

Digitale Techniek

van probleemstelling tot realisatie

deel 2

Digitale Techniek

van probleemstelling tot realisatie

deel 2

A.P. Thijssen, H.A. Vink

5e geheel herziene druk

© VSSD

Eerste druk 1982

Vijfde druk 1999-2008

Uitgegeven door de VSSD

Leeghwaterstaat 42, 2628 CA Delft, The Netherlands

tel. +31 15 278 2124, telefax +31 15 278 7585, e-mail hlf@vssd.nl

internet: www.vssd.nl/hlf

URL over dit boek: www.vssd.nl/hlf/e026.htm

Alle rechten voorbehouden. Niets uit deze uitgave mag worden verveelvoudigd, opgeslagen in een geautomatiseerd gegevensbestand, of openbaar gemaakt, in enige vorm of op enige wijze, hetzij elektronisch, mechanisch, door fotokopieën, opnamen, of op enige andere manier, zonder voorafgaande schriftelijke toestemming van de uitgever.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher.

ISBN 978-90-407-1838-0

NUR 959

Trefwoord: digitale techniek

Voorwoord

Met het verschijnen van deze vijfde druk van deel 2 van het boek “Digitale Techniek, van probleemstelling tot realisatie” is deze serie weer compleet. Hetgeen ons voor ogen stond bij deze herziening is uitstekend verwoord in het voorwoord van deel 1. In dit tweede deel zijn dezelfde doelstellingen nagestreefd: een duidelijke en didactisch verantwoorde inleiding in het vakgebied van de Digitale Techniek, gezien vanuit een praktisch standpunt. Geschiktheid voor zelfstudie was hierbij een der vanzelfsprekende uitgangspunten.

De mogelijkheden die de digitale techniek ons biedt lijken onbeperkt. De aantallen componenten per chip zijn de laatste jaren omhoog geschoten, in grote wedijver met de klokfrequentie van processoren en ASIC's. De prijs van de componenten doet het omgekeerde. Kortom, een echt succesverhaal. Toch verloopt een en ander in de praktijk niet probleemloos, onder meer te zien aan stickers die mobiele telefoons verbieden in een omgeving met storingsgevoelige apparatuur of apparatuur waarvan mensenlevens kunnen afhangen.

Waarom is deze apparatuur zo storingsgevoelig? Een van de oorzaken is het feit dat externe elektromagnetische velden invloed kunnen hebben op de werking van digitale schakelingen. Daarom zijn er EMC eisen geformuleerd, waaraan apparaten moeten voldoen ten aanzien van hun invloed op apparatuur in hun omgeving. Meestal tracht men dit op te lossen door de schakeling of een heel apparaat af te schermen, ofwel in te blikken. Dit inblikken lost het probleem aan de buitenkant op. Wat blijft is dat werkende schakelingen ook invloed op zichzelf hebben en dat los je niet op door extern blik. In dit deel wordt daarom veel aandacht geschonken aan de potentiële storingsgevoeligheid van digitale schakelingen. Zo worden er richtlijnen geformuleerd voor het ontwerp van geheugenelementen ten einde de intervallen, waarin de in te lezen data beschikbaar moet zijn, te minimaliseren. Het afvangen van redundantie, teneinde onvoorspelbaar gedrag na een storing te vermijden, past ook in het streven naar een lagere storingsgevoeligheid.

Tijdens cursussen wordt door ons vaak de vraag gesteld wat er met een bepaalde maatregel opgelost wordt. Het blijkt vervolgens veel lastiger te zijn ook te formuleren wat er niet opgelost wordt. Zo kan met een lagere voedingsspanning de dissipatie sterk verminderd worden. Toch kent een lagere voedingsspanning ook nadelen, waarvan de lagere DC/AC noise margins de belangrijkste is. Nu kan de dissipatie ook verminderd worden door de klok van die delen af te schakelen, die op dat moment niet gebruikt worden. Het bekritisieren van bepaalde oplossingen en/of ontwerp-stijlen en het genereren van alternatieven krijgt in dit boek veel aandacht. Het directe doel hiervan is het aankweken van een kritische ontwerp-houding. Ervaring bij verschillende bedrijfs cursussen heeft geleerd dat we zonder meer durven te stellen dat dit een van de sterke punten van dit boek is.

Vanaf het begin van de Digitale Techniek heeft men geprobeerd bestaande (deel)oplossingen opnieuw te gebruiken. Nu was de complexiteit van de modules in het begin zeer gering en de externe eigenschappen van zo'n module konden gemakkelijk uit het schema worden afgeleid. Naarmate de module-omvang echter toeneemt schiet deze benadering te kort. De ontwerper wil slechts de externe eigenschappen van de module kennen en kan deze niet meer afleiden uit een gedetailleerd schema op poortniveau. Daartoe moet het nodige gestandaardiseerd worden ten aanzien van de interfacebaarheid van afzonderlijke modules. Dit deel van het vakgebied is tot nu toe sterk onderbelicht. Het lijkt wel of de meeste ontwerpers niet verder willen of kunnen kijken dan de aansluitpunten van het eigen ontwerp. Deze tekst draagt op veel plaatsen bij om de discussie over modulair ontwerpen op een hoger niveau te brengen.

De volgende vraag moge dienen om deze problematiek toe te lichten. “Op welk niveau documenteer je een standaardmodule?” Kun je dit bijvoorbeeld doen op VHDL niveau, middels de specificatie van de module in VHDL waarvan gebleken is dat deze succesvol te converteren is in een concreet ontwerp? Ervaring met bijvoorbeeld programmable gate arrays leert dat het uiteindelijke resultaat in termen van oppervlak en maximale frequentie sterk afhankelijk is van de volgorde van compileren van de VHDL beschrijving. We zien een grote invloed op de uiteindelijke parameters van de resulterende schakeling.

Aan de lezer bieden deze boeken een eigentijdse en relevante introductie in het vakgebied van de Digitale Techniek. De behandelde stof is ruim voldoende voor de gemiddelde ontwerper van digitale schakelingen. Er zit voldoende diepgang in om discussies over relevante onderwerpen zoals timing op een hoog niveau te kunnen voeren. Na afloop van bedrijfs cursussen horen we vaak van cursisten bij de evaluatie dat ze nu “kunnen meepraten met de oude rotten in het vak”. Dat stemt tot veel voldoening. Maar “rust roest”. Daarom wordt commentaar van de gebruiker blijvend op prijs gesteld. We gaan daar gaarne op in.

Voor een eerste introductie cursus bevat dit boek te veel. Achterin staat een studie-advies voor een eerste ronde op TH-niveau. We hebben geprobeerd de inhoud van de eerdere drukken vrijwel integraal op te nemen, los van de uitbreidingen. Alleen geheugenwerking heeft meer aandacht gekregen, vanwege het grote belang van dit onderwerp in relatie tot timing en de daarop gebaseerde robuustheid van de resulterende schakeling. Wel is overall de tekst aangepast en verbeterd. Docenten kunnen daarom desgewenst probleemloos overschakelen van de (voorlopig nog verkrijgbare) derde druk naar deze vijfde druk. Binnen dezelfde studiebelasting blijken studenten veel meer aan te kunnen dan met eerdere drukken.

Aangevuld met een goede cursus VHDL, een cursus IC-technologie en een cursus Computerarchitectuur en Embedded Systems biedt deze cursus Digitale Techniek een uitstekende basis voor een loopbaan in het vakgebied van de IC-ontwerper. Vanuit de hier aangeboden cursus Digitale Techniek kunt u ook de discussie over wat VHDL voor u kan oplossen en wat niet, op een zinvolle wijze voeren. We wensen u toe dat uw cursussen VHDL en computerarchitectuur ook aan dit criterium voldoen.

Delft/Pijnacker en Eindhoven/Valkenswaard
augustus 2000
Loek Thijssen en Hans Vink

Stelling

In veel leerplannen is het vak Digitale Techniek onderbelicht. Uit het feit dat de basisbegrippen uit de Digitale Techniek, nullen en enen, zo eenvoudig zijn mag niet de conclusie getrokken worden dat Digitale Techniek een gemakkelijk vak is.

Inhoudsopgave

9 Geheugenelementen 1: latches

9.1	Geheugenwerking	13
	<i>Realisatie van geheugenwerking</i>	15
	<i>Onbedoelde geheugenwerking</i>	16
9.2	Het specificeren van geheugenwerking	17
	<i>Toestandsdiagrammen</i>	18
9.3	Het logisch ontwerp van geheugenelementen	19
	<i>Realisatie met AND en OR poort</i>	21
	<i>Realisatie met NAND's</i>	21
	<i>Realisatie met NOR's</i>	22
9.4	Symbolen voor latches	23
9.5	Functietabellen voor latches	24
9.6	Timing parameters van latches	26
	<i>Het data transfer model</i>	26
	<i>Duur van de set/reset opdracht</i>	29
	<i>Storingsgevoeligheid</i>	29
9.7	De ingangscombinatie SR = 11	30
9.8	Opdrachtcoderingen en overgangsverschijnselen	32
9.9	Gated latches	34
	<i>Gated S-R latches</i>	34
	<i>Gated D latches</i>	35
	<i>Functietabellen van gated latches</i>	36
9.10	Timing parameters van gated latches 1	37
	<i>Timing van de gated D latch</i>	38
	<i>Samenvatting timing van de gated D latch</i>	40
9.11	Timing parameters van gated latches 2	42
	<i>Timing van de gated S-R latch</i>	42
	<i>Referentiesignalen voor de timing</i>	44
9.12	Toepassing: de anti-denderschakeling	44
9.13	Latches en gated latches in transistoren	47
	<i>Statische en dynamische geheugenwerking</i>	47
	<i>Gated latches in NMOS</i>	49
	<i>Gated latches in CMOS</i>	50
9.14	RAM geheugens	52
	<i>Dynamische RAM's</i>	57
	Literatuur	57
	Opgaven hoofdstuk 9	58

10 Geheugenelementen 2: flip-flops

10.1	Het meester-en-slaaf principe	69
	<i>Het meester-en-slaaf principe</i>	74
10.2	De D flip-flop	77
10.3	De geheugenwerking van de D flip-flop	79
	<i>De level mode beschouwingwijze</i>	80
	<i>De clock mode beschouwingwijze</i>	81

	<i>Symbolen</i>	84
10.4	Instelsignalen voor D flip-flops	85
10.5	De J-K flip-flop	87
	<i>De logische werking van de J-K flip-flop</i>	90
	<i>Symbolen voor pulse-triggered flip-flops</i>	92
10.6	Overige logische types flip-flops	93
10.7	Flip-flops met enable faciliteiten	95
10.8	Flip-flop timing 1	96
	<i>Kloksignalen</i>	96
	<i>Timing van datasignalen</i>	97
	<i>Timing van instelsignalen</i>	99
10.9	Flip-flop timing 2	100
	<i>Het effect van een terugkoppeling van uitgang Q naar de ingang</i>	102
10.10	Flip-flops in transistoren	104
	<i>Ingangscondities</i>	105
	<i>Het intern overnemen en doorgeven van data</i>	108
	<i>Storingsgevoeligheid van de uitgangsconstructie</i>	108
	<i>De uitgangsreactie</i>	110
	<i>Samenvatting timing van flip-flops</i>	110
	Literatuur	111
	Opgaven hoofdstuk 10	111

11 Het sequentiële machine model

11.1	Sequentiële schakelingen	125
	<i>Het ontwerptraject van clock mode schakelingen</i>	125
	<i>Van toestandsdiagram naar schakeling</i>	128
11.2	Het model voor combinatorische schakelingen	131
11.3	Het model voor sequentiële schakelingen	133
	<i>Don't cares</i>	137
	<i>Inpassing van combinatorische schakelingen</i>	137
11.4	Het Mealy en het Moore model	138
	<i>Afspraken voor toestandstabellen en diagrammen</i>	139
	<i>Notatie-afspraken voor de clock mode</i>	142
11.5	Clock mode FSM realisatiestructuren	142
	<i>Dynamische aspecten van Moore type schakelingen</i>	147
11.6	Algorithmic State Machines	148
	Literatuur	152
	Opgaven hoofdstuk 11	153

12 Level mode sequentiële schakelingen

12.1	Introductie level mode sequentiële schakelingen	155
	<i>De level mode</i>	155
	<i>Beperkingen aan de toepassing van latches</i>	156
	<i>Toestandscoderingen</i>	157
12.2	Het ontwerp van een gated D latch	160
	<i>NAND realisatie van het geheugenelement</i>	161
	<i>NOR realisatie van het geheugenelement</i>	163

<i>Overlappende termen</i>	164
<i>Een vergelijking met set-reset latches</i>	165
12.3 Het ontwerp van een scan flip-flop	165
<i>Ontwerp 1: Flip-flop met een selector ervoor</i>	166
<i>Ontwerp 2: Two-port flip-flop</i>	166
12.4 Het schakelen van kloksignalen	170
<i>Level mode edge-triggered klokschakelaars</i>	172
<i>Het omschakelen tussen verschillende klokken</i>	174
12.5 De level mode interpretatie van het sequentiële machine model	175
<i>Afspraken voor de level mode</i>	176
Literatuur	178
Opgaven hoofdstuk 12	179
13 Clock mode sequentiële schakelingen	
13.1 Introductie clock mode sequentiële schakelingen	185
<i>Spelregels voor de clock mode</i>	186
13.2 Ontwerpvoorbeeld: een druktoetsinterface	188
<i>De realisatie</i>	196
13.3 Ontwerpvoorbeeld: een reeksgenerator	197
13.4 Het opstellen van toestandsdiagrammen	200
13.5 Equivalentie van toestanden	202
13.6 Overbodige toestanden	205
13.7 Resetten	208
13.8 De keuze van de toestandscodering	210
<i>Het aantal toestandscoderingen</i>	211
<i>Het benutten van structuur in de probleemstelling</i>	211
<i>One-hot assignments</i>	212
<i>Combinatie van toestandscodering en uitgangscodering</i>	213
<i>Recente ontwikkelingen</i>	213
13.9 De clock mode interpretatie	214
<i>Afspraken voor de clock mode</i>	214
13.10 De afbeelding op hardware	218
<i>Programmeerbare logica</i>	219
<i>Testbaarheid en observeerbaarheid</i>	221
Literatuur	222
Opgaven hoofdstuk 13	224
14 Registers	
14.1 De interne structuur van registers	237
<i>De structuur in de opbouw</i>	239
<i>Overbodige functies</i>	241
14.2 Enkele ontwerpdetails	242
<i>Invloed clock buffer</i>	242
<i>Implementatie inhibit functie</i>	243
<i>Implementatie van de selectoren/multiplexers</i>	245
<i>Implementatie van de set- en resetfunctie</i>	246
14.3 Toepassingen van registers	248

	<i>vermenigvuldigen en delen</i>	248
	<i>Tellen met registers tot n</i>	249
	<i>Tellen met schuifregisters tot 2n</i>	251
	<i>Tellen bij parallel-serie omzetting</i>	253
14.4	Modulo 2 teruggekoppelde registers	254
	<i>Getallen modulo m</i>	254
	<i>Modulo 2 terugkoppeling van de registerinhoud</i>	255
14.5	Terugkoppelpolynomen	257
14.6	Pseudo-random eigenschappen van maximum-lengte reeksen	259
	<i>Registerconfiguraties</i>	260
14.7	Signature analysers	263
	Literatuur	266
	Opgaven hoofdstuk 14	266

15 Tellers

15.1	Inleiding	273
	<i>De modulaire structuur van tellers</i>	274
15.2	Asynchrone binaire tellers	275
15.3	Synchrone binaire tellers	278
15.4	Synchrone binaire op/neer tellers	282
	<i>Omschakelbare op/neer tellers</i>	283
15.5	Tellers met enable ingang	284
	<i>Opbouw van het carry netwerk</i>	285
	<i>De enable implementatie</i>	287
15.6	4n-Bit tellers	288
15.7	Standaard 4-bit tellers 1	292
	<i>Functietabellen</i>	292
15.8	Standaard 4-bit tellers 2	295
	<i>De 4-bit binaire teller SN74LS163A</i>	295
	<i>Timing eigenschappen</i>	297
	<i>Lusvorming</i>	297
	<i>De 4-bit op/neer teller SN74LS169A</i>	298
	<i>Functietabellen</i>	300
	<i>De 4-bit BCD op/neer teller SN74LS668</i>	301
15.9	Parameterspecificatie van 4n-bit tellers	303
	<i>Structuuronafhankelijke parameters</i>	303
	<i>Structuurafhankelijke parameters</i>	303
15.10	Pseudo clock mode inpassing van tellers	306
	<i>Pseudo clock mode synchrone tellers</i>	307
	Literatuur	309
	Opgaven hoofdstuk 6	310

16 Het Datapad - besturing ontwerpmodel

16.1	Datapad en besturing	319
	<i>Notatie-afspraken</i>	320

16.2	Ontwerpvoorbeeld: digitale ‘one shot’	322
	<i>De architectuur van de schakeling</i>	324
	<i>De implementatie</i>	326
	<i>De realisatie</i>	329
16.3	Spelregels ten aanzien van het ontwerpen volgens het datapad-besturing model	329
16.4	Ontwerpvoorbeeld: instelbare spanningsgenerator	331
	<i>Het programma van eisen</i>	331
	<i>De architectuur van de schakeling</i>	332
	<i>De implementatie</i>	333
16.5	Vervolg ontwerpvoorbeeld	337
	<i>De realisatie</i>	337
	<i>De definitieve besturingspecificatie</i>	343
16.6	Moore - Mealy en andere perikelen	346
	<i>Proces- en klokfase-synchronisatie</i>	347
	<i>Variabele reactietijden</i>	352
16.7	De besturingsopzet voor parallelle processen	353
	<i>Snelle deelprocessen</i>	354
16.8	Van toestandsdiagram naar schakeling	355
	<i>Realisatie in PLA's, PAL's of PROM's</i>	356
	Literatuur	358
	Opgaven hoofdstuk 16	359

17 Timing bij data - overdracht

17.1	Uitgangspunten	369
	<i>Samenvatting timing modellen</i>	369
17.2	Directe data-overdracht tussen identieke flip-flops	371
	<i>Directe data-overdracht</i>	371
	<i>Kwaliteit van een flip-flop ontwerp</i>	374
	<i>Keuze voor een snelle of een langzame technologie</i>	374
17.3	Directe data-overdracht tussen flip-flops met verschillende timing parameters	375
	<i>Conclusies directe data-overdracht</i>	377
17.4	Indirecte data-overdracht	378
	<i>Conclusies indirecte data-overdracht</i>	381
17.5	De maximale klokfrequentie	381
	<i>Conclusies maximale klokfrequentie</i>	383
	<i>Maximale klokfrequentie bij pulse-triggered timing</i>	383
	<i>Toegestane propagatietijd in de logica</i>	384
17.6	Data-overdracht tussen flip-flops met verschillende timing systemen	385
	<i>Data-overdracht tussen positive en negative edge-triggered flip-flops</i>	385
	<i>Data-overdracht tussen edge-triggered en pulse-triggered flip-flops</i>	388
	<i>Conclusies data-overdracht bij verschillende timing systemen</i>	389
17.7	Flip-flop ontwerpen en marges voor clock skew	390
	<i>Flip-flop ontwerp en maximale klokfrequentie</i>	390
	<i>Marges voor clock skew en maximale klokfrequentie</i>	393
	<i>Conclusies voor het ontwerp van flip-flops</i>	393

17.8	Interne en externe timing parameters van MSI componenten	394
	<i>Logica rond flip-flops</i>	394
	<i>Signaalpropagatie door combinatorische logica</i>	395
	<i>Interne en externe setup en hold time</i>	396
	<i>Logica in de kloklijn</i>	399
	<i>Uitgangsparementers</i>	401
	<i>Samenvatting</i>	401
17.9	Data lockout flip-flops	402
	<i>Edge-triggered flip-flops met data lockout</i>	402
	<i>Data lockout timing en boundary scan</i>	403
17.10	Tweefasige klokken	404
17.11	Retiming	406
	<i>Retiming en pipelining</i>	409
	<i>Retiming en redundante toestanden</i>	411
	<i>Conclusies retiming</i>	412
17.12	Samenvatting en conclusies	413
	<i>Meten of simuleren</i>	413
	Literatuur	415
	Opgaven hoofdstuk 17	416

18 Het kloksysteem

18.1	Taken van het kloksysteem	431
	<i>De begrippen elektrisch en logisch gedefinieerd</i>	431
	<i>De opzet van een eenvoudig kloksysteem</i>	433
18.2	Klokgeneratoren en hun parameters	435
18.3	Klokkrequentie en dissipatie	437
18.4	Klok distributie	440
18.5	Klok buffers	441
	<i>Maatregelen te verbetering van de interface-baarheid</i>	443
18.6	Synchronisatie van externe signalen	445
18.7	Procesfase-synchronisatie	451
	Literatuur	453
	Opgaven hoofdstuk 18	454

Appendix Symbolen 461

Deze appendix is niet in het boek opgenomen, maar te raadplegen op URL
<http://www.vssd.nl/hlf/e025.html>

Antwoorden van de opgaven 463

Studie-advies 464

Index 465

9

Geheugenelementen 1: latches

9.1 Geheugenwerking

In de eerstvolgende hoofdstukken houden we ons bezig met het ontwerpen van sequentiële schakelingen, dat zijn schakelingen met geheugenwerking. We beginnen met een bespreking van geheugenelementen, de componenten die geheugenwerking mogelijk maken. Er zijn twee types geheugenelementen, latches en flip-flops. De eigenschappen van beide types verschillen sterk. Daarom behandelen we ze in twee aparte hoofdstukken. Daarna komen enkele hoofdstukken die het ontwerpen van sequentiële schakelingen toelichten.

In de statische toestand wordt het uitgangssignaal van een combinatorische schakeling geheel bepaald door de aangeboden ingangscombinatie. Bij sequentiële schakelingen is dit niet zo. *Sequentiële schakelingen hebben als kenmerk dat bij tenminste één ingangscombinatie het uitgangssignaal (of een of meer van de uitgangssignalen) in de statische toestand verschillende waarden kan bezitten.* De huidige ingangscombinatie legt het uitgangssignaal dan niet volledig vast. Ook enkele van de in het verleden aangeboden ingangscombinaties, alsmede de volgorde waarin ze aangeboden zijn, beïnvloeden de waarde van het huidige uitgangssignaal. Dit is alleen mogelijk als een sequentiële schakeling *geheugenwerking* bezit. Om enig inzicht in geheugenwerking te krijgen werken we het volgende voorbeeld uit.

Voorbeeld

De schakeling in figuur 9.1.a is opgebouwd uit twee rondgekoppelde NOR poorten. De externe ingangen zijn aangeduid met S (*setingang*) en R (*resetingang*). De uitgang van de schakeling is aangeduid met Z.

Bij de ingangscombinatie $SR = 10$ liggen alle signaalwaarden in de schakeling vast (nadat de schakeling voldoende tijd heeft gehad om zich in te stellen). De uitgang Z neemt daarbij de waarde $Z = 1$ aan. Zie figuur 9.1.b. Wordt nu na enige tijd de ingangscombinatie $SR = 00$ aangeboden, dan houdt de uitgang de waarde $Z = 1$. Immers, de uitgang van poort B zet op de bovenste ingang van poort A de signaalwaarde 1. Deze 1-waarde is *dominant* over de signaalwaarde op de andere ingang S van de NOR poort A. Het 0 worden van S heeft daarom geen invloed op de uitgangswaarde $Z = 1$. De schakeling blijft in de in figuur 9.1.c beschreven toestand staan.

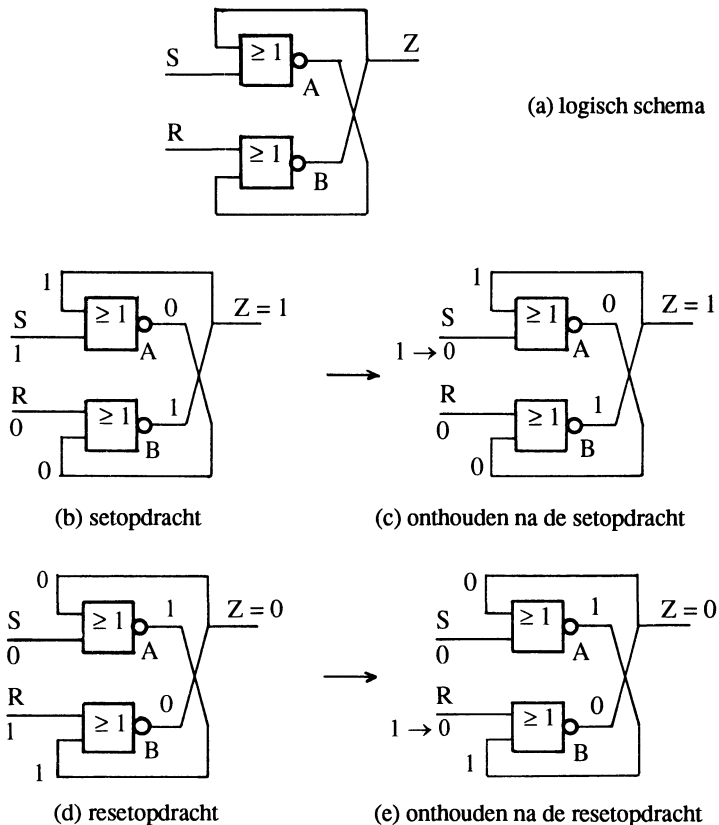
Ook het aanbieden van $SR = 01$ legt alle signaalwaarden vast. Nu wordt of blijft de uitgang $Z = 0$. Wordt vervolgens de ingangscombinatie $SR = 00$ aangeboden, dan blijft $Z = 0$. Hiervoor geldt eenzelfde redenatie als voor $Z = 1$. Zie figuur 9.1.d/e. Bij $SR = 00$ heeft de beschreven schakeling dus twee mogelijke uitgangswaarden, $Z = 0$ en $Z = 1$ (statische

toestand). Bij $SR = 00$ vertoont de schakeling *geheugenwerking*. Welke uitgangswaarde optreedt hangt van de vorige ingangscombinatie af:

$SR = 10$ maakt $Z = 1$ \leftrightarrow setcombinatie

$SR = 01$ maakt $Z = 0$ \leftrightarrow resetcombinatie

De schakeling in figuur 9.1 kan dus gebruikt worden om geheugenwerking te realiseren.



Figuur 9.1. Schakeling met geheugenwerking.

We hebben nog niet gezegd wat we met de ingangscombinatie $SR = 11$ kunnen doen. In paragraaf 9.7 zullen we motiveren dat deze combinatie beter niet toegepast kan worden. Voorlopig verbieden we daarom het toepassen van de ingangscombinatie $SR = 11$. \square

Naamgeving van signalen

In het bovenstaande voorbeeld is er sprake van een *setopdracht* (gecodeerd/gegeven met de ingangscombinatie $SR = 10$), van een *resetopdracht* ($SR = 01$) en van een *onthoudopdracht* ($SR = 00$). Hiervan afgeleid noemt men hetingangssignaal S wel het *setsignaal* en R het *resetsignaal* van het geheugenelement. Deze namen zijn verwarrend, omdat in feite de