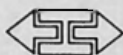


1/0

• VIVAT

Orgaan van de
studievereniging Informatica

INTER-ACTIEF



jaargang 2, nummer 6

GEMISTE KANIS

**ALS JE ALS BIJNA AFGESTUDEERDE ELEKTRONICUS NOG NIET HEBT
GESPROKEN MET AT&T EN PHILIPS TELECOMMUNICATIE: 06-022 2762**

Orgaan van de
studievereniging
informatica Interactief.

TW-gebouw A231
Technische Hogeschool
Twente
Postbus 217
7500 AE Enschede.

Redactie.

Marin de Mie
Hanke Niestern
Peter Schnitger
Clazien Wezeman

Medewerkers.

Anco Smit
Paul Oude Luttighuis
Marcel van Voelen

Kopij.

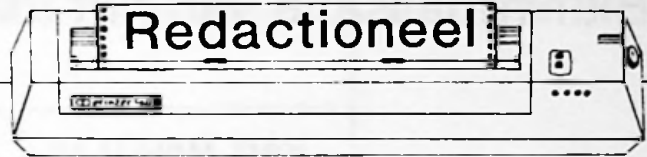
In de kopijbus bij de
Interactiefkamer of
sturen naar de redactie
van I/O Vivat, TW A-231,
THT, Postbus 217,
7500 AE Enschede.

Tel. I/A kamer: 3756
Tel. hoofdred.:
053-895105 (fin)
053-339270 (red)

Het lidmaatschap van
Interactief kost per
studiejaar 15 gulden.

Advertenties:

P. Schnitger,
Campulaan 67-215,
tel. 053-895105.



Daar is dan weer nummer zes van jullie zo geliefde lijfblad I/O Vivat. Het laatste nummer voor de vakantie, de tijd schiet voort. Tentamens komen en gaan en de vakantie staat voor de deur. De tijd van hitte en grote verveling is aangebroken. Om hierin enige verlichting aan te brengen, publiceren wij in dit nummer een heel interessant artikel van de hand van Bert Molenkamp, medewerker van de vakgroep CAP. Vooral voor degenen die het vak 'Uitrusting van digitale systemen' hebben gedaan is dit een erg boeiend artikel.

Namens de redactie wil ik iedereen die dit studiejaar een bijdrage heeft geleverd aan I/O Vivat zeer hartelijk bedanken.

Verder wil ik nog een grote blijk van waardering uitspreken voor het vele werk dat in dit studiejaar door het bestuur, de diverse commissies en alle overige actieve leden is verricht. Zij hebben zich allen bijzonder goed ingezet voor de vereniging, o.a. door het organiseren van een groot lustrum, vele borrels en excursies. Hulde!!!

Tot slot wens ik iedereen een hele prettige vakantie en veel succes bij eventuele tentamens!

Clazien

Inhoud

2 Bitminimalisatie van het microgeheugen.
auteur: Bert Molenkamp.

15 Prijspuzzel

15 Wist-u-dat. (AGI)

16 Het lijkt op de campus (slot). E. Kwien.

Bitminimalisatie van het microgeheugen

Bert Molenkamp



1. Bert Molenkamp heeft in juni 1981 zijn HTS-Elektrotechniek studie afgerond. Zijn eindopdracht, "het zoeken naar een testmethode voor magnetische logica verwerkings-modules", heeft hij uitgevoerd bij Holec Control Systems te Hattem.

In maart 1986 heeft hij zijn TH-Elektrotechniek studie afgerond. De D-opdracht is uitgevoerd bij de vakgroep CAP en had als onderwerp "minimalisatie van het microgeheugen".

Sinds 1 april 1986 is hij werkzaam bij deze vakgroep en zal zich toeleggen op het testen van VLSI (Very Large Scale Integration) - schakelingen.

2 Inleiding

In de literatuur zijn veel artikelen te vinden over het minimaliseren van microgeheugens. Agerwala [AGERWALA76] geeft een overzicht van de verschillende methodes om te komen tot een minimale oplossing. Er wordt meestal uitgegaan van een microgeheugen dat alleen gevuld is met de logische niveaus hoog en laag. Toch kunnen er veel niveaus don't care zijn. eerst zal worden aangegeven hoe de don't care's kunnen ontstaan, hiervoor is het nodig de opbouw van het microgeheugen te kennen.

Het microgeheugen kan op vele manieren zijn opgebouwd, we beperken ons tot de opbouw geschetst in figuur 2.1. Het microgeheugen bestaat uit een verzameling van microwoorden en een microwoord is vervolgens weer opgebouwd uit een testveld, adresveld en poort- en inleessignalen. Het testveld is gecodeerd zodat slechts op één testconditie kan worden getest, poort- en inleessignalen zijn niet gecodeerd [BLAAUW76]. Is het huidige microadres adres i dan wordt bijvoorbeeld getest op het gelijk nul zijn van het resultaat van een optelling. Is dit resultaat nul dan wordt vervolgd met adres j , in alle andere situaties wordt adres $i+1$ als het volgende microadres genomen. Indien in het testveld geen testvoorwaarde is opgenomen wordt altijd vervolgd met het volgende microadres, m.a.w. de logische niveaus in het adresveld zijn don't care.

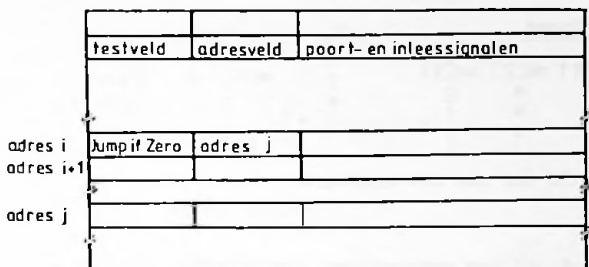
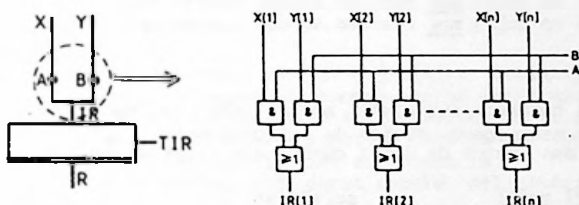


fig. 2.1 De opbouw van een microgeheugen



▽ AFLΔBESCHRIJVINGΔFIG22

```

[1] n
[2] a ^ AND; v OR; ~ NOT;
[3] n ← ASSIGNMENT
[4] n
[5] a INGANG REGISTER
[6] IR ← (XAA) v (YAB)
[7] n REGISTER
[8] IR ← (RAN)TIR v (IRAT)IR

```

▽

fig. 2.2 een register uit het datapad

De poort- en inleessignalen gaan naar het datapad. In figuur 2.2 is aangegeven waarvoor deze signalen worden gebruikt. Is het inleessignaal van een register (TIR) hoog dan wordt het register ingelezen met het niveau aan de ingang (IR). Is het inleessignaal laag dan blijft de registerinhoud ongewijzigd. In dat geval is het niveau van IR onbelangrijk, zodat de poortsignalen A en B don't care zijn.

3 bitminimalisatie

Om het aantal kolommen in het microgeheugen te minimaliseren moeten de don't care's optimaal worden gebruikt. In dit hoofdstuk is een minimalisatiemethode uitgewerkt waarmee dit wordt bereikt.

3.1 Definieren van begrippen

	sc[1]	sc[2]	sc[3]
w[1]	1	*	0
w[2]	0	*	1
w[3]	1	1	1

fig. 3.1 voorbeeld van een microgeheugen

Het microgeheugen bestaat uit een verzameling van woorden w[1] t/m w[n]. Een woord bestaat vervolgens weer uit een verzameling van subcommando's. Het microgeheugen, getoond in fig. 3.1, heeft totaal drie woorden waarvoor geldt:

```
w[3]={sc[1],sc[2],sc[3]}
w[2]={sc[3]} en sc[2] mag hieraan worden toegevoegd
w[1]={sc[1]} en sc[2] mag hieraan worden toegevoegd
```

Een don't care ("*") betekent dus niets anders dan dat het subcommando mag voorkomen in het microwoord. Moeten de subcommando's sc[1] en sc[2] worden samengenomen dan moeten de don't care's als volgt worden ingevuld:

	sc[1]	sc[2]	sc[3]		sc[12]	sc[3]	
w[1]	1	1	0	-->	w[1]	1	0
w[2]	0	0	1		w[2]	0	1
w[3]	1	1	1		w[3]	1	1

fig. 3.2 Minimaliseren van microgeheugen

Het is duidelijk dat we door een andere keuze te doen voor de don't care's ook de subcommando's sc[2] en sc[3] kunnen samenvoegen. Het minimum aantal nieuwe subcommando's blijft twee. Het volgend voorbeeld geeft duidelijk aan dat we subcommando's niet willekeurig mogen combineren indien we een minimale oplossing wensen.

	sc[1]	sc[2]	sc[3]	sc[4]
w[1]	1	*	1	0
w[2]	0	*	*	1
w[3]	1	0	*	*

fig. 3.3a: het te minimaliseren microgeheugen

	sc[1]	sc[23]	sc[4]
w[1]	1	1	0
w[2]	0	*	1
w[3]	1	0	*

fig. 3.3b: eerste poging samenvoegen van de subcommando's sc[2] en sc[3]

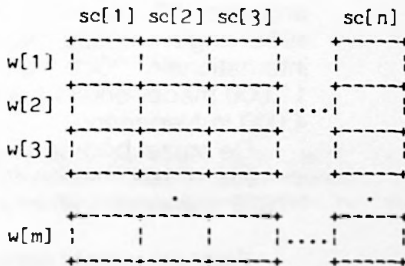
	sc[13]	sc[2]	sc[4]
w[1]	1	*	0
w[2]	0	*	1
w[3]	1	0	*

|
v

	sc[13]	sc[24]
w[1]	1	0
w[2]	0	1
w[3]	1	0

fig. 3.3c: tweede poging
 samenvoegen van de kolommen
 sc[1] met sc[3] en
 sc[2] met sc[4]

Uit het voorbeeld beschreven in fig. 3.3 blijkt duidelijk dat de eerste poging niet optimaal is immers de tweede poging geeft een kleiner aantal subcommando's. De eerste poging is dus afgewezen omdat de tweede beter bleek te zijn. Zo zouden we alle mogelijkheden kunnen proberen. Het zal de lezer duidelijk zijn dat dit niet efficiënt is. Voor een efficiënte methode zijn eerst enkele definities en stellingen nodig.



Het Microgeheugen als matrix M met:
 $M[i;j] = 1$; sc[j] behoort zeker tot w[i]
 $= 0$; sc[j] behoort zeker niet tot w[i]
 $= *$; don't care

fig. 3.4 Het microgeheugen als matrix.

Definitie 3.1

Een subcommando sc[j] is niet gedefinieerd in woord w[i] indien $M[i;j]$ don't care is.

Definitie 3.2

Een subcommando sc[j] is gedefinieerd in woord w[i] indien $M[i;j]$ geen don't care is

OCÉ: BETROUWBAAR EN INNOVEREND

Océ-van der Grinten N.V., opgericht in Venlo in 1877, is sinds 1920 werkzaam op het gebied van kopieerprocédés. Zowel voor de technische als de kantoormarkt. Het gaat hierbij om bijzonder interessante markten, omdat de hierbij gebruikte technieken een snelle ontwikkeling doormaken.

Océ speelt daarin en in de informatieverwerking een belangrijke rol. Tachtig procent van de Océ omzet wordt met kopieeractiviteiten gerealiseerd. Op het gebied van de tekenkamerreprografie is Océ de grootste ter wereld; in de kantoorreprosector is de Océ Groep de enige Europese onderneming die met apparatuur uit geheel eigen ontwikkeling en produktie op de markt is, en met snel groeiend succes. Mede door gedegen research (met circa 800 medewerkers in die sector!) en door de ontwikkeling van nieuwe technologieën is Océ thans uitgegroeid tot één van de grootste bedrijven op dit gebied. Océ breidt ook zijn activiteiten uit op het gebied van hardware en software voor tekstverwerkende apparatuur.

De Océ Groep is met eigen bedrijven werkzaam in 25 landen, verspreid over alle werelddelen. In

vele andere landen wordt samen- gewerkt met licentienemers en wederverkopers, zodat Océ produkten in ongeveer 90 landen op de markt worden gebracht.

Het hoofdkantoor van de Océ Groep is gevestigd te Venlo waar ook het grootste deel van de research en de produktie is gehuisvest, evenals de internationale marketing afdelingen. In 12 landen heeft Océ produktiebedrijven.

De snelle groei heeft consequenties gehad voor de personeelsbezetting. In 1945 waren er ongeveer 50 mensen in dienst, in 1958 ongeveer 600. Nu heeft de internationale Océ groep zo'n 11.000 medewerkers (van wie zo'n 4.000 in Nederland).

De omzet bedroeg f 16 miljoen in 1958, f 162 miljoen in 1968 en f 1828 miljoen in 1984.

Wanneer u meer wilt weten over onze organisatie en eventuele mogelijkheden voor u, bel of schrijf ons dan.

Océ-Nederland B.V., afdeling coördinatie personeelzaken,
Postbus 101, 5900 MA Venlo.
Telefoon 077-92222, toestel 2740.



Definitie 3.3

De subcommando's $sc[i]$ en $sc[j]$ zijn compatibel indien voor alle woorden, behorend tot het microgeheugen, waarvoor zowel $sc[i]$ als $sc[j]$ gedefinieerd zijn geldt dat $M[k;i]=M[k;j]$
notatie: $sc[i] @ sc[j]$

Definitie 3.4

De subcommando's $sc[i]$ en $sc[j]$ zijn incompatibel indien deze niet compatibel zijn.
notatie: $sc[i] \sim @ sc[j]$

De voorgaande definities toegepast op het voorbeeld in fig.3.3 geeft de volgende compatibele en incompatibele subcommando's.

```
compatibel   sc[1] @ sc[3]
              sc[2] @ sc[3]
              sc[2] @ sc[4]
incompatibel sc[1] ~@ sc[2]
              sc[1] ~@ sc[4]
              sc[3] ~@ sc[4]
```

In de literatuur worden bij bitminimalisatie van een microgeheugen ook begrippen als compatibel en incompatibel gebruikt ([Grasselli70],[Jayasri76] e.a.), deze zijn dan anders gedefinieerd.

Definitie 3.5

Een compatible class (afk: CC) is een verzameling van paarsgewijs compatibele subcommando's.

Definitie 3.6

Een maximal compatible class (afk: MCC) is een CC met de eigenschap dat toevoeging van een willekeurig subcommando, uitgezonderd de subcommando's in de CC, geen paarsgewijs compatibele verzameling meer oplevert.

Definitie 3.7

Een incompatible class (afk: IC) is een verzameling van paarsgewijs incompatibele subcommando's.

Definitie 3.8

Een maximal incompatible class (afk: MIC) is een IC met de eigenschap dat toevoeging van een willekeurig subcommando, uitgezonderd de subcommando's in de IC, geen paarsgewijs incompatibele verzameling meer oplevert.

Definitie 3.9

Onder het samenvoegen van twee compatibele subcommando's $sc[i]$ en $sc[j]$ wordt verstaan het maken van een nieuw subcommando $sc[ij]$ die als volgt gedefinieerd is:

- 1: voor alle woorden $w[k]$, behorend tot het microgeheugen, waarin $sc[i]$ gedefinieerd is, geldt:
 $M[k;ij]=M[k;i]$
- 2: voor alle woorden $w[k]$, behorend tot het microgeheugen, waarin $sc[j]$ gedefinieerd is, geldt:
 $M[k;ij]=M[k;j]$
- 3: Voor alle woorden $w[k]$, behorend tot het microgeheugen, waarin zowel $sc[i]$ als $sc[j]$ niet gedefinieerd zijn, geldt:
 $M[k;ij]=M[k;i]=M[k;j]=*$ (don't care).

Bij het voorbeeld van fig. 3.3 hebben we het samenvoegen van compatibele subcommando's al onbewust toegepast.

Twee incompatibele subcommando's $sc[i]$ en $sc[j]$ kunnen niet samengevoegd worden omdat er tenminste één woord $w[k]$ is waarin beide subcommandos gedefinieerd zijn en $M[k;i] < M[k;j]$. Hieruit volgt dat elementen uit een MIC ook niet samengevoegd kunnen worden, zodat de minimale oplossing tenminste #MIC kolommen heeft. Ginsburg [Ginsburg59] definieert dan ook een NMIC (Numerically MIC) als een MIC met het grootste aantal elementen. Er is dus een ondergrens gevonden voor het minimum aantal kolommen na minimalisatie, namelijk #NMIC.

Het definiëren van een compatible class is zinvol, dit blijkt direkt uit stelling 3.10.

Stelling 3.10

Alle subcommando's behorend tot dezelfde CC kunnen samengevoegd worden tot één nieuw subcommando.

Bewijs

Stel $CC\text{-oud} = \{sc[1], \dots, sc[n]\}$.

De twee subcommando's $sc[i]$ en $sc[j]$, behorend tot $CC\text{-oud}$, kunnen samengevoegd worden tot $sc[ij]$ omdat ze compatibel zijn.

Vervolgens kan een $CC\text{-nieuw}$ worden geconstrueerd door:

$$CC\text{-nieuw} = \{sc[ij]\} \vee CC\text{-oud} \setminus \{sc[i], sc[j]\}$$

$CC\text{-nieuw}$ heeft dus een kardinaliteit van $n-1$, aangezien de kardinaliteit van $CC\text{-oud}$ n is, kunnen we door herhaald samenvoegen een CC overhouden met slechts één element, het nieuwe subcommando.

Daarvoor is het echter wel nodig dat we kunnen aantonen dat $CC\text{-nieuw}$ weer een compatible class is!

Aangezien $CC\text{-oud}$ een compatible class is, is zeker

$$CC\text{-hulp} = CC\text{-oud} \setminus \{sc[i], sc[j]\}$$

een compatible class. Volstaan kan dus worden met het aantonen dat alle subcommando's uit $CC\text{-hulp}$ compatibel zijn met $sc[ij]$.

Stel echter dat voor $sc[q]$, behorend tot $CC\text{-hulp}$, geldt:

$sc[q] \neq sc[i,j]$, dan volgt uit def. 3.3 dat er een woord $w[k]$ is waarin zowel $sc[q]$ en $sc[i,j]$ gedefinieerd zijn met $M[k;i,j] \neq M[k;q]$. Aangezien $M[k;i,j]$ is ontstaan door het toepassen van def. 3.9 kan dus worden geconcludeerd dat $M[k;i,j]=M[k;i]$ indien $sc[i]$ gedefinieerd is in $w[k]$ (evenzo geldt dit voor $sc[j]$). Maar dan geldt ook dat $M[k;i]$ of $M[k;j]$ ongelijk is aan $M[k;q]$. M.a.w. $sc[i]$ of $sc[j]$ is incompatibel met $sc[q]$ en dit is tegenstrijdig met de aanname dat CC-oud een CC is.

Indien een minimum aantal CC's bepaald kan worden zodanig dat iedere subcommando tenminste eenmaal voorkomt dan hebben we ons doel bereikt: het minimaliseren van het aantal kolommen.

Uit het voorbeeld beschreven in fig. 3.3 hadden we al geconcludeerd dat de subcommando's niet willekeurig samengevoegd kunnen worden indien een minimale oplossing gewenst is.

We gaan nu naar een bijzondere klasse van compatibele subcommando's kijken.

Definitie 3.11

Subcommando $sc[j]$ heet implicierend compatibel t.o.v. subcommando $sc[i]$ indien:

1: $sc[i] \neq sc[j]$

2: In alle woorden $w[k]$ waarin subcommando $sc[j]$ gedefinieerd is, is ook subcommando $sc[i]$ gedefinieerd.

notatie: $sc[i] \supseteq sc[j]$.

Indien geldt dat $sc[i] \supseteq sc[j]$ dan hoeft niet te gelden dat $sc[j] \supseteq sc[i]$, geldt deze laatste betrekking ook dan volgt uit punt twee van def. 3.11 dat voor alle woorden $w[k]$ geldt dat $M[k;i]=M[k;j]$ (m.a.w. de kolommen zijn gelijk!).

Het zal duidelijk zijn dat door het samenvoegen van deze subcommando's tot $sc[i,j]$, voor $sc[i,j]$ moet gelden dat $M[k;i,j]=M[k;i]$.

Praktisch gezien komt dit neer op het verwijderen van kolom $sc[j]$ en dat vervolgens alleen nog de naam van kolom $sc[i]$ gewijzigd moet worden in $sc[i,j]$. Het zal duidelijk zijn dat door deze samenvoeging de minimale oplossing nog altijd te bereiken is.

Stelling 3.12

Indien subcommando $sc[j]$ implicierend compatibel is t.o.v. subcommando $sc[i]$ dan kunnen we de subcommando's altijd samenvoegen tot $sc[i,j]$ zonder dat hierdoor de minimale oplossing wordt verhinderd.

Bewijs

Veronderstel dat de volgende verzameling van CC's een minimale oplossing is: $\{CC1, CC2, \dots, CCn\}$

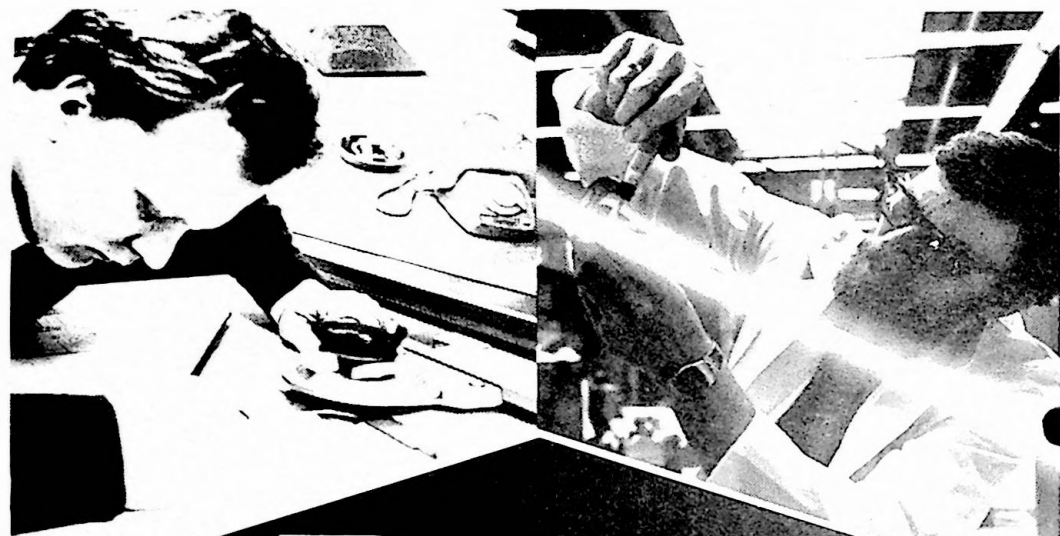
Indien geldt dat $sc[j]$ implicierend compatibel is met $sc[i]$, $sc[i]$ behoort tot CCi en $sc[j]$ behoort tot CCj dan moet worden aangetoond dat er ook een minimale oplossing is waarbij $sc[j]$ en $sc[i]$ tot dezelfde CC behoren.

Voor alle subcommando's $sc[p]$ behorend tot $CCi \setminus \{sc[i]\}$ geldt (def. 3.3):

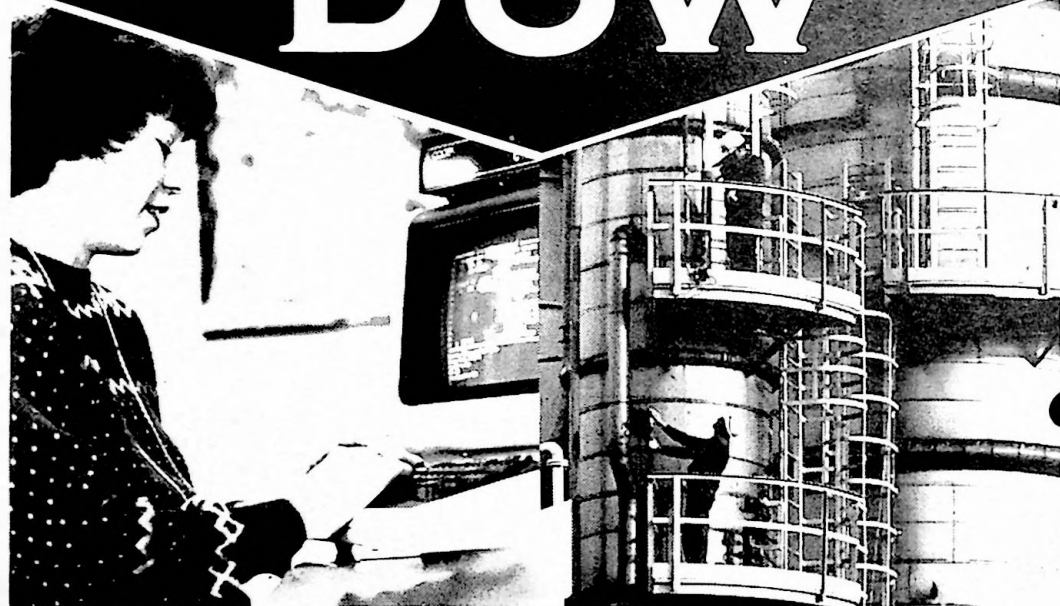
$M[k;p]=M[k;i]$ voor alle woorden $w[k]$ waarin $sc[p]$ en $sc[i]$ gedefinieerd zijn.

Uit de definitie van implicierend compatibel (def. 3.11) volgt dat $M[k;i]$ tenminste gedefinieerd is indien $M[k;j]$ gedefinieerd is, zodat ook voor alle subcommando's $sc[p]$ behorend tot $CCi \setminus \{sc[i]\}$ moet gelden:

$M[k;p]=M[k;j]$ voor alle woorden $w[k]$ waarin $sc[p]$ en



DOW



DOW, CREATIEF MET CHEMIE.

Dow Chemical (Nederland) B.V. - een dochtermaatschappij van het Amerikaanse chemische concern The Dow Chemical Company - heeft vestigingen in Terneuzen, Rotterdam en de Botlek. Met zijn 28 fabrieken en 2450 werknemers is Terneuzen de grootste Dow-vestiging buiten de Verenigde Staten. Dow heeft in Nederland 2800 werknemers en behaalde in 1984 een omzet van 4,9 miljard gulden. Ruim 90 procent van de productie wordt geëxporteerd.

Dow Chemical (Nederland) B.V., Postbus 48, 4530 AA Terneuzen. Tel. 01150-71234.

sc[j] gedefinieerd zijn.

Hieruit volgt dat de vereniging van CC_i met {sc[j]} een compatible class is.

Een andere minimale oplossing is hiermee gevonden:

{CC₁, ..., CC_i ∪ {sc[j]}, ..., CC_j \ {sc[j]}, ..., CC_n}

Dit kan worden herhaald voor alle implicierend compatibele subcommando's.

3.2 methode om te minimaliseren

We hadden al in paragraaf 3.2 vastgesteld dat een minimum aantal CC's gezocht moet worden waarin tenminste alle subcommando's eenmaal voorkomen.

Door alle CC's in rijen en alle subcommando's in kolommen te plaatsen hebben we een zogenaamd "prime implicant table" gekregen [McCluskey62].

We kunnen twee type CC's onderscheiden, n.l. CC's die een MCC zijn en CC's die geen MCC zijn. Dit onderscheid is ook gemaakt in tabel 3.1

	sc[1]	sc[2]	sc[3]	...	sc[n]
alle CC's die geen MCC zijn.	CC[1]	x	x		
	CC[2]		x	x	
	...				
	CC[i]				
alle MCC's	MCC[1]	x	x	x	
	MCC[2]				
	...				
	MCC[n]				

tabel 3.1 prime implicant table

Een kruis in de tabel geeft aan dat het desbetreffende subcommando voorkomt in de CC.

McCluskey formuleerde het probleem als volgt: "For a given implicant table find a smallest set of rows such that every column of the table has a marked entry in at least one row of the set"

Om een oplossing te vinden worden de volgende drie begrippen gehanteerd:

- essentiële rij
- rij-dominantie
- kolom-implicatie

essentiële rij

komt een subcommando in slechts één CC voor, in de kolom staat dan slechts één kruis, dan is deze CC essentieel, immers bij verwijdering van deze CC wordt het desbetreffende subcommando niet meer gedekt.

rij-dominantie

Indien rij i tenminste op alle plaatsen als rij j een kruis heeft, kan rij j worden verwijderd. We zeggen dat rij j gedomineerd wordt door rij i.

rijdominantie kan goed worden gebruikt. Aan een CC die geen MCC is kan nog een subcommando worden toegevoegd. Dit kan net zolang herhaald worden

tot de CC uiteindelijk een MCC is geworden. Deze MCC is al in de tabel aanwezig. Hieruit volgt direkt dat een CC tenminste gedomineerd wordt door een MCC. We kunnen dus alle CC's die geen MCC zijn uit de tabel verwijderen.

Hierdoor wordt het aantal rijen in de tabel aanzienlijk kleiner.

kolom-implicatie

Indien er een kolom i is die tenminste in dezelfde rijen als kolom j een kruis heeft staan, dan kunnen we kolom i verwijderen. We zeggen dat kolom j kolom i impliceert.

In def. 3.11 is het begrip implicerend compatibel gedefinieerd. Indien $sc[i] \supseteq sc[j]$ betekent dit eigenlijk niets anders dan dat $sc[i]$ tenminste in dezelfde woorden gedefinieerd is als het subcommando $sc[j]$. Dit heeft direkt tot gevolg dat subcommando's waarmee $sc[i]$ compatibel is zeker $sc[j]$ compatibel zal zijn. Maar dit heeft dan weer tot gevolg dat alle MCC's waarin $sc[i]$ voorkomt ook $sc[j]$ in moet voorkomen. Er is dus sprake van kolomimplicatie indien in de prime implicant table een van de MCC's zijn opgenomen.

Het omgekeerde, kolom i impliceert kolom j waaruit dan zou moeten volgen dat $sc[i]$ implicerend compatibel is t.o.v. $sc[j]$, geldt niet.

In vele gevallen zal herhaald toepassen van zoeken naar essentiële rij, rij-dominantie en kolom-implicatie direkt leiden tot een minimale oplossing. Echter er zijn situaties waarin deze begrippen niet direkt leiden tot een minimale oplossing, in dat geval wordt de prime implicant table cyclisch genoemd.

3.3 Samenvatting

Met de volgende procedure kan het minimum aantal kolommen worden bepaald:

- 1: bepaal alle compatibele paren, indien er een implicerend compatibel paar is kunnen deze direkt worden samengevoegd (stelling 3.12)
- 2: bepaal vervolgens alle MCC's.
- 3: vervolgens de prime implicant table opstellen en deze minimaliseren
- 4: het minimum aantal CC's is hiermee bepaald. De kolommen die overeenkomen met de subcommando's uit een CC kunnen vervolgens worden samengevoegd.

Aangezien het niet noodzakelijk is dat een subcommando in meer dan een CC voorkomt kunnen de redundante subcommando's uit de CC's worden verwijderd.

Hier ontstaat een keuzemogelijkheid, zal ik het subcommando uit $MCC[i]$ of uit $MCC[j]$ verwijderen? Dit maakt voor het samenvoegen van de kolommen niet uit, maar wel indien nog andere minimalisatiemethoden toegepast gaan worden.

```

1      1      2      3      4      5
1      0      0      0      0      0
1100000001022222022222222222222200000010000202011
100000000022222222222222222222000000000000202011
1000000000222222222222222222220000000000001021211
0000222222010010012222222222010000000000000000001
10000011001000010012222222200000000000000000000001
1000001100000100001222222222100000000010000000001
0011000100001000022222222220000010100000000000001
0000222222100001222222222200100001000000000000001
00110001010000100001222222222100001000010000000001
0000222222100001001222222222001000000001000000001
00002222220100000001002222220200000000001110000001
0000222222202222220222201102222202000010000000202001
0010000110001000001221001222000000000111000000001
00010001010000001000102222202000000000011100000001
1000001100222202222222222222020000000000000202001
1000001100000100222100222222100000000000100000001
0000222222000001002222222201000000000100001000001
0011001010000100001222222221000000000010001000001
0000222222010000100122222222100000000001000000001
00110010112222022222222222212000000000001202001
001100110022221222222222222212010000000000202001
1000001100222222222222222222000000000000202001
001100110022222022222222222213010000000000202001
0010001100000100001222222220100000000100001000001
1000001100000100122222222010100000010000000001
10110011002222022222222222220201000000000202001
000022222200000222001222220000000000000101000001
100000000022220222222222222222020000000000202010

```

fig. 4.1 Het te minimaliseren microgeheugen. ('2' = don't care)

```

88 MCC's
22 27 29 30      41 24 27 29 30
22 24 27 29 30  37 48 50 47 28
5 4 47 25 24 28 29 30  13 24 27 29 30
5 4 23 25 24 28      8 24 27 29 30
14 5 4 25 24 28 29 30  2 47 28 29 30
14 24 27 28 29 30      46 28 29 30
35 5 4 47 28 29 30      38 28 29 30
20 24 27 28 29 30      34 28 29 30
20 23 24 27 28      32 28 29 30
19 25 26 20 29 30      4 22 29 30
19 23 25 24 20      31 47 28
11 18 22 25 24 29 30  45 23 28
9 24 27 28 29 30      44 29 30
9 21 24 27 29 30      17 28
7 23 24 28 29 30      1 28
7 21 25 24 29 30      52
49 25 24 20 29 30      43
39 40 14 28 29 30
34 25 24 28 29 30
33 25 24 28 29 30
31 25 24 28 29 30
15 25 24 28 29 30
12 25 24 28 29 30
10 22 24 24 29 30
3 24 27 28 29 30
42 24 27 29 30

```

fig. 4.2 De mcc's indien geen gebruik wordt gemaakt van de eigenschap 'implicierend compatibel'

```

88 MCC's
14 10 24
7 21 24
49 24
34 24
31 24
31 24
19 24
14 24
15 24
12 24
10 24
9 21
52
51
46
45
44
43
42
41
40
38
36
35
32
20
17
13
8
4
3
2
1
48 impl. comp. met 50
47 impl. comp. met 50
39 impl. comp. met 40
37 impl. comp. met 50
30 impl. comp. met 32
29 impl. comp. met 32
28 impl. comp. met 31
27 impl. comp. met 41
25 impl. comp. met 26
24 impl. comp. met 41
23 impl. comp. met 45
22 impl. comp. met 4
14 impl. comp. met 40
6 impl. comp. met 35
5 impl. comp. met 35

```

fig. 4.3 De mcc's indien gebruik wordt gemaakt van de eigenschap 'implicierend compatibel'

Toelichting:
Een rij in fig. 4.2 en fig. 4.3 geeft de elementen van een MCC. De elementen komen overeen met de kolomnummers uit het microgeheugen.
Conclusie:
1. het aantal te bepalen MCC's is verminderd.
2. het aantal elementen waaruit een MCC bestaat is kleiner.

4. Resultaten

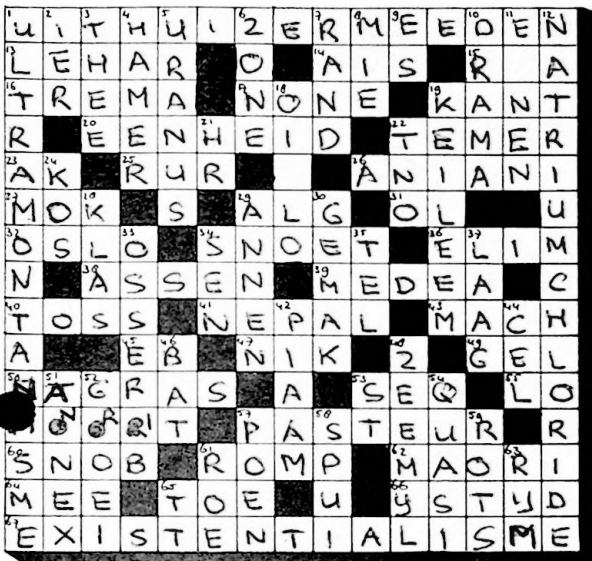
Voor de tweedejaars studenten informatika van TH-Twente wordt het college "Uitrusting van digitale systemen" gegeven. De studenten moeten voor de afronding hiervan een eenvoudige processor ontwerpen waarbij de besturing plaats vindt d.m.v. een microgeheugen. De afmetingen van het microgeheugen zijn circa 20 rijen en 40 kolommen. Op deze microgeheugens is de hier beschreven minimalisatie methode toegepast, resultaat: 6 tot 20 procent minder geheugen nodig. Hierbij moet worden opgemerkt dat de studenten veelal bewust of onbewust bitminimalisatie hadden toegepast door het samennemen van poortsignalen.

Vooraf het implicerend compatibel zijn van subcommando's vereenvoudigd het bepalen van de MCC's. Een voorbeeld van een microgeheugen met daarin don't care's is gegeven in fig. 4.1. Tabel 4.2 geeft de MCC's die ontstaan indien niet van de eigenschap 'implicerend compatibel' gebruik wordt gemaakt terwijl in tabel 4.3 juist wel deze eigenschap wordt gebruikt.

5. Literatuur

- [Ginsburg59] S.Ginsburg,"A technique for the reduction of a given machine to a minimal state machine", IRE Trans. on Electronic Comp.,sept. 1959, pp 346-355.
- [McCluskey62] I.B.Pyne,E.J.McCluskey,"The reduction of redundancy in solving prime implicant tables", IRE Trans. on Electronic Comp., aug. 1962, pp 473-482
- [Grasselli70] A.Grasselli,U.Montassari,"On the minimizing of read only memories in microprogrammed digital computers",IEEE Trans. on comp., nov. 1970, pp 962-973
- [Jayasri76] Totadri Jayasri, Dhruba Basu,"An approach to organizing microinstructions which minimizes the width of the control store word",IEEE Trans. on comp.,may 1976, pp 514-521
- [Agerwala76] T.Agerwala,"microprogram optimization: A Survey",IEEE Trans. on comp., oct. 1976, pp 962-973
- [Blaauw76] G.A.Blaauw,"Digital System Implementation", Prent. Hall,1976.

K
E K
A L O



horizontaal.

1. plaats in Groningen
13. componist
14. verhoogde toon
15. buurtschap bij Dieren
16. leesteken
17. interval
19. zijde
20. afgerond geheel
22. zeurkous
23. aankomend 25 riviertje in West-Duitsland
26. rijstmesje
27. vaatwerk
29. wier
31. oosterlengte
32. plaats in Noorwegen
34. gezicht
36. rustoord
38. plaats in Drenthe
39. mychologisch figuur
40. oppoof
41. deelvan Azie
43. eenheid van snelheid
45. getijde
47. hoofdgebaar
49. haarsmeersel
50. ergroen
53. het volgende (lat)
55. soort onderwijs
56. toegangsweg
57. 19e-eeuws weten schapper
60. parvenu
61. tors
62. oorspronkelijke bewoner
64. honingdrank
65. gesloten
66. geologisch tijdperk
67. filosofische richting.

verticaal.

1. extreme pausgezindheid
2. Europeaan
3. drank.
5. planeet.
6. aardgordel.
7. vreemde munt.
8. meelprodukt.
9. boom
10. toneel.
11. dorpje in Luxemburg.
12. keukenzout.
18. plaats bij Venray.
19. grondsoort.
21. heer.
22. onderzoeksinstituut.
24. Grieks eiland.
28. schoollokaal.
29. plaats in Drenthe
30. gerief.
33. vleesgerecht.
34. vreemde munt.
35. moment.
37. gemeen
42. plaatsje in Friesland bij Workum.
44. kleine ruimte.
46. slaghout.
48. afstandsmaat.
51. behorende bij.
52. was.
53. heilige.

PUZZEL

inzendingen voor
1 juli maken kans op
leuke Inter-Actiefitas.

Verticaal.

54. schijnbaar.
57. geld.
58. kolk, sluis.
59. steenmassa.
61. strafwerktuig.
63. bevroren dauw of mist.
65. motorraces.

Wist-U-dat?

Binnenkort worden weer de boekenbestellijsten aan iedereen toegestuurd. Je moet dan ruim van te voren je boeken bestellen. Soms blijkt helaas halverwege een trimester dat je sommige boeken niet of nauwelijks nodig hebt.

Dit is vooral erg vervelend als het ook nog om dure boeken gaat.

Er is ooit eens geprobeerd om een boekenfonds op te richten voor eerste- en tweedejaarsboeken, maar dit bleek helaas geen haalbare kaart.

Voor de studenten zijn er natuurlijk verschillende mogelijkheden om hier zelf iets aan te doen.

Vooraf eerstejaars studenten wordt daarom zeer sterk aanbevolen eens bij ouderejaars studenten over de verschillende mogelijkheden navraag te doen. Er zijn genoeg alternatieve mogelijkheden om goedkoop aan je studiematerialen en boeken te komen. Voor meer informatie kan je ook contact op nemen met een van de AGI-leden. Namen en telefoonnummers contact op nemen met een van de AGI-leden. Namen en telefoonnummers worden op het Inter-Actiefbord gehangen.

De Afdelingsgroep Infomatica. (AGI)



Chapter 14.

Frank zaapte nog eens flink. Het bekertje koffie voor hem schreeuwde het uit om opgedronken te worden. Frank voldeed aan dat verzoek en brandde acuut zijn tong. "...!" (kan echt niet door de censuur heen! red.). Vanochtend was hij nog enthousiast zijn bed uitgesprongen om lekker vroeg in het rekencentrum te kunne bliepen.

Daar schuilt op zich niets kwaads in, maar dat moet je dan wel doen op een ochtend wanneer je DEC niet plat ligt. Anders kun je je humeur voor de rest van de dag wel vergeten. En het humeur van Frank was, na anderhalf uur tevergeefs proberen in te loggen, ver te zoeken. En terwijl hij in de wc zijn tong onder de kraan hield, dacht hij, "Ik ben niet boos. Echt niet. Ik ben alleen heel erg verdrietig."

Slot.

In de verte klonk een schrapend geluid. Een lichte waas hing over wat eens een vijver was. De zon begon zijn eerste stralen over het woeste woestijnlandschap te verspreiden.

Het was een vreselijke warme zomer geweest, maar nu, zo tegen het einde ervan, was de hitte al een paar dagen enigszins dragelijk geweest. Ook deze dag beloofde weer redelijk warm te worden.

Beneden in de bunkers hielden airconditioners de hitte buiten de deur. Hier waren een flink aantal mensen druk bezig met allerhande zaken. Eentje lag echter op de grond te slapen.

Langzaam begon zijn arm heen en weer te bewegen, alsof zijn hand iets zocht om in te drukken. Toen Harald echter niets vond opende hij zijn oogjes. Met een ruk schoot hij over-eind, niet goed beseffend waar hij

zich bevond. Nog geen twee seconden geleden dacht hij nog dat hij in zijn bed lag en halfdromend probeerde zijn wekkerradio aan te zetten. Maar wat hij nu zag was in ieder geval niet zijn slaapkamer.

Het bleek dat men hem uit het verre verleden naar deze plaats had geteletransporteerd (lustrumvideo?). Een paar uur daarvoor had men geprobeerd iemand naar het verleden te sturen, maar bij het terughalen moest toch echt iets verkeerd zijn gegaan.

Harald had echter niemand in zijn buurt gezien op het moment dat hij van de weg in het gras belandde.

De voorbereidingen om Harald terug te transporteren naar zijn eigen tijd werden op dat moment reeds getroffen. Harald zou nu de opdracht van zijn voorganger op moeten nemen. Hij werd naar de transportruimte gebracht en het aftellen kon beginnen. ...3, 2, 1... Een grote flits en weg was de arme jongen.

Verschrikt merkte een van de medewerkers op dat de wijzers van de machine nog net zo stonden ingesteld als toen de voorganger van Harald was weggeflitst.

Epiloog.

Het was half zeven in de ochtend. Willem sleepte het tamelijk levenloze lichaam, dat hij had zien liggen toen hij langs het WB-gebouw in de richting van de Horstlindelaan reed, door het gras. Hij legde het lijk voor-zichtig op de achterbank van zijn auto en reed weg, zich afvragend waar Gerard was. Hij bedacht zich tevens dat Gerard zich vergist had, want dit was de Horstlindelaan helemaal niet. En hij had toch met Gerard afgesproken dat hij het lijk en Gerard daar zou oppikken...

Op de Hengelosestraat werd de auto van Willem door een politiebuisje klem gereden. Intussen stond Gerard naast het lijk op de Horstlindelaan met smart te wachten op zijn maat die hem toch zou helpen om het lijk op de een of andere wijze kwijt te raken...

Moraal van dit verhaal:

Wie 's ochtends uitslaapt, mist het spannendste gedeelte van de dag.

E. Kwien.

SIGNAAL VOOR INFORMATICI

Het bedrijf

Hollandse Signaalapparaten B.V. (kortweg Signaal) is een onderdeel van het Philips concern. Signaal ontwikkelt en produceert uiterst geavanceerde computergestuurde defensiesystemen, verkeersleidingsystemen voor de lucht- en scheepvaart en militaire telecommunicatie-apparatuur. De Signaal systemen vinden hun weg over de gehele wereld.

Voorbeelden van door Signaal gerealiseerde systemen: het Area- en Approach Control Radar Systeem op Schiphol, in Rotterdam het begeleidingssysteem voor de scheepvaart op de Nieuwe Waterweg en de Goalkeeper, een volautomatisch wapensysteem dat schepen beveiligd tegen anti-ship missiles.

Signaal heeft vestigingen in Hengelo (Ov.), Huizen (N.H.), Den Haag en Apeldoorn. Bij Signaal werken 5400 medewerkers, waarvan 4300 in de hoofdvestiging te Hengelo.

Veel mogelijkheden voor informatici

Voor de Onderzoekafdeling en voor de Software-Ontwikkelingsafdelingen, neemt Signaal jaarlijks een beperkt aantal jonge TH-ingenieurs aan.

Afgestudeerden van vrijwel alle vakgroepen van de Onderafdeling der Informatica zullen bij Signaal een passende aansluitende functie kunnen vinden. In Hengelo, Huizen, Apeldoorn of Den Haag.

Stages en afstudeeropdrachten

Stages en afstudeeropdrachten zijn een goede manier om het bedrijf te leren kennen. Signaal (de hoofdvestiging in Hengelo ligt op 15 minuten fietsen van de TH) biedt jaarlijks een aantal studenten de gelegenheid bij het bedrijf een stage te volgen of meer diepgravend aan een afstudeeropdracht te werken.

Informatie en sollicitatie

Uiteraard kunnen wij u in deze advertentie slechts beperkt informeren over de mogelijkheden voor informatici bij Signaal.

Nadere inlichtingen kunt u telefonisch verkrijgen bij de heer T.W. Haak, tel. 074-482551. Aan hem kunt u ook uw sollicitatie richten.

Hollandse Signaalapparaten B.V., Postbus 42, 7550 GD Hengelo (Ov.).



SIGNAAL



**U WEET
TE WEINIG
OVER SHELL.**

**SHELL WEET
TE WEINIG
OVER U.**

**DAAR KUNNEN WE
IETS AAN DOEN.**

Want Shell doet natuurlijk aanmerkelijk meer dan de brandstof leveren waardoor zo'n beetje de helft van de mensheid zich vervoeren kan. Van een multinational op topniveau verwacht u meer. En terecht. Wat we meer doen, vertellen we u graag. In uw belang en zeker ook in het onze. Want alleen al in Nederland hebben we elk jaar 200 academici nodig van topniveau. Voordat we daarover echt met u praten, kunnen we het volgende aanbieden:

Een toekomst bij Shell.

als u bijna aan afstuderen toe bent, loont het de moeite onze brochure "uw toekomst bij Shell" aan te vragen (070 - 77 6600).

Vakantiecursussen.

speciaal voor studenten in hun laatste studiejaar. U leert Shell van nabij kennen, en andersom.

Stages.

bij werkmaatschappijen in Nederland. Soms ook daarbuiten, zoals in de USA, Canada, Singapore of Australië. Eveneens voor studenten in hun laatste studiejaar. Een bouwsteen in uw studie én een perfecte methode om Shell te leren kennen. Uw stagecoördinatoren zijn op de hoogte.

*Publicaties
en films.*

een wat vrijblijvende maar niettemin leerzame methode om meer over ons en onze visie op de wereld te weten te komen. Inlichtingen: Shell Nederland B.V., afdeling PAG/4, Postbus 1222, 3000 BE Rotterdam.

*Excursies
en lezingen.*

wat intensiever, eventueel op maat en zeker to the point. Onze afdeling Onderwijscontacten, Postbus 1222, 3000 BE Rotterdam, vertelt u graag over onderwerpen en mogelijkheden.

Wij hebben ons zegje gezegd. De volgende stap is aan u.



SHELL DOET MEER DAN U DENKT