



Academie voor Technology, Innovation &  
Society Delft  
Academie voor ICT & Media

# Digitale System Engineering 2

**Week 2 – Toestandsmachines (vervolg)**

**Jesse op den Brouw**

**DIGSE2/2016-2017**

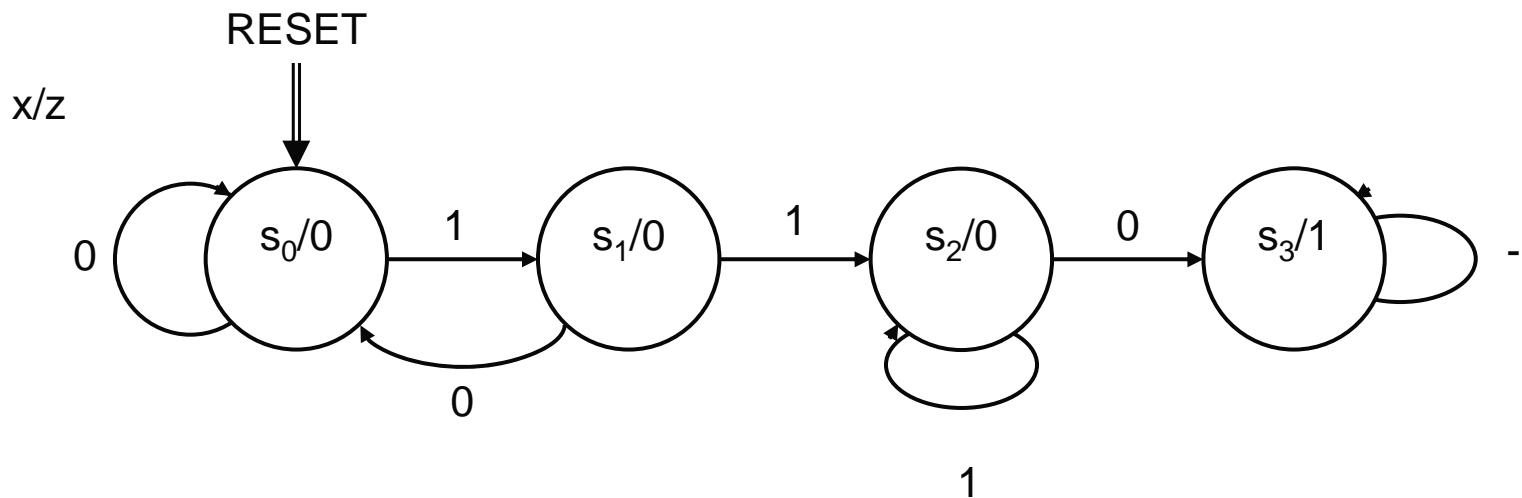
**DE HAAGSE**  
HOGESCHOOL

# Herkenningautomaat

- Een typische sequentiële machine is een *herkenningautomaat* of *patroonherkenner*.
- Herkenningautomaten spelen een belangrijke rol in het vakgebied berekenbaarheidstheorie (*theory of computation*) dat zich bezighoudt met hoe efficiënt problemen kunnen worden opgelost d.m.v een algoritme (en of ze eigenlijk wel oplosbaar zijn).
- De automaten spelen een belangrijke rol bij het opstellen en herkennen van zogenaamde talen (*languages*). Denk hierbij aan het parsen van computertalen of zogenaamde *regular expressions*.

# Herkenningautomaat

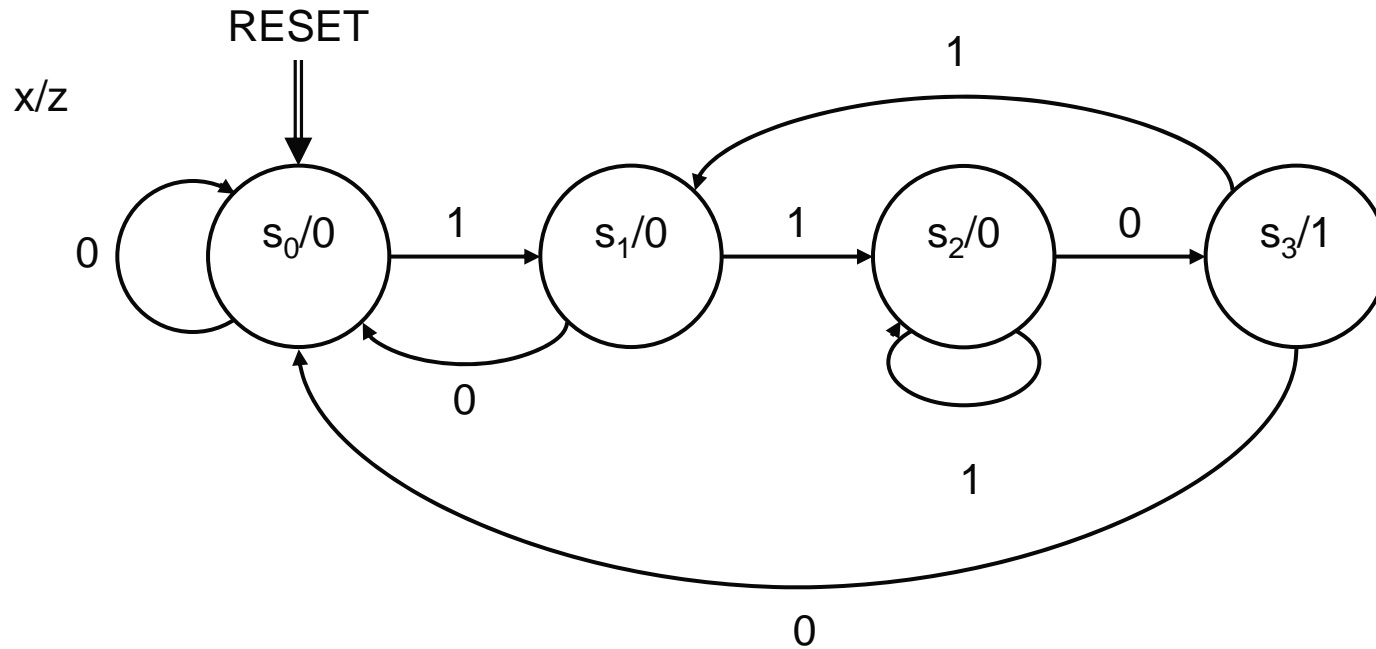
- Ontwerp een machine voor het herkennen van het patroon (of reeks) 110. De machine geeft een 1 af en stopt met herkennen.



0010110011001111000  
000000011111111111

# Herkenningautomaat

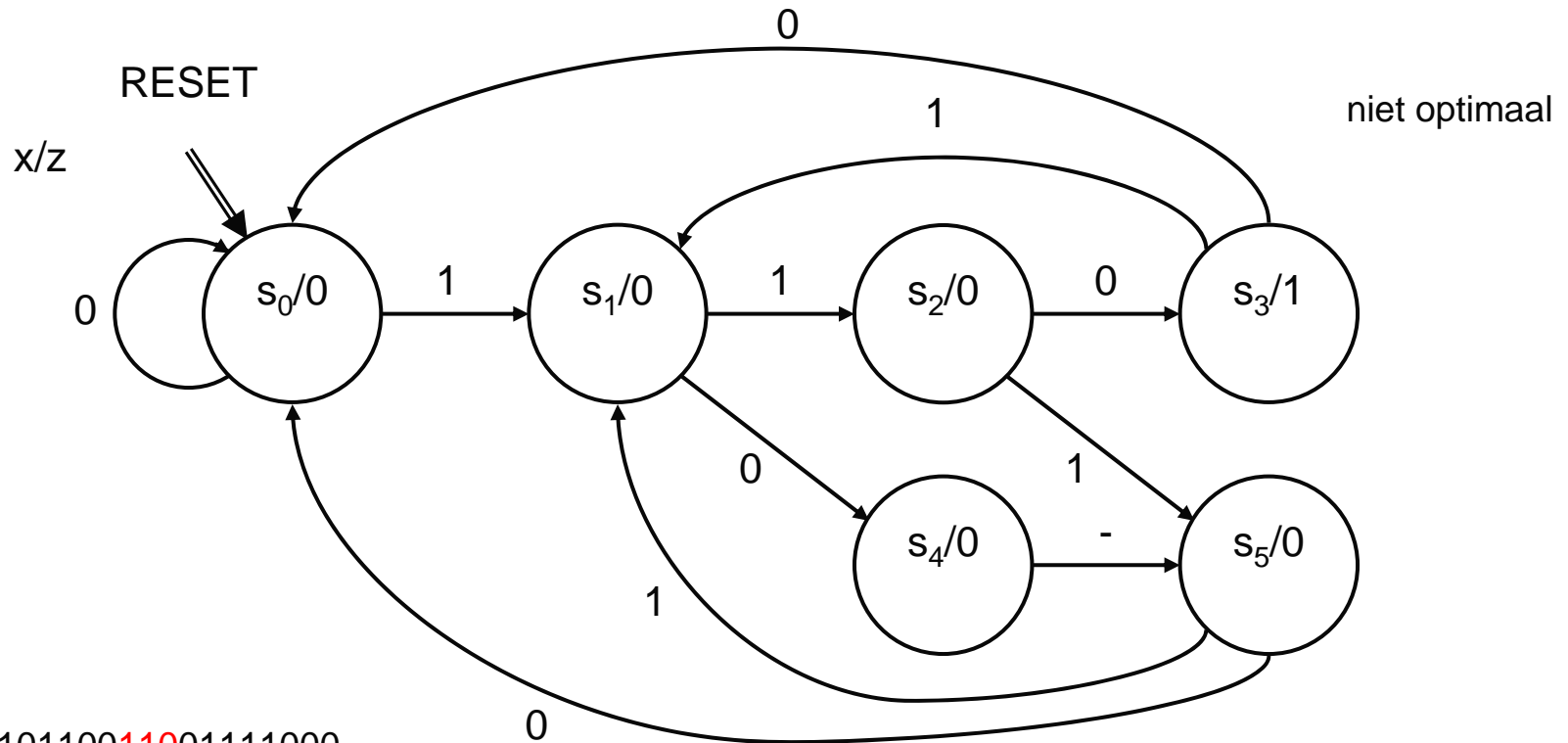
- Ontwerp een machine voor het doorlopend herkennen van het patroon (of reeks) 110. De machine moet bij herkenning een 1 afgeven.



0010110011001111000  
0000000100010000010

# Herkenningautomaat

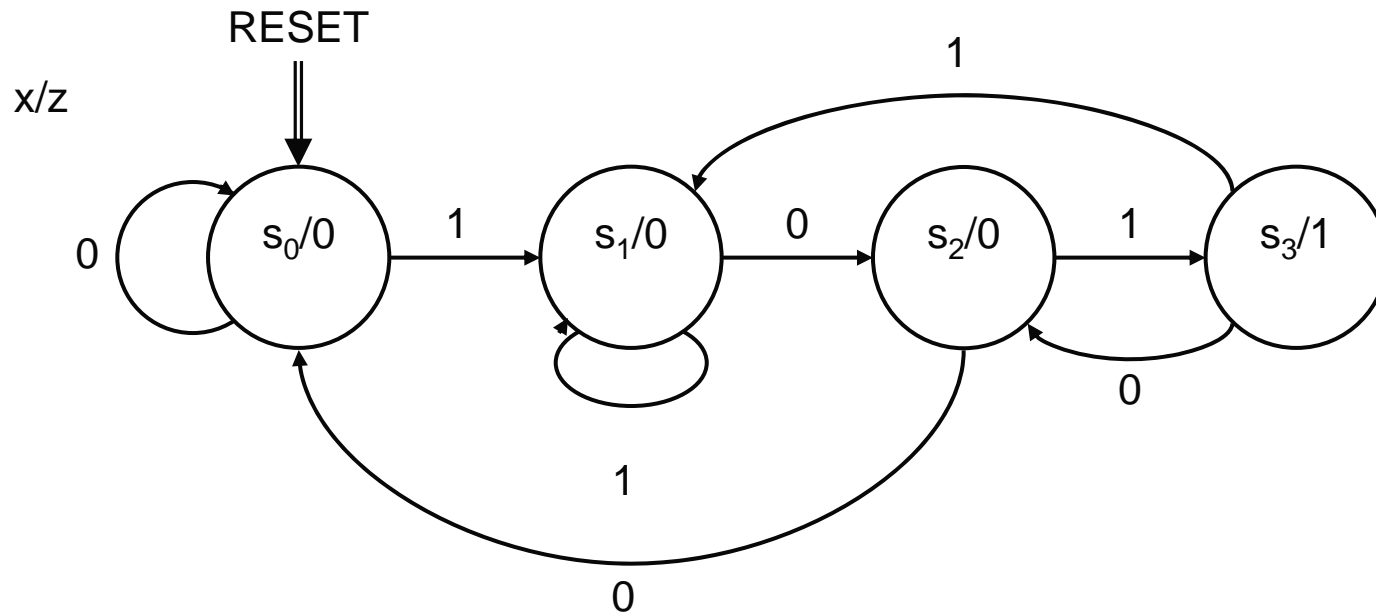
- Ontwerp een machine voor het bloksgewijs herkennen van het patroon (of reeks) 110. De machine begint met herkennen bij een 1 en moet bij herkenning een 1 afgeven.



0010110011001111000  
00000000000010000000

# Herkenningautomaat

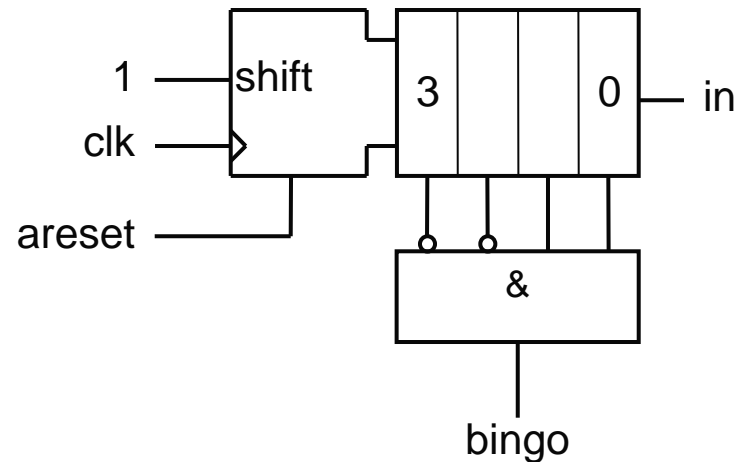
- Ontwerp een machine voor het doorlopend herkennen van het patroon (of reeks) 101. Overlappingsen zijn mogelijk. De machine moet bij herkenning een 1 afgeven.



0010100101011001111  
0000010000101000000

# Herkenningautomaat

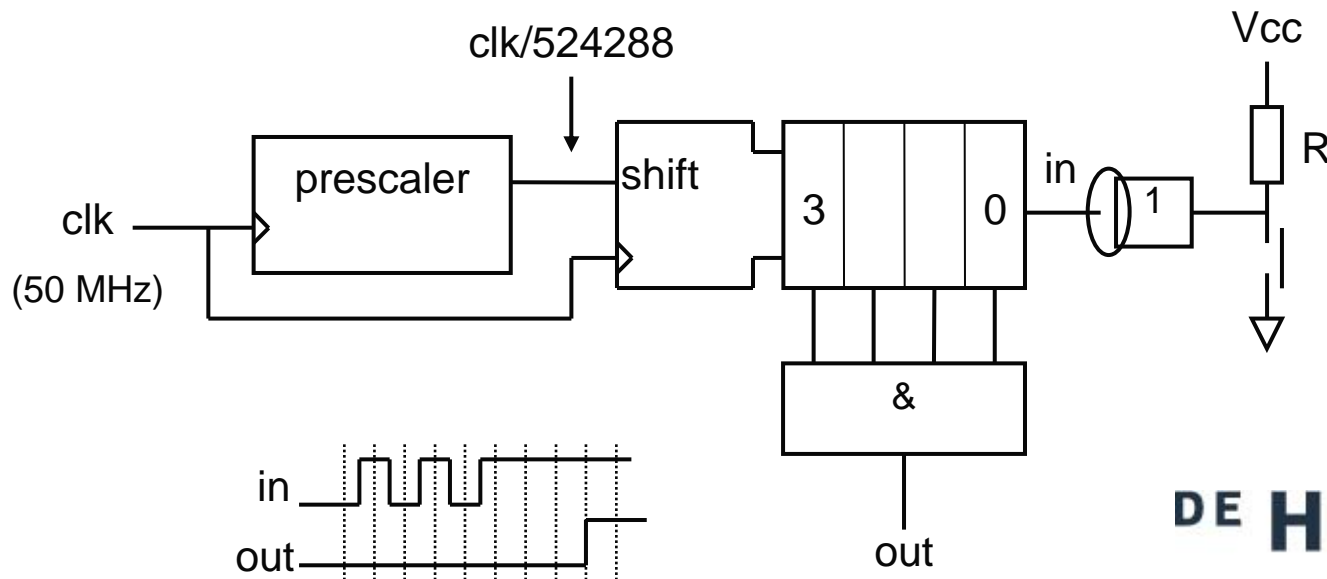
- Een herkenningautomaat kan gemaakt worden met behulp van een *schuifregister* en een AND-poort.
- Als voorbeeld het herkennen van 0011:



- Wat kan er mis gaan bij het aanbieden van 1100?

# Ontdenderen drukknop

- Een mechanische drukknop of schakelaar heeft last van *denderen*. Bij overgaan stuitert de kontaktveer. Dit levert meerdere pulsen op aan een ingang. Een herkenningsautomaat biedt uitkomst.
- Het denderen duurt 10 ms – 40 ms. Een systeemklok is erg snel, dat zou een te groot schuifregister opleveren. Een *prescaler* biedt uitkomst.





# Opgaven

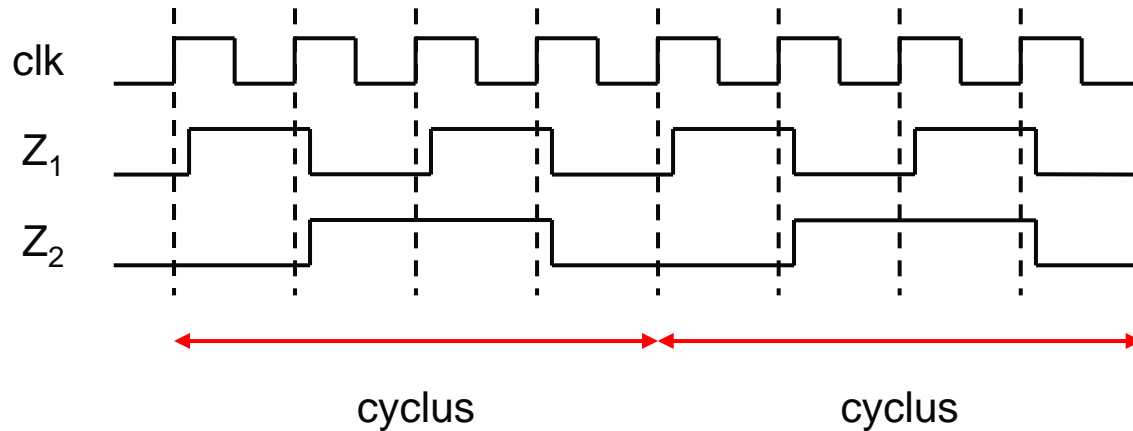
- Ontwerp een machine voor het herkennen van het patroon (of reeks) 1001. De machine geeft een 1 af en stopt met herkennen.
- Ontwerp een machine voor het doorlopend herkennen van het patroon (of reeks) 1001. De machine moet bij herkenning een 1 afgeven. Ontwerp zowel een Moore- als Mealy-machine.
- Ontwerp een Mealy-machine voor het doorlopend herkennen van het patroon (of reeks) 1001. Overlappingsen zijn mogelijk. De machine moet bij herkenning een 1 afgeven. Geef de vergelijkingen voor de opvolgertoestand en de uitgang als gekozen wordt voor de binaire telcode als toestands codering.

# Opgaven

- Ontwerp een Mealy-machine voor het doorlopend herkennen van het patroon (of reeks) 100 met een minimaal aantal toestanden.
- Gegeven de eerdere machine voor het overlappend herkennen van 101. Geef de toestandsfuncties indien one-hot codering wordt gebruikt.
- Eerder is het ontwerp van een machine voor het bloksgewijs herkennen van het patroon (of reeks) 110 besproken. Een studente beweert dat het toestandsdiagram van deze machine compacter kan worden getekend omdat toestand  $s_0$  en  $s_5$  identiek zijn en samengenomen kunnen worden. Klopt deze bewering? Motiveer het antwoord.
- Ontwerp een Mealy-machine voor het bloksgewijs herkennen van het patroon (of reeks) 110 met een minimaal aantal toestanden.

# Timingdiagram

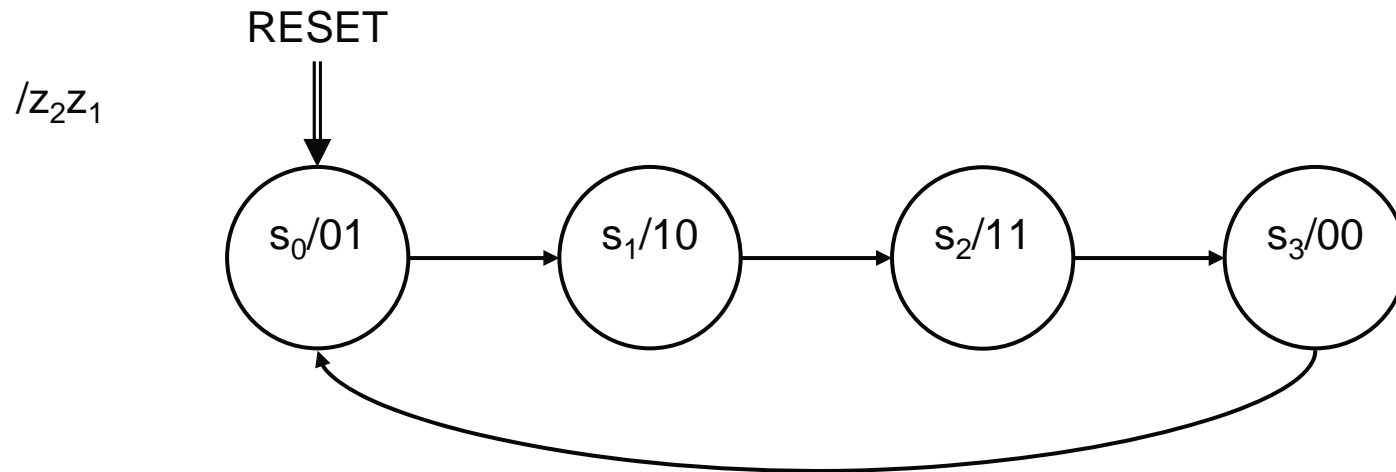
- Een toestandsmachine is ook te beschrijven met behulp van een timingdiagram.



- Wat is het bijbehorende toestandsdiagram?

# Timingdiagram

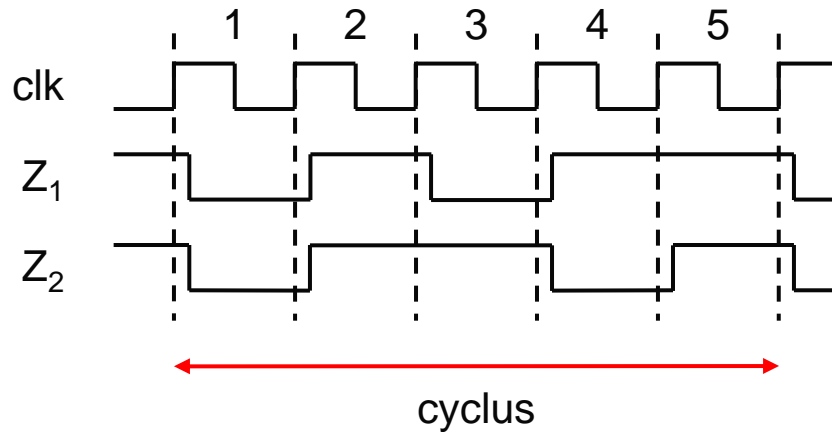
- De waarden van de uitgangen herhalen zich elke vier klokcycli. De machine is dus met vier toestanden te beschrijven. Er is geen ingang.



- Goed beschouwd is dit een *teller*.

# Timingdiagram

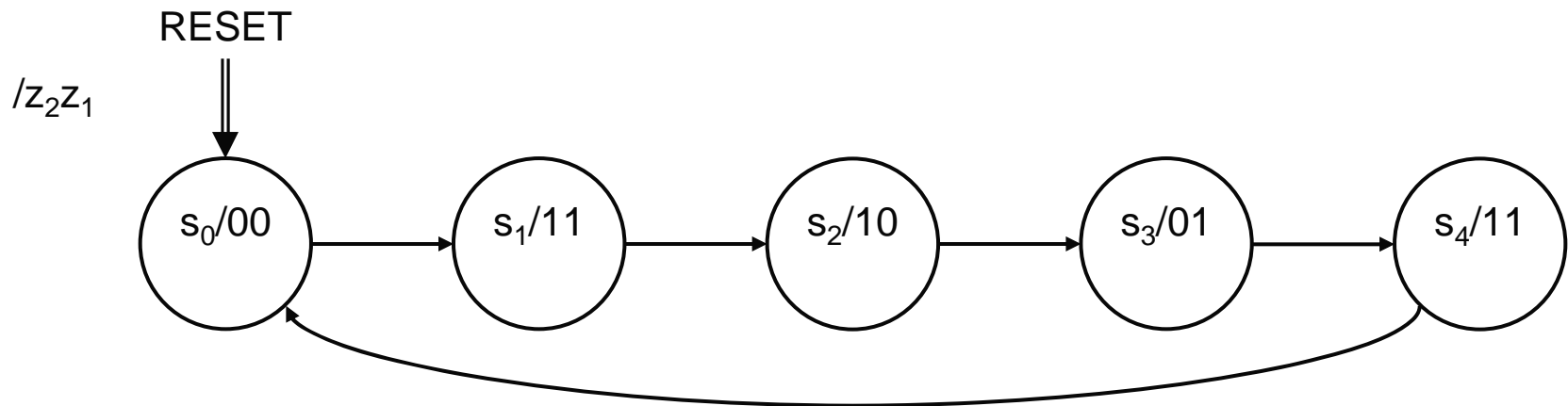
- Hieronder is weer een tijddiagram gegeven.



- Maar nu zijn de uitgangen tijdens twee klokcycli identiek. De uitgangswaarde  $z_2z_1 = 11$  komt twee keer voor.
- Wat is het bijbehorende toestandsdiagram?

# Timingdiagram

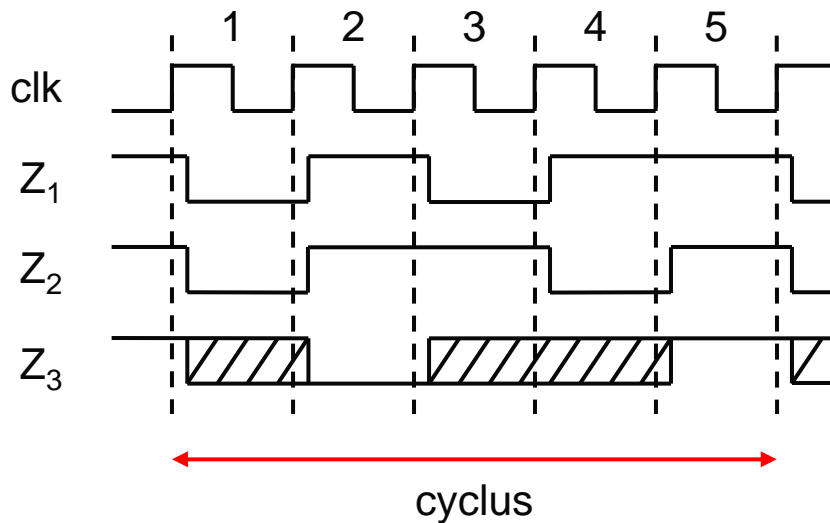
- De waarden van de uitgangen herhalen zich elke vijf klokcycli. De machine is dus met vijf toestanden te beschrijven. Er is geen ingang.



- De machine is te ontwerpen met behulp van een teller en uitgangslogica.

# Timingdiagram

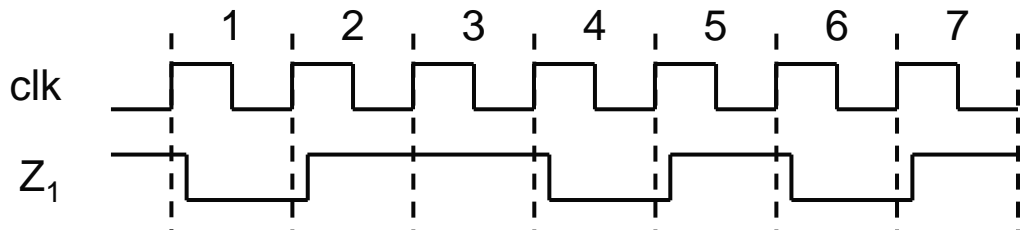
- Door toevoeging van één flipflop is de machine zonder uitgangsglogica te ontwerpen.



- Bij drie van de vijf standen is de waarde van Z<sub>3</sub> niet belangrijk en kan als don't care opgegeven worden, de waarde wordt niet uitgevoerd.

# Opgave

- In de vorige slide is een machine gespecificeerd d.m.v. een timingdiagram. Geef de functies voor de Next State Logic en de Output Logic, de toestands codering ligt gedeeltelijk vast.
- Ontwerp een machine die onderstaande timingdiagram moet uitvoeren met gebruikmaking van extra flipflops en zonder uitgangslógica.



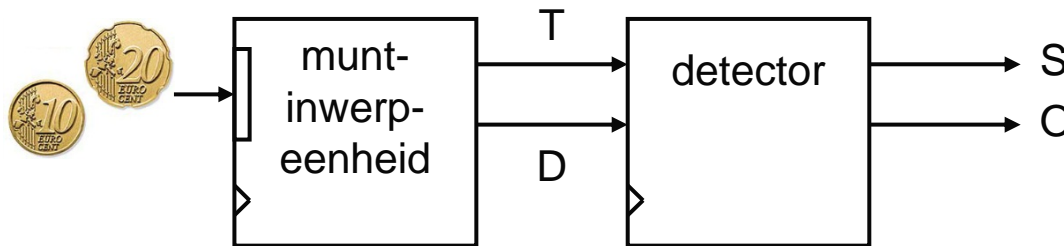


# Snoepautomaat

- In het volgende voorbeeld wordt een machine ontwikkeld die een snoepautomaat bestuurt.
- De snoepautomaat accepteert alleen muntstukken van 20 cent (T) en 10 cent (D).
- Er is 30 cent nodig om snoepgoed vrij te geven.
- Als er meer dan 30 cent wordt ingeworpen moet de machine geen snoepgoed vrijgeven maar aanduiden dat er teveel geld is ingeworpen.

# Snoepautomaat

- De munt-inwerpeenheid geeft twee signalen af, T voor een muntstuk van 20 cent en D voor een muntstuk van 10 cent.

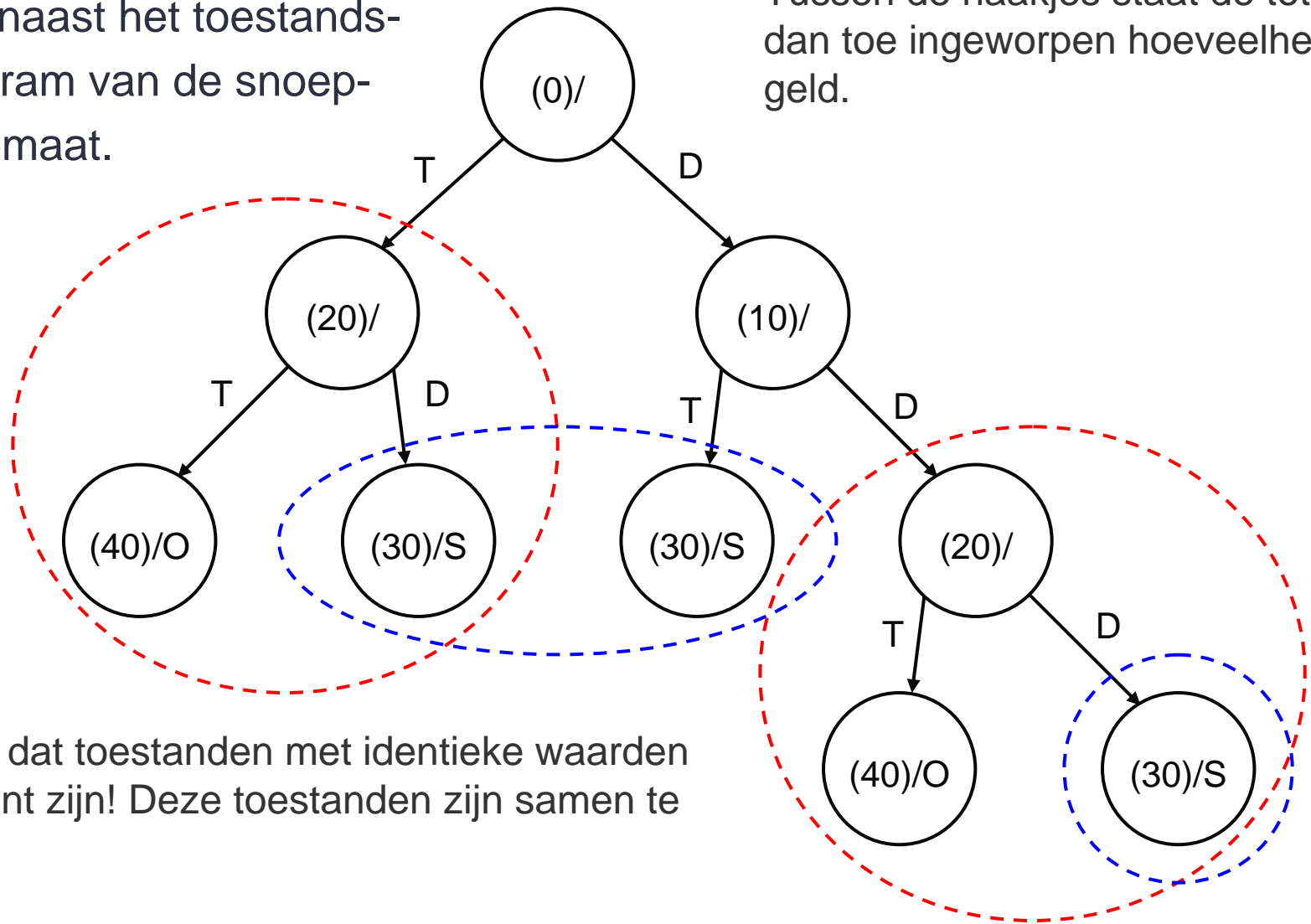


- De eenheid kan slechts één munt te gelijk accepteren. Na detectie van een muntstuk wordt het betreffende signaal één klokcyclus geactiveerd. De signalen T en D kunnen dus nooit tegelijk geactiveerd worden.
- Het signaal S wordt geactiveerd als 30 cent is ingeworpen. Het signaal O wordt geactiveerd als meer dan 30 cent is ingeworpen.

# Toestandsdiagram snoepautomaat

- Hiernaast het toestandsdiagram van de snoepautomaat.

Tussen de haakjes staat de tot dan toe ingeworpen hoeveelheid geld.



Merk op dat toestanden met identieke waarden equivalent zijn! Deze toestanden zijn samen te nemen.

# Snoepautomaat

- Het toestandsdiagram kan met minder toestanden getekend worden door het hanteren van de volgende regels:

$$T + T = 40$$

$$D + D = T$$

$$T + D = 30$$

$$D + T = 30$$

$$D + D + T = 40$$

$$D + D + D = 30$$

- Combinaties die leiden tot de hoeveelheid van 30 zijn dus identiek, net als voor 40. Verder is te zien dat  $10+10 = 20$ . Meer dan 40 komt niet voor, want dan is minstens al een hoeveelheid van 30 ingeworpen.

# Toestandsdiagram snoepautomaat (beter)

- Er wordt gebruik gemaakt van het feit dat  $D + D = T$  en dat  $D + T = T + D$

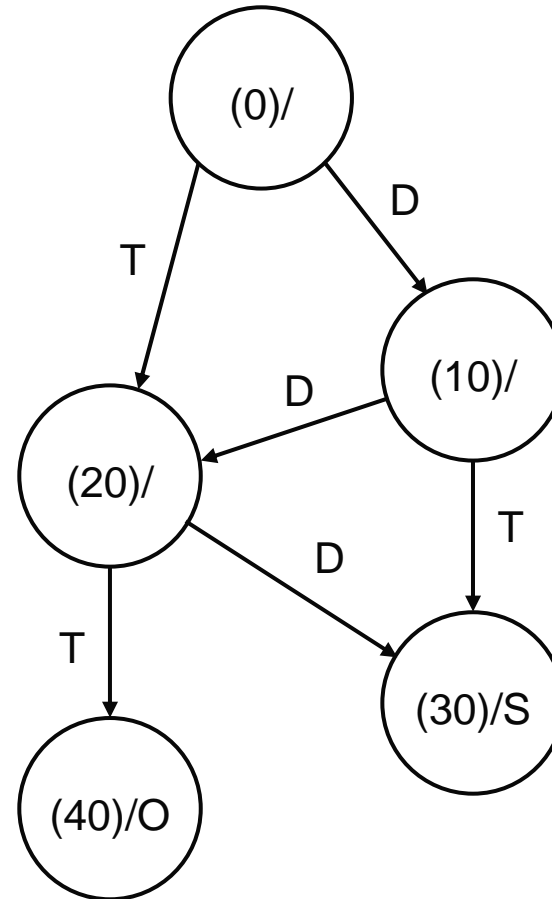
$$D = 10$$

$$T = D + D = 20$$

$$T + D = D + T = 30$$

$$T + T = 40$$

- Het aantal toestanden is verminderd tot 5.



# Opgaven

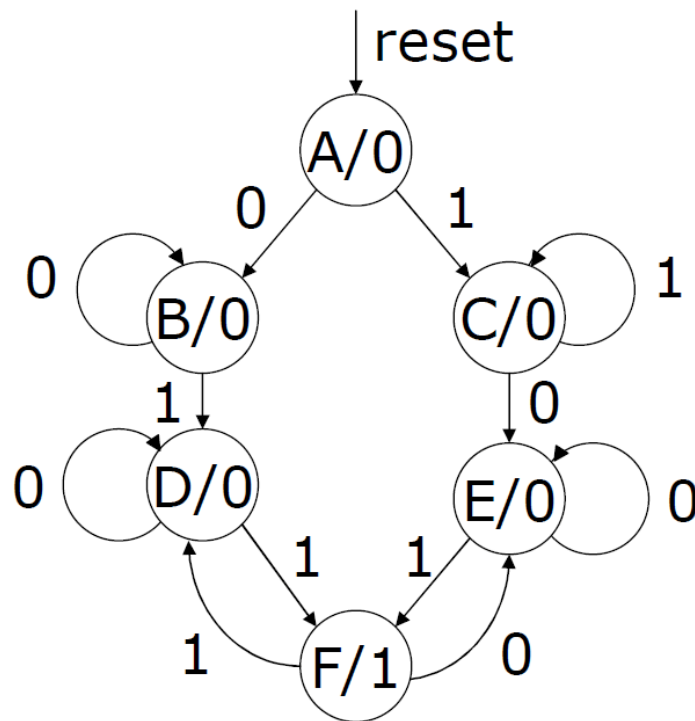
- Ontwerp een toestandsdiagram voor een snoepmachine die 35 cent moet herkennen. Toegestane munten zijn 5 cent (S) en 10 cent (D). Bij dit bedrag moet snoepgoed worden geretourneerd. Bij teveel moet een signalering geactiveerd worden.
- Hoeveel unieke toestanden heeft bovenstaande machine?
- Wat is het hoogste bedrag dat in de machine kan worden ingeworpen (dat is niet het bedrag die de machine uiteindelijk moet herkennen).
- Stel dat een snoepautomaat munten van 5 cent (S), 10 cent (D) en 20 cent (T) accepteert. Wat is het hoogste bedrag dan kan worden ingeworpen? Hoeveel unieke toestanden heeft deze snoepautomaat?

# Toestandminimalisatie

- Net als bij combinatoriek is het mogelijk het aantal toestanden te minimaliseren.
- Het idee is te zoeken naar *equivalente toestanden*.
  - Twee toestanden zijn equivalent indien voor elke mogelijke reeks ingangswaarden dezelfde reeks uitgangswaarden wordt geproduceerd.
- Formele procedure: Paull - Unger (1959)
- In de praktijk wordt dit weinig gebruikt, zeker als het aantal toestanden gering is. Een goede ontwerper zorgt meestal voor een minimale of bijna-minimale oplossing.

# Opgave

- Minimaliseer het aantal toestanden van de volgende machine.



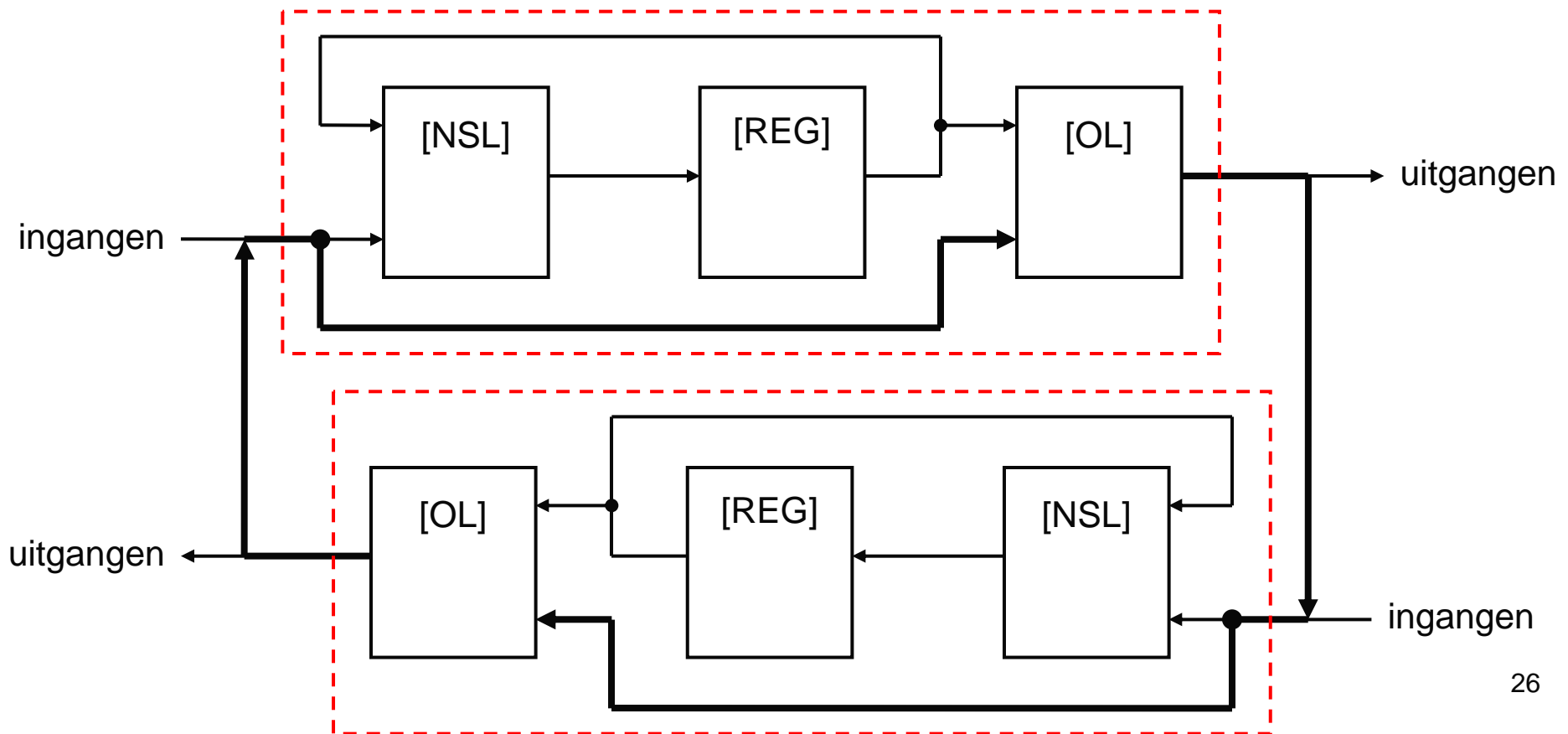


# Tijdgedrag FSM

- De uitgangen van de Mealy-machine veranderen op de actieve klokflank én de ingangen.
- Dat houdt in dat de uitgangen twee sets vertragingen hebben.
  - Ten opzichte van de actieve klokflank (synchroon)
  - Ten opzichte van de ingangen (asynchroon)
- Moore-machines hebben dat probleem niet, de uitgangen veranderen altijd synchroon met de klokflank.
- Het tijdgedrag van synchrone machines kunnen d.m.v. de bekende procedures doorgerekend worden.

# Tijdgedrag FSM

- Rondgekoppelde Mealy-machines kunnen last hebben van asynchrone terugkoppeling. Moore-machines hebben dat probleem niet.



# Referenties

- Digitale Techniek, Deel 2 – A.P. Thijssen ea – 5<sup>e</sup> druk, 2000
- Fundamentals of Digital Logic with VHDL Design – Brown, 3<sup>rd</sup> Ed, 2008, ISBN 9780071268806
- Digital Design: Principles and Practices – John F. Wakerly, 4<sup>th</sup> Ed, 2006, ISBN 0-13-173349-4
- Digitale Systemen, Ontwerpen met behulp van VHDL – W. Dolman, 1<sup>e</sup> druk, 2011, ISBN 978-90-484-2020-9
  
- M. C. Paull and S. H. Unger "Minimizing the number of states in incompletely specified sequential functions", *IRE Trans. on Electronic Computers*, vol. 8, pp.356 - 366 1959, DOI [10.1109/TEC.1959.5222697](https://doi.org/10.1109/TEC.1959.5222697)



Academie voor Technology, Innovation &  
Society Delft  
Academie voor ICT & Media

De Haagse Hogeschool, Delft  
+31-15-2606311  
J.E.J.opdenBrouw@hhs.nl  
www.dehaagsehogeschool.nl

**DE HAAGSE**  
HOGESCHOOL