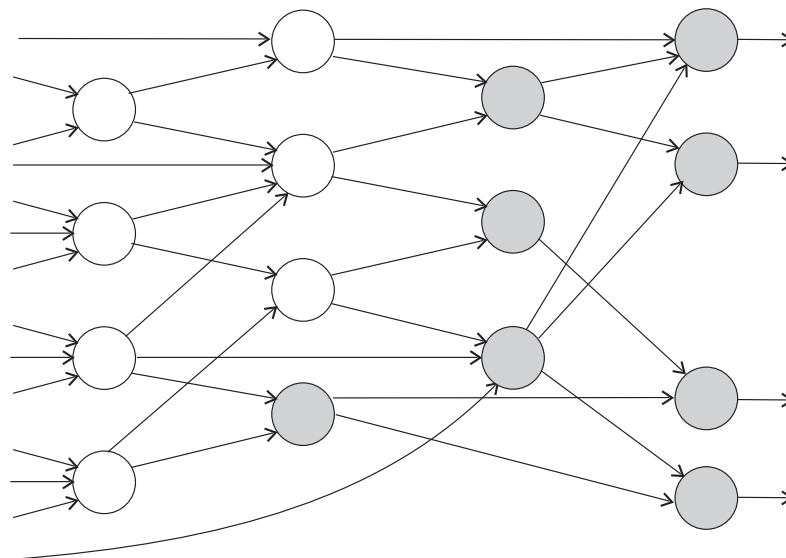


tentamen 23 juni 2000

In pijplijnen worden vaak vergrendelschakelingen (*latches* of *flipflops*) gebruikt, om de volgorde van de bewerkingen goed te regelen. Deze schakelingen worden door kloksignalen gestuurd en daarmee wordt vastgelegd wanneer de resultaten van acties beschikbaar moeten zijn en wanneer ze aan een volgende bewerkingseenheid worden aangeboden. Het is daarom vaak handiger de functies van die vergrendelschakelingen niet zozeer als geheugenelementen te zien, maar veeleer als middelen om signalen te vertragen en beschikbaarheid te synchroniseren. Bij de meest gebruikte schakelfamilies, dat wil zeggen bij statische logica, is dat dan ook precies de functie.

Logische families die door een klok gestuurd worden kunnen ook zonder vergrendelen een pijplijn vormen, en in deze toets gaan we dit laten zien voor een vorm van dynamische logica die de laatste jaren weer erg populair begint te worden, namelijk *domino-logica*. Om het eenvoudig te houden kiezen we voor een zogenaamde echte twee-fasen-klok. Het principe is uitbreidbaar tot meerfasen-systemen.



Figuur 1: Een twee-fasen domino-pijplijn

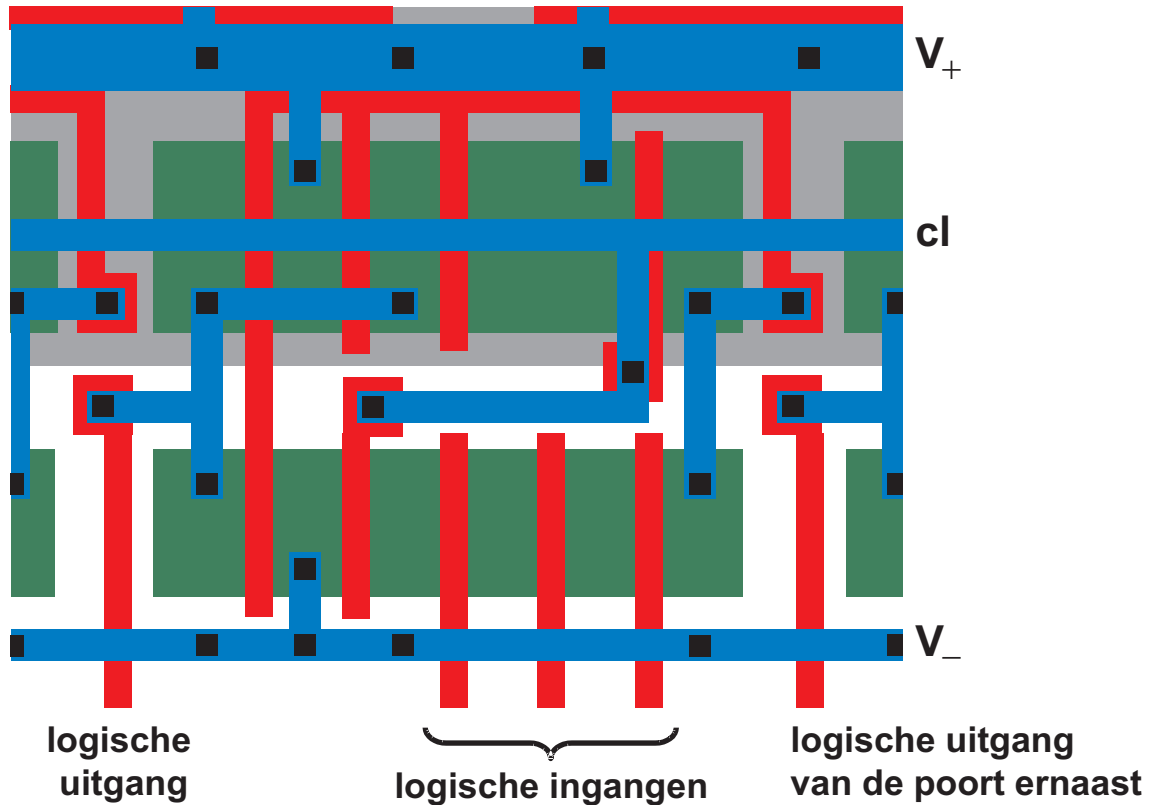
In figuur 1 is zo'n twee-fasen-pijplijn als een netwerk van *domino*-poorten weergegeven. De open cirkeltjes stellen poorten voor die door de eerste fase ϕ_1 van de klok gestuurd worden, terwijl de grijze cirkeltjes door de tweede fase ϕ_2 gestuurd worden. Nooit mag in zo'n netwerk een domino-poort met ϕ_2 rechtstreeks een invoersignaal voor een poort met ϕ_1 produceren. Verder zal het duidelijk zijn dat de complexiteit van de poorten en de paden in het netwerk eisen stelt aan de duur van de niveaus van de klok-fasen. Echter de volgorde van de overgangen van de beide kloksignalen is daar onafhankelijk van.

Vraag 1: Geef tijddiagrammen van de beide kloksignalen die qua volgorde van overgangen correct zijn! Geef ook aan hoeveel *clock skew*¹ in de door U getekende tijddiagrammen acceptabel is voor de diverse overgangen.



¹ Clock skew is de afwijking van de nominale tijdstippen voor de overgangen

Bij het ontwerpen van zo'n pijplijn gaat het er natuurlijk vooral om dat de frequentie van de beide kloksignalen hoog genoeg kan zijn. Zoals gezegd hangt deze frequentie af van de complexiteit van de poorten en de paden in het netwerk. Bij een gegeven netwerk, waarin elke poort dus een door logische synthese bepaalde functie moet uitvoeren, kan men dus slechts proberen de poorten zo snel mogelijk te maken en dit kan alleen door de dimensionering van de transistoren als we aannemen dat de vertraging van de verbindingen verwaarloosd mag worden. Echter om "layout-technische" redenen is de vrijheid daar ook zeer beperkt.



Figuur 2: Layout van een domino-poort

In figuur 2 is de layout van één zo'n poort gegeven. De lengte van alle transistoren is minimaal, terwijl de breedte van de nmos-transistoren vijf maal deze minimale lengte is. De pmos-transistoren zijn nog eens vijfentwintig procent breder. De enige vrijheid die overblijft is een aantal van deze pmos-transistoren parallel te schakelen. In de figuur is de pull-up van de invertor uitgevoerd met drie transistoren in parallel.

Vraag 2: Wat is de logische functie van de poort in figuur 2?

3-input-and

We veronderstellen dat de capaciteit van de *sense-node* van de poort gedomineerd wordt door de gate-capaciteiten van de invertor. Alle overige capaciteiten van dit knooppunt en van alle interne knooppunten verwaarlozen we daarom. Verder gaan we uit van de volgende gegevens:

$$\begin{array}{llll}
 \mu_n = 0,06m^2/Vs & c_{ox} = 10^{-3}F/m^2 & L_{min} = 0,5\mu & V_+ = 3V \\
 \mu_p = 0,024m^2/Vs & V_i = 0,32V & W_{min} = 1,0\mu & V_- = 0V \\
 \gamma = 0,3V^{\frac{1}{2}} & & V_{tno} = 0,5V & V_{tpo} = -0,5V
 \end{array}$$

c_{ox} is gate-capaciteit per oppervlakte-eenheid.

De equivalente weerstand van een geleidende transistor is omgekeerd evenredig met de breedte en de mobiliteiten. Aangezien alleen gate-capaciteiten meegenomen moeten worden, kunnen we de totale vertraging modelleren als de som van twee RC-tijden met een geschikt gekozen evenredigheidsconstante r .

$$t = (n + 1) \frac{r}{\mu_n W} c(\alpha + 1)W + \frac{r}{\mu_p \alpha W} c f_o W = rc \left[\frac{n + 1}{\mu_n} + \frac{f_o}{\mu_p \alpha} \right]$$

De afgeleide naar α is dus

$$\frac{dt}{d\alpha} = rc \left[\frac{n + 1}{\mu_n} - \frac{f_o}{\mu_p \alpha^2} \right].$$

Het minimum wordt dus bereikt voor

$$\alpha^2 = \frac{\mu_n}{\mu_p} \frac{f_o}{n + 1}$$

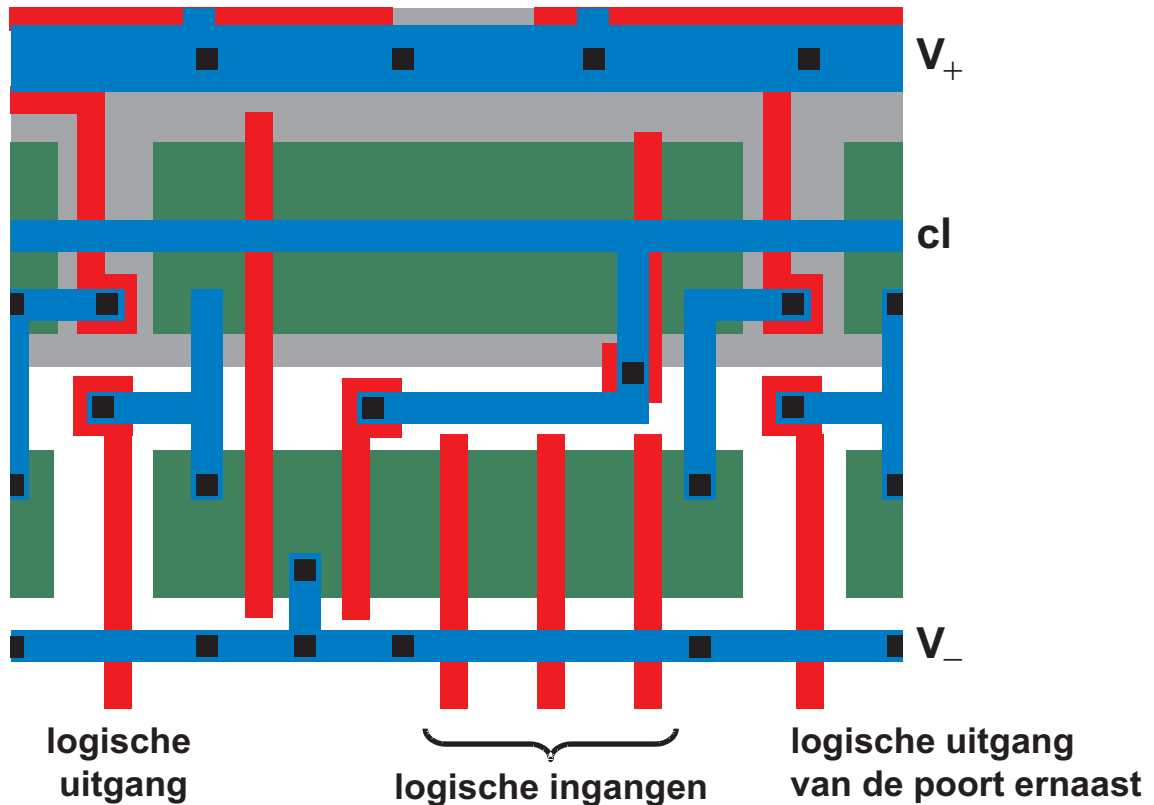
Vraag 3: Wat is het aantal pmos-transistoren in de pull-up van de invertor voor de snelste uitvoering van deze poort als de uitgang met 10 transistoren in andere poorten verbonden wordt?

$$N = 2$$

Vraag 4: Kunt U ook een algemene formule voor N , het optimale aantal pmos-transistoren in de invertor-pull-up, geven? Zo ja, geef deze formule dan hiernaast?

$$N = 0.8 \sqrt{\frac{\mu_n}{\mu_p} \frac{f_o}{n + 1}}$$

Vraag 5: Het aantal pmos-transistoren in de pull-up van de invertor in figuur 2 is oneven. Wat moet er aan de layout van de schakeling veranderen als het optimale aantal even is?



De layout van de poort is zo gemaakt dat ze eenvoudig naast elkaar geplaatst kunnen worden. Alleen het netwerk van nmos-transistoren hangt af van de functie van de poort en het aantal pmos-transistoren kan ook per poort verschillen. Door “spiegelen” kan nog een rij gevormd worden die dan de n-well met de eerste deelt. De hele pijplijn kan dan gevormd worden door een aantal van die dubbelrijen met bedrading tussen de rijen en met een tweede metaal-laag over de rijen heen.

Vraag 6: Waarom is de metaallijn van de positieve voeding tweemaal zo breed als die van de lage voedingslijn?

Vraag 7: Waarom is de kloklijn niet (ook) over de nmos-transistoren gelegd?