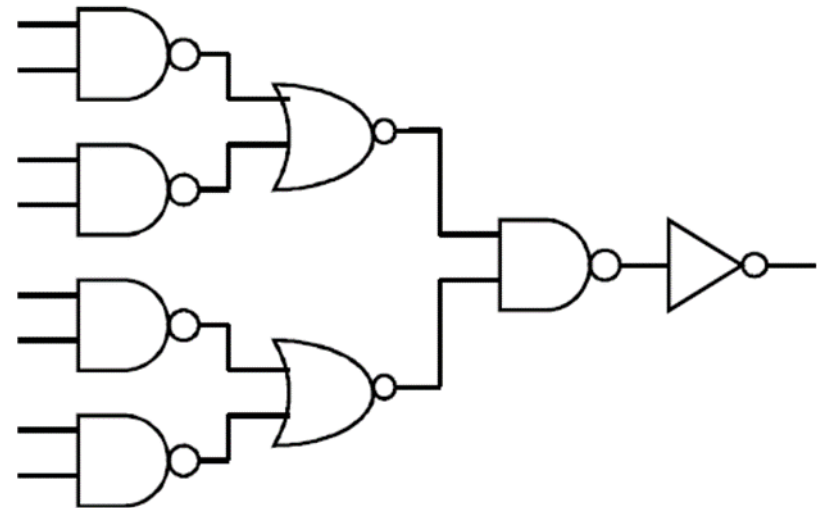
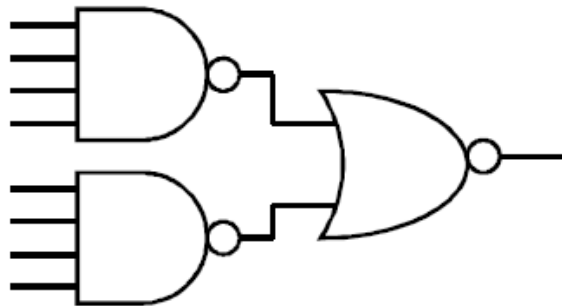
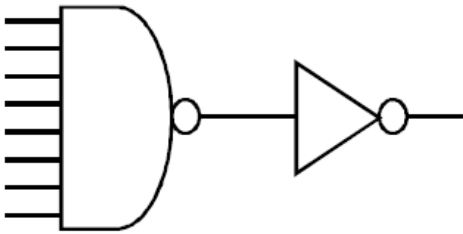
- Het meest kritische ingangssignaal zo dicht mogelijk bij de uitgang plaatsen



Ontwerptechnieken voor snellere poorten

Techniek 4 – herstructureren van de logica

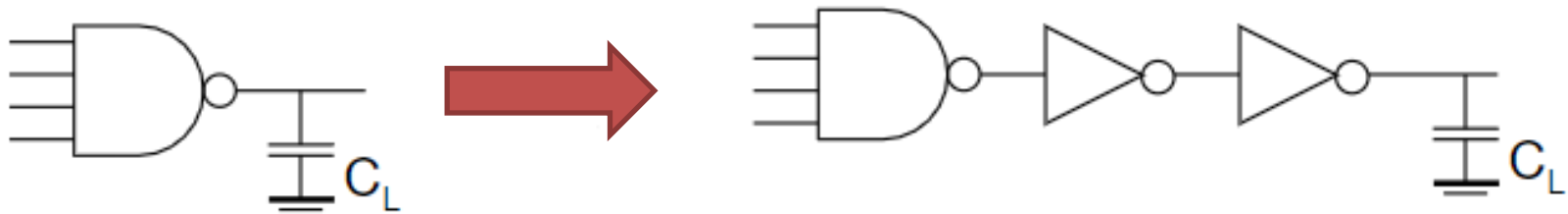
- $F = \text{AND8 poort}$
- Kleinste oppervlakte voor de 8-input NAND en 2x4-input NAND
- Snelste poort is de combinatie van de 2-ingangspoorten



Ontwerptechnieken voor snellere poorten

Techniek 5 – Bij grote fan-out extra bufferen

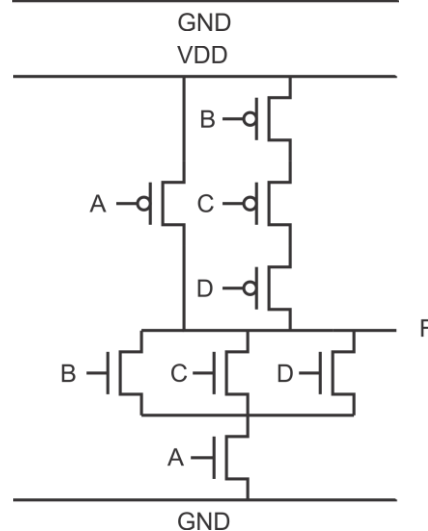
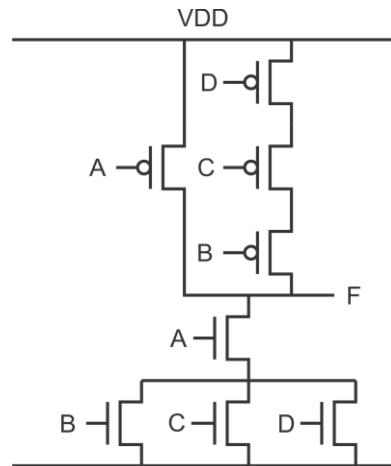
- Focus op fan-in en minimaliseer fan-out



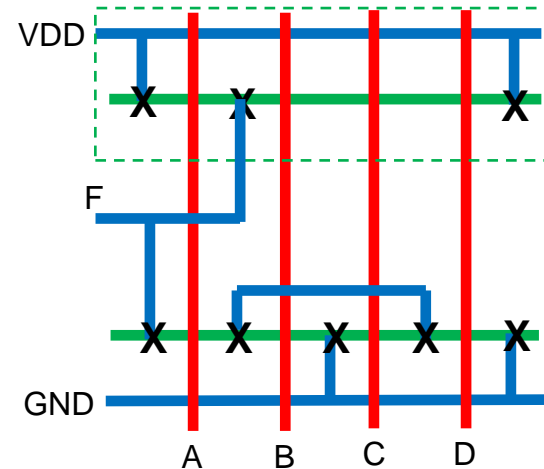
Ontwerptechnieken voor snellere poorten

Techniek 6 – layout is cruciaal!

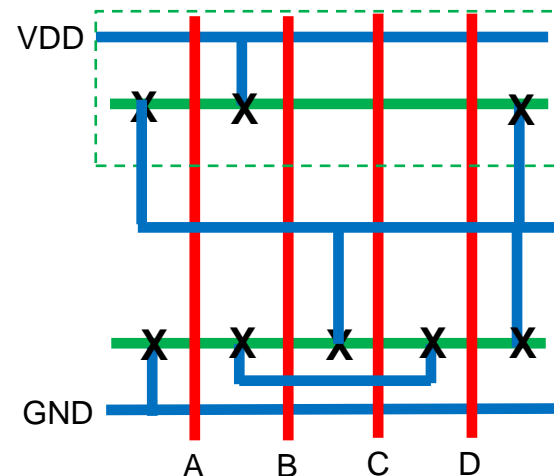
- $F = \overline{A(B+C+D)}$



Welke kiezen we als A de eerst veranderende ingang is?



2 diffusiecapaciteiten voor F

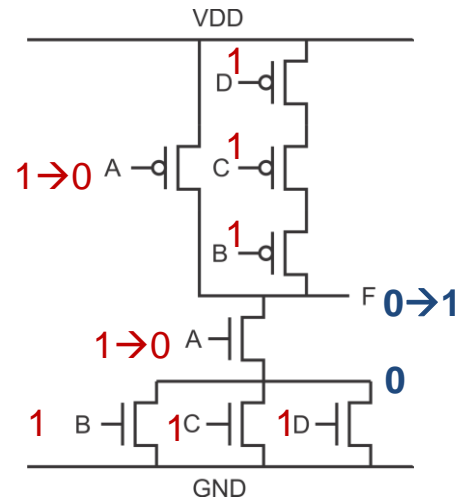


4 diff capaciteiten voor F

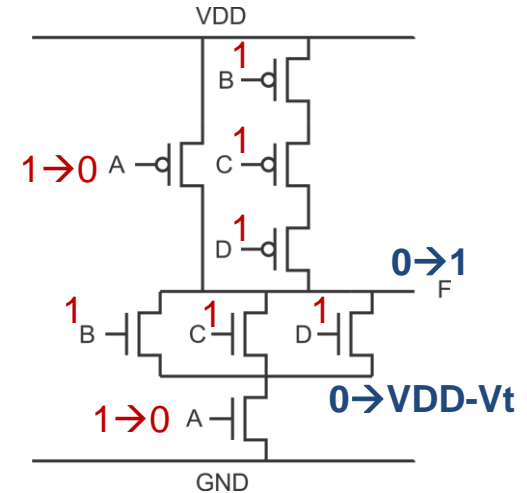
Ontwerptechnieken voor snellere poorten

Techniek 6 – layout is cruciaal!

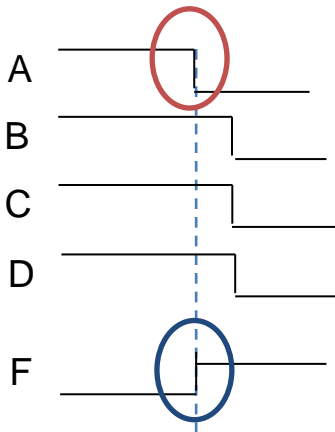
- $F = \overline{A(B+C+D)}$



Enkel uitgangscap
wordt opgeladen
3 drain caps

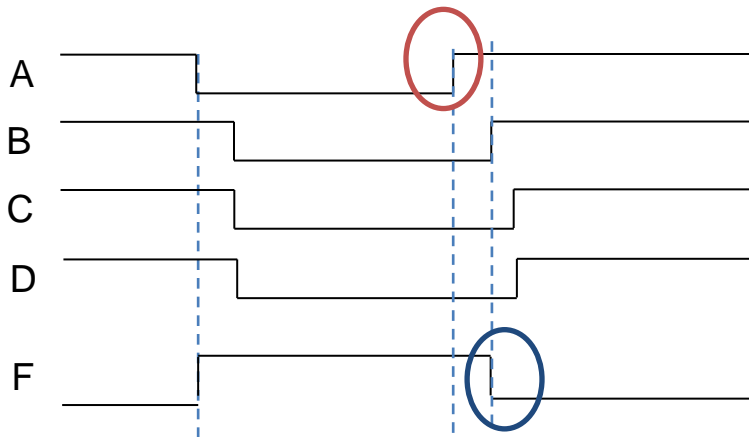
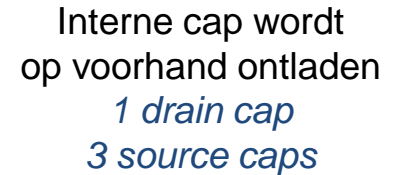
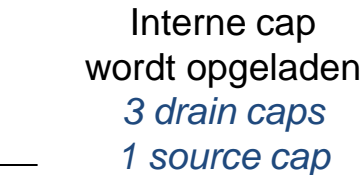


Interne en uitgangscap
worden opgeladen
6 drain caps
3 source caps



[illegible]

- $F = \overline{A(B+C+D)}$



Vertraging van complexe CMOS poorten

- Vertraging van de invertor

$$t_p = t_{p0} \left(1 + \frac{f}{\gamma} \right)$$

- t_{p0} = *intrinsic delay*
 - f = *effectieve fanout, ofwel electrical effort* ($f = C_{ext}/C_{gin}$)
 - γ = *technology factor* ($= C_{int}/C_{gin}$)
- *Uitbreiding naar een complexe poort*

$$t_p = t_{p0} \left(p + \frac{gf}{\gamma} \right) = t_{p0} d$$

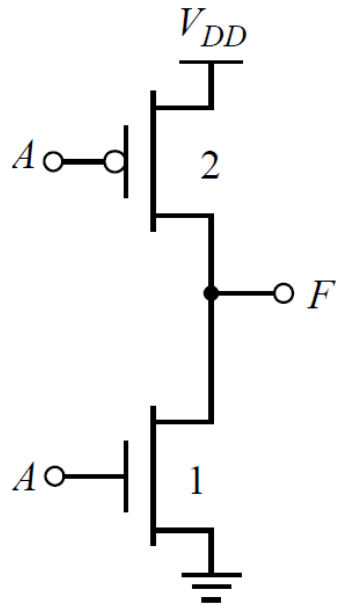
- f = *electrical effort*
 - g = *logical effort, ratio van deingangscapaciteit vs de inverter capaciteit wanneer deze berekend zijn om dezelfde stroom te geven dan de invertor*
 - p = *parasitic delay ofwel de ratio tussen de intrinsieke delay van de complexe poort vs de invertor*
 - d = *relatieve poortvertraging tov de intrinsieke delay van de referentie invertor*

Parameters voor een 2-input NAND en NOR

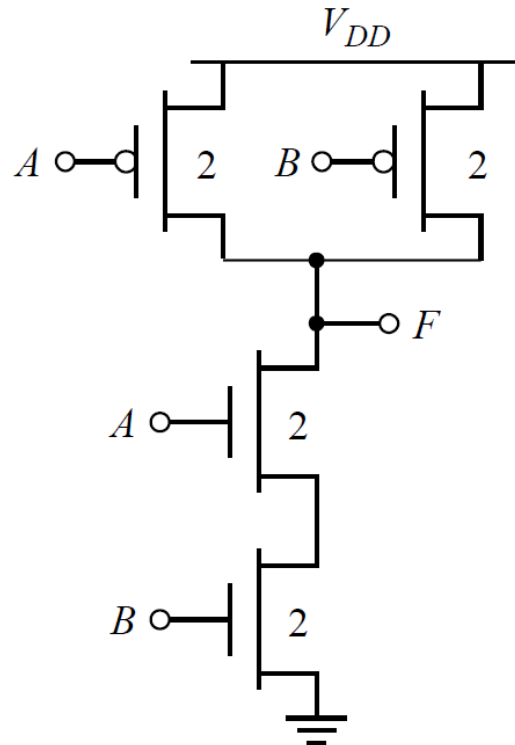
g = ratio van de ingangscapaciteit vs de invertor

p = ratio tussen de intrinsieke delay vs de invertor

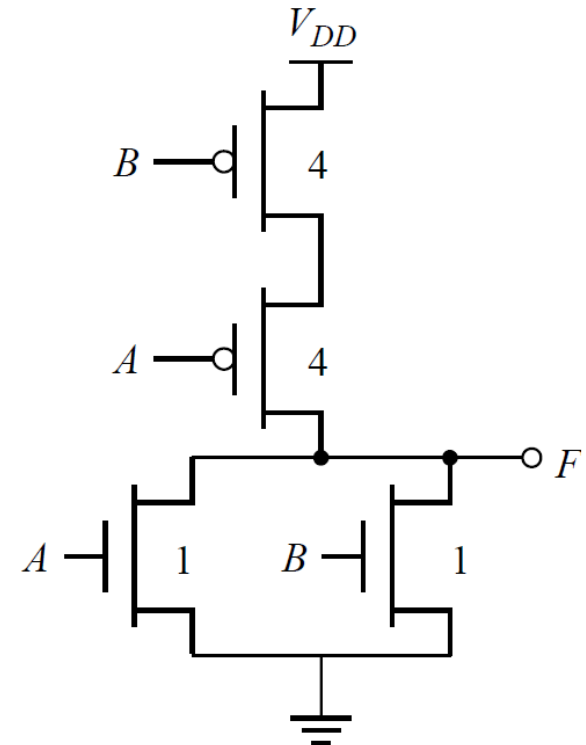
$$t_p = t_{p0} \left(p + \frac{gf}{\gamma} \right) = t_{p0} d$$



- $g = 1$
- $p = 1$



- $g = 4/3$
- $p = 2$



- $g = 5/3$
- $p = 2$

Complexere NAND/NOR

- Bereken dit nu eerst eens voor een 3-input NOR/NAND
- Daarna algemener voor een n-input NAND/NOR
- Vul de tabel verder in:
- We gaan er vanuit dat de ratio tussen PU en PD = 2

Poort	p	g
Invertor	1	1
NAND2	2	4/3
NAND3		
NAND-n		
NOR2	2	5/3
NOR3		
NOR-n		

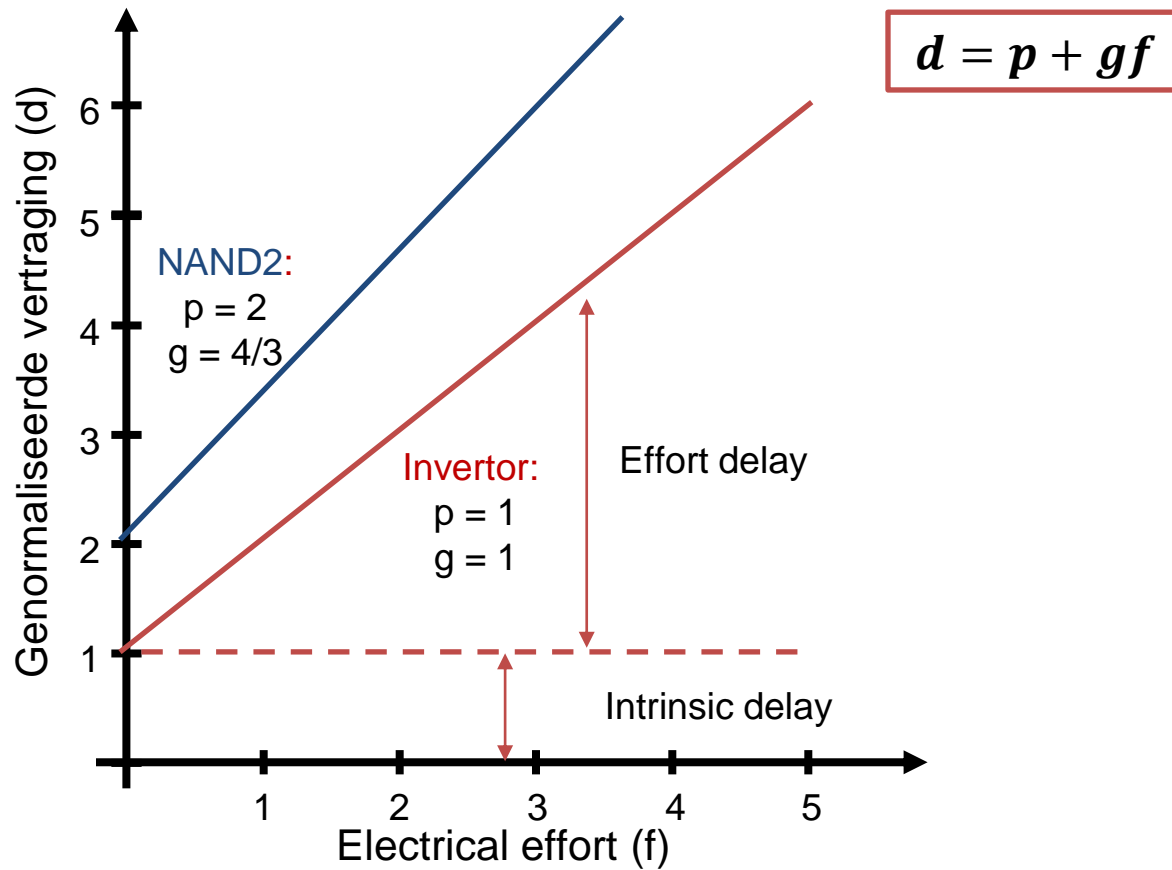
Complexere NAND/NOR

- Parasitic delay
 - Onafhankelijk van de 'load'
 - Onafhankelijk van 'W'
 - Wordt groter bij grotere poort
- Logical effort
 - Ratioingangscap vs invertor
 - Onafhankelijk van 'W'
 - Wordt groter bij stijgende complexiteit

Voor $W_p/W_n = 2$

Poort	p	g
Invertor	1	1
NAND2	2	4/3
NAND3	3	5/3
NAND-n	n	$(n+2)/3$
NOR2	2	5/3
NOR3	3	7/3
NOR-n	n	$(2n+1)/3$

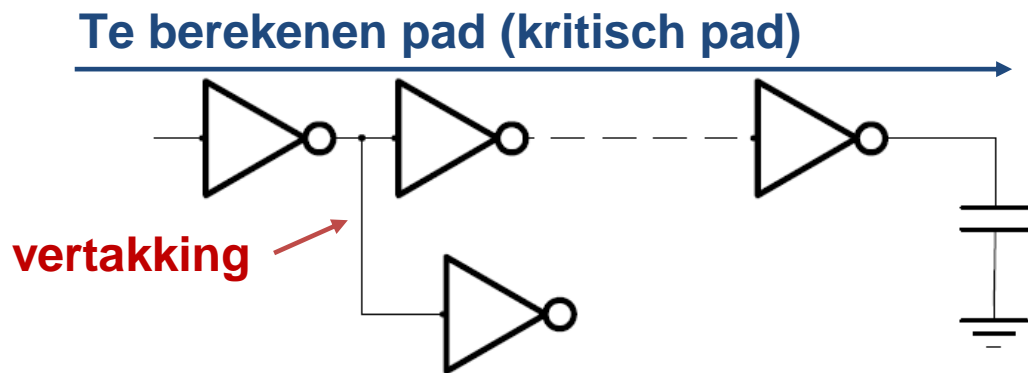
NAND2 vs INV



Branching effort (vertakkingen)

- Branching effort ofwel effort door vertakkingen tussen stages in het pad
 - Een deel van de uitgangsstroom wordt gebruikt om de volgende stage op te laden
 - Een deel van deze stroom wordt gebruikt om een vertakking (off-path) op te laden

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

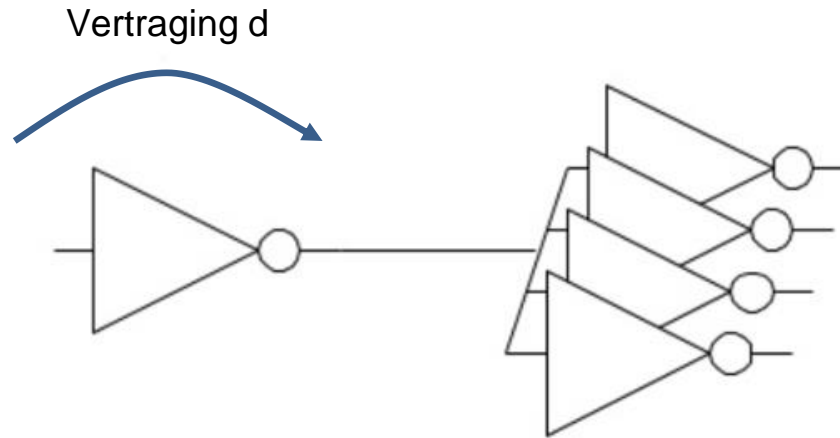


$C_{on\ path}$ = load cap van de poort op het pad

$C_{off\ path}$ = load cap van de aansluitingen op de vertakkingen

Bereken de F04 van een inverter

- F04 = fan-out of 4

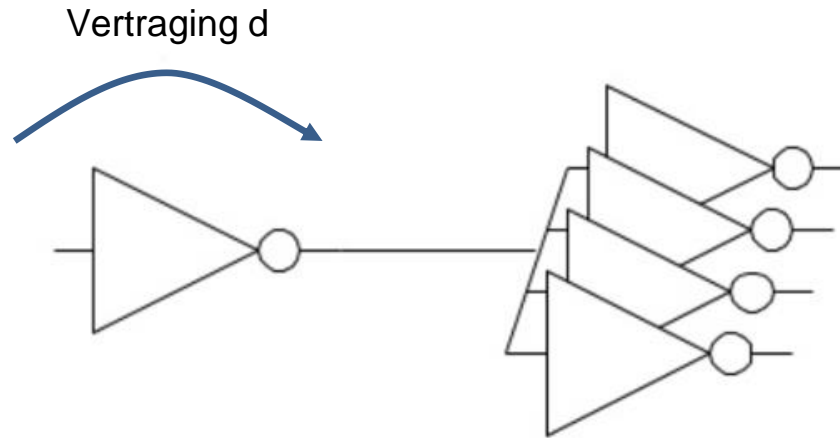


Logical effort
Electrical effort
Parasitic delay
Poortvertraging

F04 vertraging is ongeveer

Bereken de F04 van een inverter

- F04 = fan-out of 4



Logical effort	$g = 1$
Electrical effort	$f = 4$
Parasitic delay	$p = 1$
Poortvertraging	$d = 5$

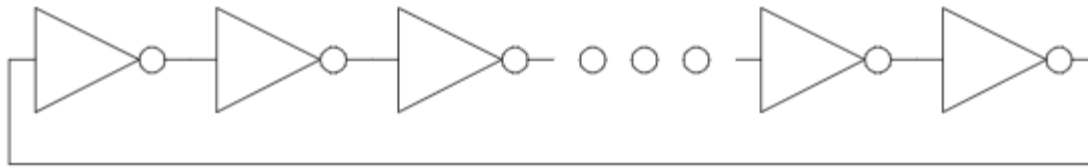
F04 vertraging is ongeveer
200 ps in 0.6 μ m tech
60 ps in 180 nm tech

Veel designers kennen de F04 vertraging van hun technologie en kunnen daarmee vergelijken.

Ring oscillator

- Bereken de snelheid van een N-stage ring oscillator

Ring oscillator bestaat uit oneven aantal stages (31)



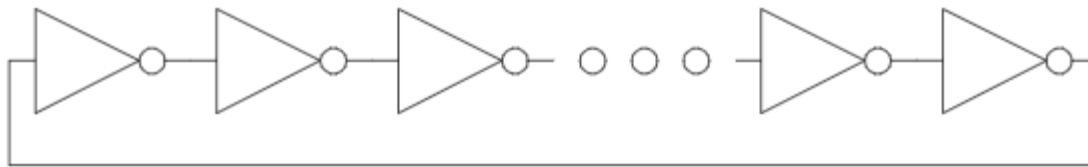
Logical effort
Electrical effort
Parasitic delay
Poortvertraging
Aantal stages
Frequentie

0.6 μ m tech

Ring oscillator

- Bereken de snelheid van een N-stage ring oscillator

Ring oscillator bestaat uit oneven aantal stages (31)



Logical effort $g = 1$

Electrical effort $f = 1$

Parasitic delay $p = 1$

Poortvertraging $d = 2$

Aantal stages N

Frequentie $f_{\text{osc}} = \frac{1}{2Nd} = \frac{1}{4N}$

0.6 μm tech

31 stages

Frequentie = $\frac{1}{2Nt_{p0}d} = 200\text{MHz}$

Vertraging van een multi-stage netwerk

- Algemene vertraging van een pad D

$$D = \sum_{i=1}^N (p_i + g_i f_i) = \sum_{i=1}^N p_i + \sum_{i=1}^N h_i = \sum_{i=1}^N d_i$$

- Stage effort: $\mathbf{h_i = g_i f_i}$
- Electrical effort van het pad: $\mathbf{F = C_{out}/C_{in} = \prod_{i=1}^N f_i}$
- Logical effort van het pad: $\mathbf{G = \prod_{i=1}^N g_i}$
- Branching effort van het pad: $\mathbf{B = \prod_{i=1}^N b_i}$
- Pad effort in total: $\mathbf{H = GFB}$

Optimale effort per stage

- We veronderstellen opnieuw dat elke stage dezelfde effort doe

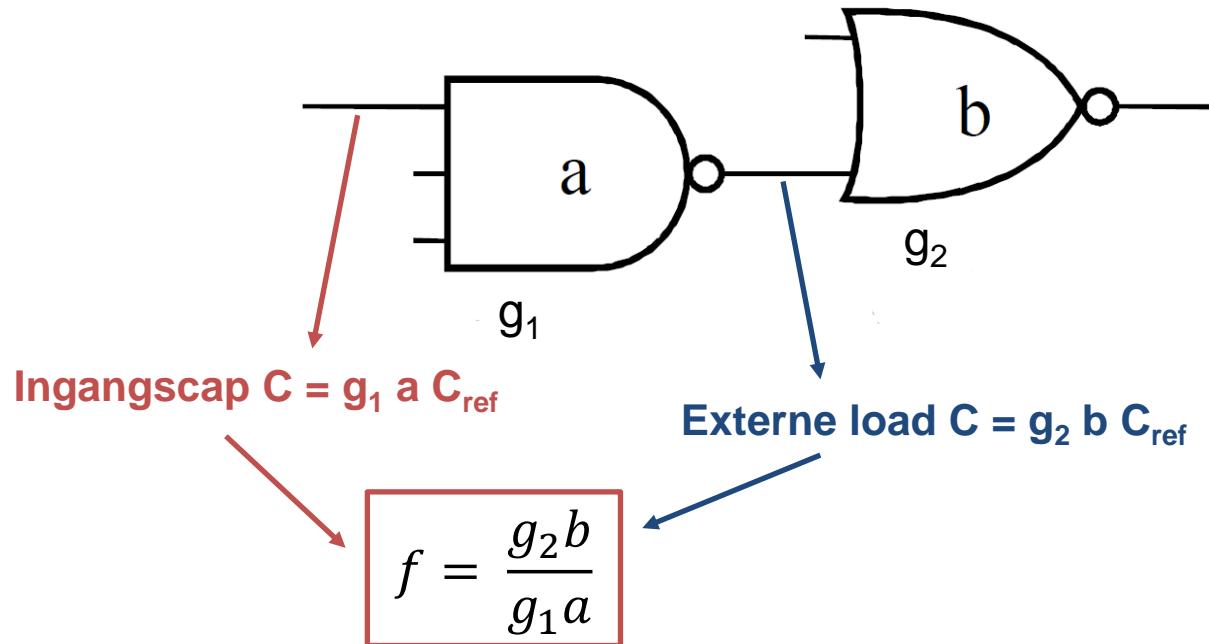
$$h^N = H \rightarrow h = \sqrt[N]{H}$$

- Stage efforts: $h = g_1 f_1 = g_2 f_2 = \dots$
 - Dit betekent ook: $g_1 C_{\text{ext},1} / C_{g,1} = g_2 C_{\text{ext},2} / C_{g,2} = \dots$
- Effective fanout van elke stage: $f_i = h/g_i$
- Minimum pad delay

$$\hat{D} = \sum_{i=1}^N (p_i + g_i f_i) = N \sqrt[N]{H} + P$$

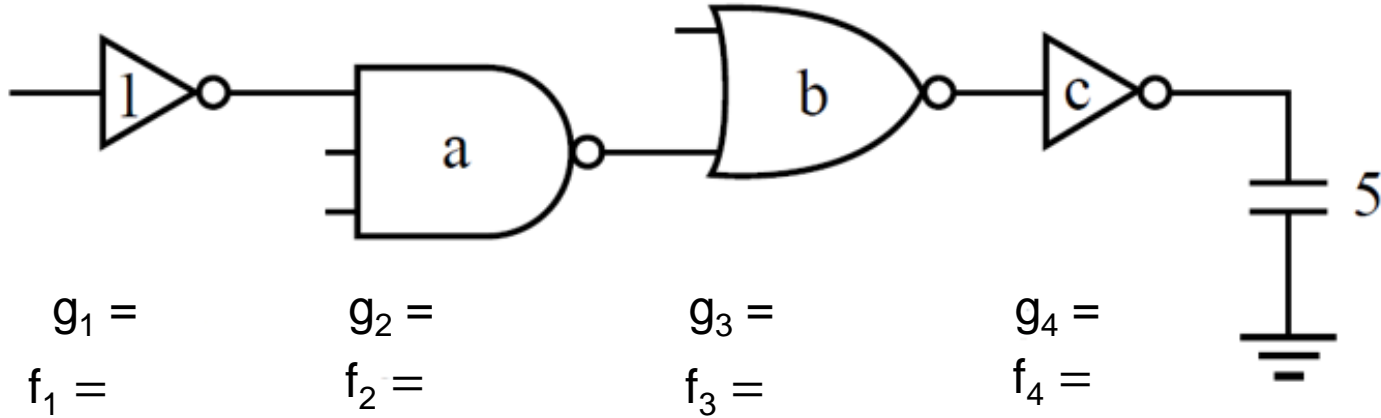
Electrical effort

- Electrical effort
= ratio tussen external load en ingangscapaciteit van een poort
- We veronderstellen dat een poort met grootte '1' ($a=1$) dezelfde *drive* heeft als een minimale inverter met ingangscap C_{ref}



Optimalizatie van een pad

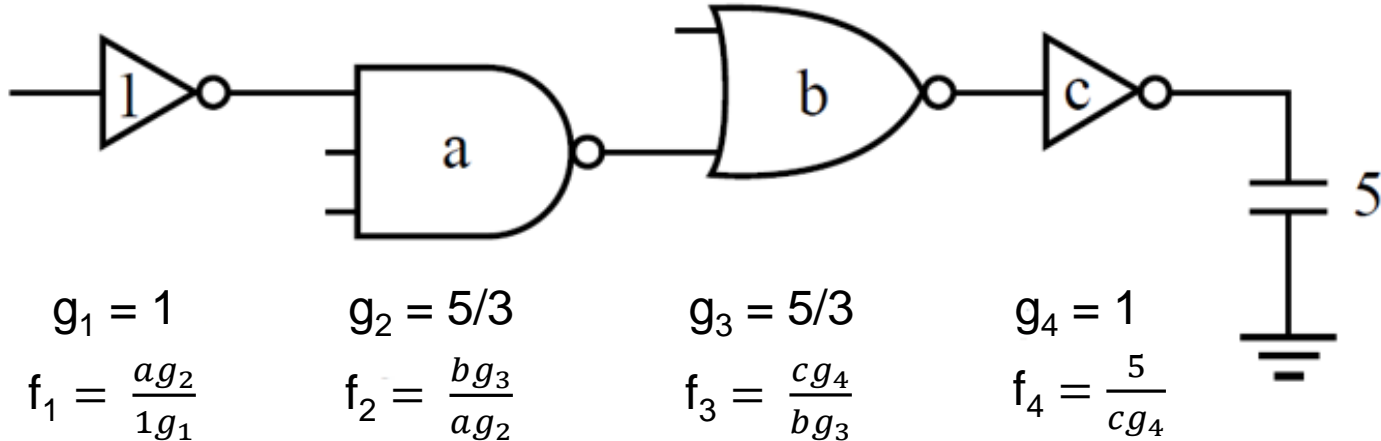
f = electrical effort \rightarrow ratio tussen external load cap van een poort tov deingangscap



- \rightarrow Zoek f , rekening houdend dat de stage effort voor elke stage dezelfde moet zijn
- \rightarrow Reken uit om a , b , c te vinden!

Optimalizatie van een pad

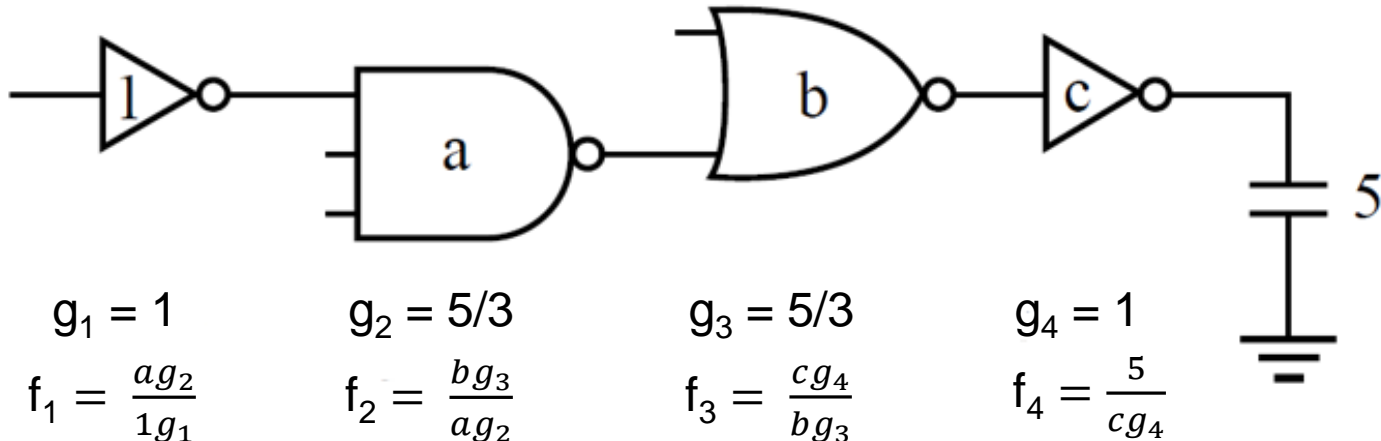
f = electrical effort \rightarrow ratio tussen external load cap van een poort tov deingangscap



- \rightarrow Zoek f , rekening houdend dat de stage effort voor elke stage dezelfde moet zijn
- \rightarrow Reken uit om a , b , c te vinden!

Optimalizatie van een pad

f = electrical effort \rightarrow ratio tussen external load cap van een poort tov de ingangscap



\rightarrow Zoek f , rekening houdend dat de stage effort voor elke stage dezelfde moet zijn

Effective fanout

Pad logical effort

Vertakkingen (geen)

Pad effort

Optimale stage effort

(Parasitaire delay)

(Minimale delay)

$$F = C_L / C_{g,1} = 5$$

$$G = 1 \times 5/3 \times 5/3 \times 1 = 25/9$$

$$B = 1$$

$$H = GFB = 125/9 = 13.9$$

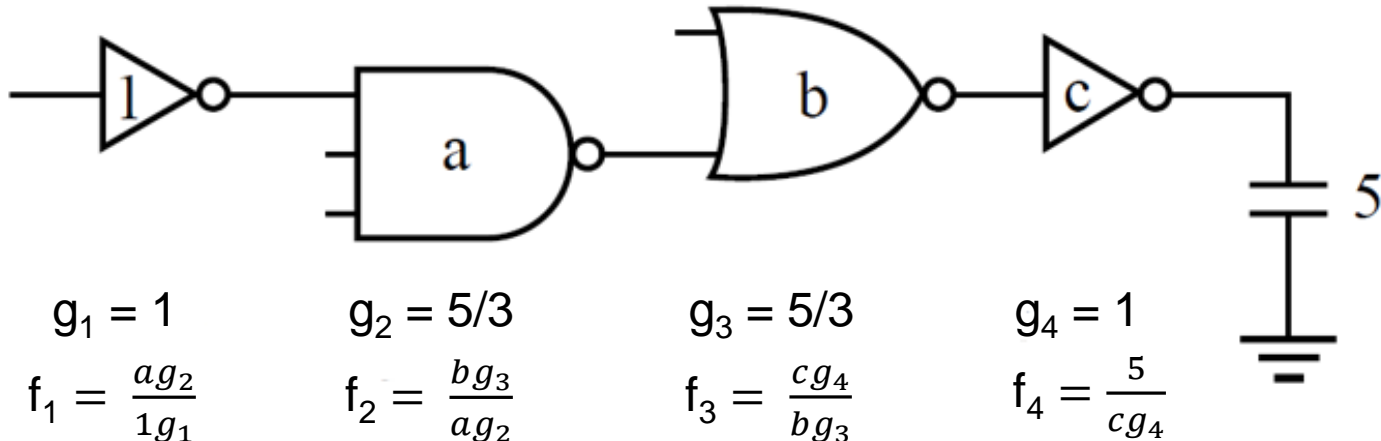
$$\hat{h} = \sqrt[4]{13.9} = 1.93$$

$$P = 1 + 3 + 2 + 1 = 7$$

$$\hat{D} = P + N \hat{h} = 7 + 4 \times 1.93 = 14.72$$

Optimalizatie van een pad

f = electrical effort \rightarrow ratio tussen external load cap van een poort tov deingangscap



\rightarrow Zoek f , rekening houdend dat de stage effort voor elke stage dezelfde moet zijn

Optimale stage effort

$$\hat{h} = \sqrt[4]{13.9} = 1.93$$

Met de optimale stage effort kunnen we f bepalen:

Electrical effort

$$f = \frac{\hat{h}}{g_i}$$

$$f_1 = \frac{1.93}{1} = 1.93$$

$$f_2 = \frac{1.93}{5/3} = 1.16$$

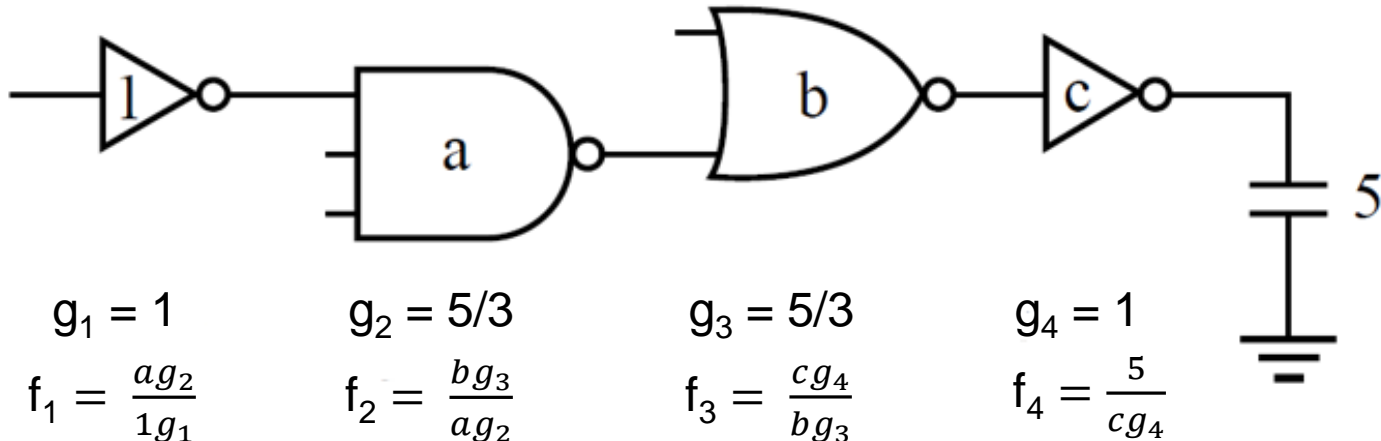
$$f_3 = \frac{1.93}{5/3} = 1.16$$

$$f_4 = \frac{1.93}{1} = 1.93$$

Invertoren krijgen een grotere elektrische effort omdat ze beter zijn om loads aan te sturen.

Optimalizatie van een pad

f = electrical effort \rightarrow ratio tussen external load cap van een poort tov deingangscap



- \rightarrow Zoek f , rekening houdend dat de stage effort voor elke stage dezelfde moet zijn
- \rightarrow Reken uit om a , b , c te vinden!

$$f_1 = \frac{1.93}{1} = 1.93$$

$$f_2 = \frac{1.93}{5/3} = 1.16$$

$$f_3 = \frac{1.93}{5/3} = 1.16$$

$$f_4 = \frac{1.93}{1} = 1.93$$

$$a = \frac{f_1 g_1}{g_2} = 1.158$$

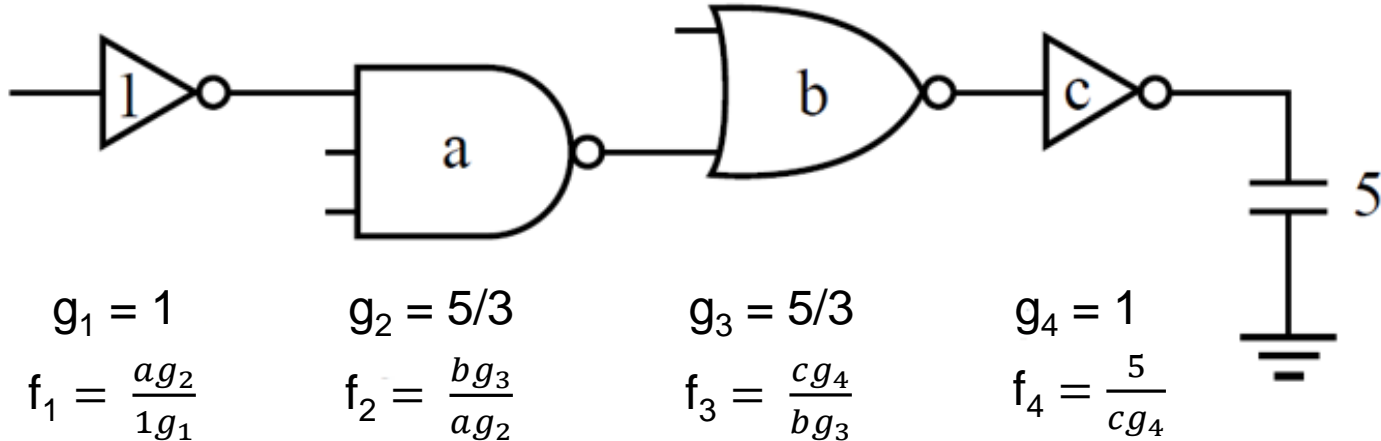
$$b = \frac{a f_2 g_2}{g_3} = 1.34$$

$$c = \frac{f_3 b g_3}{g_4} = 2.59$$

$$c = \frac{5}{f_4 g_4} = 2.59$$

Optimalizatie van een pad – simpele methode

f = electrical effort \rightarrow ratio tussen external load cap van een poort tov de ingangscap



\rightarrow Optie om verder te gaan zonder f te berekenen

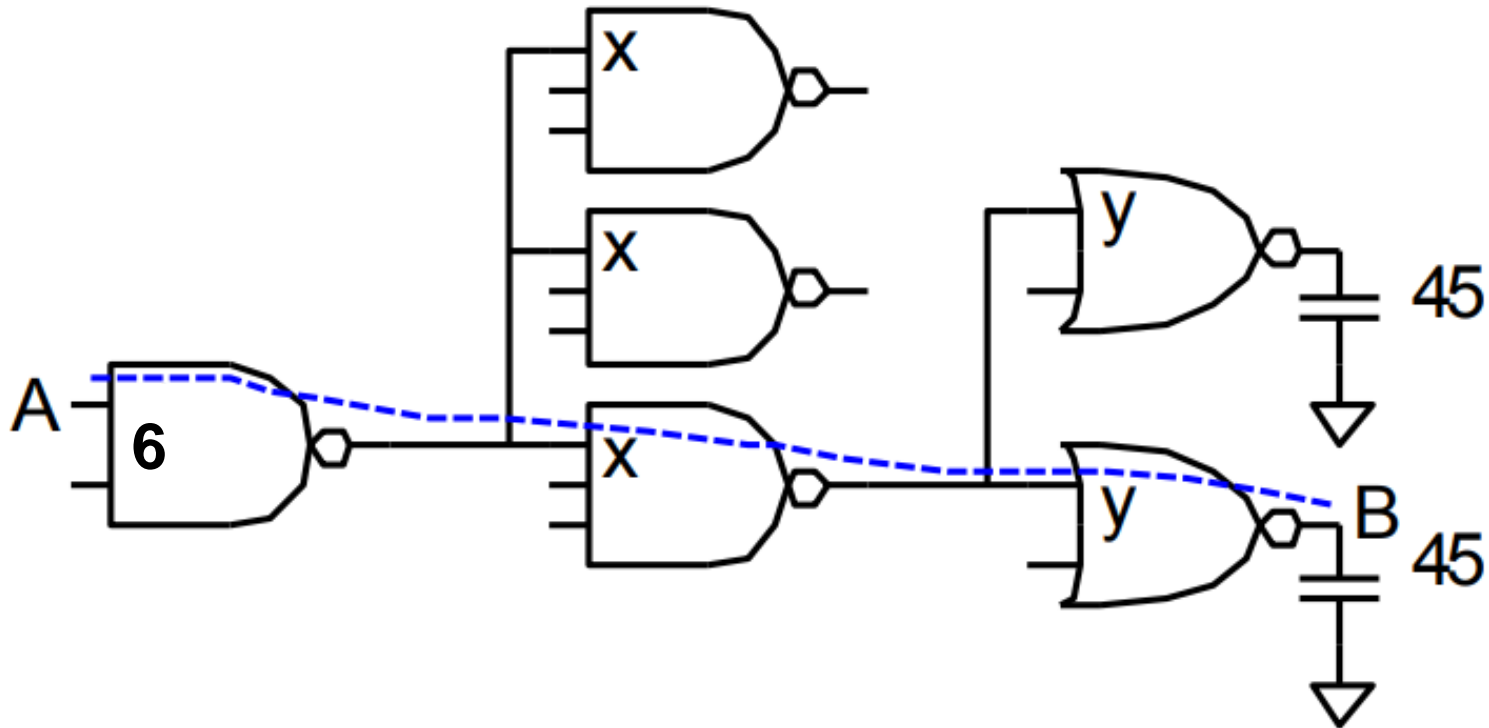
$$f_1 = \frac{ag_2}{1g_1} = \frac{\hat{h}}{g_1} \xrightarrow{\text{yields}} \mathbf{a = \frac{\hat{h}}{g_2} = 1.158}$$

$$f_2 = \frac{bg_3}{ag_2} = \frac{\hat{h}}{g_2} \xrightarrow{\text{yields}} \mathbf{b = \frac{a\hat{h}}{g_3} = 1.34}$$

$$f_3 = \frac{cg_4}{bg_3} = \frac{\hat{h}}{g_3} \xrightarrow{\text{yields}} \mathbf{c = \frac{b\hat{h}}{g_4} = 2.59}$$

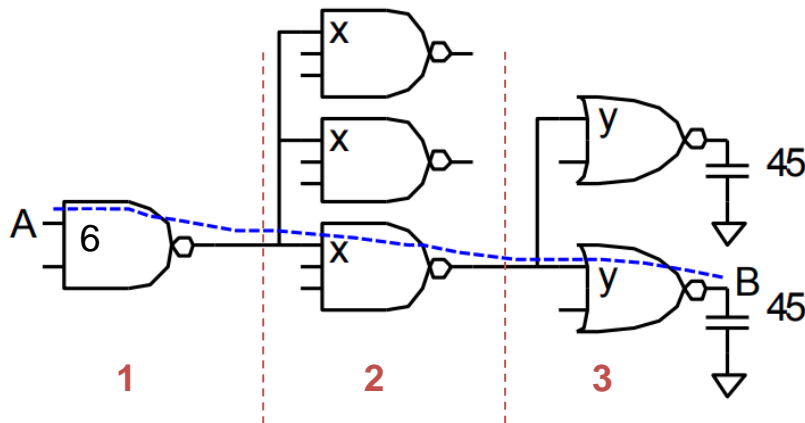
Oefening met vertakking

Bereken x en y zodat je de minste vertraging krijgt van A naar B



Oefening met vertakking

Bereken x en y zodat je de minste vertraging bekomt van A naar B



$$g_1 = 4/3 \quad g_2 = 5/3 \quad g_3 = 5/3$$

$$f_1 = \frac{b_1 x g_2}{6 g_1} \quad f_2 = \frac{b_2 y g_3}{x g_2} \quad f_3 = \frac{45}{y g_3}$$

$$b_1 = \frac{x+2x}{x} = 3 \quad b_2 = \frac{y+y}{y} = 2$$

Effective fanout

Pad logical effort

Vertakkingen (geen)

Pad effort

Optimale stage effort

(Parasitaire delay)

Minimale delay

$$F = C_L / C_{g,1} = 45/8$$

$$G = 100/27$$

$$B = 3 * 2 = 6$$

$$H = GFB = 125$$

$$\hat{h} = \sqrt[3]{125} = 5$$

$$P = 2+3+2 = 7$$

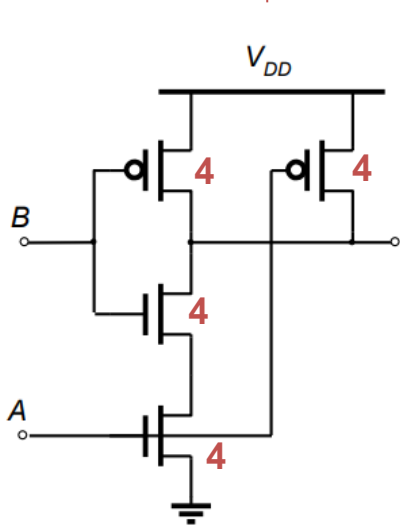
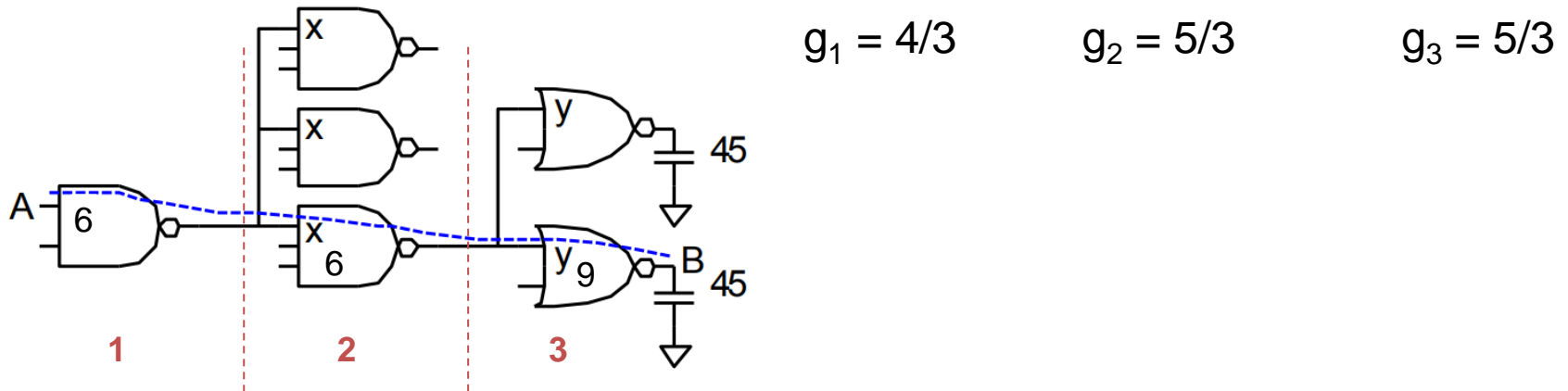
$$\hat{D} = P + N \hat{h} = 7 + 3 * 5 = 22 = 4.4 \text{ FO4}$$

$$f_1 = \frac{b_1 x g_2}{6 g_1} = \frac{\hat{h}}{g_1} \text{ yields } x = \frac{6 \hat{h}}{b_1 g_2} = 6$$

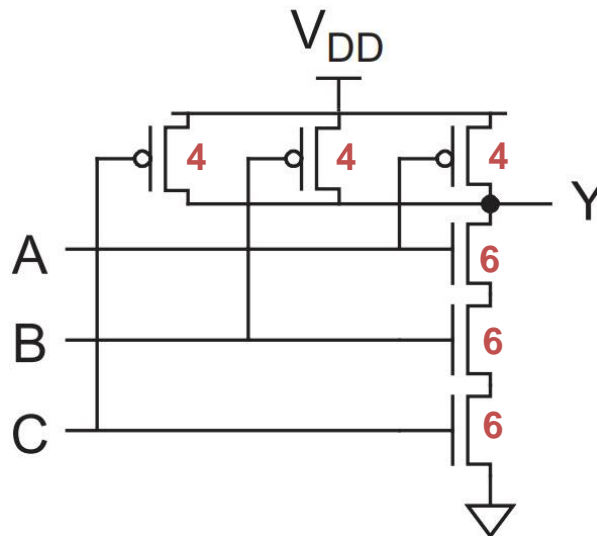
$$f_3 = \frac{45}{y g_3} = \frac{\hat{h}}{g_3} \text{ yields } y = \frac{45}{\hat{h}} = 9$$

Oefening met vertakking

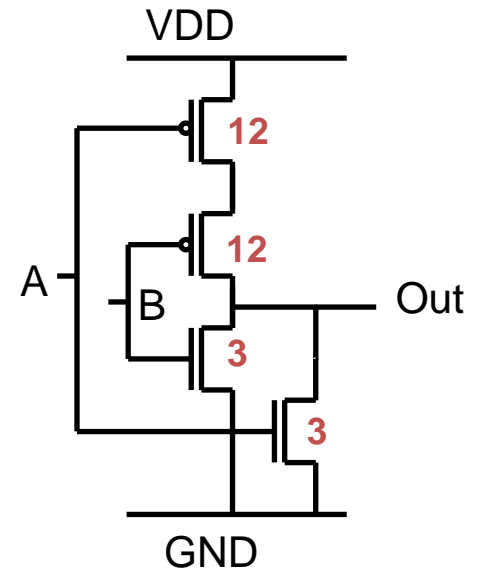
Nu zetten we deze nog om naar transistorgroottes voor dezelfde stroom



$$C_{g,1} = 4/3 * 6 = 8$$



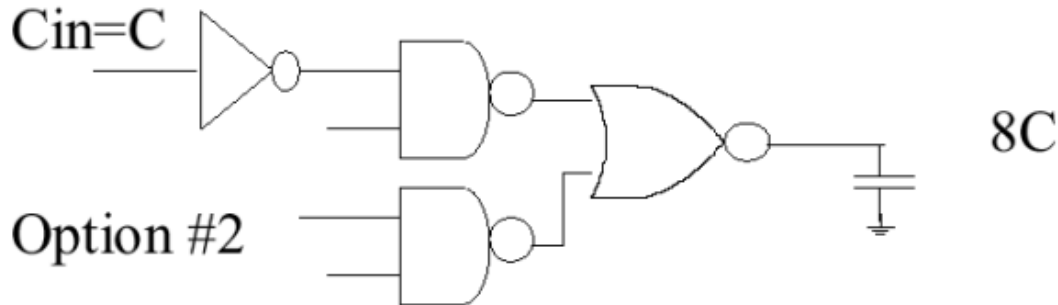
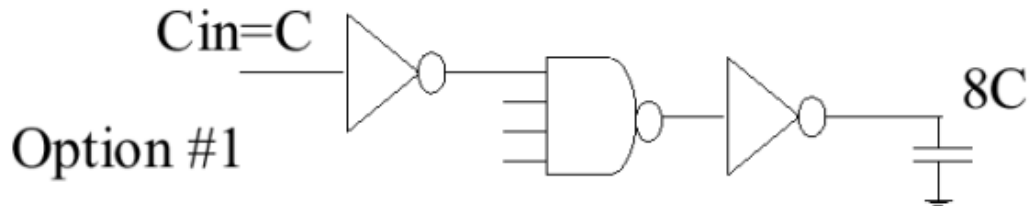
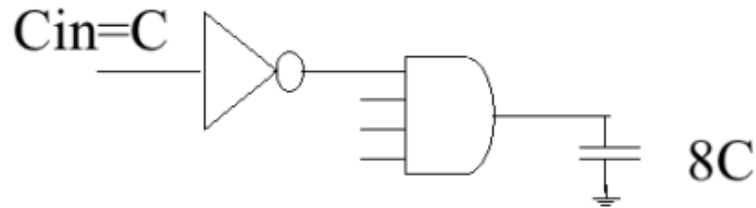
$$C_{g,2} = 5/3 * 6 = 10$$



$$C_{g,3} = 5/3 * 9 = 15$$

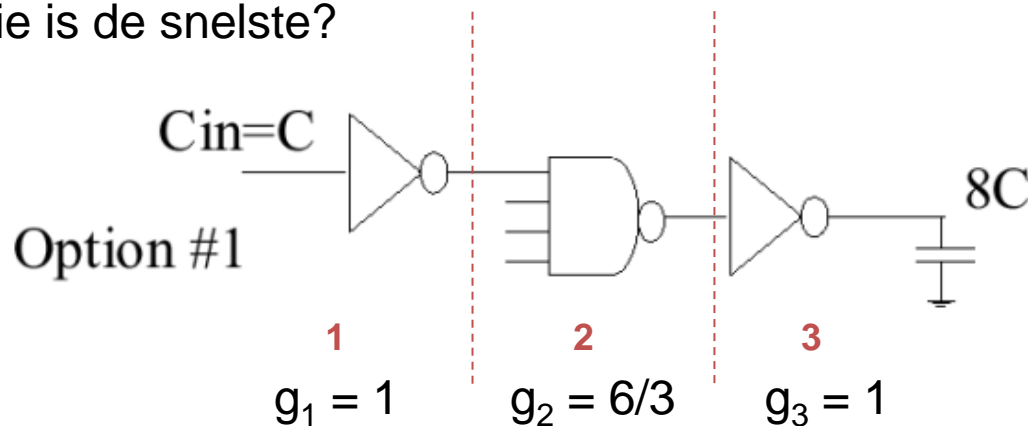
Extra oefening

Welke optie is de snelste?



Extra oefening

Welke optie is de snelste?



Effective fanout

Pad logical effort

Vertakkingen (geen)

Pad effort

Optimale stage effort

(Parasitaire delay)

Minimale delay

$$F = C_L / C_{g,1} = 8$$

$$G = 6/3 = 2$$

$$B = 1$$

$$H = GFB = 16$$

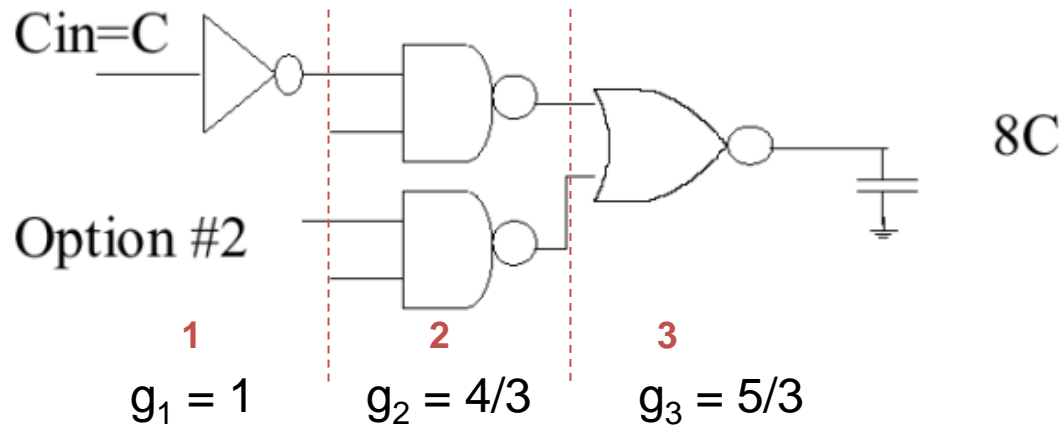
$$\hat{h} = \sqrt[3]{16} = 2.5$$

$$P = 1 + 4 + 1 = 6$$

$$\hat{D} = P + N \hat{h} = 6 + 3 \cdot 2.5 = 13.5$$

Extra oefening

Welke optie is de snelste?



Effective fanout

$$F = C_L / C_{g,1} = 8$$

Pad logical effort

$$G = 20/9$$

Vertakkingen (geen)

$$B = 1$$

Pad effort

$$H = GFB = 160/9$$

Optimale stage effort

$$\hat{h} = \sqrt[3]{H} = 2.60$$

(Parasitaire delay)

$$P = 1 + 2 + 2 = 5$$

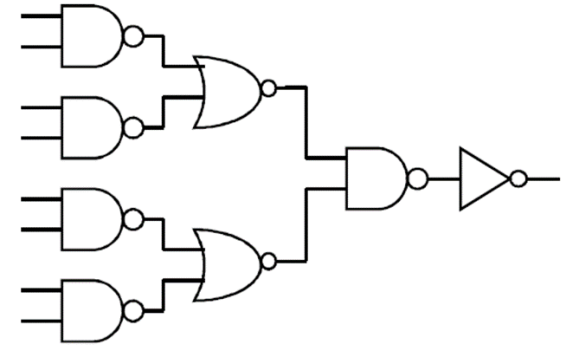
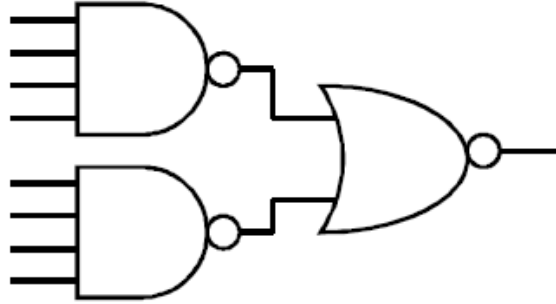
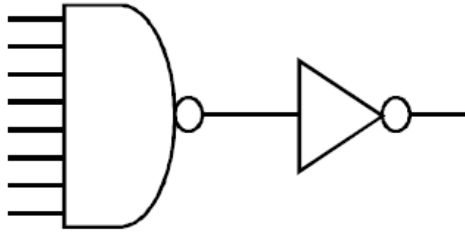
Minimale delay

$$\hat{D} = P + N \hat{h} = 5 + 3 \cdot 2.6 = 12.8$$

Optie 2 is de snelste

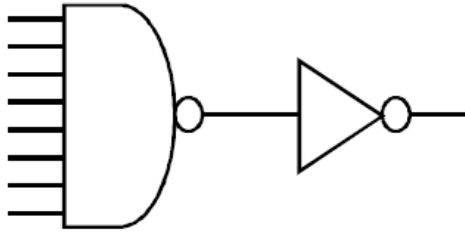
Welke is de snelste?

Welke optie is de snelste als $C_{in} = C$ en $C_{out} = 8C$?

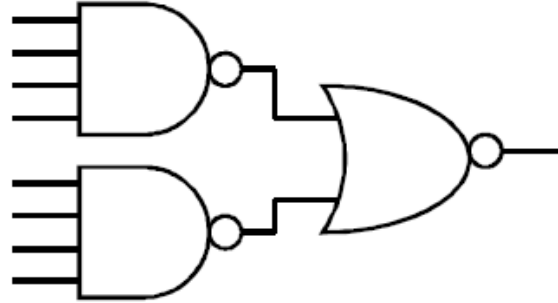


Welke is de snelste?

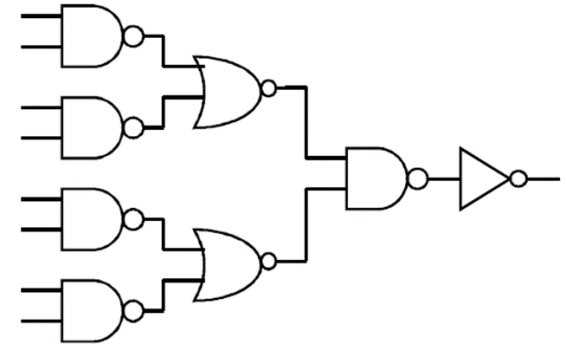
Welke optie is de snelste als $C_{in} = C$ en $C_{out} = 8C$?



$$\begin{aligned}F &= C_L/C_{g,1} = 8 \\G &= 10/3 \\B &= 1 \\H &= GFB = 80/3 \\\hat{h} &= \sqrt[2]{H} = 5.16 \\P &= 8+1 = 9 \\\hat{D} &= P+N \hat{h} = 19.32\end{aligned}$$



$$\begin{aligned}F &= C_L/C_{g,1} = 8 \\G &= 6/3 \cdot 5/3 \\B &= 1 \\H &= GFB = 240/9 \\\hat{h} &= \sqrt[2]{H} = 5.16 \\P &= 4+2 = 6 \\\hat{D} &= P+N \hat{h} = 16.32\end{aligned}$$

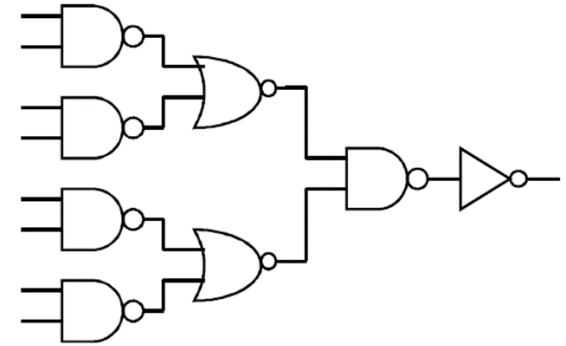
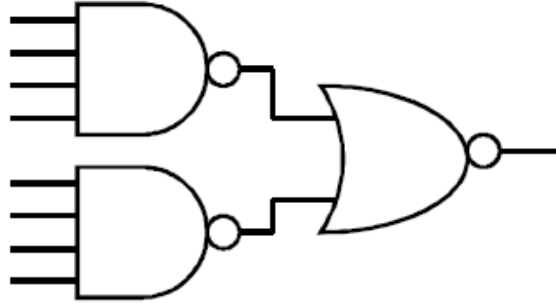
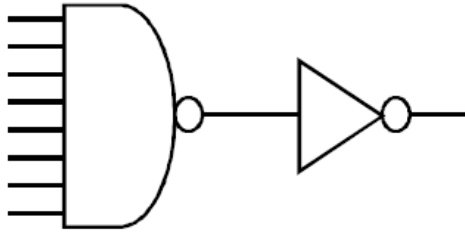


$$\begin{aligned}F &= C_L/C_{g,1} = 8 \\G &= 4/3 \cdot 5/3 \cdot 4/3 \\B &= 1 \\H &= GFB = 640/27 \\\hat{h} &= \sqrt[4]{H} = 2.21 \\P &= 2+2+2+1 = 7 \\\hat{D} &= P+N \hat{h} = 15.82\end{aligned}$$

Welke optie is de snelste als $C_{in} = C$ en $C_{out} = 25C$?

Welke is de snelste?

Welke optie is de snelste als $C_{in} = C$ en $C_{out} = 25C$?



$$F = C_L/C_{g,1} = 25$$

$$G = 10/3$$

$$B = 1$$

$$H = GFB = 250/3$$

$$\hat{h} = \sqrt[2]{H} = 9.12$$

$$P = 8+1 = 9$$

$$\hat{D} = P+N \hat{h} = 27.25$$

$$F = C_L/C_{g,1} = 25$$

$$G = 6/3 \cdot 5/3$$

$$B = 1$$

$$H = GFB = 83.33$$

$$\hat{h} = \sqrt[2]{H} = 9.12$$

$$P = 4+2 = 6$$

$$\hat{D} = P+N \hat{h} = 24.25$$

$$F = C_L/C_{g,1} = 25$$

$$G = 4/3 \cdot 5/3 \cdot 4/3$$

$$B = 1$$

$$H = GFB = 74$$

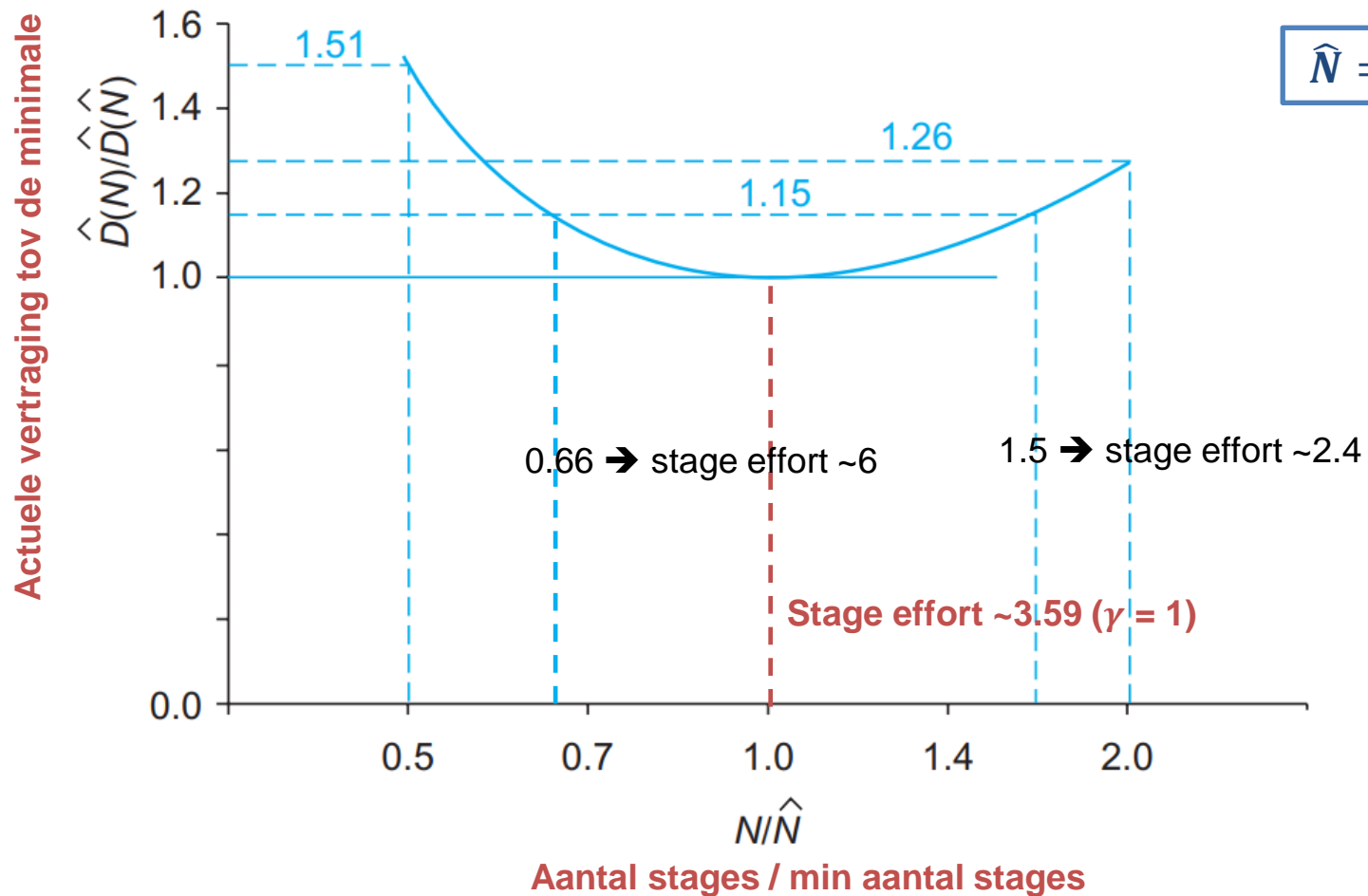
$$\hat{h} = \sqrt[4]{H} = 2.93$$

$$P = 2+2+2+1 = 7$$

$$\hat{D} = P+N \hat{h} = 18.73$$

Is dit nu het ideale aantal stages?

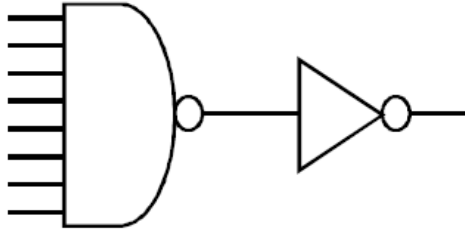
Gevoeligheid van de delay



- Vertraging is binnen 15% als stage effort tussen 6 en 2.4 ligt.
- Wij kiezen opnieuw voor stage effort 4, delays zijn 2% tov min

Welke is de snelste?

Bereken de snelste optie voor de drie gevallen:



$$F = C_L / C_{g,1} = 25$$

$$G = 10/3$$

$$B = 1$$

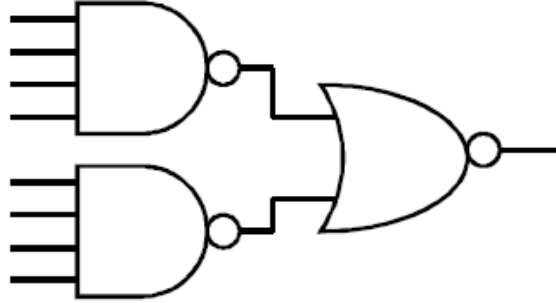
$$H = GFB = 250/3$$

$$\hat{N} =$$

$$\hat{h} =$$

$$P = 8 + 1 = 9$$

$$\hat{D} = P + N \hat{h} =$$



$$F = C_L / C_{g,1} = 25$$

$$G = 6/3 * 5/3$$

$$B = 1$$

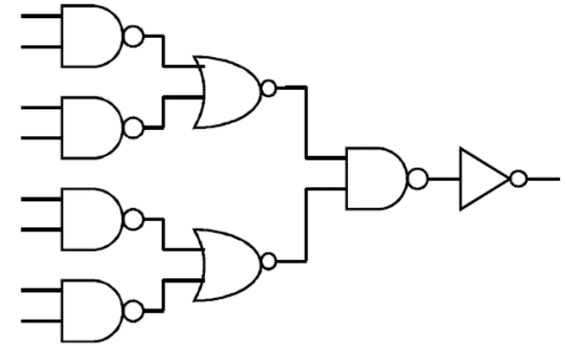
$$H = GFB = 83.33$$

$$\hat{N} =$$

$$\hat{h} =$$

$$P = 4 + 2 = 6$$

$$\hat{D} = P + N \hat{h} =$$



$$F = C_L / C_{g,1} = 25$$

$$G = 4/3 * 5/3 * 4/3$$

$$B = 1$$

$$H = GFB = 74$$

$$\hat{N} =$$

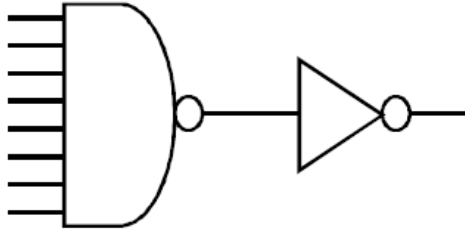
$$\hat{h} =$$

$$P = 2 + 2 + 2 + 1 = 7$$

$$\hat{D} = P + N \hat{h} =$$

Welke is de snelste?

Bereken de snelste optie voor de drie gevallen:



$$F = C_L / C_{g,1} = 25$$

$$G = 10/3$$

$$B = 1$$

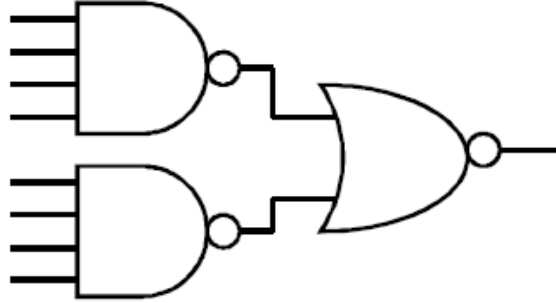
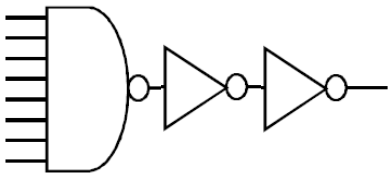
$$H = GFB = 250/3$$

$$\hat{N} = \log_4 H = 3.2$$

$$\hat{h} = \sqrt[3]{H} = 4.37$$

$$P = 8 + 1 = 9$$

$$\hat{D} = P + N \hat{h} = 22.10$$



$$F = C_L / C_{g,1} = 25$$

$$G = 6/3 * 5/3$$

$$B = 1$$

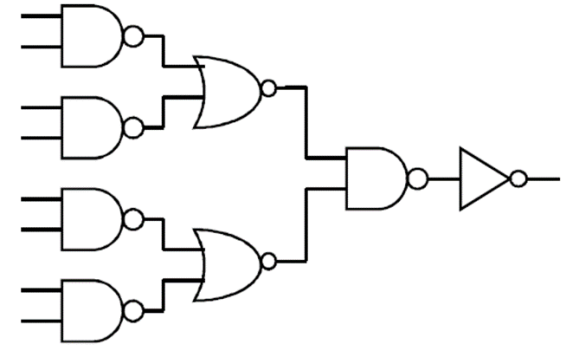
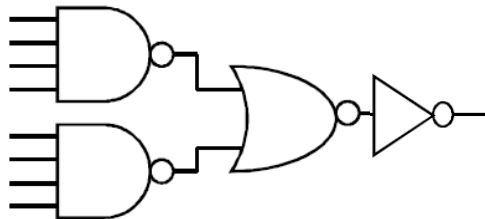
$$H = GFB = 83.33$$

$$\hat{N} = \log_4 H = 3.2$$

$$\hat{h} = \sqrt[3]{H} = 4.37$$

$$P = 4 + 2 = 6$$

$$\hat{D} = P + N \hat{h} = 19.10$$



$$F = C_L / C_{g,1} = 25$$

$$G = 4/3 * 5/3 * 4/3$$

$$B = 1$$

$$H = GFB = 74$$

$$\hat{N} = \log_4 H = 3.1$$

$$\hat{h} = \sqrt[4]{H} = 2.93$$

$$P = 2 + 2 + 2 + 1 = 7$$

$$\hat{D} = P + N \hat{h} = 18.73$$

Invertoren voegen
niets toe aan H

Limieten van logical effort

- Kip of het ei probleem
 - Je moet eerst G berekenen
 - Maar je kent het aantal stages niet zonder G
- Eenvoudig model
 - Houdt bv geen rekening met rise time effecten
- Extra iteratie nodig als er interconnects zijn
- Focust op max snelheid
 - Niet echt op min area/power voor een gegeven delay

Conclusie – logical effort

- Logical effort is zinvol om over vertragingen te denken
 - Numerische logical effort karakteriseert poorten
 - NANDs zijn sneller dan NORs in CMOS
 - Het snelste pad bekom je bij een stage effort van 4
 - Maar is niet zo gevoelig voor aantal stages of groottes
 - Minder stages is niet onmiddellijk snellere paden
 - Invertoren en NAND2 zijn het beste om grote caps aan te sturen