

MASTER

Thermische modelvorming van een miniatuur smeltpatroon in PSPICE

van Herel, J.P.

*Award date:*  
1992

[Link to publication](#)

**Disclaimer**

This document contains a student thesis (bachelor's or master's), as authored by a student at Eindhoven University of Technology. Student theses are made available in the TU/e repository upon obtaining the required degree. The grade received is not published on the document as presented in the repository. The required complexity or quality of research of student theses may vary by program, and the required minimum study period may vary in duration.

**General rights**

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain

**Take down policy**

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.



Technische Universiteit Eindhoven  
Faculteit der Elektrotechniek  
Vakgroep Elektrische Energiesystemen.

---

Thermische modelvorming  
van een  
miniatur smeltpatroon  
in PSPICE

Afstudeerverslag

EG/92/610

J.P. van Herel

**Hoogleraar:** Prof. ir. G.C. Damstra

**Begeleiders:** M. Sc. X.Z Meng,  
ir. J.G.J. Sloot

Eindhoven,  
25 aug 1992

---

De faculteit der Elektrotechniek van de Technische Universiteit Eindhoven aanvaardt geen  
verantwoordelijkheid voor de inhoud van stage- en afstudeerverslagen.

---

# Samenvatting

Het afstudeerwerk is verricht in het kader van het onderzoek naar betrouwbaarheid van smeltpatronen, geleid door een promovendus. Hierbij wordt een onderverdeling gemaakt in diverse soorten patronen: hoogspanningspatronen, substraatpatronen en miniatuurpatronen.

Voor het opstellen van een model ten behoeve van de betrouwbaarheid vormt het thermisch gedrag van een patroon tijdens variërende belastingstromen een belangrijke rol. Tijdens het afstudeerwerk is geprobeerd een beeld te krijgen van de temperatuurverdeling over de smeltdraad in een miniatuurpatroon, bij pulsformige stromen.

In eerste instantie is getracht deze temperatuurinformatie te verkrijgen door het direct meten van temperatuur met behulp van geschikte meetsystemen. Door optredende complicaties is dit niet uitvoerbaar gebleken.

Er is een andere methode ontwikkeld voor het verkrijgen van de gewenste informatie door gebruikmaking van stroom door en de spanning over de patroon tijdens belasting. Hiervoor zijn een meetopstelling en software ontwikkeld.

Daarnaast is een eerste opzet gemaakt voor de ontwikkeling van een thermisch model van de smeltdraad, inclusief directe omgeving. Hierin zijn aspecten als geleiding, straling en convectie opgenomen. Dit model is omgewerkt naar een elektrisch equivalent met discrete componenten en in PSPICE gesimuleerd. Met de resultaten kan een idee worden gevormd van de temperatuurverdeling over de draad, als functie van de tijd.

# Inhoud

Samenvatting	2
Voorwoord	4
Inleiding	5
<b>I PSPICE Model</b>	<b>6</b>
<b>1 Modelvorming</b>	<b>6</b>
1.1 Lijst van symbolen	6
1.2 Inleiding	7
1.3 Inleiding warmte-theorie	8
1.4 Het thermisch model	9
1.5 Inleiding elektrische modellering	11
1.5.1 Draadsegmentering	11
1.5.2 Vastleggen referentie	12
1.5.3 Segment opbouw	12
1.5.4 Modellering eindcaps	14
1.5.5 Aanpassing gebruik half segment	14
1.6 Het elektrisch model	15
1.6.1 Warmte injectie: $I_{inj}(t,V)$	15
1.6.2 Opslag van warmte: $C(V)$	16
1.6.3 Thermische geleidingsweerstand: $R_{th}(V)$	17
1.6.4 Convectie weerstand: $R_{conv}(V)$	18
1.6.5 Stralingsterm $R_{rad}(V)$	24
<b>2 PSPICE componenten modellering</b>	<b>28</b>
2.1 PSPICE model: $R_{th\_Ni\_ax}$	28
2.1.1 PSPICE model: $R_{th\_glas\_rad}$	30
2.1.2 PSPICE model: $R_{rad}$	31
2.1.3 PSPICE model: $R_{conv}$	33
2.1.4 PSPICE model: $C_{ni}$	34
2.1.5 PSPICE model: $C_{lucht}$	36
2.1.6 PSPICE model: $C_{glas}$	37
2.1.7 PSPICE model: $I_{inj}$	38
2.1.8 PSPICE model: Opbouw segment	41

2.1.9	PSPICE model: Eindcap . . . . .	44
2.1.10	PSPICE model: Opbouw draad . . . . .	44
2.1.11	PSPICE model: Constanten . . . . .	46
2.1.12	PSPICE model: Stuurcodes . . . . .	46
<b>3</b>	<b>Simulaties</b>	<b>48</b>
3.1	Constante stroom: controle I-t curve . . . . .	48
3.1.1	Uitvoering . . . . .	49
3.1.2	Resultaten . . . . .	50
3.2	Pulsstromen . . . . .	55
3.2.1	Uitvoering . . . . .	55
3.2.2	Resultaten . . . . .	56
<b>4</b>	<b>Conclusies en aanbevelingen</b>	<b>59</b>
4.1	Conclusies It-curve . . . . .	59
4.1.1	Correctie It-curve . . . . .	60
4.1.2	Straling versus convectie . . . . .	61
4.2	Conclusies pulsstromen . . . . .	62
4.3	Aanbevelingen . . . . .	64
<b>5</b>	<b>PSPICE: het gereedschap</b>	<b>66</b>
5.1	Programmabeschrijving . . . . .	66
5.1.1	PSPICE versus MicroCap III . . . . .	66
5.2	PSPICE bronnen . . . . .	67
5.2.1	Tijdgestuurde bronnen . . . . .	67
5.2.2	Spanningsgestuurde bronnen . . . . .	68
5.2.3	Stroomgestuurde bronnen . . . . .	68
5.2.4	Niet lineaire bronnen: Polynoom . . . . .	69
5.2.5	Niet lineaire bronnen: Tabel . . . . .	70
5.3	Blackboxes in PSPICE . . . . .	71
5.3.1	Blackbox syntax . . . . .	71
5.3.2	Blackboxes en parameters . . . . .	72
5.4	Constanten en formules in PSPICE . . . . .	73
5.5	Modellenbouw . . . . .	73
5.5.1	Modellering R(V) . . . . .	74
5.5.2	Modellering C(V) . . . . .	76
<b>II</b>	<b>Meetsysteem</b>	<b>79</b>
	Inleiding	79
<b>7</b>	<b>Het pulscircuit</b>	<b>80</b>
7.1	Theoretische beschouwing . . . . .	81
7.1.1	Stroom $i(t)$ . . . . .	81
7.1.2	Condensatorspanning $U_c(t)$ . . . . .	83
7.2	Berekening $I^2t$ . . . . .	85

<b>8</b>	<b>Het meetcircuit</b>	<b>87</b>
8.1	Het meetprincipe . . . . .	87
8.2	Het circuit . . . . .	88
8.2.1	De differentiaalversterker . . . . .	88
8.2.2	De differentiaalingangen . . . . .	92
8.3	De digitizer . . . . .	93
8.3.1	Kwantisering . . . . .	93
8.3.2	Signaal optimalisering . . . . .	94
8.3.3	Toepassen van offset stroom . . . . .	96
<b>9</b>	<b>BACOS Software</b>	<b>99</b>
9.1	Beschrijving . . . . .	99
9.2	Systeemeisen . . . . .	99
9.3	Gebruik software . . . . .	100
9.3.1	Handleiding BACOS . . . . .	100
9.3.2	Specificatie van ASCII-uitvoerfile . . . . .	102
9.3.3	Specificatie van Matlab-uitvoerfile . . . . .	102
9.4	Ontwerp software . . . . .	103
9.5	Uitbreiding software . . . . .	103
<b>III</b>	<b>Literatuur</b>	<b>104</b>
<b>10</b>	<b>Literatuurstudie</b>	<b>104</b>
10.1	Doel literatuurstudie . . . . .	104
10.2	Conclusies literatuurstudie . . . . .	105
10.3	Definitieve literatuurlijst . . . . .	106
10.3.1	Bronnen VUBIS . . . . .	106
10.3.2	Bronnen EEA en CCA . . . . .	107
<b>IV</b>	<b>Bijlagen van deel 1</b>	<b>110</b>
<b>A</b>	<b>Literatuurlijst</b>	<b>110</b>
<b>B</b>	<b>Materiaalgegevens Nikkel</b>	<b>112</b>
B.1	Soortelijke elektrische weerstand van Nikkel $\rho$ . . . . .	113
B.2	Soortelijke warmte Nikkel $c$ . . . . .	114
B.3	Thermische geleidbaarheid Nikkel $\lambda$ . . . . .	116
B.4	Eigenschappen lucht. . . . .	117
B.5	Materiaalgegevens glas. . . . .	118
<b>C</b>	<b>Afmetingen patroon 217.315</b>	<b>119</b>
<b>D</b>	<b>PSPICE listing NI2.CIR</b>	<b>120</b>
<b>E</b>	<b>It-curven data</b>	<b>128</b>

---

<b>V</b>	<b>Bijlagen van deel 2</b>	<b>129</b>
<b>F</b>	<b>Schema differentiaalversterker</b>	<b>129</b>
<b>G</b>	<b>Koppelen meetcircuit</b>	<b>131</b>
<b>H</b>	<b>BAKKER formaat</b>	<b>133</b>
<b>I</b>	<b>MATLAB formaat</b>	<b>137</b>
<b>J</b>	<b>Pascal TYPEN overzicht</b>	<b>141</b>
<b>K</b>	<b>Listings BACOS</b>	<b>142</b>
<b>L</b>	<b>Listings BACOS</b>	<b>145</b>
<b>M</b>	<b>Listings BACOS</b>	<b>162</b>

# Voorwoord

Nu de verslaglegging grotendeels achter de rug is en ik bezig ben met het schrijven van dit voorwoord, realiseer ik me steeds meer de waarde van het afstudeerwerk.

Het doel, het verkrijgen van inzicht in de temperatuurverdeling over de smeltdraad, is bereikt. Het simulatiemodel is gestructureerd opgezet, zodat latere verfijningen, wijzigingen en uitbreidingen eenvoudig kunnen worden doorgevoerd.

Ook het feit dat kan worden gesimuleerd met willekeurige stromen, maakt het model zeer waardevol.

Ik heb met veel plezier aan de afstudeeropdracht gewerkt. Vooral het multi-disciplinaire karakter van het onderwerp (combinatie elektrotechniek, fysica en informatica) heeft een extra dimensie aan het geheel gegeven.

Het zelfstandig tot een goed resultaat brengen van een dergelijk onderwerp geeft aan dat je in staat bent je ook op minder bekend terrein te bekwamen.

Rest mij de verschillende mensen uit de vakgroep EG te bedanken, die allen op hun manier steeds behulpzaam waren bij het vinden van oplossingen bij optredende problemen.

In het bijzonder wil ik promovendus X.Z. Meng bedanken voor de dagelijkse begeleiding en de steeds weer nieuwe ideeën. Ik heb gemerkt dat verschillende taal en cultuur geen belemmering hoeven te zijn voor een goede samenwerking.

Daarnaast dank ik Prof. Ir. G.J. Damstra en Ir. J.P.J. Sloot voor de begeleiding en de heren L. Vermeij, P. van Rietschoten en J. Jaspar van Littelfuse Nederland voor de samenwerking en het ter beschikking stellen van materiaal en gegevens.



# Inleiding

Ten behoeve van een duidelijke structuur in de verslaglegging bestaat dit afstudeerwerk uit drie delen.

Deel 1 vormt het meest belangrijke deel van het afstudeerwerk en beschrijft in detail de thermische modelvorming van de patroon. Uitgaande van een thermisch model wordt een elektrisch-netwerk analogon ontwikkeld, waarmee het mogelijk wordt om dit door te rekenen met netwerkanalyse programmatuur.

Conclusies en aanbevelingen worden gegeven en er is een beknopte uiteenzetting van de mogelijkheden van PSPICE, het netwerkanalyse programma dat bij de berekeningen is gebruikt.

Deel 2 beschrijft de theorie en praktische uitvoering van een testcircuit dat wordt gebruikt voor het doen van metingen aan de miniatuur smeltpatronen enerzijds en het eraan gekoppelde meetcircuit en de bijbehorende dataverwerking anderzijds.

Deel 3 bevat de verslaglegging van de literatuurstudie die aan het begin van het afstudeerperiode is uitgevoerd. Resultaten van deze studie hebben voor een groot deel het vervolg van de werkzaamheden bij het afstuderen bepaald.

Het afstudeerwerk is verricht op de Technische Universiteit Eindhoven, bij de faculteit E, vakgroep EG in de periode december 1991 tot augustus 1992 onder begeleiding van de promovendus M. Sc. X.Z. Meng en de heren Prof. ir. G.J. Damstra en ir. J.G.J. Sloot.

# Deel I. PSPICE Model

## 1. Modelvorming

*Dit hoofdstuk beschrijft de modelvorming van de smeltdraad in een miniatuur smelt patroon in detail.*

*Uitgaande van de thermische wetten (Fourier, Newton) wordt een elektrisch analo gon opgesteld voor simulatie in PSPICE.*

### 1.1 Lijst van symbolen

$\gamma$	$\frac{kg}{m^3}$	specifieke massa
$\epsilon$	-	emissiviteit
$\lambda$	$\frac{W}{mK}$	warmtegeleidingscoëfficiënt
$\mu$	$\frac{kg}{m\dot{s}}$	dynamische viscositeit
$\nu$	$\frac{m^2}{s}$	kinematische viscositeit
$\rho$	$\Omega m$	soortelijke elektrische weerstand
$\sigma$	$\frac{W}{m^2 K^4}$	constante van Stefan Boltzmann ( $5,67 \cdot 10^{-8}$ )
$c$	$\frac{J}{kgK}$	soortelijke warmte
$h$	$\frac{W}{m^2 K}$	convectie overdrachtcoëfficiënt
$q_i$	W	warmte toevoer in segment
$q_s$	W	warmte opslag in element
$q_k$	W	warmte transport t.g.v. geleiding
$q_c$	W	warmte transport t.g.v. convectie
$q_r$	W	warmte transport t.g.v. straling

$T_d$	K	temperatuur van het draadsegment
$T_o$	K	temperatuur van de omgeving
A	$m^2$	oppervlak (cilindervormige) segment doorsnede
S	$m^2$	oppervlak (cilindervormige) segment wand
l	m	lengte van (cilindervormige) segment
L	m	lengte van totale draad
$r_d$	m	straal van de draad
$r_g$	m	straal binnenkant glas
$r_o$	m	straal buitenkant glas
n	-	aantal segmenten in het model
Gr	-	Grashof getal
Pr	-	Prandtl getal
Ra	-	Rayleigh getal
Nu	-	Nusselt getal

## 1.2 Inleiding

Om inzicht te krijgen in de verdeling van temperatuur over de metalen smeltdraad in een miniatuurpatroon in axiale richting, bij een opgedrukte elektrische stroom  $i(t)$ , is een simulatiemodel ontwikkeld. Dit model is gebaseerd op een artikel van K.W. Awkward [1] en geheel omgewerkt voor het gebruik van transiënte stromen.

Er wordt gebruik gemaakt van het analogon tussen de thermische eigenschappen van de draad inclusief omgeving enerzijds en een elektrisch netwerkmodel anderzijds.

Voor het doorrekenen (simuleren) van het elektrisch model is gebruik gemaakt van een bestaand software pakket met de naam PSPICE (PC version of Simulation Program with IC Emphasis). Dit pakket is toonaangevend op het gebied van elektrische netwerksimulaties op Personal Computers en biedt een groot scala aan mogelijke netwerkcomponenten en gebruikersfuncties. Zie hoofdstuk 5 voor een meer uitgebreide beschrijving van PSPICE.

Omdat bij de modellering materiaaleigenschappen kunnen worden opgegeven, is het model in principe geschikt voor ieder soort materiaal, waarvan minimaal de volgende parameters bekend zijn:

### 1. Als functie van temperatuur:

- Soortelijke weerstand  $\rho$ ;
- Soortelijke warmte  $c$ ;
- Thermische geleidbaarheid  $\lambda$ .

### 2. Bij kamertemperatuur:

- Soortelijke dichtheid  $\gamma$ ;
- Afmetingen.

Simulaties kunnen worden uitgevoerd bij excitatie met een willekeurige elektrische stroomvorm (mits binnen PSPICE te definiëren).

### 1.3 Inleiding warmte-theorie

Wanneer een stroom door een geleider wordt gestuurd, zal ten gevolge van de elektrische weerstand van het materiaal energie in warmte worden omgezet. Deze energie zorgt voor temperatuurstijging ten opzichte van de omgeving met als gevolg warmtetransport in het materiaal. De ontwikkelde warmte per tijdseenheid wordt aangeduid met  $q_i$ .

$$q_i = i^2(t) \cdot r(t) \quad (1.1)$$

Een deel van de ontwikkelde warmte zal worden gebruikt voor opslag van warmte in het materiaal. De hoeveelheid opgeslagen warmte per volume  $\Delta x \Delta y \Delta z$  per seconde is gelijk aan:

$$q_s = \gamma C \Delta x \Delta y \Delta z \frac{\partial T}{\partial t} [W] \quad (1.2)$$

Hierin is  $\gamma$  het soortelijk gewicht van het materiaal [ $\frac{kg}{m^3}$ ] en  $C$  de soortelijke warmte [ $\frac{J}{kgK}$ ].

Warmtetransport ontstaat zodra er sprake is van temperatuurverschillen (in het materiaal of tussen materiaal en omgeving). Dit transport is mogelijk in een drietal varianten: **conductie** (geleiding), **convectie** en **straling** (radiatie). De bijdrage van iedere variant aan het totale warmtetransport is per situatie verschillend.

De basisvergelijking voor warmtegeleiding in een materiaal staat bekend als *de wet van Fourier* en luidt voor één dimensie als volgt:

$$q_k = -\lambda A \frac{\partial T}{\partial x} [W] \quad (1.3)$$

Hierbij is  $T(x)$  de lokale temperatuur [ $K$ ],  $x$  de afstand (positief in de richting van de warmtestroom) [ $m$ ],  $A$  het oppervlak waar doorheen de warmtestroom plaatsvindt [ $m^2$ ] en  $\lambda$  een evenredigheidsconstante: thermische geleidbaarheid [ $\frac{W}{mK}$ ]. Het minteken geeft aan dat de warmtestroom zich verplaatst van de hoge naar de lage temperatuur.

Convectie treedt op bij een overgang tussen een materiaal en een vloeistof (gas) en is een moeilijk te modelleren proces. Er bestaan empirische relaties, geldig voor verschillende geometrische configuraties waarbij ondermeer onderscheid gemaakt wordt tussen vrije- en gedwongen convectie, in vrije- of begrensde ruimte.

De mate van warmte afgifte door convectie is vastgelegd in een door Isaac Newton voorgestelde wet:

$$q_c = hS\Delta T [W] \quad (1.4)$$

Hierbij is  $S$  het oppervlak waardoor de warmtestroom zich verplaatst [ $m^2$ ],  $h$  de coëfficiënt van convectie [ $\frac{W}{m^2K}$ ] en  $\Delta T$  het temperatuurverschil tussen oppervlak en omringend medium [ $K$ ]. De parameter  $h$  karakteriseert de eigenschappen van de betreffende soort convectie en moet per situatie worden bepaald. In de meeste gevallen is  $h$  temperatuurafhankelijk.

Een derde mogelijkheid voor warmtetransport wordt gevormd door straling. Deze vorm van transport treedt op bij alle voorwerpen met een temperatuur boven  $0K$ . Planck vond voor een perfecte straler (zwarte straler) de volgende relatie:

$$q_r = \sigma ST^4 [W]$$

Hierin is  $\sigma = 5.67 \cdot 10^{-8} [\frac{W}{m^2 K^4}]$  de constante van Stefan-Boltzmann, S het stralingsoppervlak [ $m^2$ ] en T de temperatuur van de straler [K].

In praktijk zal er geen sprake zijn van een zwarte straler en is een correctie nodig. Dit gebeurt door de introductie van een emissiviteits-factor  $\epsilon$ . Deze factor is altijd kleiner dan 1 en temperatuur afhankelijk. Wanneer daarbij de temperatuur van de omgeving gelijk aan  $T_2$  is, volgt uit de wet van Planck voor de netto warmtestroom van 1 naar 2:

$$q_r = \sigma(S_1\epsilon_1T_1^4 - S_2\epsilon_2T_2^4)[W] \quad (1.5)$$

Hierbij wordt aangenomen, dat zowel oppervlak 1 als oppervlak 2 zichzelf niet zien (en daardoor eigen straling absorberen). Een platte plaat voldoet hier aan, een inwendige wand van een cilinder niet. Deze laatste situatie doet zich juist bij deze modellering voor, zodat uitdrukking 1.5 niet direct mag worden gebruikt.

Door het introduceren van een geometrie-factor kan echter alsnog een uitspraak worden gedaan over de grootte van het optredend warmtetransport. Dit komt in paragraaf 1.6.5 aan de orde.

Samenvattend kan de volgende energiebalans worden opgesteld, waaraan steeds zal worden voldaan:

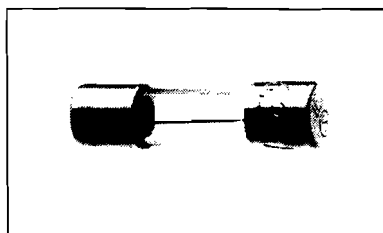
$$\begin{aligned} \text{toevoer} &= \text{opslag} + \text{geleiding} + \text{convectie} + \text{straling} \\ q_i &= q_s + q_k + q_c + q_r \end{aligned} \quad (1.6)$$

Bij de modellering zal voor iedere term een juiste interpretatie moeten worden gevonden.

## 1.4 Het thermisch model

Bij het opstellen van een thermisch model voor de miniatuurpatroon is het de bedoeling om de werkelijkheid zo goed mogelijk te benaderen. Dat betekent dat de diverse warmtestromen, zoals aangegeven in paragraaf 1.3, op een of andere manier ook in het model moeten worden opgenomen.

Het moge duidelijk zijn dat deze stap een van de belangrijkste (en ook moeilijkste) onderdelen is van de gehele simulatie.



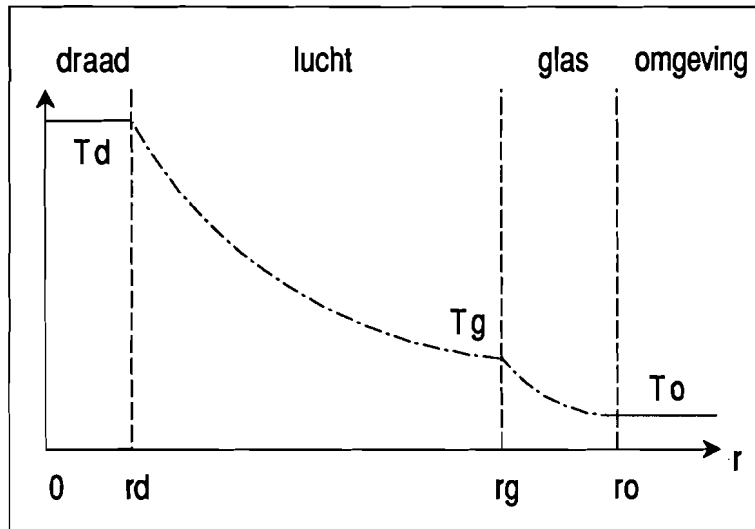
Figuur 1.1: Afbeelding van een miniatuurpatroon met luchtvulling (lengte 2cm).

In figuur 1.1 is een afbeelding te zien van een miniatuurpatroon, zoals deze veelvuldig in de kleinere elektrotechnische apparaten te vinden is.

De patroon bestaat uit een metalen draad, ingesoldeerd tussen twee metalen einden, de *caps* genaamd. Afhankelijk van de ontwerpstroom (rating) van de patroon, is er sprake van een enkele materiaalsoort of een legering. Ook de diameter van de draad hangt hier mee samen.

Om de draad heen zit een glazen cilinder, ingeklemd tussen de metalen caps. De dunne metaal-draad zit daarmee in de cilinder opgeborgen. Het glas vormt een hechte verbinding met de caps, zodat een solide geheel ontstaat.

Bij de modellering van de miniatuurpatroon zal het wenselijk zijn om ook de invloed van de glazen cilindrische omhulling te beschouwen. Naast de draad dient daarom ook de lucht en het glas te worden gemodelleerd. Dit is in afbeelding 1.2 verduidelijkt. De afbeelding toont een dwarsdoorsnede van de patroon. Achtereenvolgens zijn van binnen naar buiten de nikkeldraad, lucht, glas en omgeving te onderscheiden. Per gebied is steeds een indicatie van het temperatuurniveau aangegeven.



Figuur 1.2: Een dwarsdoorsnede van miniatuurpatroon met daarin de diverse temperatuurgebieden.

Voor de modellering van de miniatuurpatroon zijn per segment de volgende aannames gemaakt (werkend vanuit hart van de draad naar omgeving):

1. In het draadsegment ( $0 < r < r_d$ ):
  - er is sprake van één draadtemperatuur in radiale richting;
  - opname van warmte door opgenomen elektrisch vermogen;
  - geleiding van warmte in axiale richting (eendimensionaal);
  - opslag van warmte.
2. De luchtgevulde ruimte tussen draad en glaswand ( $r_d < r < r_g$ ):
  - temperatuur daalt van draadtemperatuur naar glaswandtemperatuur;
  - vrije convectie in radiale richting;
  - opslag van warmte in de lucht.
3. In het glas ( $r_g < r < r_o$ ):
  - temperatuur daalt van glaswandtemperatuur naar omgevingstemperatuur;
  - geleiding van warmte in radiale richting;
  - opslag van warmte.
4. De omgevingslucht buiten het glas ( $r \geq r_o$ ):
  - temperatuur blijft op omgevingstemperatuur.

## 5. Tussen draad en omgeving:

- straling.

## 1.5 Inleiding elektrische modellering

Om het thermisch model geschikt te maken voor simulatie in PSPICE, wordt het omgewerkt naar een elektrisch equivalent.

Hierbij wordt het volgende analogon tussen thermisch en elektrisch model gehanteerd:

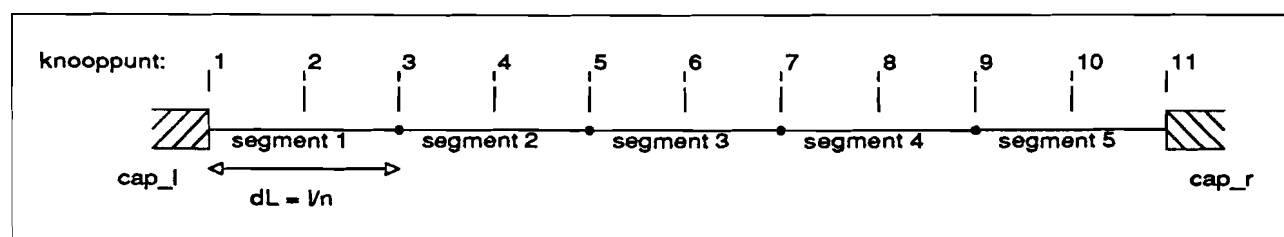
temperatuur $T$	$\Rightarrow$	spanning $V$
warmtestroom $q$	$\Rightarrow$	elektrische stroom $I$
thermische weerstand $R_{th}$	$\Rightarrow$	elektrische weerstand $R_{el}$
thermische opslag $C_{th}$	$\Rightarrow$	elektrische opslag $C_{el}$

Gebruik makend van de in de vorige paragraaf uiteengezette aannames en het geldend analogon volgt:

- De door de elektrische stroom in de draad opgewekte hoeveelheid warmte zal door een elektrische bron met het karakter van een stroombron worden verwezenlijkt;
- Opslag van warmte in de draad, lucht en in het glas worden gemodelleerd met een elektrische condensator;
- Warmtegeleiding in de draad en het glas, alsmede convectie en straling wordt voorgesteld met een elektrische weerstand;

Per elektrische component zal het juiste gedrag van het thermische equivalent moeten worden nagegaan. Het juist dimensioneren ervan bepaalt immers het al dan niet slagen van de simulatie. Dit wordt in paragraaf 1.6 verder uiteengezet.

### 1.5.1 Draadsegmentering



Figuur 1.3: Het in  $n$  segmenten opdelen van de metalen draad ( $n=5$ ).

Om het temperatuurgedrag van de draad te kunnen simuleren is het noodzakelijk om de draad in de lengterichting op te delen in een aantal draad-segmenten (diskretiseren). De temperatuur per segment wordt hierbij verondersteld constant te zijn. Naar mate het aantal segmenten toeneemt, zal een betere benadering ontstaan van het continue temperatuurverloop.

Men dient wel te bedenken, dat hierdoor het model groter wordt, wat zal leiden tot langere rekestijden. Er moet dus een compromis worden gesloten tussen de gewenste nauwkeurigheid en het aantal segmenten.

In verband met symmetrie is voor het aantal segmenten  $n$  een oneven waarde gekozen. De consequenties hiervan worden belicht in paragraaf 1.5.5.

Figuur 1.3 toont de metalen draad, die is opgedeeld in een 5-tal segmenten ( $n=5$ ), ieder met een lengte  $l = \frac{L}{n}$ . De oneven knooppunten 1 t/m 11 markeren de overgang tussen de diverse segmenten. Even knooppunten 2 t/m 10 markeren het midden van een segment.

De uiteinden van de draad (bij knooppunt 1 en 11) worden verbonden met het model van de caps. Het gedrag van de temperatuur van de caps wordt bepaald door het model ervan dat in paragraaf 1.5.4 worden beschreven.

Voor wat betreft de relatie tussen het aantal segmenten ( $n$ ), aantal knopen ( $t$ ) en het knooppuntnummer in het midden van de draad ( $m$ ), geldt het volgende:

$$t = 2n + 1 \quad \text{en} \quad m = n + 1$$

### 1.5.2 Vastleggen referentie

Wanneer er in een elektrisch circuit sprake is van een spanning, wordt dit gegeven t.o.v. een spanningsreferentie of nulniveau. In de analogie met temperatuur betekent dit automatisch dat er dan ook sprake moet zijn van een referentie in temperatuur. Een spanning gelijk aan de referentiespanning impliceert een temperatuur gelijk aan de referentietemperatuur.

Dit gegeven heeft voor de modellering tot de volgende afspraak geleid:

Een spanning gelijk aan de referentiespanning (= aarde) wordt aangeduid met  $V = 0V$  en komt overeen met een temperatuur gelijk aan kamertemperatuur  $T = 300K$ .

Bij aanvang van de simulatie hebben alle knooppunten een spanning gelijk 0 Volt, hetgeen overeenkomt met een temperatuur van 300 Kelvin: de draad is op omgevingstemperatuur.

Knooppunt 0 is in het model steeds aarde.

### 1.5.3 Segment opbouw

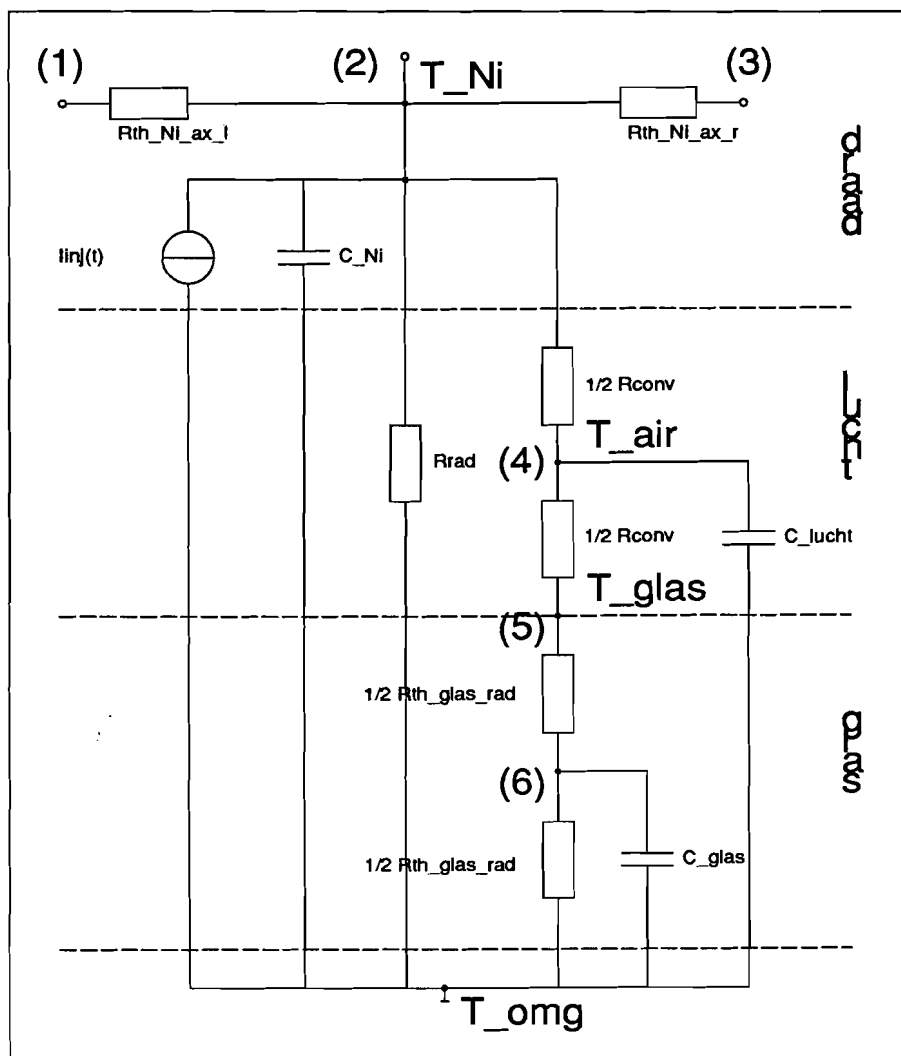
Per draadsegment geldt een elektrisch netwerk zoals getekend in figuur 1.4. Hierin zijn alle in paragraaf 1.4 genoemde componenten opgenomen. In de figuur zijn ter verduidelijking 4 verschillende gebieden aangegeven, gescheiden door een stippellijn. Van boven naar beneden zijn dit: de draad, de lucht, het glas en de omgeving (lucht).

Alle componenten in het segment zijn op een bepaalde manier tussen de verschillende lagen geschakeld. Voor een gedetailleerde beschrijving van iedere component wordt verwezen naar paragraaf 1.6.

Het segment kan worden gezien als een blackbox met een zestal aansluitingen naar buiten, genummerd 1 t/m 6.

De betekenis van de knooppunten is als volgt:





Figuur 1.4: Elektrisch analoog van thermisch model van een draadsegment.

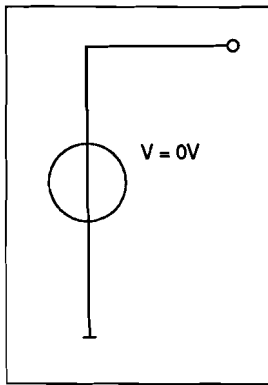
1. is de linkerkant van het draadsegment en wordt, bij aaneensluiting van meerdere segmenten, gekoppeld aan knooppunt 3 van het naastliggende segment;
2. is het hart van het segment. De spanning op dit knooppunt komt overeen met de temperatuur van het segment. Waarden van thermische-, convectie en stralingsweerstand, capaciteit en elektrische vermogensdissipatie worden berekend aan de hand van de spanning op dit knooppunt (de temperatuur van het segment);
3. is de rechterkant van het draadsegment en wordt, bij aaneensluiting van meerdere segmenten, gekoppeld aan knooppunt 1 van het naastliggende segment;
4. is een intern knooppunt met een spanning overeenkomend met een temperatuur ergens in de lucht tussen draad en glazen wand. De waarde ervan hangt ondermeer af van de gekozen convectie-verhouding en vertegenwoordigt een soort gemiddelde temperatuur. De temperatuur wordt gebruikt bij de berekening van de hoeveelheid warmte die wordt opgeslagen in de lucht;

5. is een intern knooppunt met een spanning die overeenkomt met de temperatuur op het scheidingsvlak van de lucht en de glazen wand;
6. is een intern knooppunt met een spanning die overeenkomt met de temperatuur ergens in het glas. De waarde wordt ondermeer bepaald door de materiaalgegevens van het glas.

Met betrekking tot de modellering kunnen nog de volgende opmerkingen worden gemaakt:

- In de draad is alleen warmtegeleiding in axiale richting gemodelleerd. Door de geringe diameter van de draad zal de draad in radiale zin egaal opgewarmd worden verondersteld. Hierdoor vindt geen warmtegeleiding in radiale richting plaats;
- In de lucht is er geen vorm van warmte-uitwisseling in een richting evenwijdig aan de geleider;
- In het glas is er geen vorm van warmte-uitwisseling in een richting evenwijdig aan de geleider. Dit effect is verwaarloosd, daar de temperatuurgradiënt in die richting klein gebleken is.

#### 1.5.4 Modellering eindcaps



Figuur 1.5: Modelvorming van een metalen eindcap.

In het model worden de uiteinden van de draad op een constante potentiaal  $V = 0V$  gebracht. De eindcaps blijven op kamertemperatuur. In het model is in plaats van een directe verbinding naar aarde gekozen voor een spanningsbron, met een bronspanning  $V = 0V$ . Hierdoor is het in een later stadium mogelijk, om de eindcaps een bepaalde vaste temperatuur te geven.

Blijkt in een later stadium uit experimenten dat de temperatuur van de eindcaps gedurende het experiment niet constant verondersteld mag worden, dan kan deze spanningsbron worden vervangen door bijvoorbeeld een capaciteit of iets dergelijks.

#### 1.5.5 Aanpassing gebruik half segment

Omdat de miniatuurpatroon, zoals getekend in figuur 1.3, in axiale richting symmetrie vertoont, is het voor de modellering voldoende om slechts één helft te beschouwen: één cap en een halve draad.

In dat geval zal het middelste segment (nummer  $\frac{(n+1)}{2}$ ) in lengte worden gehalveerd, wat betekent dat de componentwaarden in dit segment wijzigen. Hierbij gelden de volgende veranderingen (zie ook figuur 1.4):

- De thermische weerstand aan de rechterkant van het centrale knooppunt ( $R_{th\_Ni\_ax\_r}$ ) vervalt. De linker weerstand halveert;
- Omdat de grootte van het segment is gehalveerd, betekent dit voor de capaciteiten van draad, lucht en glas eveneens een halvering. Het volume halveert immers;
- De dissipatie van elektrische energie halveert, omdat door halveren van de segmentlengte ook de elektrische weerstand van het segment halveert;
- De thermische weerstand in het glas verdubbelt, omdat het geleidend oppervlak halveert;
- De waarden van convectie- en stralingsweerstand verdubbelen, omdat door de halvering van de segmentlengte de omhullende oppervlakken eveneens halveren.

Door halvering van het model ontstaat er bijna een halvering in het aantal knooppunten en zullen simulaties een kortere rekentijd hebben, zonder verlies van informatie. Het is nu echter niet meer mogelijk om te simuleren met twee eindcaps op verschillende temperatuur (asymmetrisch geval).

## 1.6 Het elektrisch model

In deze paragraaf zullen alle componenten waaruit het segment is opgebouwd in detail worden besproken en worden formules afgeleid. In hoofdstuk 2 is uiteengezet hoe de formules in PSPICE termen moeten worden gerealiseerd.

### 1.6.1 Warmte injectie: $I_{inj}(t,V)$

Met de opgedrukte stroom  $i(t)$  door de draad, zal ten gevolge van de elektrische weerstand ervan per tijdseenheid een hoeveelheid energie in de vorm van warmte worden geïnjecteerd. Dit vindt plaats over de gehele lengte van de draad. In het model zal deze injectie van warmte plaatsvinden in de segmenten en wel op de knooppunten 2, 4, ...  $t/m$   $2n$  van figuur 1.4. De grootte van dit vermogen hangt af van de waarde van de elektrische stroom en elektrische segment weerstand. De elektrische stroom is afhankelijk van de tijd, terwijl de elektrische weerstand afhankelijk is van de temperatuur van het segment. Het geïnjecteerde vermogen is daarom afhankelijk van tijd en segment temperatuur volgens:

$$P_{inj}(t,T) = i(t)^2 \cdot R_{el,segment}(T) = i(t)^2 \cdot \frac{\rho(T)l}{A} = i(t)^2 \cdot \frac{\rho(T)L}{n\pi r_d^2}$$

Voor de bepaling van de elektrische geleidbaarheid  $\rho$  als functie van de temperatuur is gebruik gemaakt van referentie [4]. In bijlage B.1 is deze relatie zowel grafisch als in tabelvorm weergegeven.

De totale lengte  $L[m]$  en straal van de draad  $r_d[m]$  volgen uit de afmetingen van de draad (zie ook bijlage C).

Bij de modellering wordt volgens het analogon het geïnjecteerde vermogen (de warmtestroom) door een stroombron  $I_{inj}$  gerealiseerd. De grootte temperatuur gaat hierbij over in spanning.

Hiermee ontstaat voor de stroombron de volgende formule:

$$I_{inj}(t, V) = i(t)^2 \cdot \frac{\rho(V)L}{n\pi r_d^2} \quad (1.7)$$

**Opmerking:** Omdat voor de spanning  $V = 0$  als referentie is gekozen en dit overeenkomt met kamertemperatuur (300K), is in formule 1.7  $\rho(0)$  de elektrische geleidbaarheid bij kamertemperatuur. Ditzelfde doet zich ook voor bij alle andere fysische grootheden als  $\lambda$ ,  $c$  en  $h$  die in de nog volgende formules worden gebruikt.

In het elektrisch model wordt de stroombron  $I_{inj}$  tussen aarde en het centrale segment knooppunt geplaatst. De spanning over de bron is een maat voor de temperatuur van het segment en wordt bij de berekening van de stroom als zodanig gebruikt (zie figuur 1.4).

In paragraaf 2.1.7 wordt uiteengezet hoe de realisering van formule 1.7 in PSPICE tot stand is gekomen.

### 1.6.2 Opslag van warmte: $C(V)$

Voor de bepaling van een algemene uitdrukking voor de elektrische capaciteit uit het analogon met thermische capaciteit voor nikkel, lucht en glas, wordt uitgegaan van vergelijking 1.2:

$$q_s = \gamma c \Delta x \Delta y \Delta z \frac{\partial T}{\partial t}$$

Uit het analogon  $q_s \rightarrow I$  en  $T \rightarrow V$  volgt een relatie tussen spanning en stroom als bij een capaciteit, zodat  $\gamma c \Delta x \Delta y \Delta z \rightarrow C_{el}$ .

Omdat de soortelijke warmte  $c$  temperatuurafhankelijk is, volgt voor de elektrische capaciteit:

$$C_{el} = C_{el}(V) = \gamma \cdot c(V) \cdot \Delta x \Delta y \Delta z$$

Dit gaat over in:

$$C_{el}(V) = \gamma \cdot volume \cdot c(V)$$

Een juiste uitdrukking voor het volume hangt af van de geometrie en wordt nog gegeven.

Voor de bepaling van het volume van respectievelijk de draad, de lucht en het glas wordt gebruik gemaakt van de afmetingen van de miniatuurpatroon, zoals getekend in bijlage C. Dit levert de volgende tabel met daarin de volumes en de uiteindelijke capaciteit formules:

capaciteit	specifieke warmte	bijlage	volume	capaciteit formule
CNi	$c_{Ni}$	B.2	$\pi r_d^2 l$	$\gamma_{Ni} \cdot \frac{\pi L}{n} r_d^2 \cdot c_{Ni}(V)$
Clucht	$c_{air}$	B.4	$\pi(r_g^2 - r_d^2)l$	$\gamma_{lucht} \cdot \pi(r_g^2 - r_d^2) \frac{L}{n} \cdot c_{lucht}(V)$
Cglas	$c_{glas}$	B.5	$\pi(r_o^2 - r_g^2)l$	$\gamma_{glas} \cdot \pi(r_o^2 - r_g^2) \frac{L}{n} \cdot c_{glas}(V)$

Door het plaatsen van de condensator tussen het centrale segmentknooppunt en aarde (zie figuur 1.4), is de spanning over de capaciteit direct een maat voor de temperatuur van het segment.

In paragraaf 2 wordt uiteengezet hoe de formules in PSPICE kunnen worden gerealiseerd.

### 1.6.3 Thermische geleidingsweerstand: $R_{th}(V)$

Ten gevolge van het optredend temperatuurverschil tussen twee naast elkaar gelegen segmenten, ontstaat een warmtestroom (van hoge naar lage temperatuur).

De grootte ervan wordt bepaald door zowel het temperatuurverschil als de materiaaleigenschappen en het gedrag kan worden uitgedrukt in een thermisch weerstands-equivalent. Uitgangspunt hierbij vormt de wet van Fourier (formule 1.3):

$$q_k = -\frac{\lambda(T) \cdot oppervlak}{d(lengte)} dT \xrightarrow{\text{analogon}} I = \frac{\lambda(V) \cdot oppervlak}{lengte} (V - V_0) = \frac{\lambda(V) \cdot oppervlak}{lengte} V$$

Dit levert voor het elektrisch weerstandequivalent de volgende uitdrukking:

$$R_{th}(V) = \frac{lengte}{oppervlak \cdot \lambda(V)} \quad (1.8)$$

In het model is op een tweetal plaatsen sprake van thermische geleiding: in axiale richting in de draad, tussen de verschillende segmenten en in radiale richting in het glas.

Geleiding in radiale richting in de draad is er niet, daar bij de modellering is aangenomen dat de draad (in radiale richting) een constante temperatuur heeft.

Uiteraard is er ook geleiding in radiale richting in de lucht. Deze bijdrage wordt reeds meegenomen bij de modellering van de convectie, zoals beschreven in paragraaf 1.6.4.

Voor beide modelweerstandsen gelden specifieke uitdrukkingen voor de variabelen *lengte* en *oppervlak* uit formule 1.8.

Transport van de warmte in axiale richting door de draad gaat door een cilindersegment met lengte  $\frac{l}{2} = \frac{L}{2n}$  (figuur 1.6 links). Hierbij wordt de halve segmentlengte gebruikt, omdat per segment twee aparte thermische weerstanden worden gebruikt: naar links ( $R_{th\_Ni\_ax\_l}$ ) en naar rechts ( $R_{th\_Ni\_ax\_r}$ ). Met de gegevens uit de figuur volgt voor de thermische weerstand van de draad in axiale richting:

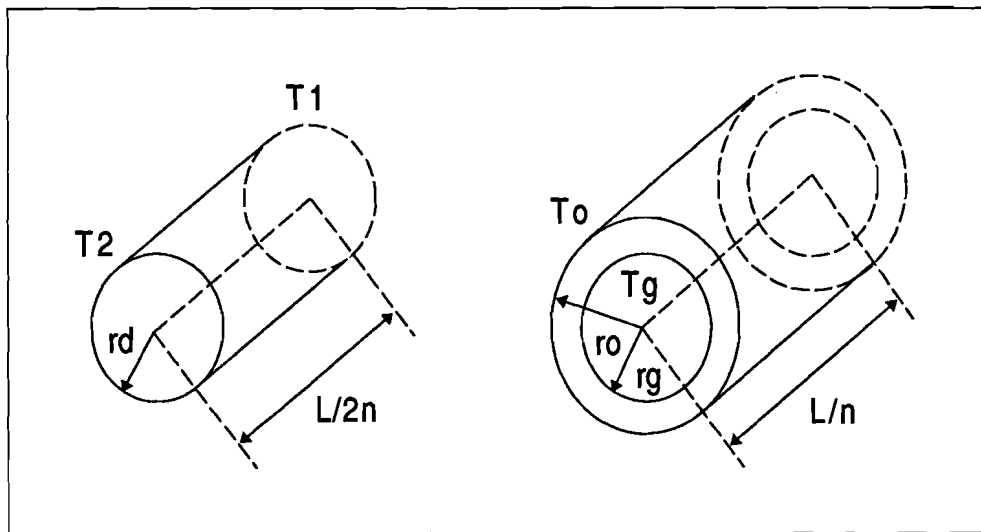
$$R_{th\_Ni\_ax\_l(r)} = \frac{L}{2\pi r^2 n \lambda(V)} \quad (1.9)$$

De thermische geleidbaarheid als functie van de temperatuur voor nikkel is weergegeven in bijlage B.3. Bij de modellering van formule 1.9 zal deze afhankelijkheid moeten worden meegenomen.

In het geval van warmtetransport door het glas in radiale richting is sprake van warmte door een uitgeholde cilindervormige structuur (figuur 1.6 rechts). Deze structuur laat zich het best beschrijven met cilindercoördinaten, waardoor formule 1.8 als volgt wordt afgeleid:

De wet van Fourier (formule 1.3) uitgedrukt in cilindercoördinaten luidt:

$$q = -\lambda(T) \cdot S \cdot \frac{dT}{dr} = -\lambda(T) \cdot 2\pi r l \cdot \frac{dT}{dr}$$



Figuur 1.6: Geometrie voor het bepalen van *lengte* en *oppervlak* voor *Rth\_Ni\_ax* (links) en *Rth\_glas\_rad* (rechts).

Splitsen van de variabelen en integreren geeft:

$$q \cdot \frac{dr}{r} = -2\pi l \cdot \lambda(T) dT \Leftrightarrow q \int_{r_g}^{r_o} \frac{1}{r} dr = -2\pi l \cdot \int_{T_g}^{T_o} \lambda(T) dT \Leftrightarrow q \ln\left(\frac{r_o}{r_g}\right) = -2\pi l \cdot \int_{T_g}^{T_o} \lambda(T) dT$$

Wil men in deze uitdrukking de integraal in het rechterlid kunnen oplossen, dan zal voor  $\lambda_{glas}(T)$  een formule moeten worden gevonden.

In bijlage B.5 staat voor glas een  $\bar{\lambda} = 1,95$  vermeld. Bij gebrek aan meer gedetailleerde informatie is deze gemiddelde waarde aangenomen. Gebruik van deze eenvoudige relatie levert:

$$q \cdot \ln\left(\frac{r_o}{r_g}\right) = -2\pi l \cdot 1,95 \cdot (T_o - T_g)$$

Voor de thermische weerstand volgt tot slot:

$$Rth_{glas\_rad} = -\frac{(T_o - T_g)}{q} = \frac{\ln\left(\frac{r_o}{r_g}\right)}{2\pi l \cdot 1,95} = \frac{n \ln\left(\frac{r_o}{r_g}\right)}{2\pi L \cdot 1,95} \quad (1.10)$$

Dit is een constante.

#### 1.6.4 Convectie weerstand: $R_{conv}(V)$

In het model moet naast thermische geleiding en capaciteit, ook convectie worden gemodelleerd. Samen met straling en geleiding vormt convectie in het model de mogelijkheid voor warmte afgifte in radiale richting.

Uitgaande van de wet van Newton, zoals genoemd in formule 1.4, kan op eenzelfde manier als bij de modellering van thermische geleidbaarheid, een weerstand voor convectie worden

gedefinieerd:

$$R_{\text{conv}}(T) = \frac{1}{S \cdot h(T)} \quad (1.11)$$

waarbij  $S$  het omhullende oppervlak is en  $h$  de convectiecoëfficiënt.

In tegenstelling tot de soortelijke warmte en elektrische- en thermische geleidbaarheid, is  $h$  als functie van temperatuur minder eenvoudig te bepalen.

In basisliteratuur over *heat transfer* (o.a. [7] t/m [13]), wordt aangegeven dat het convectiegedrag door een groot aantal factoren wordt bepaald:

- convectie is afhankelijk van de geometrie van het voorwerp (plaat, cilinder) en ook van de wijze van opstelling (vertikaal, horizontaal);
- de mate van convectie hangt af van het soort medium waarmee de warmte wordt afgevoerd (lucht, vloeistof);
- het medium kan een laminair of turbulent gedrag vertonen;
- naast pure convectie wordt ook de invloed van geleiding in radiale richting automatisch meegenomen. Beide effecten treden tegelijkertijd op en zijn als zodanig in gedane experimenten niet los van elkaar te beschouwen.

In de loop der jaren is veel onderzoek verricht naar de convectie eigenschappen van diverse geometrieën en is geprobeerd om relaties te leggen tussen de verschillende gedragingen van de convectie (inclusief radiale geleiding). Standaard formules zijn in de literatuur echter niet te vinden. Er bestaan slechts een aantal empirisch bepaalde relaties.

Hierbij is getracht om algemene relaties tussen materiaalgrootheden te verkrijgen, geldig voor een bepaalde geometrie. Dat heeft geleid tot een aantal dimensieloze grootheden, die steeds in de relaties terugkomen:

**Pr** (Prandtl getal). Geeft een relatie tussen het snelheidsveld (in het omringende medium) en het temperatuurveld (op rand van media).

$$Pr = \frac{c_p \cdot \mu}{k} \quad (1.12)$$

met:

$c_p$  : soortelijke warmte omringend medium [ $\frac{J}{kgK}$ ]

$\mu$  : dynamische viscositeit [ $\frac{kg}{ms}$ ]

$k$  : thermische geleidbaarheid omringend medium [ $\frac{W}{mK}$ ]

**Gr** (Grashof getal). Dit getal wordt gebruikt in het geval van natuurlijke convectie. Het geeft de relatie tussen opwaartse kracht en visceuze kracht en vormt het criterium voor de overgang van laminaire- naar turbulente stroming (vergelijk het Reynolds getal voor geforceerde convectie).

$$Gr_x = \frac{g \cdot \beta \cdot (T_w - T_\infty) \cdot x^3}{\nu^2} \quad (1.13)$$

met:

- $g$  : versnelling zwaartekracht [ $\frac{m}{s^2}$ ]  
 $\beta$  : thermische uitzetting van het medium. Voor een gas geldt  $\frac{1}{T_\infty}$   
 $T_w$  : temperatuur van de wand van de geleider [ $K$ ]  
 $T_\infty$  : temperatuur van het medium, ver verwijderd van de geleider [ $K$ ]  
 $\nu$  :  $= \frac{\mu}{\rho}$  kinematische viscositeit van het medium [ $\frac{m^2}{s}$ ]  
 $\rho$  : soortelijke dichtheid van het medium [ $\frac{kg}{m^3}$ ]  
 $x$  : relevante afmeting [ $m$ ]

De relevante afmeting verschilt per geometrie en is de straal in het geval van cilindergeometrie.

**Ra** (Rayleigh getal). Het produkt van Prandtl- en Grashof getal:

$$Ra = Pr \cdot Gr \quad (1.14)$$

**Nu** (Nusselt getal).

$$Nu_x = \frac{h_x \cdot x}{k} \quad (1.15)$$

met:

- $h_x$  : convectie coëfficiënt in x-richting [ $\frac{W}{m^2K}$ ]  
 $x$  : relevante afmeting [ $m$ ]  
 $k$  : thermische geleidbaarheid omringend medium [ $\frac{W}{mK}$ ]

De relevante afmeting verschilt per geometrie en is de straal in het geval van cilindergeometrie.

Wanneer het Nusselt getal als functie van temperatuur bekend is, kan hieruit met behulp van formule 1.15 de coëfficiënt van convectie worden bepaald.

## Convectiemodellen

Voor de modelvorming van convectie bij een draad zijn relaties gezocht die geldig zijn voor de volgende geometrie:

- horizontale cilindergeometrie;
- vrije convectie;
- draad in lucht;
- kleine draaddiameter (orde  $50\mu m$ ).

Om het uiteindelijk selecteren van een convectiemodel zo verantwoord mogelijk te doen, is uit de literatuur een eerste selectie gemaakt van een zestal verschillende model-relaties. Deze relaties voldoen zo goed mogelijk aan de gestelde eisen en zullen onderling worden vergeleken.

**Let op:** Alle nu volgende (empirische) convectiemodellen bevatten naast optredende convectie ook geleiding (invloeden zijn in elkaar verweven).



## Model I.

Het eerste model is gebaseerd op een afleiding zoals beschreven in Perry [13] en geeft een directe uitdrukking voor de convectie overdracht coëfficiënt:

$$h = 1,32 \cdot \left[ \frac{T_w - T_\infty}{2r} \right]^{\frac{1}{4}} \quad (1.16)$$

met

- $T_w$  : temperatuur van de wand [K]
- $T_\infty$  : temperatuur van medium, ver van wand [K]
- $r$  : straal van de geleider [m]

Met nadruk dient vermeld te worden dat dit een sterk vereenvoudigde uitdrukking is. Omwerken van  $h$  naar de convectieweerstand volgt door het invullen ervan in formule 1.11.

Holman [9] noemt deze relatie eveneens, met de opmerking dat deze relatie alleen kan worden gebruikt als

- er sprake is van laminaire stroming ( $10^4 < GrPr < 10^9$ );
- geldt dat  $r < 10\text{cm}$ ;
- er sprake is van vrije convectie;
- lucht het omringende medium is.

## Model II.

Holman [9] geeft in paragraaf 7-5 een relatie tussen de dimensieloze getallen, geldig voor een horizontale cilinder in geval van vrije convectie. Het is een uitdrukking voor een gemiddelde waarde van Nusselt getal:

$$\bar{Nu}^{\frac{1}{2}} = 0,60 + 0,387 \cdot \left[ \frac{Gr \cdot Pr}{\left[ 1 + \left( \frac{0,559}{Pr} \right)^{\frac{9}{16}} \right]^{\frac{16}{9}}} \right]^{\frac{1}{6}} \quad \text{voor } 10^{-5} < Gr \cdot Pr < 10^{12} \quad (1.17)$$

Indien de eigenschappen van de omgeving bekend zijn, kunnen de factoren  $Gr$  en  $Pr$  worden berekend als functie van de temperatuur. Omdat er in de lucht rondom de geleider geen sprake is van constante temperatuur (varieert van  $T_d$  tot  $T_o$ ), wordt een gemiddelde temperatuur gedefinieerd, aangeduid met  $T_f$ , de zogenaamde filmtemperatuur. Deze is gedefinieerd volgens

$$T_f = \frac{T_d + T_o}{2} \quad (1.18)$$

en wordt gebruikt voor de berekening van alle dimensieloze getallen. Omdat de temperatuur van de omgeving  $T_o$  constant beschouwd wordt, is de wandtemperatuur  $T_d$  de enige variabele, zodat  $Gr$  en  $Pr$  als functie van de segment temperatuur kunnen worden berekend.

Indien het produkt  $Gr \cdot Pr$  tussen de grenzen van formule 1.17 ligt, kan met die formule en na kwadrateren het Nusseltgetal als functie van de wandtemperatuur worden bepaald. Deze relatie is vastgesteld door Churchill en Chu [11].

Toepassen van formule 1.15 geeft tenslotte de uitdrukking van  $h(T_d)$ . Hieruit kan met behulp van formule 1.11 een uitdrukking voor de convectieweerstand worden gevonden.

### Model III.

Wanneer wordt voldaan aan  $10^{-6} < GrPr < 10^9$  kan de relatie uit formule 1.17 volgens Churchill [11] worden vereenvoudigd tot:

$$Nu = 0,36 + 0,518 \cdot \left[ \frac{Gr \cdot Pr}{\left[ 1 + \left( \frac{0,559}{Pr} \right)^{\frac{9}{16}} \right]^{\frac{16}{9}}} \right]^{\frac{1}{4}} \quad (1.19)$$

Evenals bij de twee voorgaande modellen kan ook nu uit het Nusselt getal de convectie overdracht coëfficiënt worden bepaald.

### Model IV.

In Churchill en Chu [11] wordt nog een uitdrukking geponeerd, die veel lijkt op de relatie uit model II:

$$Nu = 0,36 + 0,521 \cdot \left[ \frac{Gr \cdot Pr}{\left[ 1 + \left( \frac{0,442}{Pr} \right)^{\frac{9}{16}} \right]^{\frac{16}{9}}} \right]^{\frac{1}{4}} \quad (1.20)$$

Deze relatie kan worden gebruikt als er sprake is van laminaire stroming ( $GrPr < 10^9$ ).

### Model V.

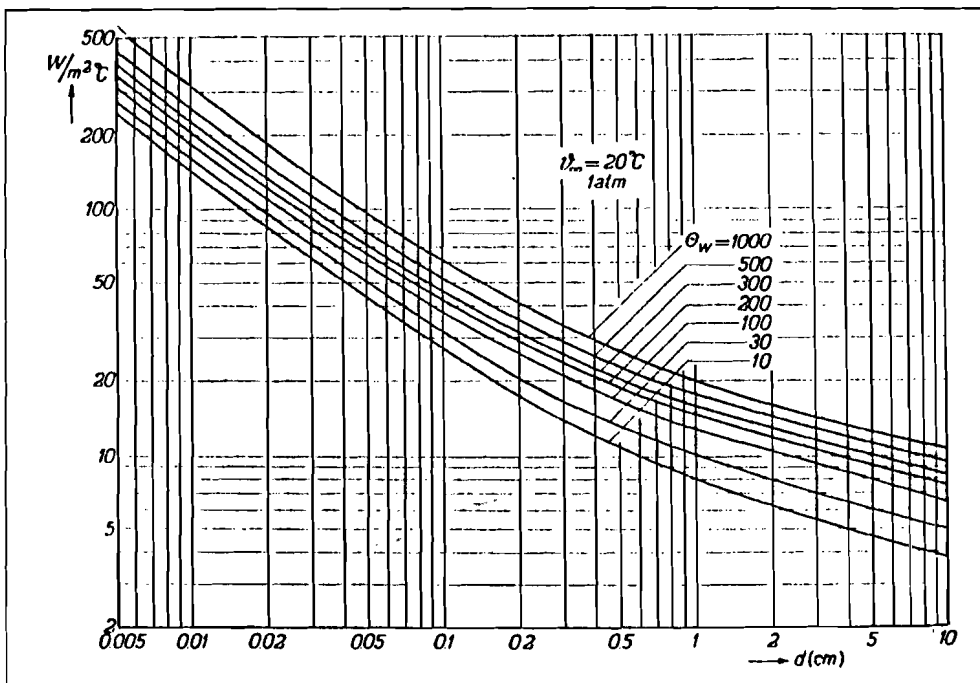
Uit Tsubouchi en Masuda [12] komt de volgende relatie:

$$Nu = 0.36 + 0.048(Gr \cdot Pr)^{\frac{1}{8}} + 0.52(Gr \cdot Pr)^{\frac{1}{4}} \quad (1.21)$$

Deze relatie kan worden gebruikt indien  $10^{-7} \leq Gr \cdot Pr \leq 10^8$ .

### Model VI.

In het artikel van W. Elenbaas [10] wordt een beschouwing over een horizontale cilinder gegeven. Uit de theorie volgt een grafiek, zoals getoond in figuur 1.7, met daarin voor diverse draaddiameters en verschiltemperaturen de waarde van  $h$ . In het artikel wordt geen formule afgeleid.



Figuur 1.7: Elenbaas grafiek met  $h(T)$  bij diverse draaddiameters.

### Selecteren convectie model

In de vorige paragraaf zijn een groot aantal verschillende uitdrukkingen voor convectie (inclusief geleiding) opgesomd. Hieruit zal één bepaald model moeten worden gekozen voor toepassing in het PSPICE-model.

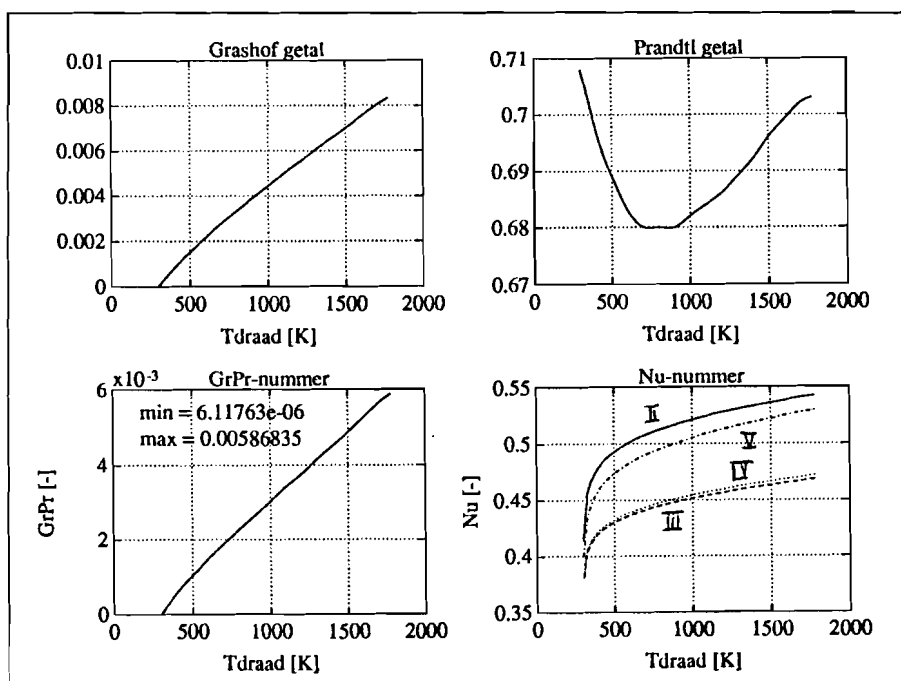
Om de keuze te kunnen motiveren, zullen allereerst de modellen met elkaar vergeleken worden. Daartoe zijn alle relaties omgerekend naar thermische convectieweerstand  $R_{conv}$ , zoals gegeven in formule 1.11.

De dimensieloze grootheden  $Gr$ ,  $Pr$ ,  $GrPr$  en  $Nu$  zijn berekend aan de hand van de gegevens voor lucht (bijlage B.4).

Gebruik makend van de specifieke afmetingen van het modelsegment ( $L$ ,  $r_d$ ,  $n$ ) volgen de curven uit figuur 1.9.

De grafiek rechts toont de convectieweerstand van model II t/m VI. Model I is weggelaten, omdat hierbij een convectieweerstand hoort die sterk afwijkt van de overige modellen (linker grafiek). Dit is te verklaren aan de hand van het verloop van  $GrPr$  in grafiek 1.8. Model I vereist een waarde  $GrPr > 10^4$ . De grafiek toont waarden die veel lager liggen. Dit model wordt daarom als niet bruikbaar beschouwd.

Uit de rechter grafiek blijkt, dat alle overige curven eenzelfde verloop hebben en redelijk overeenstemmen. Bij de definitieve keuze van convectie formule is daarom gekeken naar "een gemiddelde" en is gekozen voor model V, zoals beschreven door Tsubouchi.



Figuur 1.8: Berekende waarde van Gr, Pr, GrPr en Nu voor lucht ( $T \geq 300K$ ).

### Opsplitsen $R_{conv}$ t.b.v. Clucht

Nu het model van convectie bekend is, moet nog worden nagedacht over de plaatsing van lucht-capaciteit Clucht.

In de lucht tussen de draad en de glazen omhulling is geen sprake van constante temperatuur. Er is sprake van een (onbekend) temperatuurprofiel. Dat betekent ook dat voor ieder deelvolumen lucht een andere temperatuur geldt. Dat leidt vanwege de temperatuurafhankelijkheid van de soortelijke warmte  $c$  dan ook tot verschillende capaciteitswaarden.

Het vervangschema zou in dat geval leiden tot een parallelschakeling van een groot aantal lucht-capaciteiten en stukken convectieweerstand (T-segment), ieder met een specifieke waarde.

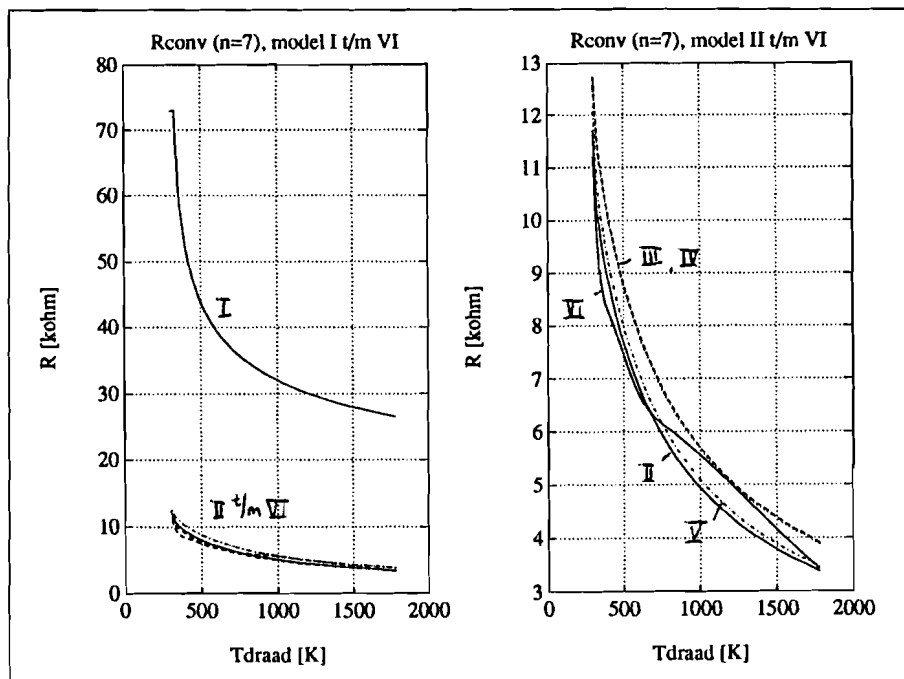
In eerste opzet volstaan met het plaatsen van een enkele capaciteit Clucht tussen draad en omgeving. Hiertoe is de convectieweerstand  $R_{conv}$  opgedeeld in twee gelijke delen, zoals weergegeven in figuur 1.10 (zie ook figuur 1.4). Mogelijk dient op een later tijdstip deze verandering te worden aangepast in verband met de geldende cilindergeometrie.

Capaciteit Clucht wordt gedimensioneerd op het gehele segment-volume (paragraaf 1.6.2).

In het model wordt alleen de convectieweerstand opgedeeld en niet de stralingsweerstand. Dit heeft te maken met het feit dat bij de modellering van straling de invloed van de lucht in eerste instantie is weggelaten. Straling vindt plaats tussen draad en glas.

### 1.6.5 Stralingsterm $R_{rad}(V)$

In literatuur over *heat transfer* wordt uitgebreid aandacht besteed aan overdracht van warmte door middel van straling [9, h8], [7, h9].



Figuur 1.9: Berekend verloop van convectie en radiale geleidingsweerstand als functie van de draadtemperatuur voor de diverse modellen: links model I t/m VI, rechts model II t/m model VI.

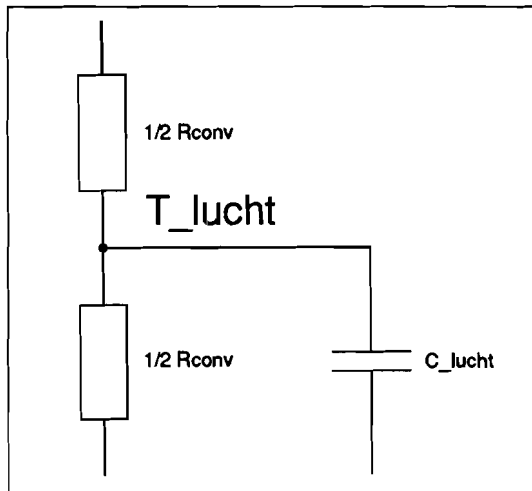
Thermische straling is een vorm van warmte overdracht door middel van elektromagnetische straling, die wordt uitgezonden door een voorwerp als gevolg van zijn temperatuur.

Zoals vastgelegd in de stralingswet van Stefan-Boltzman, gegeven in formule 1.5, is de hoeveelheid door het voorwerp uitgezonden straling met de vierde macht afhankelijk van de absolute temperatuur van dat voorwerp. Ook de invloed van materiaaleigenschappen, samengebracht in de emissiviteit  $\epsilon$  bepaalt de hoeveelheid straling.

Doordat alle voorwerpen met temperatuur  $T > 0K$  stralen, zal in de praktijk het stralende voorwerp ook straling ontvangen van andere stralende voorwerpen in de omgeving. In feite komt er een deel van de straling terug: de netto hoeveelheid uitgestraalde energie wordt kleiner.

Voor de kwantitatieve bepaling van deze hoeveelheden straling zijn in de literatuur een aantal theorieën opgesteld. Belangrijk onderdeel hierbij wordt gevormd door het gebruik van zogenaamde *vormfactoren*, waarmee het mogelijk is om verschillende geometrieën in een standaard formulevorm te plaatsen.

In Holman [9] wordt een formule afgeleid voor de netto warmte overdracht van cilinderoppervlak 1 naar cilinderoppervlak 2 ten gevolge van straling. Door de holle vorm van oppervlak 2 wordt een bepaalde hoeveelheid straling vanaf oppervlak 2 weer direct op oppervlak 2 terug gestraald. Dit vereist een modificatie van de standaard stralingsformule, in de vorm van een vormfactor  $F$ . Holman verwekt dit effect in een relatie voor cilindergeometrie en er volgt voor de netto warmte

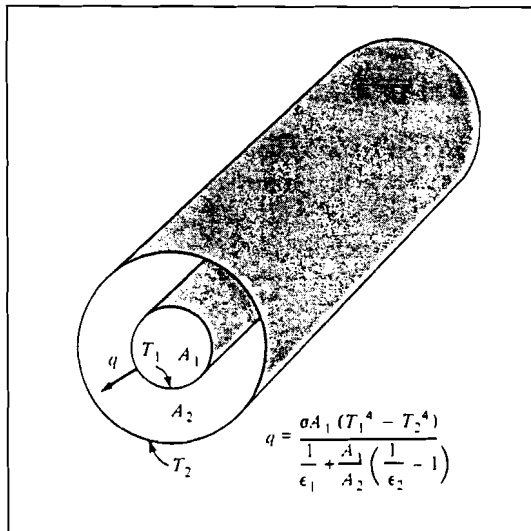


Figuur 1.10: Opsplitsen Rconv voor het plaatsen van Clucht.

overdracht:

$$q = q_{1 \rightarrow 2} = \frac{\sigma(T_1^4 - T_2^4)}{\frac{1-\epsilon_1}{\epsilon_1 S_1} + \frac{1}{S_1 F_{12}} + \frac{1-\epsilon_2}{\epsilon_2 S_2}} \quad (1.22)$$

De term  $F_{12}$  is de vormfactor, die per geometrie verschilt. Deze factor geeft aan, welk deel van de door oppervlak 1 uitgezonden energie door oppervlak 2 wordt opgevangen.



Figuur 1.11: Schets van draad (1) met glasomhulling (2) van de miniatuurpatroon, voor bepaling van stralingsterm. Tussen draad en glas zit lucht.

Figuur 1.11 toont een situatieschets van de metalen draad ( $S_1, T_1, \epsilon_1$ ) en de glazen cilindrische omhulling ( $S_2, T_2, \epsilon_2$ ). De invloed van de lucht ertussen wordt niet beschouwd. Aangenomen wordt, zoals beschreven op bladzijde 457 van Kreith [7], dat de lucht transparant is voor straling. Omdat oppervlak van de draad geheel door het glas oppervlak wordt omsloten, kan worden aangetoond dat vormfactor  $F_{12}$  uit formule 1.22 gelijk aan 1 is, zodat volgt:

$$q = \frac{\sigma(T_1^4 - T_2^4)}{\frac{1-\epsilon_1}{\epsilon_1 S_1} + \frac{1}{S_1} + \frac{1-\epsilon_2}{\epsilon_2 S_2}} = \frac{\sigma(T_1^4 - T_2^4)}{\frac{1}{\epsilon_1 S_1} + \frac{1}{\epsilon_2 S_2} - \frac{1}{S_2}} = \frac{\sigma S_1 (T_1^4 - T_2^4)}{\frac{1}{\epsilon_1} + \frac{S_1}{S_2} \left( \frac{1}{\epsilon_2} - 1 \right)} \quad (1.23)$$

Formule 1.23 kan nog verder worden vereenvoudigd:

- Voor glas gelden waarden van emissiviteit in de orde van 0.90 – 0.95 (bijlage B.5). Dit houdt in dat voor de modellering van straling, het glas nauwelijks invloed heeft. De straling gaat schijnbaar ongehinderd door naar de omgeving. Ter vereenvoudiging wordt daarom aangenomen:

1.  $\epsilon_2 = 1$ ;
2.  $T_2 = T_o$ ;

- Wanneer voor oppervlak  $S_2$  het oppervlak van de glazen cilinder wordt genomen, geldt reeds dat voor miniatuur patronen het quotiënt van  $S_2$  en  $S_1$  in de orde van 1000 ligt. De factor  $\frac{S_1}{S_2}$  gaat daardoor naar nul.

De overdracht van straling vindt nu plaats tussen de draad en de omgeving, juist op de overgang van glas en omgevingslucht.

Deze vereenvoudiging levert voor de uiteindelijke stralingsformule de volgende uitdrukking:

$$q = \frac{\sigma S_1 (T_1^4 - T_2^4)}{\frac{1}{\epsilon_1} + 0} = \sigma \epsilon_1 S_1 (T_d^4 - T_o^4) \quad (1.24)$$

Om in het model uitdrukking 1.24 als elektrische weerstand te modelleren, is herschrijven ervan voldoende:

$$\begin{aligned} q &= \sigma \epsilon_1 S_1 (T_d^4 - T_o^4) \\ &= \sigma \epsilon_1 S_1 (T_d^2 + T_o^2)(T_d^2 - T_o^2) \\ &= \sigma \epsilon_1 S_1 (T_d^2 + T_o^2)(T_d + T_o) \cdot (T_d - T_o) \end{aligned}$$

Uit deze uitdrukking kan de weerstand worden gehaald:

$$R_{\text{rad}}(T_d, T_o) = \frac{T_d - T_o}{q} = \frac{1}{\sigma \epsilon_1 S (T_d^2 + T_o^2)(T_d + T_o)} = \frac{\text{constante}}{(T_d^2 + T_o^2)(T_d + T_o)}$$

Gebruik makend van het analogon van thermisch en elektrisch model, volgt voor de elektrische stralingsweerstand:

$$R_{\text{rad}}(V) = \frac{\text{constante}}{((300 + V)^2 + 300^2)((300 + V) + 300)} \quad (1.25)$$

Immers:

- De omgevingstemperatuur wordt constant verondersteld (300K);
- In de formule dient de absolute temperatuur van het draadsegment te worden gebruikt, niet de toename in temperatuur, dus  $(V + 300)$ .

## 2. PSPICE componenten modellering

*Uitgaande van het elektrisch analagon, opgesteld in het vorige hoofdstuk, worden in dit hoofdstuk alle componenten in PSPICE code omschreven. Daarnaast wordt aangegeven, welke extra PSPICE-code nodig is voor de simulatie.*

Alle componenten worden als een n-poort beschouwd. Dat wil zeggen dat er sprake is van een blackbox met n in- en/of uitgangen. De opbouw van ieder van deze blackboxes wordt eerst apart besproken.

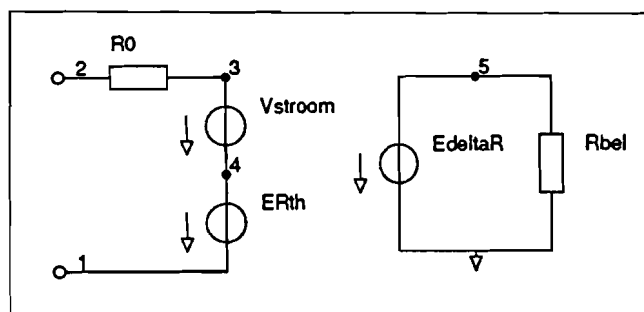
In een later stadium worden de blackboxes aan elkaar geschakeld. Groot voordeel bij het gebruik van blackboxes is, dat een eenmaal gedefinieerde blackbox meerdere malen in de schakeling kan worden gebruikt. Dit kan de hoeveelheid benodigde invoer voor de beschrijving van het model aanzienlijk beperken.

Voor een uiteenzetting van het gebruik van blackboxes in PSPICE wordt verwezen naar paragraaf 5.3 op bladzijde 71.

Alle PSPICE code is voorzien van extra commentaar.

Alle nu volgende berekeningen zijn gebaseerd op gegevens van de minatuur smelt patroon, type 217.315 van de firma Littelfuse. Gegevens van deze patroon zijn opgenomen in bijlage C van dit verslag.

### 2.1 PSPICE model: Rth\_Ni\_ax



Figuur 2.1: Blackbox: Rth\_Ni\_ax.

Voor de modellering van de thermische weerstand in axiale richting in de nikkeldraad (Rth\_Ni\_ax weerstand), zoals weergegeven in formule 1.9 zal gebruik gemaakt worden van de PSPICE constructie als beschreven in paragraaf 5.5.1, op bladzijde 74.

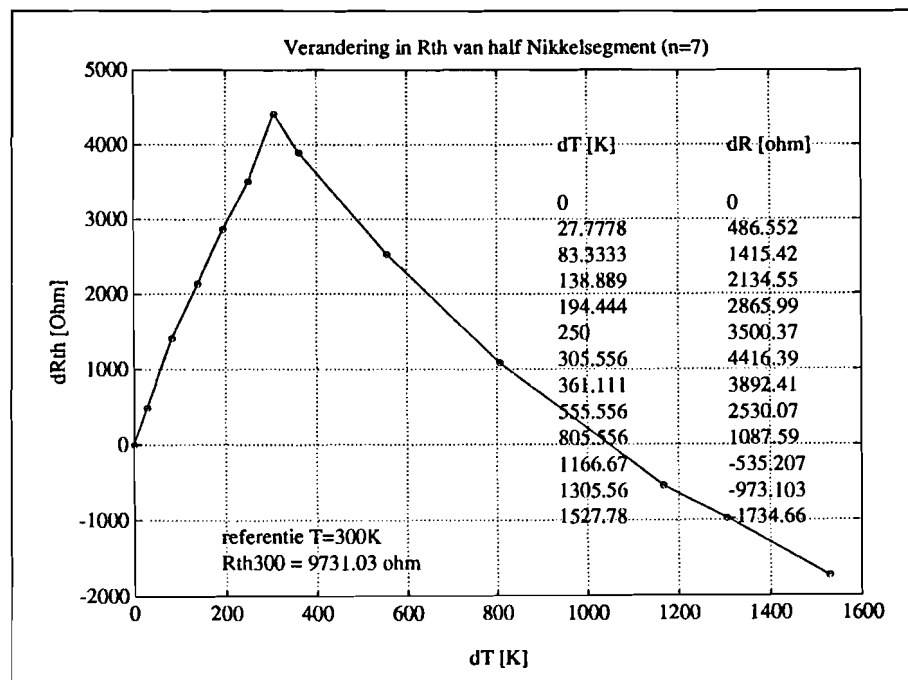
In deze constructie wordt uitgegaan van de equivalente weerstand, bepaald bij kamertemperatuur (300 K): Rth\_Ni\_ax\_300. Door het in serie plaatsen van een component, die zich als weerstand gedraagt en waarvan de weerstandswaarde beïnvloedbaar is, kan de totale weerstand worden veranderd.

De mate van verandering is afhankelijk van de spanning over de totale weerstand en kan zowel positief als negatief zijn.



Voor de bepaling van het juiste verloop van  $R_{th\_Ni\_ax}$  als functie van de spanning geldt:

- De berekening is gebaseerd op een model van 7 segmenten ( $n=7$ ). Dit houdt in dat voor de bepaling van  $R_{th\_Ni\_ax}$  een lengte van  $\frac{L}{2.7}$  moet worden gehanteerd, daar per segment steeds twee weerstanden worden gebruikt (zie figuur 1.4);
- De thermische geleidbaarheid  $\lambda$  als functie van de temperatuur is gegeven in bijlage B.3;
- Alle relevante afmetingen van de smeltpatroon type 217.315 zijn opgenomen in bijlage C.



Figuur 2.2: Berekende waarden t.o.v. 300K van  $R_{th\_Ni\_ax}$  als functie van toename van de temperatuur.

Grafiek 2.2 toont de resultaten van de berekening. De waarden van de weerstand zijn uitgedrukt in een verandering t.o.v.  $R_{th\_Ni\_ax\_300}$ . Omdat in de grafiek de STIJGING in temperatuur wordt getoond, kan deze stijging in temperatuur van het segment direct worden vertaald in de spanning op knooppunt 2 van het segment (zie segmentmodel in figuur 1.4).

Door gebruik te maken van de Tabel-functie van PSPICE, zoals beschreven in paragraaf 5.2.5 op bladzijde 70 kunnen de grafiepunten eenvoudig worden overgenomen in het model.

Het PSPICE Blackbox-model ziet er (voorzien van commentaar) als volgt uit:

```
* =T H E R M I S C H E   G E L E I D I N G   a x i a a l===== *
*
*   2 -----XXXX----- 1
*   (+)      Rth_Ni_ax   (-)
*
```

```

.PARAM Rth_Ni_ax_300 = 8996.54; Rth_Ni_ax voor half segment bij n=7

.SUBCKT Rth_Ni_ax 2 1
R0      2 3  { Rth_Ni_ax_300 * (7/n) }
Vstroom 3 4  0V
ERth    4 1  VALUE { V(5)*I(Vstroom) * (7/n) }
EdeltaR 5 0  TABLE { V(2,1) } =
*      dT      dRth,ax
+      0.0      0.0
+      27.7778  449.827
+      83.3333  1308.59
+      138.889  1973.43
+      194.444  2649.67
+      250.000  3236.17      ; tabel geeft verandering in Rth_Ni_ax aan tov 300K
+      305.556  4083.05
+      361.111  3598.62      ; dT is temperatuurverschil met 300K
+      555.556  2339.1       ; dRth is weerstandsverschil met Rth_Ni_ax_300
+      805.556  1005.5
+      1166.67  -494.81
+      1305.56  -899.654
+      1527.78  -1603.73
Rbel    5 0  1MEG
.ENDS Rth_Ni_ax

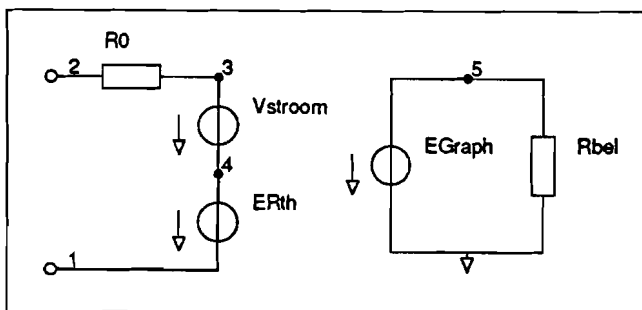
```

Opmerkingen:

- In de definitie wordt de totale weerstand met een factor  $(7/n)$  vermenigvuldigd. Bij een segment-aantal anders dan 7 ontstaat op die manier een juiste correctie op de tabel-data, die is gebaseerd op  $n = 7$ . De weerstand is immers lineair afhankelijk van de segmentlengte;
- De factor  $n$  zal elders in PSPICE-code moeten worden gedefinieerd (zie paragraaf 2.1.11).

### 2.1.1 PSPICE model: Rth\_glas\_rad

Omdat bij de afleiding van formule 1.10 naar een equivalent van spanningsafhankelijke weerstand is toegewerkt, gaat deze modellering op eenzelfde manier als in de vorige paragraaf. De PSPICE structuur is hetzelfde, alleen de getallen veranderen.



Figuur 2.3: Blackbox: Rth\_glas\_rad.

In die formule is voor de thermische geleidbaarheid  $\lambda$  van het glas een richtwaarde genomen (gemiddelde), omdat gegevens omtrent de afhankelijkheid in temperatuur niet beschikbaar zijn.

Dit houdt in dat de weerstand constant is, voor iedere spanning, zodat volstaan zou kunnen worden met het definiëren van een vaste weerstand  $R$  tussen het aansluitpunt van het blackbox (1) en het gemeenschappelijk aardknooppunt (0) (figuur 2.3).

Er is echter gekozen voor de opzet met gebruik van de tabelfunctie, zodat spanningsafhankelijkheid later alsnog eenvoudig kan worden aangebracht.

Gebruik makend van de gegevens uit bijlage C volgt voor de thermische weerstand:

$$R_{th\_glas\_rad} = \frac{n \ln\left(\frac{r_o}{r_g}\right)}{2\pi L \cdot 1,95} = \frac{7 \ln\left(\frac{2,25 \cdot 10^{-3}}{1,5 \cdot 10^{-3}}\right)}{2\pi \cdot 0,0175 \cdot 1,95} = 13,237\Omega$$

Tot slot nog de PSPICE-code:

```
* =T H E R M I S C H E   G E L E I D I N G glas radiaal===== *
*
*   2 ----XXXX---- 1
*   (+) Rth_glas   (-)
*
.PARAM Rth_glas_rad300 = 13.237 ; Rth van glas voor segment bij n=7

.SUBCKT Rth_glas_rad 2 1 PARAMS: factor = 1
R0      2 3 {factor * Rth_glas_rad300 * (n/7) }
Vstroom 3 4
ERth    4 1 VALUE { V(5)*I(Vstroom) * (n/7) * factor}
EGraph  5 0 TABLE { V(2,1) } =
*   dT      dRth_glas_rad
+   0        0
+  1700      0
Rbel    5 0 1MEG
.ENDS Rth_glas_rad
```

In paragraaf 2.1.8 wordt het gebruik van de parameter *factor* belicht. In paragraaf 5.3.2 staat de geldende PSPICE syntax beschreven.

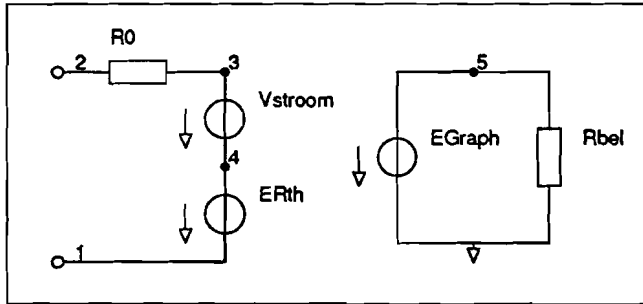
## 2.1.2 PSPICE model: Rrad

De eigenschappen van straling zijn in paragraaf 1.6.5 gedefinieerd in termen van weerstand, zodat bij het PSPICE model gebruik gemaakt kan worden van  $R(V)$ .

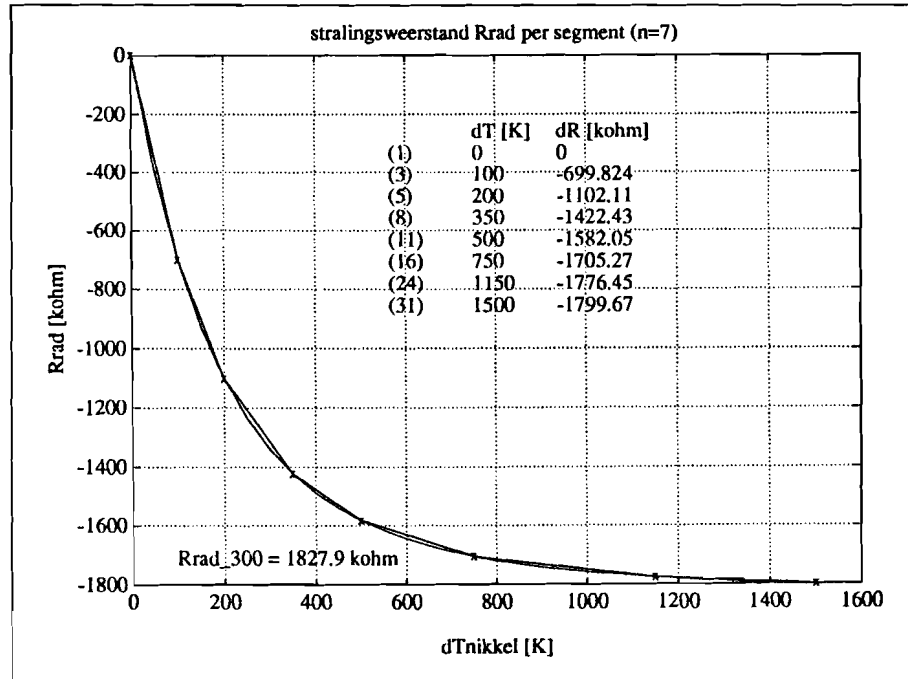
Allereerst wordt een numerieke waarde voor de constante term uit formule 1.25 bepaald. Voor  $n=7$  geldt:

$$\text{constante} = \frac{1}{\sigma \epsilon S} = \frac{1}{5,67 \cdot 10^{-8} \cdot \frac{0,41+0,045}{2} \cdot \pi d^{\frac{0,0175}{7}}} = 197,4 \cdot 10^{12}$$

De stralingsweerstand kan nu eenvoudig als functie van de spanning worden berekend. In grafiek 2.5 is het resultaat weergegeven. De resultaten worden in de PSPICE tabel opgenomen. De PSPICE-code wordt hiermee gelijk aan:



Figuur 2.4: Blackbox: Rrad.



Figuur 2.5: Berekende waarden van Rrad t.o.v. 300K als functie van de toename in de temperatuur.

```

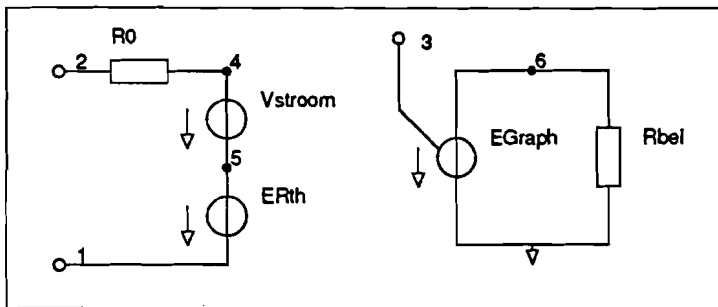
* =S T R A L I N G S W E E R S T A N D===== *
*
*      2 ----XXXX---- 1
*      (+)  Rrad  (-)
*
.PARAM Rrad_300 = 1.828E6 ; stralingsweerstand bij 300K, n=7

.SUBCKT Rrad 2 1 PARAMS: factor = 1
R0      2 3 {factor * Rrad_300 * (n/7) }
Vstroom 3 4 0V
ERth    4 1 VALUE { factor * V(5)*1E3*I(Vstroom) * (n/7) }
EGraph  5 0 TABLE { V(2,1) } =
*  dT    dR_rad [kOhm]
+  0      0
    
```

```

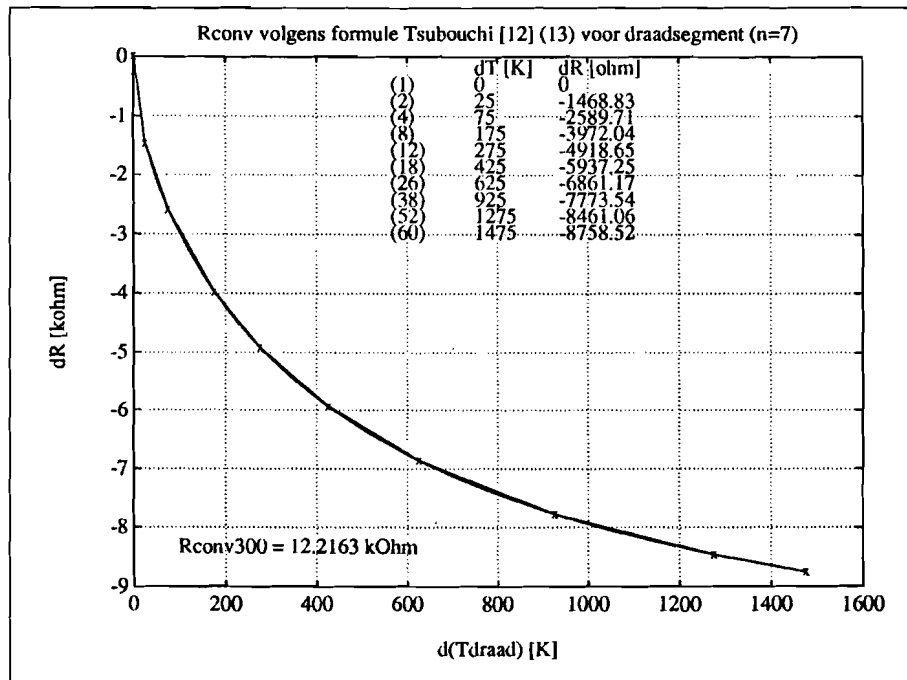
+ 100      -699.824
+ 200      -1102.11
+ 350      -1422.43
+ 550      -1616.62
+ 750      -1705.27
+ 1150     -1776.45
+ 1400     -1794.77
Rbel      5 0 1MEG
.ENDS Rrad
    
```

### 2.1.3 PSPICE model: Rconv



Figuur 2.6: Blackbox: Rconv

Met formule 1.21 als basis kan de convectieweerstand worden berekend met formule 1.11, als functie van de verandering van draadtemperatuur ten opzichte van de omgevingstemperatuur.



Figuur 2.7: Berekende waarden t.o.v. 300K van Rconv als functie van toename in temperatuur.

De resultaten zijn weergegeven in figuur 2.7 en worden verwerkt in de PSPICE-code:

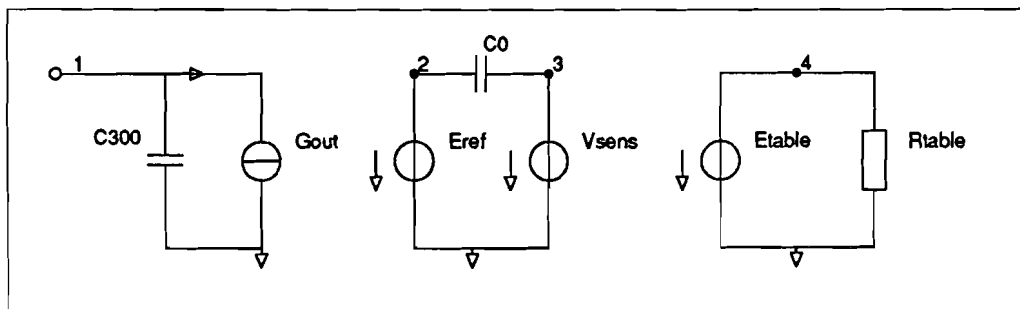
```

* =C O N V E C T I E===== *
*
* 2 ----XXXX---- 1          3: meetklem voor draadtemperatuur
* (+)  Rconv  (-)
*
.PARAM Rconv_300 = 13478.6 ; Rconvectie bij 300K, n=7

.SUBCKT Rconvectie 2 1 3 PARAMS: factor = 1
R0      2 4 { factor * Rconv_300 * (n/7) }
Vstroom 4 5 0V
ERth    5 1 VALUE { factor * V(6)*I(Vstroom) * (n/7) }
EGraph  6 0 TABLE { V(3) } =
*  dT      dRconv
+  0        0.0
+  25      -1934.43
+  75      -3027.91 ; tabel geeft verandering in Rconv aan tov 300K
+  175     -4448.19
+  275     -5445.80 ; dT is temperatuurverschil met 300K
+  425     -6533.07 ; dRconv is weerstandsverschil met Rconv_300
+  625     -7529.50
+  925     -8521.12
+  1275    -9272.89
+  1500    -9642.41
Rbel    6 0 1MEG
.ENDS Rconvectie

```

### 2.1.4 PSPICE model: Cni

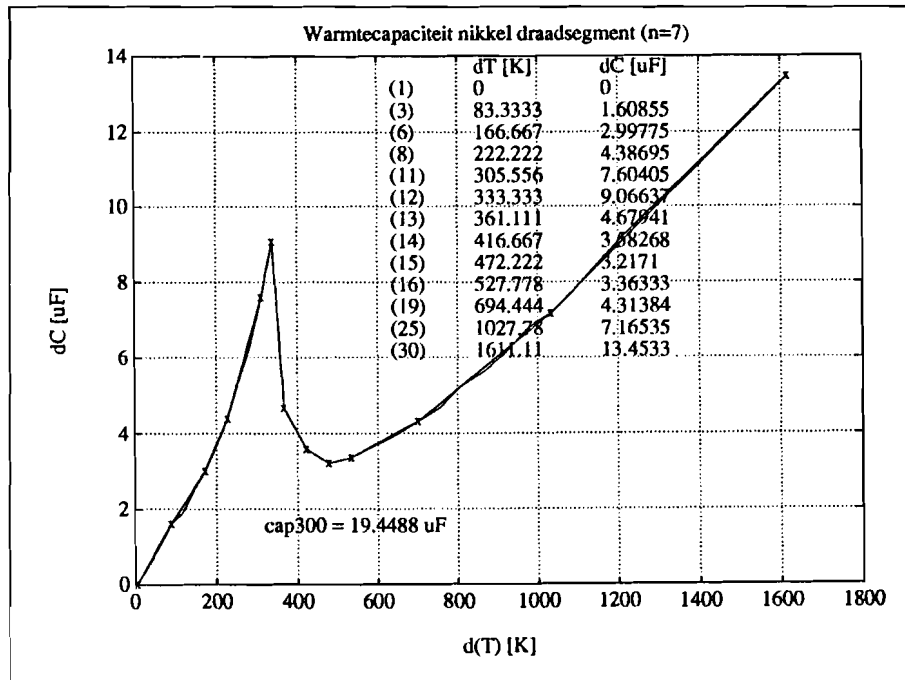


Figuur 2.8: Blackbox: Cni

Voor de realisering van het blackbox-model Cni uit figuur 2.8 wordt gebruik gemaakt van het model van een spanningsafhankelijke capaciteit, zoals besproken wordt in paragraaf 5.5.2 op bladzijde 76.

Uitgangspunt voor de berekening vormt de formule uit paragraaf 1.6.2, waarmee de capaciteit als functie van de spanning kan worden afgeleid. Het verloop van soortelijke warmte als functie van temperatuur voor nikkel is weergegeven in bijlage B.2.

Ook nu wordt als referentie de capaciteit bij  $V = 0V$  gebruikt. Bij hogere spanning wordt de capaciteit vergroot door het parallel plaatsen van extra capaciteit.



Figuur 2.9: Berekende waarden t.o.v. 300K van  $C_{Ni}$  als functie van de toename in de temperatuur.

In grafiek 2.9 staan de numerieke waarden vermeld, zoals deze uit de berekening volgen. Wanneer dit wordt ingevuld in het 'geraamte' de PSPICE-structuur voor  $C(V)$ , dan volgt:

```
* =W A R M T E O P S L A G  Nikkel===== *
*
*   1 ----||---- 0
*   (+)  CNi   (-)
```

```
.PARAM NiCap_300  = 19.4488E-6 ; equivalent warmtecapaciteit bij 300K (n=7)
```

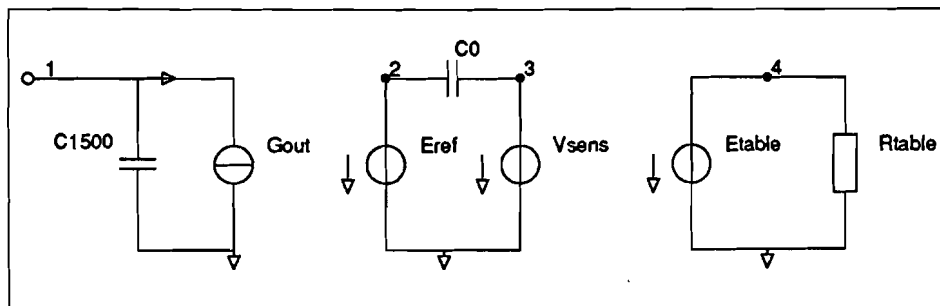
```
.SUBCKT Nicap 1 PARAMS: factor = 1
Eref  2 0      VALUE = { V(1,0) }
C0    2 3      1
Vsens 3 0      0V
C300  1 0      { factor * NiCap_300 * (7/n) }
Gout  1 0      VALUE = { factor * I(Vsens) * V(4)*1E-6 * (7/n) }
Etable 4 0 TABLE { V(1) } =
*   V(1)      dC [uF]
+   0.0       0.0
+   83.33     1.60855
```

```

+ 166.667  2.99775
+ 222.222  4.38695
+ 305.556  7.60405
+ 333.333  9.06637
+ 361.111  4.67941
+ 416.667  3.58268
+ 472.222  3.21710
+ 527.222  3.36333
+ 694.444  4.31384
+ 1027.78  7.16535
+ 1611.11  13.4533
Rtable 4 0 1MEG
.ENDS Nicap

```

### 2.1.5 PSPICE model: Clucht



Figuur 2.10: Blackbox: Clucht

Het blackbox-model van de luchtcapaciteit uit figuur 2.10 ontstaat op dezelfde manier als dat van de nikkel capaciteit. De basisformule uit paragraaf 1.6.2 bevat nu afmetingen voor het luchtvolume (zie bijlage C). In bijlage B.4 staat de soortelijke warmte van lucht vermeld. Met de gegevens kan grafiek 2.11 worden afgeleid. Hieruit volgt de PSPICE-code voor Clucht:

```

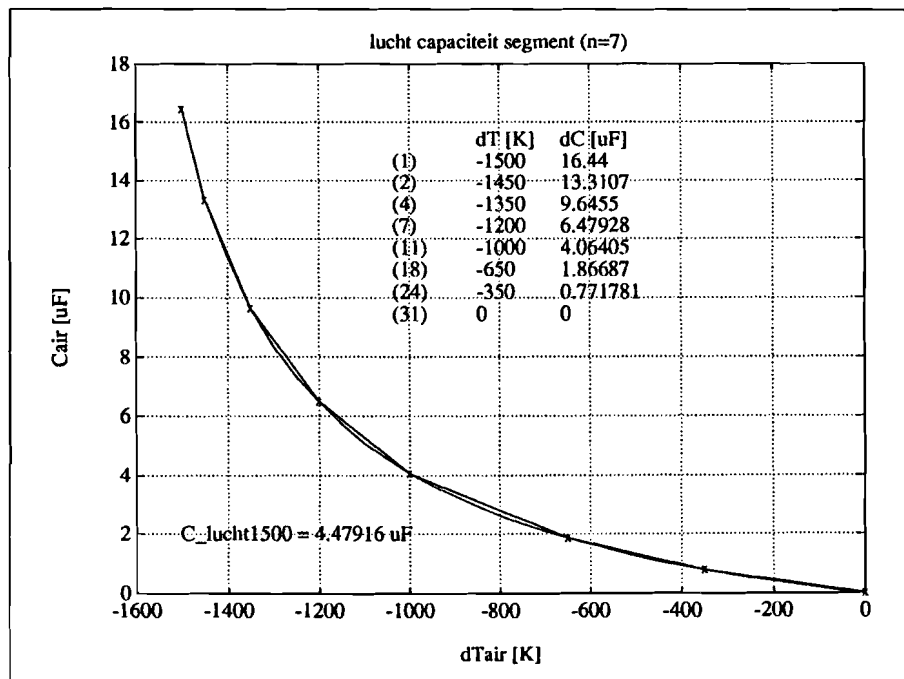
* =W A R M T E O P S L A G  lucht===== *
*
*   1 ----||---- 0
*   (+) Clucht  (-)

.PARAM LuchtCap_1500 = 4.47916E-6 ; warmtecapaciteit bij 1500K, n=7

.SUBCKT Luchtcap 1 PARAMS: factor = 1
Eref  2 0      VALUE = { V(1,0) }
C0    2 3      1
Vsens 3 0      0V
C1500 1 0      { factor * LuchtCap_1500 * (7/n) }
Gout  1 0      VALUE = { factor * I(Vsens) * V(4)*1E-6 * (7/n) }

```





Figuur 2.11: Berekende waarden t.o.v. 300K van Clucht als functie van de toename in temperatuur.

```
Etable 4 0 TABLE { V(1,0)-1500 } =
* V(1) dC [uF]
+ -1500 16.4400
+ -1450 13.3107
+ -1350 9.6455
+ -1200 6.4793
+ -1000 4.0641
+ -650 1.8669
+ -350 0.7718
+ 0 0.0
Rtable 4 0 1MEG
.ENDS Luchtcap
```

### 2.1.6 PSPICE model: Cglas

Evenals bij de thermische weerstand voor het glas, is ook van de soortelijke warmte een gemiddelde waarde aangenomen (bijlage B.5).

Hoewel ook nu kan worden volstaan met een constante waarde voor Cglas, is dezelfde structuur als bij CNi en Clucht aangehouden, wat leidt tot de volgende PSPICE-code:

```
* =W A R M T E O P S L A G glas===== *
*
* 1 ----||---- 0
```

```

* (+) Cglas (-)
*
.PARAM GlasCap_300 = 49.39E-3;      warmtecapaciteit bij 300K, n=7

.SUBCKT Glascap 1 PARAMS: factor = 1
Eref  2 0  VALUE = { V(1,0) }
C0     2 3  1
Vsens  3 0  0V
C300   1 0  { factor * GlasCap_300 * (7/n) }
Gout   1 0  VALUE = { factor * I(Vsens) * V(4) * (7/n) }
Etable 4 0  TABLE { V(1,0) } =
*   V(1)  dC [uF]
+   0.0    0
+  1700    0
Rtable 4 0  1MEG
.ENDS Glascap

```

### 2.1.7 PSPICE model: Iinj

Bij het blackbox-model voor de stroombron Iinj (formule 1.7), moet zowel de invloed van tijd als segmentpotentiaal (temperatuur) worden gerealiseerd. In het PSPICE model is dit opgedeeld in twee aparte modellen, waarvan de signalen gekoppeld zijn.

#### DC stroom

Allereerst is er het  $i(t)$ -signaal, waarmee de elektrische stroom door de miniatuurpatroon wordt voorgesteld. Dit signaal is als spanning aan de uitgang van een blackbox aanwezig. Afhankelijk van het gewenste stroomsignaal, zal het model iets wijzigen.

Voor het simuleren met een gelijkstroom van 315mA gelden de volgende PSPICE regels, waarbij een met de stroom evenredig spannings signaal  $V(1,0)$  wordt opgewekt:

```

Vstroom  1 0  PWL( (0,0) (1ns,315E-3A) (1000s,315E-3A) )
Rstroom  1 0  1MEG

```

Het model bevat een tijdgestuurde spanningsbron Vstroom waarbij de uitgangsspanning wordt beschreven volgens de 'PieceWise Linear' functie. Voor  $1ns \leq t \leq 1000s$  is dit signaal gelijk aan 315mV. In het tijdsinterval  $0 \leq t \leq 1ns$  stijgt de spanning lineair van 0V tot 315mV. Het getal 1000 is dermate groot gekozen, dat de totale simulatietijd deze waarde nooit te boven zal gaan.

Het oplopen van de spanning in de eerste nanoseconde is met opzet gedaan om er voor te zorgen dat bij het starten van de simulatie voor PSPICE geen problemen ontstaan met te grote waarden van  $\frac{dV}{dt}$ . Dit kan leiden tot afbreken van de simulatierun door numerieke instabiliteiten. De tijd van 1ns is zodanig gekozen, dat geen nadelige gevolgen ontstaan voor de simulatie.

#### Pulsstroom: halve sinus

Wanneer er sprake is van een eenmalig signaal in de vorm van een halve (gedempte) sinus, moet gebruik gemaakt worden van meerdere bronnen, zoals uit de volgende PSPICE regels blijkt:

```
Vsindemp 2 0 SIN(0V 5V 50Hz 0 0)
Rsindemp 2 0 1MEG
Vstap    3 0 PULSE(1V 10ms 1ns 1ns)
Rstap    3 0 1MEG
Estroom  1 0 VALUE = { V(2) * V(3) }
Rstroom  1 0 1MEG
```

Vsindemp beschrijft een continue, ongedempte 50Hz-sinus met een amplitude van 5V. Vstap beschrijft een pulssignaal met amplitude 1V voor  $0 \leq t \leq 10\text{ms}$  en 0V voor  $t > 10\text{ms}$ . Het gewenste halve sinus-signaal wordt verkregen met een spanningsgestuurde spanningsbron, met naam Estroom, waarbij de signalen Vsindemp en Vpuls vermenigvuldigd worden. Weerstanden zorgen voor de verplichte bronbelasting. De waarde ervan is niet relevant.

### Pulsstroom: willekeurig

Een laatste methode die kan worden gebruikt voor het verkrijgen van het stroomsignaal is het gebruik van de tabelfunctie van PSPICE. Dit is handig in die gevallen waarin het niet mogelijk is om het gedrag met een van de andere methoden vast te leggen.

```
Vstroom 1 0 PWL( (0,0) (10,5) (15,8) ... )
Rstroom 1 0 1MEG
```

of met de echte tabelfunctie:

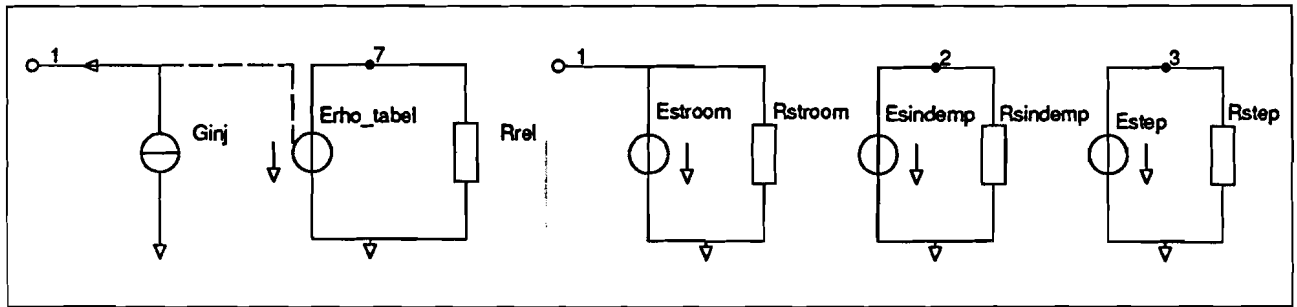
```
Estroom 1 0 TABLE { TIME } = ( (0,0) (10,5) (15,8), ... )
Rstroom 1 0 1MEG
```

Wordt als voorbeeld het DC-stroom model genomen, dan is de PSPICE blackbox-code als volgt:

```
* =S T R O O M S I G N A A L===== *
*
*      1          V(1) is het signaal van i(t)
*      + |
*      ( ) Estroom
*      |
*      ~
.SUBCKT Stroomsignaal_dc 1
Vstroom 1 0 PWL ( 0,0 1ns,315E-3 1000s,315E-3)
Rstroom 1 0 1MEG
.ENDS Stroomsignaal_dc
```

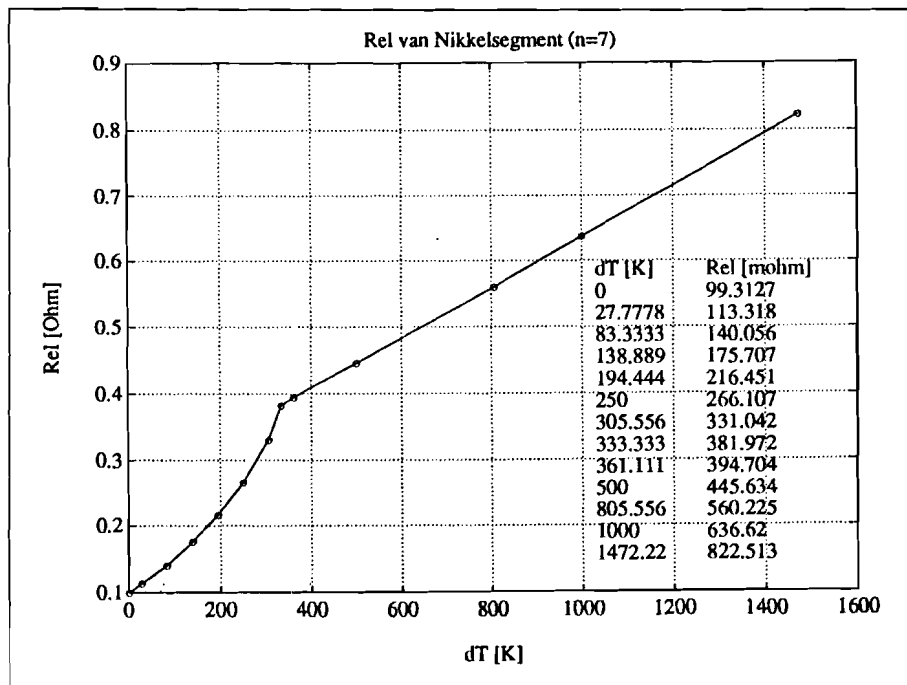
Het  $i(t)$ -blackbox-signaal wordt bij het definiëren van de blackbox lijn als ingangssignaal gebruikt.

Deze blackbox heeft twee poorten. Poort 1 is de positieve uitgang van stroombron lijn, die met de andere kant aan de gemeenschappelijke aarde (knooppunt 0) hangt. Op poort 2 (ingang) wordt het  $i(t)$  signaal aangesloten (figuur 2.12).



Figuur 2.12: Blackbox:  $I_{inj}(t,V)$ .

Naast het stroomsignaal  $i(t)$  moet ook de elektrische weerstand  $R_{rel}$  van het segment bekend zijn. Indien de karakteristieke segment afmetingen bekend zijn, kan met de grafiek van soortelijke weerstand van nikkel (bijlage B.1) de  $R_{rel}$  worden bepaald (gebaseerd op  $n=7$ ). In figuur 2.13 is  $R_{rel}$  uitgezet.



Figuur 2.13: Berekende waarden t.o.v. 300K van  $R_{rel}$  als functie van de stijging van de temperatuur.

Na kwadrateren van het stroomsignaal en vermenigvuldigen met het signaal van  $R_{rel}$ , wordt met stroombron  $G_{inj}$  de gewenste stroom gerealiseerd. De PSPICE-code is als volgt:

```
* =E L E C T R I S C H   V E R M O G E N===== *
*
```

```

*      1  2          V(1) is temperatuursignaal van segment
*      |  |
* Iinj  (^)          V(2) is i(t) signaal
*      |
*      ~

```

```

.SUBCKT Iinj 1 2 PARAMS: factor=1
Ginj 0 1 VALUE = { factor * pwr(V(2),2) * V(7)*1E-3 * (7/n) }
Xrho 7 1 0 Rel_tabel
.ENDS Iinj

```

```

.SUBCKT Rel_tabel 1 2 3
Erho_tabel 1 0 TABLE { V(2,3) } =
* dT [K] Rel [mohm]
+ 0.0 99.3127
+ 27.78 113.318
+ 83.33 140.056; Tabel geeft de weerstand van Nikkel
+ 138.89 175.707; zoals deze uit de materiaalgrafiek is berekend.
+ 194.44 216.451
+ 250.00 266.107; dT is de temperatuurtoename tov 300K
+ 305.56 331.042
+ 333.33 381.972
+ 361.11 394.704
+ 500.00 445.634
+ 805.56 560.225
+1000.00 636.620
+1472.22 822.513
Rrel 1 0 1MEG
.ENDS Rel_tabel

```

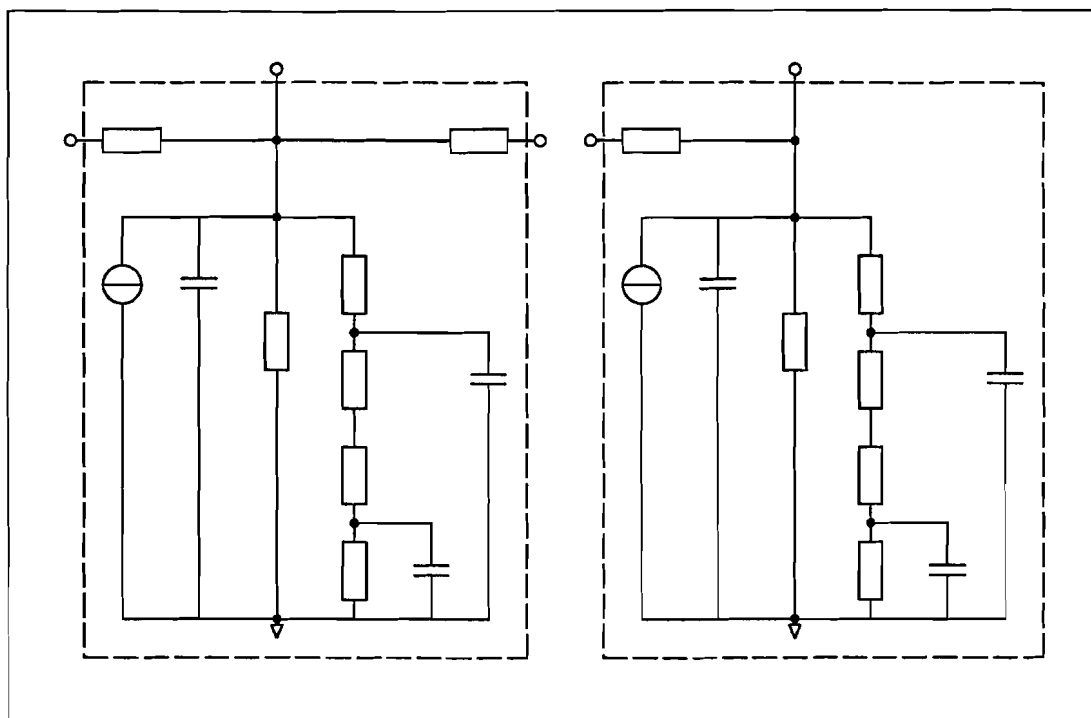
### 2.1.8 PSPICE model: Opbouw segment

Bij opbouw van het draadsegment, worden alle besproken blackboxes aan elkaar geschakeld, zoals aangegeven in figuur 1.4. Het zo ontstane netwerk vormt op zich weer een blackbox.

Zoals besproken in paragraaf 1.5.5, moeten twee blackboxes worden gedefinieerd (figuur 2.14).

- Segmenten met nummers 1 t/m  $\frac{(n-1)}{2}$  representeren een volledig draadsegment met lengte  $\frac{L}{n}$ . Alle componenten zijn gebaseerd op deze lengte en kunnen rechtstreeks in de blackbox worden gebruikt. De blackbox heeft vier aansluitingen;
- Het segment met nummer  $\frac{(n+1)}{2}$  (het middelste segment) is in lengte gehalveerd (zie figuur 1.3). De componentwaarden in dit segment zijn gewijzigd. Door gebruik te maken van een variabele met naam 'factor' die in de componentformules wordt gebruikt, kan dit eenvoudig worden gerealiseerd. De blackbox heeft drie aansluitingen.

De PSPICE-code voor het hele segment:



Figuur 2.14: Blackbox-model voor heel segment (links) en half segment (rechts).

```

* =S E G M E N T===== *
*
*          Rth_Ni          Rth_Ni
*          1 --XXXX-- 2 ---XXXX--- 3          7 --<-- i(t)
*          |
*          |-----|-----|
*          |         |         |
*          |         |CNi  X   X 1/2 Rconv
*          | Iinj (~) === X Rrad 4-----
*          |         |         X 1/2 Rconv |
*          |         |         |         === Clucht
*          |         |         |         5
*          |         |         |         |
*          |         | 1/2 Rth_glas X
*          |         |         |         |
*          |         |         |         6---
*          |         |         |         |
*          |         | 1/2 Rth_glas X === Cglas |
*          |         |         |         |
*          |         |         |         |
*          |         |         |         |
*
.SUBCKT Segment 1 2 3 4 5 6 7
Xpower 2 7 Iinj PARAMS: factor = 1
XCNi 2 Nicap PARAMS: factor = 1
Xconv1 2 4 2 Rconvectie PARAMS: factor = 0.5
    
```

```

Xconv2  4 5 2  Rconvectie  PARAMS: factor = 0.5
XClucht 4      Luchtcap     PARAMS: factor = 1
XRad    2 0    Rrad         PARAMS: factor = 1
XCglas  6      Glascap     PARAMS: factor = 1
XRglas1 5 6    Rth_glas_rad PARAMS: factor = 0.5
XRglas2 6 0    Rth_glas_rad PARAMS: factor = 0.5
Xgel_l  2 1    Rth_Ni_ax
Xgel_r  3 2    Rth_Ni_ax
.ENDS Segment

```

Omdat in de blackbox van de convectieweerstand de totale convectie in het segment wordt berekend, moet deze bij het plaatsen van de luchtcapaciteit worden opgesplitst. Dit verklaart het gebruik van 'factor = 0,5'. De convectieweerstand wordt evenredig verdeeld (figuur 1.10). Dezelfde eigenschap wordt toegepast bij het opsplitsen van de thermische geleidbaarheid in radiale richting in het glas.

In de kopregel van de segmentdefinitie worden alle knooppunten (1 t/m 6) naar buiten uitgevoerd, zodat het later mogelijk is om naast de draadtemperatuur ook een indicatie te hebben van een optredende temperatuur in de lucht en in het glas.

Voor het halve segment zijn de waarden van de parameter 'factor' gewijzigd en ontbreekt de thermische weerstand aan de rechterkant. Een verdubbeling van de 'factor' betekent een verdubbeling van de componentwaarde. De bijbehorende PSPICE-code:

```

.SUBCKT Segmenthalf 1 2 4 5 6 7
Xpower  2 7      Iinj         PARAMS: factor = 0.5
XCNi    2        Nicap        PARAMS: factor = 0.5
Xconv1  2 4 2    Rconvectie    PARAMS: factor = 1
Xconv2  4 5 2    Rconvectie    PARAMS: factor = 1
XClucht 4        Luchtcap     PARAMS: factor = 0.5
XRad    2 0      Rrad         PARAMS: factor = 2
XCglas  6        Glascap     PARAMS: factor = 0.5
XRglas1 5 6      Rth_glas_rad PARAMS: factor = 1
XRglas2 6 0      Rth_glas_rad PARAMS: factor = 1
Xgel_l  2 1      Rth_Ni_ax
Xgel_r  3 2      Rth_Ni_ax
.ENDS Segmenthalf

```

Voor de juiste aanpassing van 'factor' moet gekeken worden naar de invloed van een veranderende segmentlengte op de component. Hierbij kan de volgende tabel worden gebruikt:

component	evenredig met l	evenredig met 1/l	heel segment factor=	half segment factor=
linj	×		1	0,5
CNi	×		1	0,5
Rconv		×	0,5	1
Clucht	×		1	0,5
Rrad		×	1	2
Cglas	×		1	0,5
Rth_glas_rad		×	1	2
Rth_Ni_ax	×		1	0,5

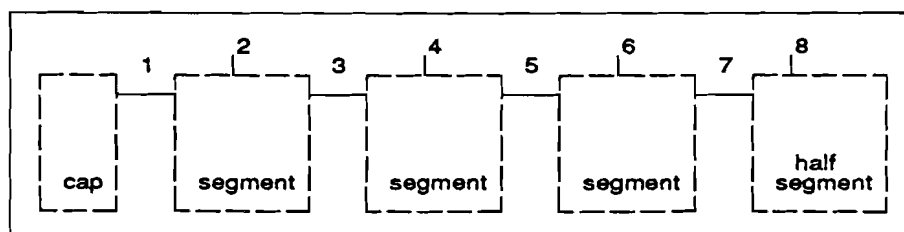
### 2.1.9 PSPICE model: Eindcap

Het realiseren van het eindcap-model uit figuur 1.5 in een blackbox is eenvoudig. Er is sprake van een enkele spanningsbron met  $V = 0V$ . In PSPICE-code wordt dit:

```
* =E I N D C A P===== *
*
*      1
*      | +
*  Vcap ( )
*      |
*      ~
.SUBCKT Eindcap 1
  Vcap 1 0 0V
.ENDS Eindcap
```

### 2.1.10 PSPICE model: Opbouw draad

In deze laatste fase wordt de blackbox van het segment uit paragraaf 2.1.8 gedupliceerd en aan elkaar gekoppeld, zoals afgebeeld in figuur 2.15.



Figuur 2.15: Draadmodel opgebouwd uit de blackboxes van de eindcap, de segmenten en het halve segment ( $n=7$ ).



De bijbehorende PSPICE-code voor  $n=5$  luidt:

```
* ----- *
*                               Opbouw helft van het 5-segmenten model                               *
* ----- *

Xcap      1                               Eindcap
Xblok1    1  2  3      21  22  23      999 Segment
Xblok2    3  4  5      41  42  43      999 Segment
Xblok3    5  6                               61  62  63      999 Segmenthalf

Xstroom   999                               Stroomsignaal_dc
```

Omdat hier geen sprake meer is van blackbox definities maar alleen een aaneenschakeling ervan, zijn alle knooppunten zogenaamde globale knooppunten, waarvan na simulatie de spanning als functie van de tijd kan worden bekeken. Interne knooppunten van blackboxes kunnen niet worden bekeken.

De nummering van de knooppunten (getallen) is willekeurig te kiezen maar moet echter overeenkomen met de volgorde van nummering bij de definitie van de blackbox. Als voorbeeld de definitie van netwerkdeel 'blok1' die de schakeling van het segment bevat.

In PSPICE code wordt dit:

```
Xblok1  1 2 3  21 22 23  999
```

De kopregel van de bijbehorende segmentdefinitie (zie paragraaf 2.1.8) luidde:

```
.SUBCKT Segment 1 2 3 4 5 6 7
```

Bij de plaatsing van segment blok1, wordt knooppunt 1 gekoppeld aan aansluitklem 1 van de blackbox, knooppunt 2 aan aansluitklem 2, ..., knooppunt 23 aan aansluitklem 6 en knooppunt 999 aan aansluitklem 7. Dit laatste is juist, omdat de blackbox van het segment op aansluitklem 7 het stroomsignaal verwacht en dat wordt in het model op knooppunt 999 afgegeven.

Door de spanningen op de verschillende knooppunten te vergelijken, kan een temperatuurprofiel als functie van de tijd worden gevonden. Hierbij representeert knooppunt  $n + 1$  het midden van de draad. Voor  $n=5$  levert dit de volgende signalen (bedenk dat toename in spanning overeenkomt met toename in temperatuur):

```
1      toename in temperatuur van eindcap;
2      toename in draadtemperatuur in het midden van segment 1;
3      toename in draadtemperatuur op grens van segment 1 en 2;
4      toename in draadtemperatuur in het midden van segment 2;
:      :
21     toename in luchttemperatuur in segment 1;
22     toename in temperatuur op rand van lucht en glas in segment 1;
23     toename in glastemperatuur in segment 1;
:      :
999    signaal van elektrische stroom  $i(t)$  door draad
```

Door toepassen van deze blackbox constructie is het eveneens zeer eenvoudig om het aantal segmenten  $n$  te vergroten. Hiervoor behoeven alleen maar meer segmenten te worden geschakeld. Uiteraard moet de waarde van  $n$  ook worden aangepast (zie paragraaf 2.1.11).

Voor de situatie  $n=7$  volgt logischerwijs:

Xcap	1							Eindcap
Xblok1	1	2	3	21	22	23	999	Segment
Xblok2	3	4	5	41	42	43	999	Segment
Xblok3	5	6	7	61	62	63	999	Segment
Xblok4	7	8		81	82	83	999	Segmenthalf
Xstroom	999							Stroomsignaal_dc

Merk op:

- Wanneer wordt verondersteld, dat de eindcap van de patroon op kamertemperatuur blijft, dan zal  $V_1$  altijd nul blijven;
- Gegeven het feit dat de draad in het midden het warmst wordt, zal voor ieder tijdstip gelden:

$$V_1 \leq V_2 \leq V_3 \dots \leq V_m = V_{n+1}$$

- Aangezien er sprake is van symmetrie in het model, is er ook sprake van symmetrie in het temperatuurprofiel. Zo zal in figuur 1.3 op bladzijde 11 gelden dat  $V_1 = V_{11}$ ,  $V_2 = V_{10}$ , etcetera. In formulevorm:

$$V_m = V_t = V_{2(n+1)-m} \quad \text{voor } m \in [1, 2, 3, \dots, n+1]$$

### 2.1.11 PSPICE model: Constanten

Zoals beschreven in paragraaf 5.4 kunnen ook constanten worden gedefinieerd. Deze constanten kunnen in de formules van het model (of blackbox) worden gebruikt.

Het PSPICE model kent slechts één constante definitie, waarmee het aantal segmenten wordt opgegeven waarin de hele draad is verdeeld.

```
.PARAM n = 7 ; aantal segmenten van HELE draad
```

Door aan de constante een andere waarde toe te kennen, ontstaat een model met andere eigenschappen.

**Let op:** Zoals in de vorige paragraaf is beschreven, dient hiermee ook het aantal geschakelde segmenten te wijzigen!

### 2.1.12 PSPICE model: Stuurcodes

In de laatste fase van de PSPICE modeldefinitie moet worden aangegeven, welk soort van simulatie is gewenst, met welke nauwkeurigheid moet worden gerekend en welke vorm van uitvoer noodzakelijk is. Hiervoor beschikt PSPICE over een aantal zogenaamde (punt)-commando's.

Voor dit model zijn de volgende puntcommando's gebruikt:

```
.PROBE
.OPTIONS RELTOL=0.01
.TRAN 10ms 1200ms 0ms 12ms
.END
```

**.PROBE** geeft aan dat de resultaten met het postprocessing programma PROBE bekeken gaan worden. Er wordt een databestand aangemaakt met daarin de resultaten, leesbaar voor PROBE. Wenst men numerieke uitvoer voor verdere verwerking in andere programma's als Matlab, dan kan het commando **.PRINT** worden gebruikt. Achter het commando wordt opgegeven welke signalen gewenst zijn. Zo geeft V(8) de spanning op knooppunt 8;

**.OPTIONS** geeft extra instellingen aan, in dit geval in verband met de rekennauwkeurigheid. Met RELTOL=0.01 wordt aangegeven dat de simulatieresultaten een maximale relatieve fout van 1% mogen hebben;

**.TRAN** geeft aan dat er sprake is van een TRANsiënt simulatie (simulatie in de tijd). Met een viertal parameters worden instellingen aangegeven. Achtereenvolgens zijn dit:

1.  $\Delta t$  tussen twee opeenvolgende tijden in de resultatenfile;
2. de totale simulatietijd;
3. het tijdstip van het eerste datapunt in de resultatenfile;
4. de maximale stapgrootte in tijd tijdens de simulatie.

**.END** markeert het einde van de PSPICE invoerfile.

Voor een meer uitgebreide uiteenzetting van de beschikbare puntcommando's binnen PSPICE wordt verwezen naar de handleiding.

Hiermee is de PSPICE modelbeschrijving van de smeltdraad compleet en kan worden gesimuleerd. In bijlage D is de volledige listing van het model opgenomen.

## 3. Simulaties

*In dit hoofdstuk wordt aan de hand van voorbeeld code uiteengezet hoe met het PSPICE model verschillende simulaties kunnen worden gedaan.*

*De meest belangrijke grafieken worden getoond.*

Het model dat in hoofdstuk 1 van deel 3 uitvoerig besproken is, is qua dimensies gebaseerd op één type miniatuurpatroon: 217.315. Deze patroon bevat een nikkeldraad met een doorsnede van  $50\mu\text{m}$  en een lengte van ongeveer 1,75 cm. De draad is omgeven door een glazen cilinder met een diameter van ongeveer 5mm. Het tussenliggend medium is lucht. Voor specifieke gegevens wordt verwezen naar bijlage C. De patroon is ontworpen voor een nominaalstroom van 315mA, 250V.

In het PSPICE model kan de vorm van de elektrische stroom door de nikkeldraad als functie van de tijd eenvoudig worden vastgelegd.

Dit maakt het mogelijk om twee speciale gevallen te bekijken:

- Allereerst wordt gesimuleerd met een gelijkstroom. De grootte van de gelijkstroom kan eenvoudig worden gewijzigd. Dit maakt het mogelijk het model te verifiëren aan de hand van de bij de patroon horende  $I_t$ -curve. Deze curve is weergegeven in bijlage E.
- Om relaties te leggen tussen de simulaties en de metingen aan het pulscircuit is het noodzakelijk om ook met pulsvormige stromen te simuleren. Als voorbeeld wordt gesimuleerd met een stroompuls die is afgeleid van een pulsvorm uit het meetcircuit.

De resultaten van de simulaties worden getoond en vergeleken met de praktijk. Conclusies die voortvloeien uit de resultaten worden in hoofdstuk 3 besproken.

### 3.1 Constante stroom: controle $I_t$ curve

Het principe voor het bepalen van een  $I_t$ -curve van een patroon is eenvoudig. Door de patroon wordt een gelijkstroom met een bepaalde waarde gestuurd, die wordt ingeschakeld op tijdstip  $t=0$ . Op het moment van doorgaan van de smeltdraad in de patroon wordt de tijd opgenomen. Dit wordt herhaald bij verschillende waarden van gelijkstroom. Ieder  $(I,t)$ -paar vormt een coördinaat in de  $I_t$ -curve.

In de praktijk worden een beperkt aantal waarden van stroom gehanteerd, uitgedrukt in 'zoveel maal de nominaalstroom'. Voor de hand liggende waarden voor dit type glaspatroon zijn  $2I_n$ ,  $3I_n$ , ... tot  $10I_n$ .

Omdat er altijd sprake is van spreiding in de eigenschappen van patronen, ondermeer veroorzaakt door de wisselende materiaaleigenschappen en tijdens de fabricage, moeten voor het verkrijgen van een representatieve  $I_t$ -curve deze meting vele malen worden herhaald.

Hierdoor ontstaat een band van  $I_t$ -curven, waarbinnen iedere patroon zal liggen. In bijlage E is dit weergegeven voor de 217.315 miniatuurpatroon van Littelfuse.

Bij het simuleren zal de stroom op een zelfde reeks waarden worden ingesteld en op  $t=0$  worden geschakeld. Vanaf dat moment wordt gesimuleerd totdat er theoretisch sprake is van doorsmelten van de draad. Dit houdt het volgende in:

- Zoals uit de simulaties blijkt, bereikt het midden van de draad de hoogste temperatuur. Het middelste segment zal daarom ook het snelst de smelttemperatuur bereiken. Het tijdstip waarop deze temperatuur wordt bereikt, is gedefinieerd als de smelttemperatuur;
- Hierbij wordt géén rekening gehouden met eventuele fysische aspecten die het tijdstip van smelten verleggen;
- De verkregen stroom-tijd combinatie wordt ingetekend in de praktische  $I_t$ -curve en de curven worden vergeleken.

### 3.1.1 Uitvoering

Als voorbeeld wordt een berekening uitgevoerd bij  $I_{DC} = 3 \cdot I_n = 3 \cdot 315mA = 945mA$ . Gebruik makend van de modeltheorie voor definitie van de stroom  $i(t)$  uit paragraaf 2.1.7, wordt de blackbox als volgt gedefinieerd:

```
* =S T R O O M S I G N A A L===== *
*
*      1          V(1) is het signaal van i(t)
*      + |
*      ( ) Estroom
*      |
*      ~
.SUBCKT Stroosignaal 1
Vdc      2 0  PWL ( 0,0 1ns,945E-3 1000s,945E-3)
Rdc      2 0  1MEG
Vstep    3 0  1V
Rstep    3 0  1MEG
Estroom  1 0  VALUE = { V(2) * V(3) }
Rstroom  1 0  1MEG
.ENDS Stroosignaal
```

Vervolgens moet een juiste simulatietijd worden gekozen voor gebruik bij het **.TRAN** commando. Eigenlijk is dit een kwestie van 'trial and error', maar als er van uit gegaan wordt dat de simulatie redelijk zal overeenkomen met de werkelijke  $I_t$ -waarden, dan is een totale simulatietijd van 300 milliseconde acceptabel.

De maximale stapgrootte wordt gesteld op 2ms, daar bij grotere waarden van tijd blijkt dat PSPICE niet altijd in staat is om zelf de tijdstap voldoende te verkleinen, als dat rekentechnisch noodzakelijk is.

*Een opmerking is hier op zijn plaats: Het is noodzakelijk gebleken om te werken met een zelf in te stellen maximale tijdstap bij het .TRAN commando (4e parameter) omdat PSPICE anders in de problemen kan komen. Dit uit zich in een melding Unable to finish transient analysis waarmee de simulatie wordt gestopt.*

*Door een te grote rekenstap worden de numerieke rekenfouten steeds groter (divergentie). Kennelijk heeft ook het PSPICE tijdstap algoritme soms problemen, vooral bij snel veranderende signalen.*

*Door de gefixeerde maximale tijdstap kunnen deze misstappen worden voorkomen. Maar het heeft wel een nadeel: In stabiele gebieden, waar de tijdstap zonder problemen groter zou kunnen zijn, blijft de maximum rekenstap van kracht. Dit leidt daardoor tot wat langere rekentijden.*

*Dergelijke eigenschappen zijn inherent verbonden aan het toepassen van een numeriek algoritme en komen ook bij andere rekenpakketten voor. Het is daarom altijd een kwestie van zorgvuldig kiezen van staptijden.*

De PSPICE stuurcodes zoals beschreven in paragraaf 2.1.12 worden daarmee als volgt:

```
.PROBE
.OPTIONS RELTOL=0.01
.TRAN 5ms 300ms 0ms 2ms
.END
```

Hierbij is gekozen voor uitvoer naar PROBE, met een meetwaarde na iedere 5ms simulatietijd en een maximale toegestane relatieve fout van 1%.

Het model is hiermee gereed en kan worden gebruikt voor simulatie.

### 3.1.2 Resultaten

In deze paragraaf worden de resultaten gegeven van de simulaties die met het model zijn uitgevoerd bij excitatie met constante stroom.

#### It-curve

Nadat het model met PSPICE is doorgerekend, kan met de postprocessor PROBE van PSPICE, het beeld uit figuur 3.1 worden verkregen.

Het beeld toont de toename in temperatuur van knooppunt 8, het midden van de draad. Bijlage B geeft voor 99% zuiver nikkel een smelttemperatuur 1728K, dus een toename van 1428K uitgaande van kamertemperatuur.

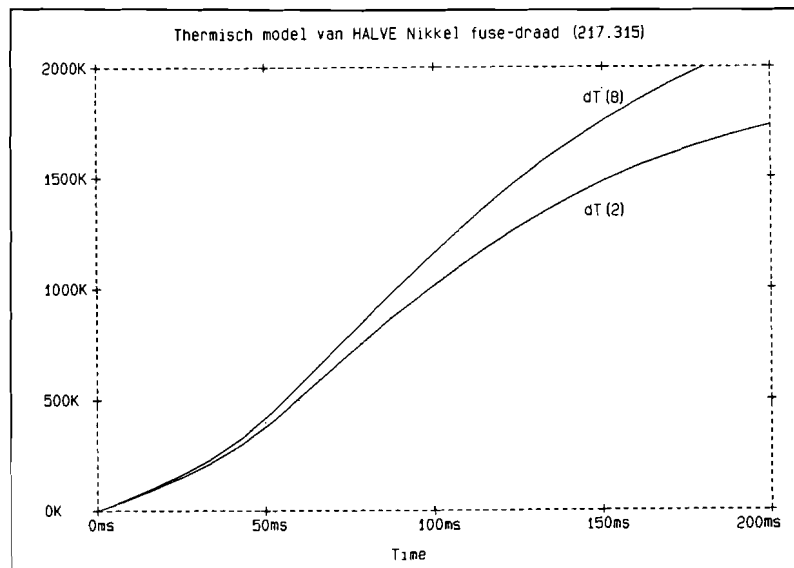
Met de cursor-optie kan worden afgelezen op welk tijdstip deze theoretische smelttemperatuur zal zijn bereikt. Er volgt  $t_{smelt} = 148,5\text{ms}$ .

In dezelfde grafiek is ook het temperatuurverloop van het meest linkse segment (segment 2) weergegeven.

Op dezelfde manier kan voor alle andere gewenste stromen het model worden doorgerekend.

Voor lagere overstromen ontstaan langere smelttijden en er is een grens, waarbij de smelttemperatuur niet meer wordt bereikt, in dit geval ongeveer bij  $I=2I_n$ .

Een voorbeeld hiervan is te zien in figuur 3.2 waarbij de simulatie is uitgevoerd met een stroom gelijk aan de nominaalstroom. De temperatuur van het midden van de draad bereikt een eindwaarde gelijk aan  $174,8 + 27 = 201,8^\circ\text{C}$ .



Figuur 3.1: Afdruk van PROBE-scherm met de toename in temperatuur van het midden van de draad V(8) en van segment 2 V(2) ( $I=3I_n$ ).

Omdat de temperatuur zijn eindwaarde langzaam nadert is het niet zinvol om te spreken over DE tijd waarop de eindwaarde wordt bereikt. Er is in dat geval gekozen voor het noteren van de tijd, waarop de stijging van temperatuur op 90% van de eindwaarde komt. Dit tijdstip is veel eenvoudiger uit de grafiek te bepalen, omdat de temperatuur gradiënt daar veel groter is. Voor de nominaalstroom wordt voor  $t_{90\%}$  een tijd gelijk aan 905ms.

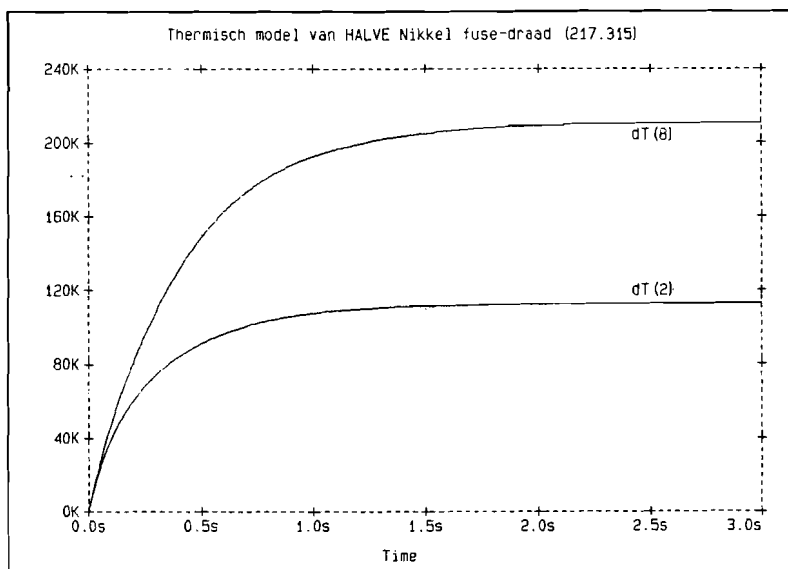
De simulatieresultaten kunnen nu in tabelvorm worden weergegeven.

I/ $I_n$	$\Delta T_{eind}$ [ $^{\circ}C$ ]	90% $t_{eind}$ [ms]
1	174,8	905
1,5	632,1	604
2	1139,0	432

Beide tabellen geven resultaten van stroomwaarden die nodig zijn voor het verkrijgen van een goede It-curve. Uiteraard is het mogelijk om ook met andere stroomwaarden de simulatie uit te voeren.

I/ $I_n$	$t_{smelt}$ [ms]
2,25	865
2,3	470
2,4	338,5
3	148,5
4	70,68
5	42,0
6	27,86
7	20,0
8	15,21
10	9,42

Nu voor een groot aantal stromen de smelttijden bekend zijn, kan een vergelijk worden gemaakt met de It-curve van de 217.315 patroon, zoals weergegeven in bijlage E. De berekende waar-



Figuur 3.2: Afdruk van PROBE-scherm met de toename in temperatuur van het midden van de draad V(8) en van segment 2 V(2), ( $I=I_n$ ).

den zijn ingetekend en met elkaar verbonden door rechte lijnstukken. Het resultaat staat in grafiek 3.3 weergegeven.

In het volgende hoofdstuk wordt het resultaat van deze simulatie uitvoeriger besproken. Hier wordt volstaan met het geven van de grafiek.

### Plaats-tijd afbeelding

Tot nu toe is alleen gekeken naar het verloop van de temperatuur in het midden van de draad (knooppunt 8). Uiteraard is het ook mogelijk om van de andere knooppunten de temperatuurstijging te bekijken.

Door de verschillende grafieken op een speciale manier te combineren, kan een beeld worden gevormd over het optredend temperatuurprofiel over de draad.

Er ontstaat dan een drie-dimensionale afbeelding zoals in grafiek 3.4 is weergegeven. Hierin staan plaats, tijd en temperatuurstijging tegen elkaar uitgezet. Na ongeveer 3s wordt een evenwicht bereikt bij een maximale draadtemperatuur van zo'n 250°C.

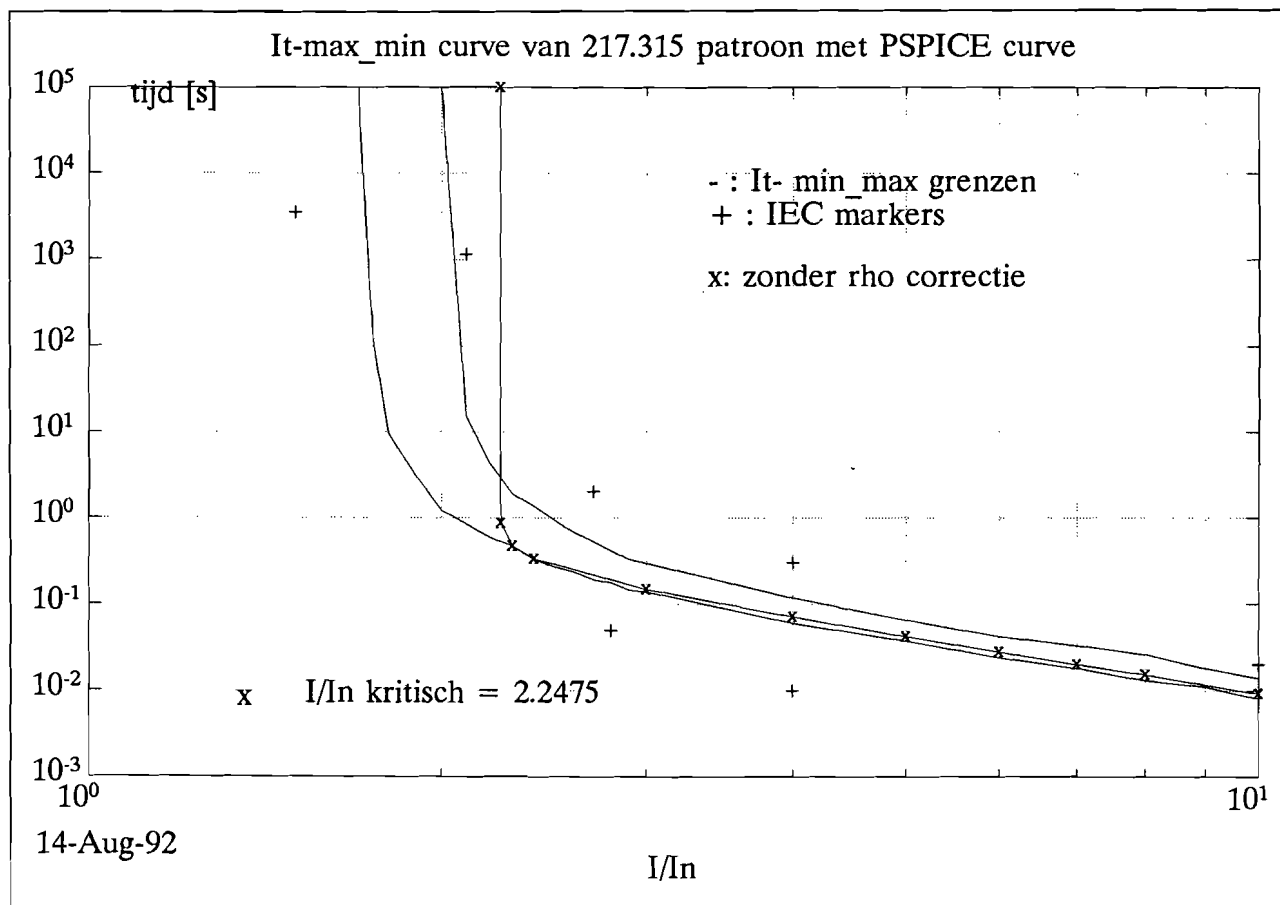
### Temperatuur in lucht en glas

Naast de draadtemperatuur kan ook het verloop van temperatuur op een plaats in de lucht en in het glas van ieder segment worden bekeken.

Als voorbeeld staat in figuur 3.5 het verloop ervan in het middelste segment (Xblok4) weergegeven, voor  $I=I_n$ . De nummers hebben de volgende betekenis:

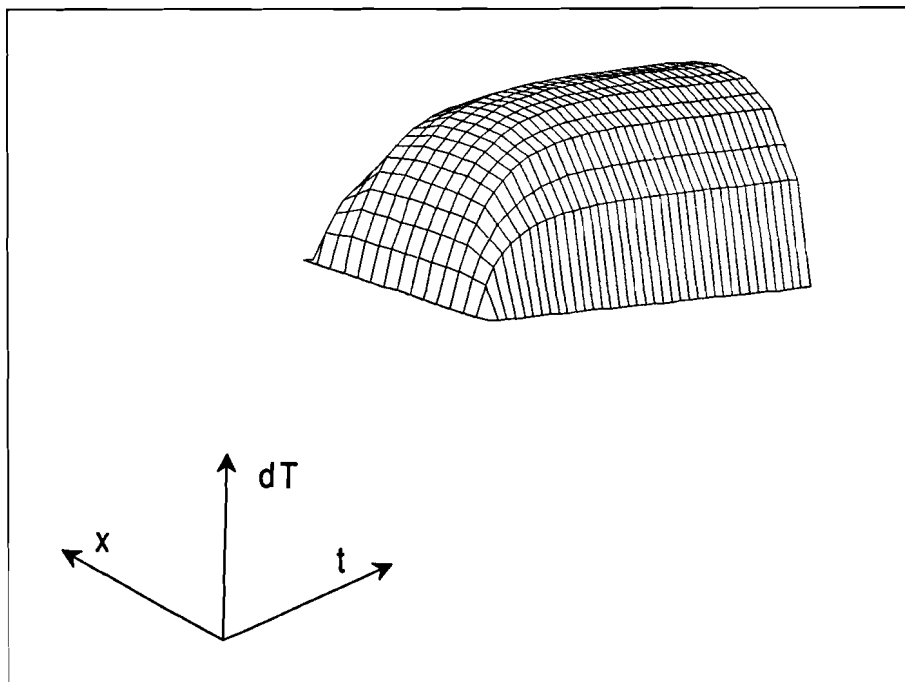
1. temperatuurstijging van draadsegment, V(8);
2. temperatuurstijging ergens in de lucht, V(81). Op dit punt is de warmtecapaciteit van de lucht aangesloten (figuur 1.10);



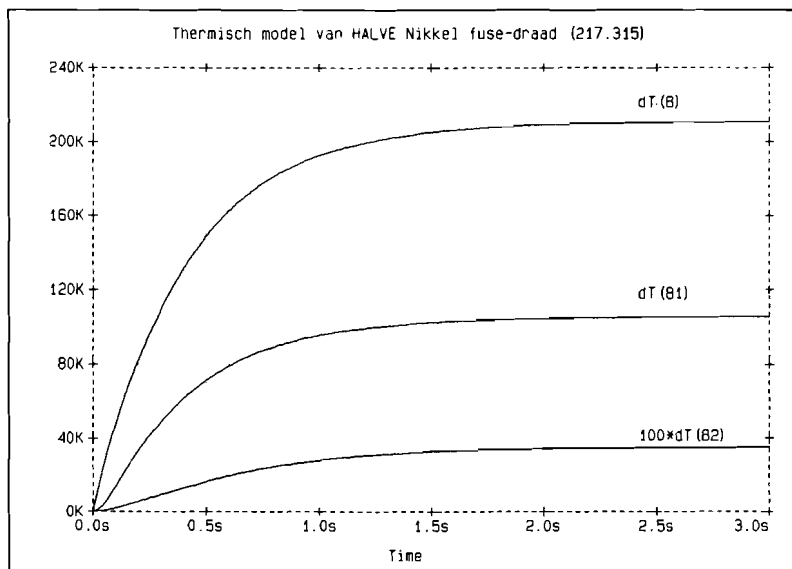


Figuur 3.3: Vergelijk van gesimuleerde It-karakteristiek met de geldende It-curven voor 217.315 patroon.

3. temperatuurstijging op overgang van lucht en glas (binnenzijde) maal 100, V(82). Deze stijging is minimaal.



Figuur 3.4: 3D weergave van de temperatuurverdeling over de draad. Er ontstaat evenwicht na  $\pm 3s$  bij een maximale temperatuur van  $\pm 250^{\circ}C$ .



Figuur 3.5: Indicatie van temperatuurverloop in nikkel, in lucht en in glas van middelste segment ( $I=I_n$ ).

## 3.2 Pulsstromen

De simulaties die gedaan zijn met de gelijkstroom kunnen zonder meer ook herhaald worden voor pulsformige stromen.

In deze paragraaf wordt gesimuleerd met een theoretisch verloop van  $i(t)$ , zoals dit bij een LRC-circuit met constante componentwaarden het geval is. In praktijk is er sprake van veranderende weerstand, waardoor de pulsform wijzigt.

In dat geval kan de simulatiestroom het best worden gedefinieerd met den PWL- of TABLE-functie van PSPICE, zodat het werkelijk gemeten stroomsignaal zo goed mogelijk kan worden benaderd.

Voor het simuleren met de theoretische pulsstroom behoeft alleen het gedeelte van de stroomdefinitie in PSPICE te worden aangepast.

### 3.2.1 Uitvoering

Als voorbeeld wordt een berekening uitgevoerd met een pulsstroom, opgewekt met het in hoofdstuk 2 beschreven circuit. Hierbij zijn de volgende componentwaarden aangehouden:

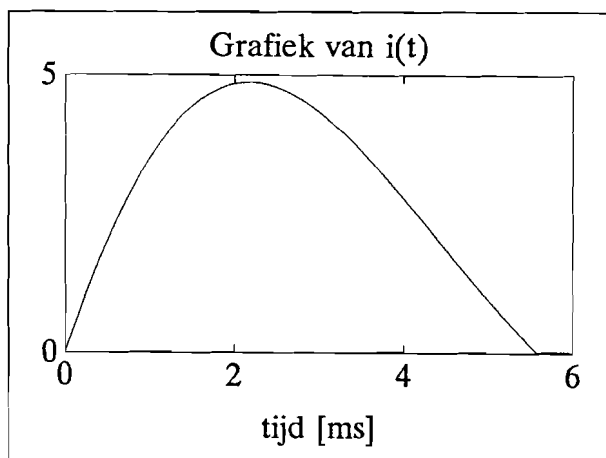
- $U_o = 10V$ ;
- $L = 2.2mH$ ;
- $C = 1.27mF$ ;
- $R = 1\Omega$ ;

Deze waarden kunnen in formules 7.2 en 7.4 worden ingevuld. Er volgt:

$$\omega = \sqrt{\frac{1}{LC} - \left(\frac{R}{2L}\right)^2} \Leftrightarrow f = 89,4772\text{Hz}$$

$$i(t) = \frac{U_o}{\omega L} e^{-\frac{R}{2L}t} \cdot \sin(\omega t) = 8,0851 e^{-204,5455t} \cdot \sin(562,2019t)\text{A}$$

$$t_{nul,I} = \frac{\pi}{\omega} = 5,588\text{ms}$$



Figuur 3.6: Stroompuls  $i(t)$  waarmee het model is doorgerekend. Voor de componentwaarden geldt:  $L=2,2mH$ ,  $C=1.27mF$ ,  $R=1\Omega$ ,  $U_c=10V$ .

Gebruik makend van de modeltheorie voor definitie van de stroom  $i(t)$  uit paragraaf 2.1.7, wordt hiermee de PSPICE code voor de blackbox van het stroomsignaal als volgt:

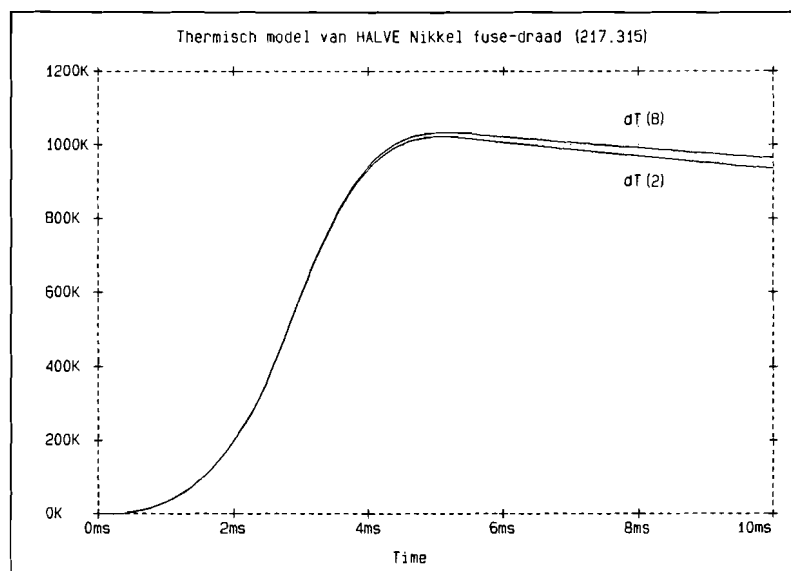
```

* =S T R O O M S I G N A A L===== *
*
*      1          V(1) is het signaal van i(t)
*      + |
*      ( ) Estroom
*      |
*      ^
.PARAM frequentie = 89.4772 ; frequentie van stroomsignaal
.PARAM amplitude  =  8.0851 ; amplitude van ongedempte sinus
.PARAM demping    = -204.5455 ; dempingsfactor

.SUBCKT Stroomsignaal 1
Vsendemp 2 0 SIN( 0V {amplitude} {frequentie} 0 {-demping} )
Rsendemp 2 0 1MEG
Vstep    3 0 PULSE( 1V 0V {1/(2*frequentie)-1e-6} 1e-6 1e-9 )
Rstep    3 0 1MEG
Estroom  1 0 VALUE = { V(2) * V(3) }
Rstroom  1 0 1MEG
.ENDS Stroomsignaal

```

### 3.2.2 Resultaten

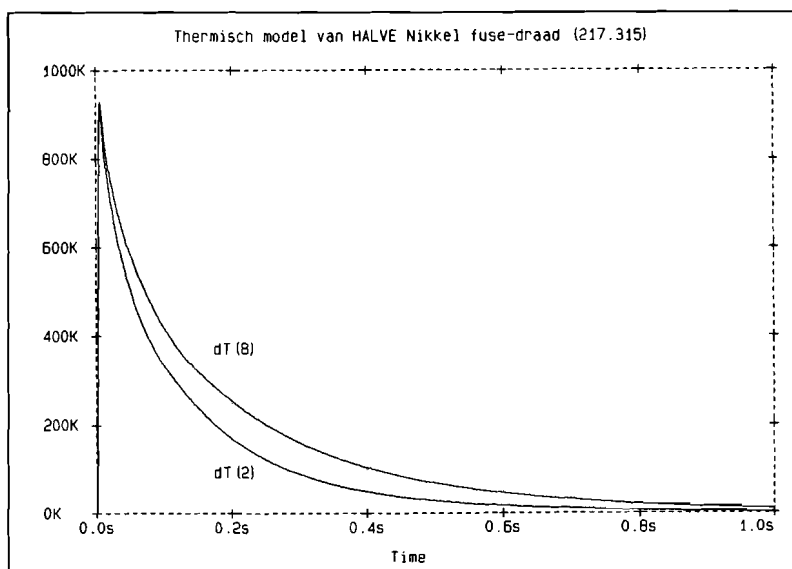


Figuur 3.7: Temperatuurstijging tijdens puls van knooppunten 2 en 8 (korte tijd).

Met het nieuwe stroomsignaal wordt gesimuleerd en met PROBE kunnen de resultaten op eenzelfde manier grafisch worden weergegeven.

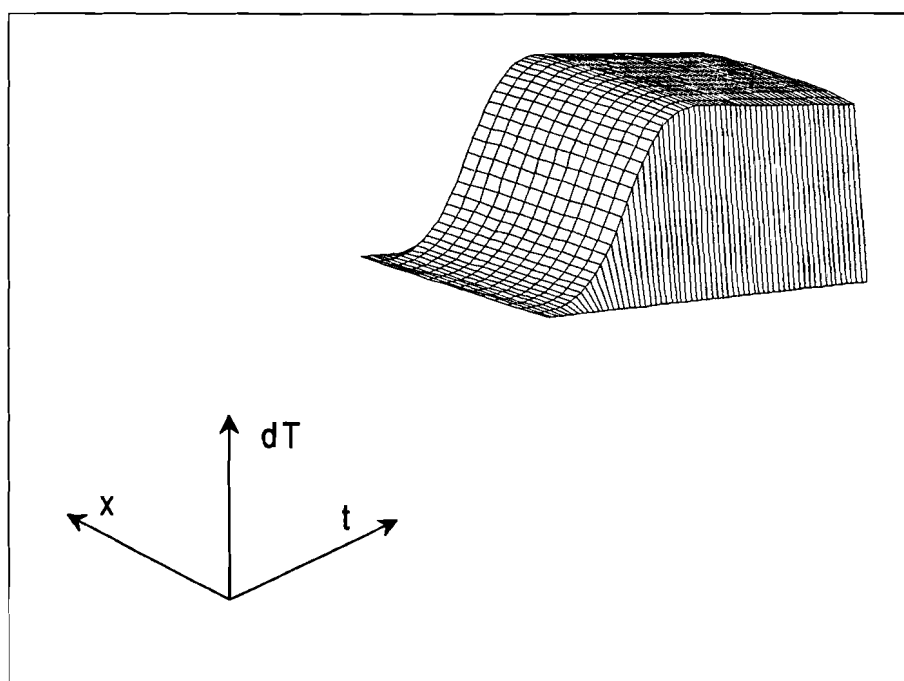
Achtereenvolgens zijn in de figuren weergegeven:

- Stroompuls  $i(t)$ , in figuur 3.6;



Figuur 3.8: Temperatuurstijging tijdens puls van knooppunten 2 en 8 (lange tijd).

- Temperatuurstijging op punten 8 (midden) en 2 in eerste 10 ms, in figuur 3.7;
- Idem, maar nu voor een groter tijdstraject, in figuur 3.8;
- Temperatuurprofiel van draad, in figuur 3.9.



Figuur 3.9: 3D temperatuurprofiel van draad tijdens puls. Maximale temperatuurstoename bedraagt  $\pm 1100^{\circ}\text{C}$  na  $\pm 6\text{ms}$ .

## 4. Conclusies en aanbevelingen

*In dit laatste hoofdstuk worden voor zover mogelijk conclusies getrokken uit de simulaties die met het model zijn gedaan en waarvan de resultaten in het vorige hoofdstuk zijn getoond.*

Alvorens de conclusies te bespreken, dient te worden opgemerkt dat in dit stadium nog geen vergelijkingsresultaten bestaan tussen de praktijkmetingen met de opstelling uit hoofdstuk 2 enerzijds en het simulatiemodel anderzijds.

Alle aandacht is in de laatste fase van het afstuderen gevestigd op het model en de bijbehorende gedetailleerde verslaglegging.

### 4.1 Conclusies It-curve

Uit de gesimuleerde It-curve van grafiek 3.3 op bladzijde 53 kunnen de volgende conclusies worden getrokken:

- De gesimuleerde curve valt niet geheel binnen de It-schaar van curven van de 217.315 patroon van Littelfuse;
- In het gebied  $I > 3In$  ligt de gesimuleerde curve voor een groot gedeelte binnen- of net onder de grens. Dit betekent dat bij de simulatie de theoretische smelttemperatuur te snel wordt bereikt. Er kan te weinig warmte worden afgevoerd, waardoor de temperatuur te snel toeneemt.

In de praktijk is er sprake van een bepaalde smelttijd, die ligt tussen het bereiken van de smelttemperatuur en het werkelijk doorgaan van de draad. Hierdoor komt de gesimuleerde curve iets hoger te liggen, wat gunstig is;

- Voor kleinere overstromen wordt de afwijking groter en ontstaat de situatie waarbij het volgens de simulatie juist langer duurt alvorens de smelttemperatuur is bereikt. Dit duidt erop dat in die situatie te veel warmte kan worden afgevoerd, zodat de temperatuur niet snel genoeg stijgt.

De genoemde conclusies duiden erop dat het model reeds een goede benadering van de draad is, maar nog niet exact juist is. Een paar oorzaken hiervan zijn wel te geven:

- De materiaaleigenschappen zoals gebruikt voor het model zijn gebaseerd op 99% zuiver nikkel. Het is echter de vraag in hoeverre het werkelijke draad materiaal hieraan voldoet;
- Onvolkomenheden in de nikkeldraad (verontreinigingen, inhomogeniteiten, wijzigingen in diameter, etc.) zijn niet te modelleren;
- Het proces voor convectie, dat voor lage overstromen zeer belangrijk wordt, is gebaseerd op een empirische formule en is daarom niet exact. Ook is de invloed van de glasomhulling op convectie onbekend;

- De invloed van de soldeerverbinding tussen draad en eindcap is niet meegenomen, evenals de warmtecapaciteit ervan;

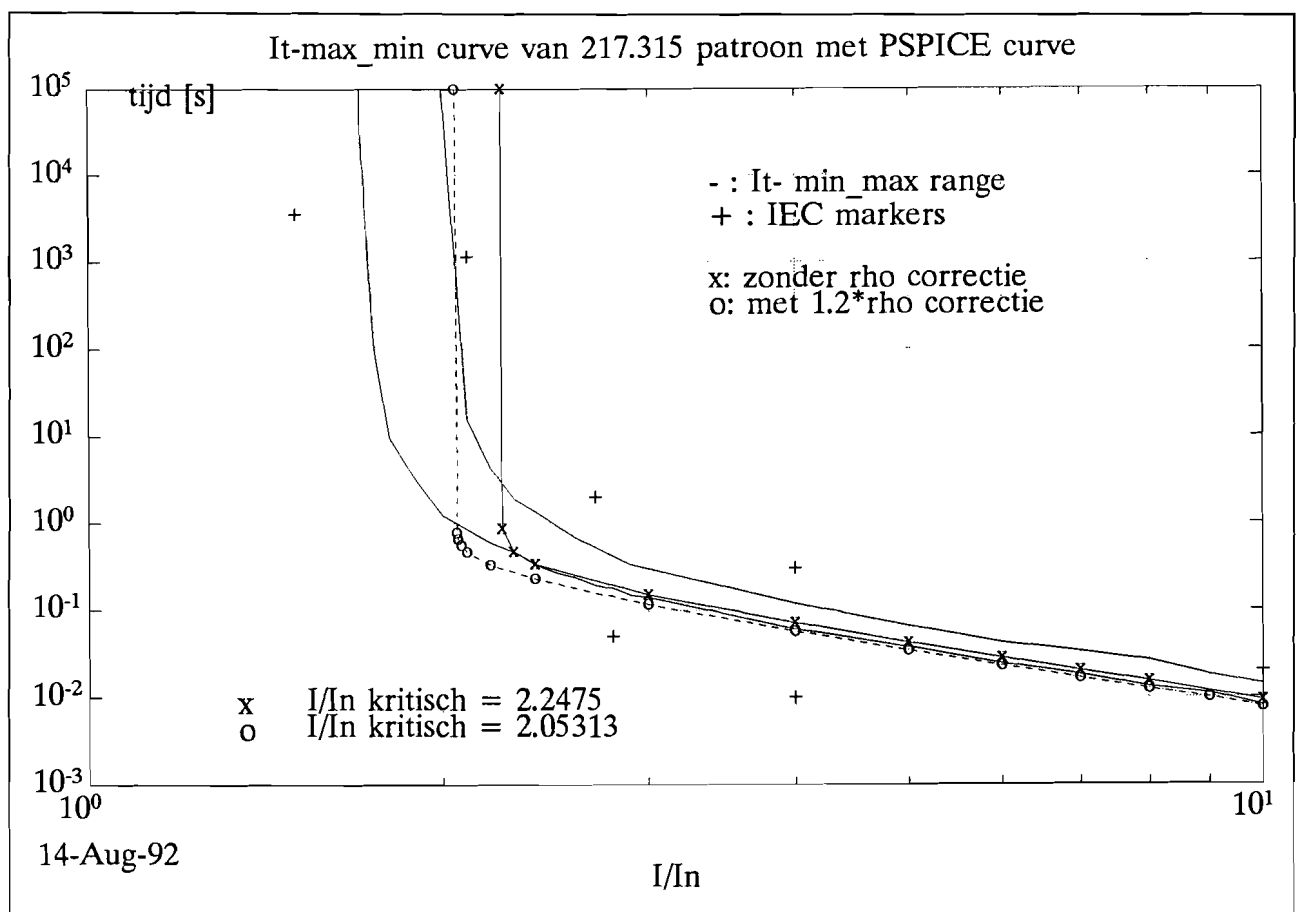
Het is niet eenvoudig deze laatste aspecten in termen van modelgrootheden te beschrijven. Dit vereist een verdere studie.

#### 4.1.1 Correctie It-curve

Het is mogelijk om een correctie uit te voeren op de gesimuleerde It-curve, die is gebaseerd op een wijziging in de elektrische geleidbaarheid van nikkel, waardoor de hoeveelheid gedissipeerde warmte in de draad wijzigt.

De correctie is gebaseerd op het verschil in elektrische weerstand tussen de gesimuleerde draad en de weerstand van de echte patronen, gemeten bij kamertemperatuur.

Hiertoe is van 10 stuks 217.315 patronen de koudweerstand bepaald en is het gemiddelde genomen:  $R_{gem} = 833,85m\Omega$ . Vergeleken met de berekende koudweerstand van de simulatiedraad (gebaseerd op 99% nikkel en de geldende draadafmetingen) is dit een factor 1,2 hoger.



Figuur 4.1: Grafiek met hierin de gecorrigeerde It-curve.



Deze correctie is doorgevoerd in het model, door de waarden uit de tabel met de elektrische weerstand per segment ( $R_{el}$ ) met een factor 1,2 te vermenigvuldigen en opnieuw een It-curve te bepalen.

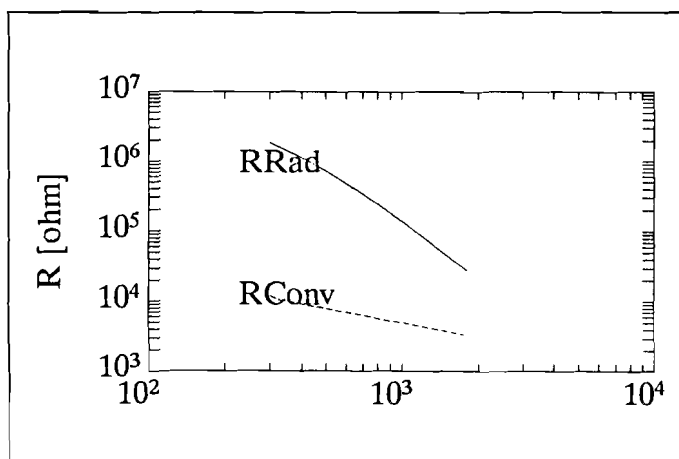
Deze nieuwe karakteristiek is in figuur 4.1 bijgetekend (samen met de eerste karakteristiek). De resultaten zijn eveneens in tabelvorm opgenomen.

I/In	$\Delta T_{eind}$ [°C]	90% $t_{eind}$ [ms]
2	472	1355
2,05	380	1425

I/In	$t_{smelt}$ [ms]	I/In	$t_{smelt}$ [ms]
2,05625	776	4	56,65
2,0625	654	5	34,15
2,075	558	6	22,64
2,1	468	7	16,61
2,2	330	8	12,35
2,4	226,2	9	9,93
3	115,6	10	7,71

Het verloop van de gecorrigeerde It-curve toont een betere benadering, vooral in het gebied van de lage overstromen. De curve is verder naar links geschoven. Ook is te zien dat de curve iets naar beneden is gezakt, verder van de curve vandaan. Hieruit kan worden geconcludeerd, dat toepassen van de correctie een verbetering is, maar nog niet voldoende. Ook andere aspecten zullen bij een vervolgstudie moeten worden onderzocht.

#### 4.1.2 Straling versus convectie



Figuur 4.2: Vergelijk van R<sub>rad</sub> en R<sub>conv</sub> uit het model.

Bij de PSPICE modelvorming zijn voor de convectie en straling modellen ontwikkeld in de vorm van een spanningsafhankelijke weerstand. Door beide karakteristieken met elkaar te vergelijken op logaritmische schaal (figuur 4.2), kan de volgende conclusie worden getrokken:

De stralingsweerstand R<sub>rad</sub> is over het gehele temperatuurgebied groter dan de convectieweerstand R<sub>conv</sub>. Dientengevolge is voor deze configuratie van de minia-

tuurpatroon de invloed van convectie groter dan die van straling, ook voor hogere temperatuur. Dit betekent dat aan de modellering van convectie de uiterste zorg moet worden besteed!

Ondanks dit feit moet men zich toch afvragen of dit niet duidt op een niet geheel correcte modellering van convectie en/of straling. De stralingsweerstand gaat immers volgens een vierde macht ( $T^4$ ) omlaag.

Dit aspect verdient bij een verder onderzoek nog enige aandacht.

## 4.2 Conclusies pulsstromen

Omdat nog geen goed vergelijk gemaakt kan worden tussen het temperatuurverloop van het PSPICE gesimuleerde model en het berekende verloop gebaseerd op de meting van spanning en stroom, is het aantal conclusies dat hieruit kan worden getrokken nog beperkt.

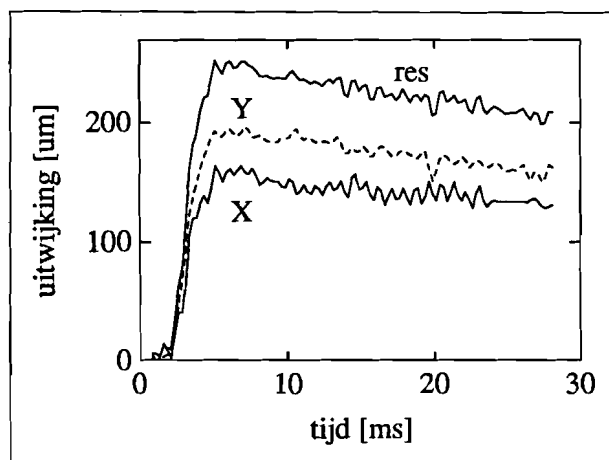
### Het temperatuurprofiel

Uit het gesimuleerde temperatuurverloop met de stroompuls uit figuur 3.6 en het bijbehorende temperatuurprofiel uit figuur 3.7 en figuur 3.8, kan worden geconcludeerd dat de draad egaal wordt opgewarmd. Omdat de eindcaps op constante temperatuur worden gehouden, treden op de overgang van cap naar draad (eerste segment) grote temperatuurgradienten op.

Een wijziging in de modellering van de eindcaps kan dit fenomeen wellicht beïnvloeden.

### Temperatuur versus uitwijking

Tijdens het uitvoeren pulstesten aan de miniatuurpatroon is gebleken dat de smeltdraad in de patroon zichtbaar een beweging uitvoert.



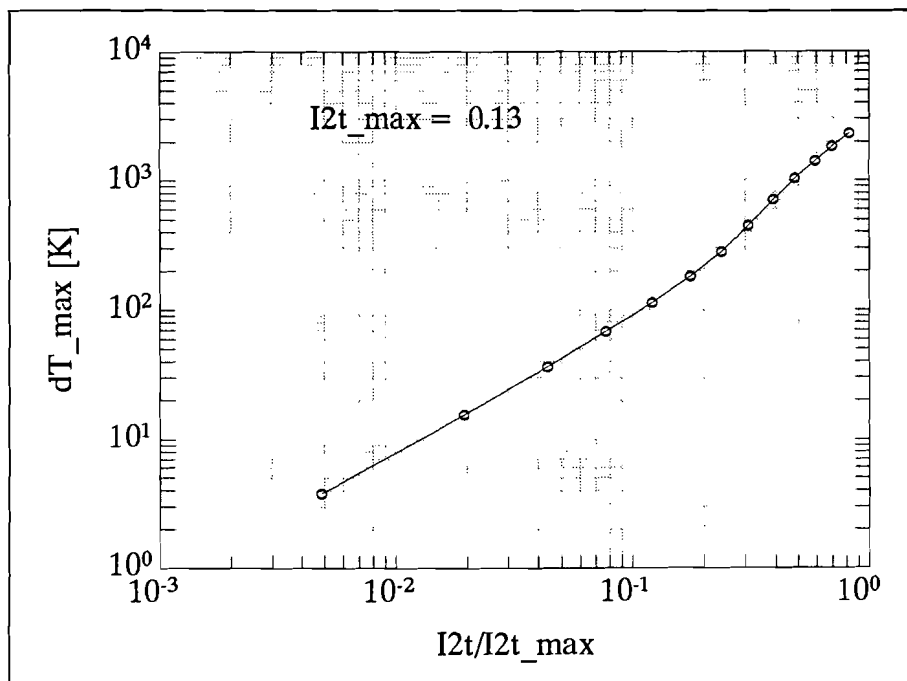
Figuur 4.3: Indikatie van de uitwijking van het midden van de smeltdraad tijdens stroompuls.

Dit fenomeen is in het kader van het promotie-onderzoek nader onderzocht en met gebruik van een snelle beeldjescamera is getracht om de beweging vast te leggen en de uitwijking te bepalen. In figuur 4.3 is het verloop van de uitwijking in het midden van de draad geschetst. Hierbij is de uitwijking berekend uit de bijdrage in zowel x- als y-richting.

Wanneer dit verloop wordt vergeleken met het verloop van de temperatuurstijging van het draadmidden tijdens de puls (figuur 3.7) blijkt een overeenkomstige vorm. Door de introductie van de thermische uitzettingscoëfficiënt  $\beta$  kan de relatieve lengtetoe name  $\beta\Delta T$  worden berekend.

De resultaten uit beide figuren spelen in het promotie-onderzoek een belangrijke rol bij de bepaling van de effecten die de lengtetoe name veroorzaken.

Door de maximale toename in temperatuur uit te zetten tegen de bijbehorende  $I^2t$  waarde van de puls kan de geldende relatie worden bepaald en worden gebruikt in het onderzoek.



Figuur 4.4: Relatie tussen  $I^2t$  en  $\Delta T$ , bepaald uit simulatie.

Hiertoe wordt uit figuur 7.5 bij gewenste  $I^2t$  de bijbehorende  $U_0$  gevonden en met bijvoorbeeld formule 7.2 de amplitudefactor  $A$  bepaald. Deze waarde wordt in het PSPICE model ingevoerd. Uit het gesimuleerde temperatuurverloop van het midden van de draad kan het maximum in toename worden afgelezen.

De zo ontstane grafiek staat in figuur 4.4 getoond.

## 4.3 Aanbevelingen

Dit hoofdstuk wordt afgesloten met een opsomming van aanbevelingen, die een goede aanzet kunnen geven voor vervolgstudies bij de verfijning van het opgezette PSPICE model. Vanwege de beperkte tijd zijn deze zaken niet meer uitgevoerd.

### Diameter wijzigen

Om meer zekerheid te krijgen over de juistheid van het model, is het raadzaam om ook nikkelpatronen met een andere draaddiameter te simuleren en de  $I_t$ -curve te vergelijken.

De besproken correctie van elektrische weerstand  $Rel$  behoeft niet geheel door wijziging van  $\rho$  te worden gerealiseerd. Het is ook mogelijk dat een wijziging in diameter noodzakelijk is.

Hiertoe moeten in het model de diverse equivalente weerstanden en capaciteiten worden gewijzigd. Dit kan relatief eenvoudig door het aanbrengen van een extra factor in de formules van het model, zoals gedaan met  $n$ , waarbij bij alle componenten de evenredigheid met de diameter wordt bijgevoegd. Een nieuw in te voeren parameter (bijvoorbeeld  $d$ ) is de verhouding tussen de nieuwe diameter en de waarde  $50 \cdot 10^{-6}$ , waarop het model gebaseerd is.

### Componentformules in PSPICE

Uit historisch gronden zijn bij de definitie van de diverse componenten de equivalente weerstand of capaciteit uit de tabel-functies buiten PSPICE berekend en later ingevuld. In deze numerieke waarden zitten de relevante afmetingen en materiaaleigenschappen.

Omdat het mogelijk is om binnen PSPICE gebruik te maken van formules, kan het rekenwerk ook binnen het model zelf worden gedaan en wel zodanig dat de tabel-functies allen nog de materiaaleigenschappen bevatten. Op die manier wordt het eenvoudig om ook voor andere materialen te simuleren, zonder dat de tabel opnieuw moet worden berekend.

### Eindcaps detail

Het model van de eindcaps kan worden verfijnd door het toepassen van bijvoorbeeld warmtecapaciteit of een voorgedefinieerd temperatuurpatroon.

### Totaalweerstand draad

Uit de berekeningen van het PSPICE model volgt niet direct het verloop van de totale draadweerstand. Met enige extra programmeerregels kan dit echter worden verwezenlijkt, zodat een vergelijk kan worden gemaakt met de berekende weerstand, gebaseerd op de metingen aan de smeltpatroon.

Hier wordt volstaan met het geven van de benodigde PSPICE code, die aan het model kan worden toegevoegd. Na simulatie is het spanningssignaal op knooppunt 207 equivalent met de totale weerstand van de halve draad (1V komt overeen met  $1\Omega$ ). De code is gebaseerd op  $n=7$ .

```
* =T O T A A L W E E R S T A N D   D R A A D===== *
*
* Berekenen van elektrische weerstand totale draad
```

\* V(207) is maat equivalent van elektrische weerstand draad

```
XRsegment1 201      0   (2,1) SegmentR
XRsegment2 202    201   (3,2) SegmentR
XRsegment3 203    202   (4,3) SegmentR
XRsegment4 204    203   (5,4) SegmentR
XRsegment5 205    204   (6,5) SegmentR
XRsegment6 206    205   (7,6) SegmentR
XRsegment7 207    206   (8,7) SegmentR
Rbelasting 207      0      1MEG

.SUBCKT SegmentR 1 2 3 4
EsegmentR      1 2    VALUE = { 2 * (V(6)+V(5))/2E3 * 2 }
Xrho_table_low 5 3 0  Rel_tabel
Xrho_table_hgh 6 4 0  Rel_tabel
.ENDS SegmentR
```

Met de blackbox 'SegmentR' wordt de elektrische weerstand van het segment berekend (spanning evenredig met weerstand). Door het aaneenschakelen van deze bronnen ontstaat de totale weerstand van het HALVE segment. Een extra factor 2 in de spanningsbron verdubbelt het totaal, waarmee alsnog de weerstand van de gehele draad wordt berekend.

Als met PSPICE het verloop van de gemiddelde weerstand gedurende de simulatie kan worden berekend, kan deze worden vergeleken met de gemiddelde weerstand, die is berekend uit de meetresultaten van V en I.

## 5. PSPICE: het gereedschap

*In dit hoofdstuk wordt een beknopte uiteenzetting gegeven van de meest belangrijke mogelijkheden van het netwerk simulatie pakket PSPICE. Het is bedoeld als extra informatie voor het verkrijgen van een beter inzicht in het gebruik van PSPICE. Technieken die bij de modelvorming zijn gebruikt, zullen hier worden beschreven.*

*Voor een uitgebreidere beschrijving van het programmapakket PSPICE kunnen naast de handleiding van PSPICE ook bijvoorbeeld het boek van Hoefler [2], Tuinenga [3], Keown [15] of Banzhof [16] worden geraadpleegd.*

### 5.1 Programmabeschrijving

Het softwarepakket PSPICE (PC version of Simulation Program with IC Emphasis), ontwikkeld door de Amerikaanse firma *MicroSim Corporation*, is gebaseerd op de reeds lang bestaande Berkley SPICE versies.

PSPICE wordt gezien als een van de toonaangevende pakketten op het gebied van simulatiesoftware.

Hart van het pakket wordt gevormd door de netwerk compiler met bijbehorende componenten bibliotheken. De compiler bevat code voor het interpreteren van de invoer bestanden en het omzetten ervan naar rekenmatrices, numerieke routines voor het oplossen van de stelsels vergelijkingen en routines voor het aanmaken van de uitvoer bestanden.

Een invoer bestand voor PSPICE is een ASCII bestand, waarin volgens een bepaalde methode het netwerk wordt omschreven. Dit bestand kan met een willekeurige editor worden aangeemaakt.

Uitvoer kan in grafische vorm worden weergegeven met een apart programma PROBE. Hiermee kunnen ook afdrukken worden gemaakt op printers en plotters.

De componentenbibliotheken bevatten karakteristieken van elektrische componenten als transistoren, dioden, etcetera. Met het programma PARTS kunnen nieuwe componenten bijgevoegd en of gewijzigd worden.

#### 5.1.1 PSPICE versus MicroCap III

Naast PSPICE bestaan er ook andere software pakketten voor het doorrekenen van elektrische netwerken op de PC. De meest bekende hiervan is MicroCap, uitgegeven door de Amerikaanse firma *Spectrum Software*.

Zonder in detail beide pakketten met elkaar te vergelijken, wordt volstaan met het opsommen van een aantal argumenten, die er toe geleid hebben om PSPICE te gebruiken in plaats van MicroCap, versie III:

- MicroCap III kent geen POLY-, TABLE-, VALUE-functie;
- Het gebruik van blackboxes is in MicroCap III beperkt;
- PSPICE is robuuster, meer flexibel;
- Uitvoerfaciliteiten zijn in MicroCap III beperkt.

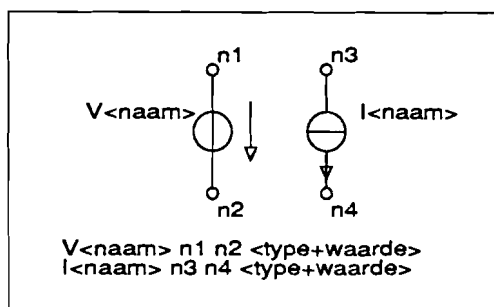
## 5.2 PSPICE bronnen

Naast R, L en C vormen stroom- en spanningsbronnen een belangrijk onderdeel van een elektrisch netwerk. PSPICE staat het gebruik van deze bronnen in diverse variaties toe:

- tijdgestuurde bronnen;
- spanningsgestuurde bronnen;
- stroomgestuurde bronnen;

De afhankelijkheid kan naast een lineair karakter ook een niet lineair karakter bezitten. De verschillende mogelijkheden zullen in het kort worden besproken.

### 5.2.1 Tijdgestuurde bronnen



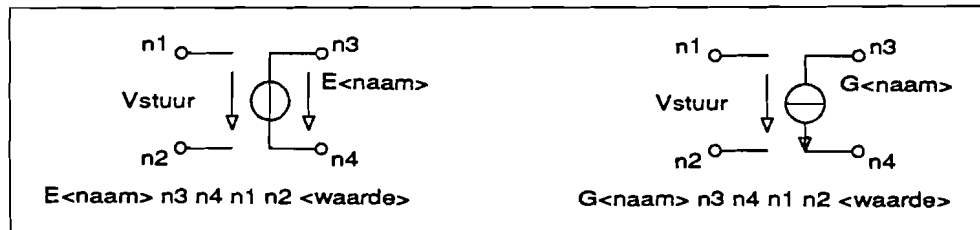
Figuur 5.1: Model van tijdgestuurde spannings- en stroombron.

De meest bekende bronnen binnen PSPICE zijn de tijdgestuurde bronnen voor spanning en stroom (figuur 5.1). Aan deze bronnen wordt een functievoorschrift toegekend met de tijd als enige variabele:  $i(t)$  en  $v(t)$ . Hierbij zijn een aantal functievoorschriften mogelijk:

<b>DC</b>	Een gelijkspanningsbron;
<b>PULSE</b>	Een pulsvormige bron, eventueel periodiek, met instelbare stijgtijden, amplituden en periode;
<b>SIN</b>	Een sinusbron met instelbare amplitude, frequentie, DC niveau en eventuele demping;
<b>EXP</b>	Een enkelvoudig exponentieel signaal;
<b>PWL</b>	Een PieceWise Linear signaal, waarbij de functie is vastgelegd door het geven van een aantal coördinaten. Tussen twee coördinaten wordt lineair geïnterpoleerd;
<b>VALUE</b>	Een vrij te definiëren functie met tijdafhankelijkheid;

Binnen PSPICE wordt een tijdafhankelijke spanningsbron aangeduid met **V**, evenzo een stroombron met **I**.

## 5.2.2 Spanningsgestuurde bronnen



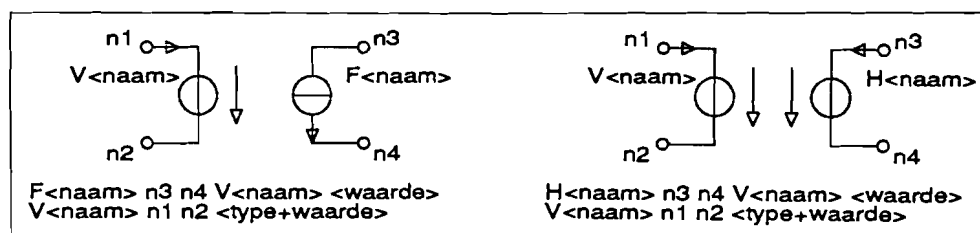
Figuur 5.2: Model van spanningsgestuurde spannings- en stroombron.

PSPICE kent een tweetal spanningsgestuurde bronnen, die worden aangeduid met de letters **G** en **E** (figuur 5.2).

**G** is een spanningsgestuurde stroombron waarbij de stroom bepaald wordt door  $I_{G-bron} = \sigma \cdot V_{stuur}$ . De stroom is lineair evenredig met de sturende spanning. De transgeleiding, uitgedrukt in Siemens, wordt met  $\sigma$  opgegeven.

**E** is een spanningsgestuurde spanningsbron waarbij de spanning van de bron bepaald wordt door  $V_{E-bron} = k_V \cdot V_{stuur}$ . De spanning van de bron is lineair evenredig met de sturende bron. Met factor  $k_V$  kan een versterking worden geïntroduceerd.

## 5.2.3 Stroomgestuurde bronnen



Figuur 5.3: Model van stroomgestuurde spannings- en stroombron.

Naast de spanningsgestuurde bronnen bestaan er binnen PSPICE ook een tweetal stroomgestuurde bronnen, aangeduid met **F** en **H**.

**F** is een stroomgestuurde stroombron waarbij de uitgangsstroom bepaald wordt door de stroom die vloeit door een (tijdafhankelijke) spanningsbron elders in het model. In formulevorm:  $I_{F-bron} = k_I \cdot I_{V-bron}$ . Hierbij kan V-bron een werkelijk aanwezige bron zijn of een fictieve bron, gedefinieerd in de tak waarin de stroom gemeten dient te worden, met een waarde van 0V. De bron met spanning nul werkt dan als een kortsluiting en beïnvloedt de schakeling verder niet (nulspanningsbron). De versterking wordt bepaald door de dimensieloze factor  $k_I$ .



**H** is een stroomgestuurde spanningsbron waarbij de uitgangsspanning bepaald wordt door de stroom die vloeit door een (tijdafhankelijke) spanningsbron elders in het model. In formulevorm:  $I_{H-bron} = \rho \cdot I_{V-bron}$ . Ook nu mag V-bron, evenals bij de F-bron een nulspanningsbron zijn. Factor  $\rho$  bepaalt de transweerstand.

### 5.2.4 Niet lineaire bronnen: Polynoom

De hiervoor besproken bronnen **E**, **F**, **G** en **H** hebben allemaal een lineair karakter:

$$\text{Signaal}_{\text{uit}} = \text{constante} \cdot \text{Signaal}_{\text{in}}$$

Het is binnen PSPICE echter ook mogelijk om de bronnen om te bouwen tot niet-lineaire bronnen. Hierbij worden één of meerdere signaal-bronnen gecombineerd tot een enkele uitgangbron volgens een polynoom-reeks. De coëfficiënten van deze polynoom zijn vrij kiesbaar.

In het geval van één enkele sturende spannings  $V_1$  bij een **E**-bron kan voor de uitgangsspanning de volgende polynoom worden opgesteld:

$$f(V_1) = p_0 + p_1 V_1 + p_2 V_1^2 + p_3 V_1^3 + p_4 V_1^4 + \dots$$

In PSPICE code wordt dit:

```
EBron 1 0 p0 p1 p2 p3 ...
```

Voor  $p_0, p_1, p_2, \dots$  moet een numerieke waarde worden ingevuld. Niet genoemde parameters krijgen de waarde 0.

De uitgangsspanning kan ook bepaald worden door een polynoom-functie van meerdere sturende bronnen, bijvoorbeeld  $V_1$  en  $V_2$ . Door het gebruik van twee ingangsbronnen wordt de polynoom als volgt:

$$\begin{aligned} f(V_1, V_2) = & p_0 + \\ & + p_1 V_1 + p_2 V_2 + \\ & + p_3 V_1^2 + p_4 V_1 V_2 + p_5 V_2^2 + \\ & + p_6 V_1^3 + p_7 V_1^2 V_2 + p_8 V_1 V_2^2 + p_9 V_2^3 + \\ & + p_{10} V_1^4 \dots \end{aligned}$$

Hierbij zijn de constante coëfficiënten  $p_0, p_1, \dots$  zelf te bepalen waardoor heel eenvoudig een niet lineair effect wordt gemaakt.

Neem als voorbeeld het opwekken van een amplitude gemoduleerd (AM)-signaal, gevormd door het vermenigvuldigen van twee sinusvormige signalen (draaggolf en signaal). Het PSPICE model kan er als volgt uitzien:

```
VDraaggolf 1 0 SIN(0V 1V 500kHz)
Vsignaal 2 0 SIN(0V 0.1V 3kHz)
EAMsignaal 3 0 POLY(2) (1,0) (2,0) 0 0 0 0 100
```

Het AM-signaal, opgewekt met de spanningsgestuurde spanningsbron **E** heeft de volgende wiskundige uitdrukking (coëfficiënten  $p_5, p_6, \dots$  zijn 0):

$$\begin{aligned} E\_AM\text{signaal} &= 0 + 0 \cdot V_1 + 0 \cdot V_2 + 0 \cdot V_1^2 + 100 \cdot V_1 V_2 \\ &= 100 \cdot V_{\text{draaggolf}} \cdot V_{\text{signaal}} \end{aligned}$$

### 5.2.5 Niet lineaire bronnen: Tabel

In PSPICE kan bij de **E**-bron en **G**-bron gebruik worden gemaakt van een tabelfunctie. In deze tabel staat voor iedere mogelijke waarde van sturende spanning een bijbehorende uitgangsspanning (**E**) of stroom (**G**). Een voorbeeld van zo'n tabelfunctie volgt. De spanningsbron is geschakeld tussen knooppunten 3 en 4:

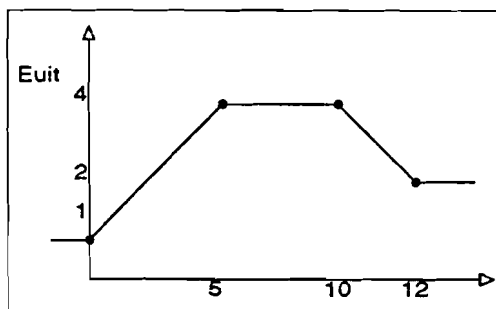
```
Euit 3 4 TABLE { V(2,1) } = 0 1 5 4 10 4 12 2
```

Door gebruik te maken van '\*' en '+' kan de PSPICE invoerregel worden verduidelijkt:

```
Euit 3 4 TABLE { V(2,1) } =
*   Vstuur  Vuit
+     0     1
+     5     4
+    10     4
+    12     2
```

- een '\*' op de meest linkse positie van een regel geeft aan, dat het een commentaarregel betreft. Deze regel wordt door PSPICE overgeslagen en maakt het mogelijk om teksten te gebruiken;
- een '+' op de meest linkse positie van een regel geeft aan, dat de regel nog bij de vorige regel hoort. PSPICE zal deze regel achter de vorige regel plakken, alvorens uit te voeren.

In dit voorbeeld is de sturende spanning  $V(2,1)$ : de spanning tussen knooppunten 2 en 1. Tijdens simulatie zal PSPICE in de kolom 'Vstuur' zoeken naar de plaats waar  $Vstuur = V(2,1)$ . Hierbij wordt tussen de opgegeven coördinaatpunten gebruik gemaakt van lineaire interpolatie. De bronspanning Euit zal een waarde aannemen zoals in kolom Vuit is opgegeven. Ook hier wordt, indien nodig, interpolatie toegepast.



Figuur 5.4: Het spanningsgedrag van de spanningsbron Euit.

In figuur 5.4 is het gedrag van de Ebron getekend. Hierin is duidelijk het karakter van lineair interpoleren te zien. Wanneer de stuurspanning beneden de laagste (0V) of hoogste (12V) waarde uit de tabel komt, dan blijft de uitgangsspanning op het niveau van de bijbehorende laagste respectievelijk hoogste uitgangsspanning steken.

Het zal duidelijk zijn, dat de tabel-functie vele mogelijkheden biedt. Op deze manier kunnen ook signalen, die niet met polynomen kunnen worden beschreven, toch worden gemodelleerd. In het model voor het bepalen van het thermisch gedrag van een draad in een patroon, zal de tabelfunctie veelvuldig worden gebruikt voor het beschrijven van de materiaalkarakteristieken (thermische geleidbaarheid, soortelijke warmte, etc.).

## 5.3 Blackboxes in PSPICE

Een feature van PSPICE die bij de modellering van de miniatuurpatroon goed bruikbaar is, is het gebruik van blackboxes.

Een blackbox kan het best worden gezien als een doos met daarin een nog nader te definiëren elektrisch netwerk waarbij gewenste signalen in of uit het netwerk kunnen worden gebracht (de aansluitklemmen van de blackbox). De blackbox gedraagt zich als een nieuwe component, met een aantal aansluitingen.

Een gedefinieerde blackbox kan worden gebruikt in een ander netwerk door het simpelweg aanbrengen van verbindingen met de aansluitklemmen. Het is ook mogelijk om eenzelfde blackbox meerdere malen in de schakeling op te nemen.

Een goed voorbeeld hiervan in het PSPICE model is de blackbox 'Segment', die aan elkaar wordt geschakeld en op die manier de smeltdraad modelleert.

Omdat blackboxes in blackboxes kunnen voorkomen, is het netwerk van blackbox 'Segment' weer opgebouwd uit andere blackboxes, etc. Zie paragraaf 2 voor een uitvoerige beschrijving van 'Segment'.

### 5.3.1 Blackbox syntax

Om het gebruik van een blackbox in PSPICE te demonstreren wordt een blackbox met naam 'parallel' gedefinieerd. De blackbox bestaat uit een parallelschakeling van twee weerstanden R1 (100 $\Omega$ ) en R2 (50 $\Omega$ ), voorgesteld als een serieschakeling van twee gelijke weerstanden R2a en R2b.

De PSPICE-code is als volgt:

```
(1) .SUBCKT parallel 1 2
(2)  R1  1  2  100
(3)  R2a 1  3   25
(4)  R2b 3  2   25
(5) .ENDS parallel
```

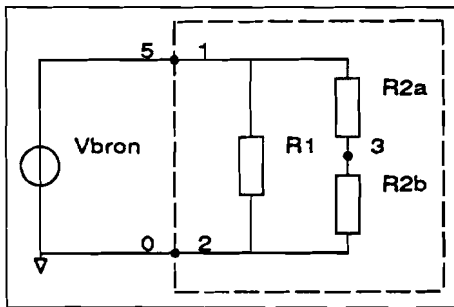
**regel 1** Kopregel van de blackbox definitie. Achtereenvolgens komen hier de PSPICE opdracht .SUBCKT, de naam van de blackbox en de nummers van de aansluitklemmen van het subcircuit te staan;

**regel 2 - 4** Definitie van het netwerk van de blackbox. Hierin kunnen naar willekeur knooppuntnummers worden gebruikt. Indien in dit geval nummer 1 of 2 wordt gebruikt, dan betekent dit koppelen aan de aansluitklemmen. Nummers, anders dan 1 of 2, zijn lokale knooppuntnummers en bestaan alleen binnen de definitie van deze blackbox (bijvoorbeeld nummer 3).

**regel 5** Voetregel van de blackbox definitie. Achtereenvolgens komen hier de PSPICE opdracht .ENDS en de naam van de blackbox.

Door het gebruik van de PSPICE-component **X** kan een gedefinieerde blackbox in de schakeling worden opgenomen. Het aansluiten van de eerder gedefinieerde blackbox 'parallel' op een bron Vbron gaat bijvoorbeeld als volgt:

- ```
(1) Vbron      1 0 5V
(2) Xweerstand 1 0 parallel
```



Figuur 5.5: Koppeling van knooppunt aan aansluitklem bij gebruik van blackboxes.

**regel 1** Schakelt een spanningsbron met DC spanning 5V tussen knooppunten 1 en 0 (aarde);

**regel 2** Schakelt de blackbox 'parallel' tussen knooppunten 1 en 0 van het netwerk. Dit betekent dat aansluitklem 1 van de blackbox wordt aangesloten op knooppunt 1 van het netwerk. Evenzo wordt aansluitklem 2 aangesloten op knooppunt 0 van het netwerk (zie ook figuur 5.5).

Omdat in het netwerk de blackbox 'parallel' meerdere malen kan worden gebruikt, is het noodzakelijk om bij ieder gebruik een unieke naam te geven. Deze naam moet worden ingevuld na de 'X'. De blackbox is in het netwerk hernoemd tot 'weerstand'. Bij nogmaals gebruik van 'parallel' zal men een andere naam dan 'weerstand' moeten gebruiken.

### 5.3.2 Blackboxes en parameters

Het is in PSPICE mogelijk om in een blackbox definitie ook parameters op te nemen. Deze parameters worden verwerkt in de componenten in de blackbox. Bij gebruik van de blackbox wordt voor de parameter een waarde opgegeven. Op die manier kan het gedrag ervan nog worden beïnvloed.

Als voorbeeld de blackbox parallel. De PSPICE listing wordt nu:

```
(1) .SUBCKT parallel 1 2 PARAMS: res1=100 res2=25
(2)  R1  1  2  { res1 }
(3)  R2a 1  3  { res2 }
(4)  R2b 3  2  { res2 }
(5) .ENDS parallel
```

In de kopregel van de definitie volgt na de standaard PSPICE aanhef het woord **PARAMS:**. Van iedere te gebruiken parameter wordt een standaard waarde gedefinieerd. Wordt van een parameter bij gebruik van de blackbox geen waarde opgegeven, dan geldt de standaard waarde. In de definitie van het blackbox netwerk (regel 2 - 4) worden de parameters gebruikt in een formule omgeving (zie paragraaf 5.4 voor beschrijving van het gebruik van parameters en formules in PSPICE).

In het gebruik van de blackbox verandert weinig:

```
(1) Vbron      1 0 5V
(2) Xweerstand 1 0 parallel PARAMS: res1=20 res2=60
```

Bij de aanroep worden de gewenste waarden van res1 en res2 doorgegeven. Omdat beide parameters expliciet worden genoemd, zullen de standaard waarden hier niet gelden.

## 5.4 Constanten en formules in PSPICE

In de invoer van PSPICE kan gebruik gemaakt worden van constanten. Hierbij wordt een getalwaarde toegekend aan een variabelenaam. Het gebruik van deze naam in een formule elders in de invoer betekent voor PSPICE het invullen van de bijbehorende waarde.

Het toekennen van een benadere waarde van  $\pi$  aan de variabele pi gaat met

```
.PARAM pi = 3.14159265
```

De gedefinieerde constanten kunnen in een formule worden gebruikt bij de definitie van de spannings- en stroombronnen in combinatie met VALUE =. Hierbij wordt de formule tussen accolade-open '{' en accolade-sluiten '}' geplaatst. Een voorbeeld:

```
Esignaal VALUE = { 2 / pi * SIN( V(2,1) ) }
```

Het voorbeeld toont dat in de formulering gebruik gemaakt kan worden van getallen en operatoren als optellen, aftrekken, vermenigvuldigen, delen, worteltrekken (SQRT), kwadrateren (PWR) en goniometrische operatoren (SIN, COS, TAN).

Naast spanningen als V(2,1) mogen ook stromen door V-bronnen worden gebruikt in formules. Als voorbeeld: I(Vsensor).

## 5.5 Modellenbouw

In deze paragraaf wordt een tweetal modellen als blackbox ontworpen, welke ook gebruikt worden bij het opstellen van het simulatiemodel. De werking van de modellen wordt in detail besproken.

Achtereenvolgens worden behandeld:

**R(V)** Een model van een spanningsafhankelijke weerstand. Omdat binnen PSPICE geen model van een spanningsafhankelijke weerstand bestaat, wordt met een klein netwerk het gedrag van een dergelijke weerstand nagebootst;

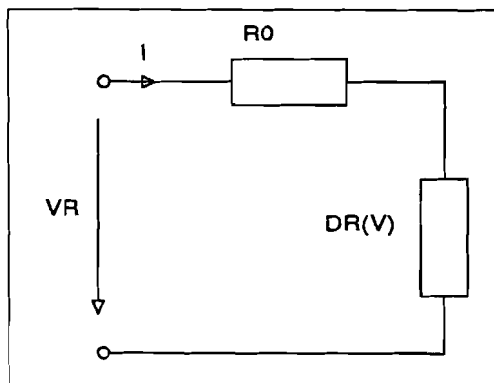
**C(V)** Een model voor een spanningsafhankelijke capaciteit. Hoewel binnen PSPICE de spanningsafhankelijkheid van een capaciteit met een eerste en tweede orde term kan worden ingegeven, is dit voor deze modellering onvoldoende en is een uitgebreider model opgesteld.

### 5.5.1 Modelling $R(V)$

Voor het PSPICE model uit hoofdstuk 4 is een weerstandmodel nodig, waarin weerstand afhankelijk is van spanning (temperatuur in het thermisch model is spanning in het elektrisch analogon). Een verandering in spanning over de weerstand leidt direct tot een verandering in waarde van weerstand.

Bij de modellering zal gebruik gemaakt worden van de spanningsgestuurde spanningsbron (**E**-bron) en de tabelfunctie, zoals in paragraaf 5.2.5 is besproken.

#### Principe



Figuur 5.6: Het netwerkmodel van de spanningafhankelijke weerstand.

De spanningsafhankelijke weerstand wordt voorgesteld als een serieschakeling van twee weerstanden  $R_0$  en  $\Delta R(V)$ , zoals aangegeven in figuur 5.6:

- Weerstand  $R_0$  heeft een vaste waarde, gebaseerd op de weerstandswaarde bij  $V_R = 0V$ ;
- Weerstand  $\Delta R(V)$  is het spanningsafhankelijke deel en heeft een waarde gelijk aan  $\Delta R(V) = R(V) - R_0$ .

Omdat in PSPICE geen spanningsafhankelijke weerstand bestaat, zal  $\Delta R(V)$  alsnog moeten worden omschreven met beschikbare PSPICE componenten.

Dit gebeurt met een spanningsgestuurde spanningsbron (**E**). Het principe is als volgt:

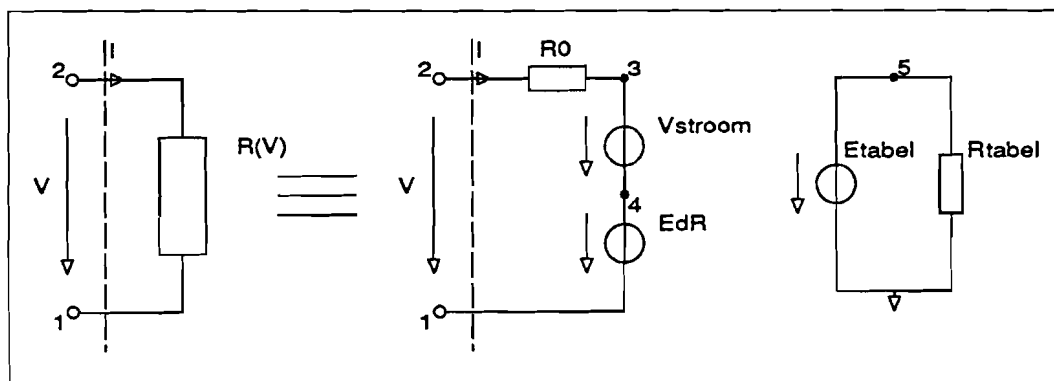
Gesteld dat voor weerstand  $\Delta R(V)$  een waarde  $\Delta R$  gewenst is en er vloeit op dat moment een stroom  $I$  doorheen, dan moet de **E**-bron dus een spanning  $E = dR \cdot I$  opdrukken.

De waarde van  $\Delta R$  is gerelateerd aan spanning  $V_R$  over de weerstand en is van verschillende externe factoren afhankelijk (materiaaleigenschappen, afmetingen, etc.). Door gebruik te maken van de tabelfunctie in PSPICE kan de afhankelijkheid op een eenvoudige manier worden omschreven.

#### PSPICE model

Figuur 5.7 toont het PSPICE model voor de realisering van  $R(V)$ . Hierbij is  $R(V)$  geplaatst tussen de klemmen 2 en 1. Voor de modellering is een drietal extra knooppunten nodig. Vanwege het 'Blackbox karakter' van het model is de nummering van deze knooppunten niet relevant. In dit geval is doorgenummerd met knooppunten 3,4 en 5.

Het bijbehorende stuk PSPICE code is als volgt:



Figuur 5.7: PSPICE netwerkmodel voor een spanningsafhankelijke weerstand.

```

(1) .SUBCKT RV 2 1
(2)  R0      2  3  { R_0 }
(3)  Vstroom 3  4  0V
(4)  EdR     4  1  VALUE { V(5) * I(Vstroom) }

(5)  Etabel  5  0  TABLE { V(2,1) } =
(6)  *  V(2,1)  dR
(7)  +      0    1
(8)  +      5    4
(9)  +     10    4
(10) +     12    2
(11) Rtabel  5  0  1MEG
(12) .ENDS RV

```

**regel 1 en regel 12** Definiëren een blackbox (subcircuit) RV met twee aansluitklemmen, die binnen de blackbox genummerd zijn met 1 en 2. Dit zijn dit de aansluitklemmen van  $R(V)$ . Zie paragraaf 5.3 voor een beschrijving van subcircuits in PSPICE;

**regel 2** Er wordt een vaste weerstand  $R_0$  gedefinieerd tussen knooppunten 2 en 3 van het model. De waarde ervan wordt gegeven door de constante  $R_0$  die elders in PSPICE is gedefinieerd. Zie paragraaf 5.4 voor het gebruik van constanten in PSPICE;

**regel 3** Voor het meten van de stroom door weerstand  $R_0$  wordt een bron met spanning  $V = 0V$  gebruikt, die in serie is geschakeld. Deze nul-spanningsbron beïnvloedt de schakeling niet en maakt het mogelijk om in de definitie van  $EdR$  (regel 4) de stroom  $I(Vstroom)$  te gebruiken. Het op deze manier gebruiken van een nul-spanningsbron wordt binnen PSPICE veelvuldig toegepast. De uitdrukking  $I(R_0)$  is niet toegestaan in een formule omgeving;

**regel 4** De spanningsgestuurde spanningsbron  $EdR$  vormt het hart van het model. Door het gebruik van `VALUE` wordt het mogelijk om een formule te hanteren. In deze formule mogen naast constanten ook spanningen op knooppunten en stromen door  $V$ -bronnen worden opgegeven.

De afleiding van de formule is als volgt:

$$\begin{aligned} E_{dR} &= V(5) \cdot I(Vstroom) \\ &= E_{tabel} \cdot I(Vstroom) \end{aligned}$$

De inwendige weerstand van  $E_{dR}$  volgt uit:

$$R_{E_{dR}} = \frac{E_{dR}}{I(Vstroom)} = E_{tabel}$$

Hieruit volgt dat  $E_{tabel}$  de eenheid moet hebben van weerstand. Met andere woorden: In de tabel kan in de tweede kolom de gewenste verandering in weerstand worden gezet, bij de spanning over  $R(V)$  (de eerste kolom);

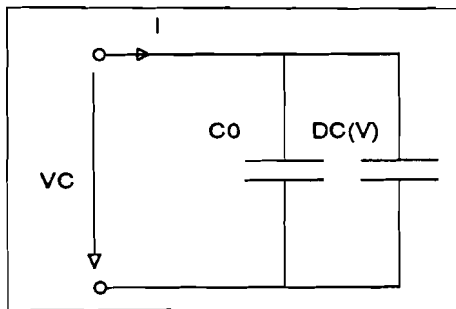
**regel 5 - regel 10** De **E**-bron met de tabelfunctie voor  $dR$  is tussen de knooppunten 5 en 0 geplaatst;

**regel 11** Omdat knooppunt 5 niet mag zweven moet een tweede component worden aangesloten. Er is gekozen voor een weerstand. De weerstandswaarde is niet relevant, daar deze de signaalspanning op knooppunt 5 niet beïnvloedt.

### 5.5.2 Modelling $C(V)$

Voor het modelleren van de spanningsafhankelijke capaciteit wordt gebruik gemaakt van een klein netwerk. Evenals bij  $R(V)$  wordt ook nu gebruik gemaakt van gestuurde bronnen.

#### Principe



Figuur 5.8: Netwerkmodel voor de spanningsafhankelijke capaciteit.

De spanningsafhankelijke capaciteit wordt voorgesteld als een parallelschakeling van twee capaciteiten  $C_0$  en  $\Delta C(V)$ , zoals getekend in figuur 5.8:

- Capaciteit  $C_0$  heeft een vaste waarde, gebaseerd op de weerstandswaarde bij  $V_c = 0V$ ;
- Capaciteit  $\Delta C(V)$  is het spanningsafhankelijke deel en heeft een waarde gelijk aan  $\Delta C(V) = C(V) - C_0$ .

Capaciteit  $\Delta C(V)$  moet alsnog worden omschreven met beschikbare PSPICE componenten.

Dit gebeurt door het introduceren van een spanningsgestuurde stroombron ( $G$ ). Het principe hierbij is als volgt:



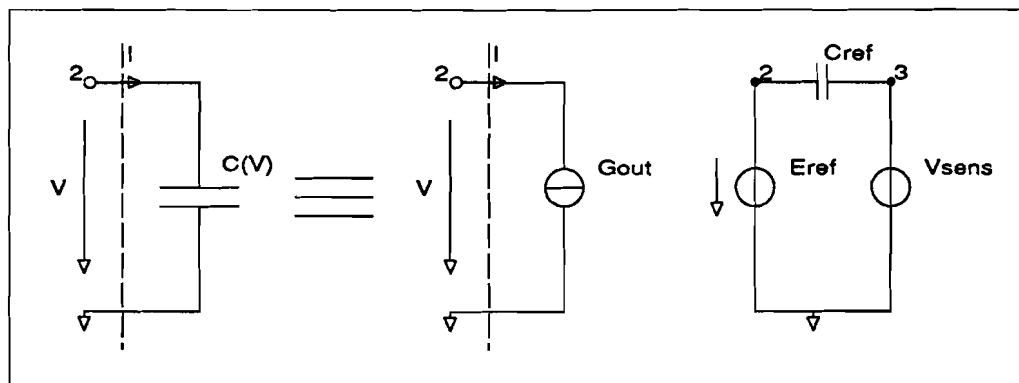
Gesteld dat voor capaciteit  $\Delta C(V)$  een waarde  $\Delta C$  gewenst is en dat de spanning  $V_C$  over  $C(V)$  een gradient  $\frac{dV}{dt}$  heeft, dan volgt voor de stroom door  $\Delta C(V)$ :

$$i_{\Delta C(V)} = \Delta C \cdot \frac{dV}{dt}$$

De stroom wordt opgedrukt met de G-bron.

De waarde van  $\Delta C$  zal afhangen van verschillende externe factoren (materiaaleigenschappen, afmetingen, etc.). Door gebruik te maken van de tabelfunctie kan deze afhankelijkheid eenvoudig worden omschreven.

### PSPICE model



Figuur 5.9: PSPICE netwerkmodel voor een spanningsafhankelijke capaciteit.

Figuur 5.9 toont het PSPICE model voor de realisering van  $C(V)$ . Hierbij is  $C(V)$  geplaatst tussen de klemmen 1 en 0 van de blackbox. Omdat 0 het gemeenschappelijk aardpunt van de schakeling is en daarom ook buiten de blackbox direct beschikbaar is, is deze niet naar buiten uitgevoerd. In de blackbox zijn nog een aantal extra (interne) knooppunten nodig (2,3,4).

Het bijbehorend stuk PSPICE code is als volgt:

```
(1) .SUBCKT CV 1
(2) Eref 2 0 VALUE = { V(1,0) }
(3) Cref 2 3 1
(4) Vsens 3 0 0V
(5) C0 1 0 { C_0 }
(6) Gout 1 0 VALUE = { I(Vsens) * V(4)*1E-6 }
(7) Etable 4 0 TABLE { V(1,0) } =
(8) * V(1,0) dC [uF]
(9) + 0 0
(10) + 5 10
(11) + 7 15
(12) + 12 20
(13) Rtable 4 0 1MEG
(14) .ENDS CV
```

**regel 1 en regel 14** Definiëren een blackbox (subcircuit) CV met een enkele aansluitklem, die binnen de blackbox genummerd is met 1. Zie paragraaf 5.3 voor een beschrijving van subcircuits in PSPICE;

**regel 2 - regel 4** Definitie van een netwerk waarin met een referentie-condensator van 1F het spanningssignaal over condensator C(V) wordt gedifferentieerd. De optredende stroom in het circuit wordt gebruikt bij de definitie van de G-bron in regel 6. Er geldt:

$$i_{Cref} = i_{Vsens} = Cref \cdot \frac{dEref}{dt} = 1 \cdot \frac{dV_C}{dt} = \frac{dV_C}{dt}$$

Bron Vsens op regel 4 heeft een spanning 0V en wordt gebruikt voor de bepaling van de stroom door capaciteit Cref;

**regel 5 en regel 6** Definitie van het uitgangscircuit. C0 is de standaard capaciteit bij een spanning  $V_C = 0V$  en is in PSPICE bekend onder de naam C\_0. Parallel daaraan wordt de G-bron geplaatst, die zich gedraagt als  $\Delta C(V)$  en zorgt voor het spanningsafhankelijk karakter van de totale C(V). In formulevorm geldt:

$$i_G = I(Vsens) \cdot (V(4) \cdot 10^{-6}) = \frac{dV_C}{dt} \cdot \Delta C$$

De stroombron gedraagt zich alsof er sprake is van een capaciteit  $\Delta C$ ;

**regel 7 - 12** In deze tabelfunctie wordt de relatie gelegd tussen de spanning  $V_C$  over de capaciteit C(V) en de waarde van benodigde extra capaciteit  $\Delta C$ ;

**regel 13** Omdat knooppunt 4 niet mag zweven moet een tweede component worden aangesloten. Er is gekozen voor een weerstand. De weerstandswaarde is niet relevant, daar deze de signaalspanning op knooppunt 4 niet beïnvloedt.

De besproken blackboxmodellen van R(V) en C(V) zullen bij de PSPICE modellering in hoofdstuk 4 veelvuldig worden gebruikt.

## Deel II. Meetsysteem

### Inleiding

Zoals uit de literatuurstudie, besproken in deel 3, is gebleken, kan niet rechtstreeks de temperatuur van de draad worden gemeten en zal naar alternatieven moeten worden gezocht. Dit deel van het afstudeerverslag beschrijft de diverse aspecten van een alternatief.

In hoofdstuk 1 wordt het pulscircuit beschreven. Met dit circuit wordt een stroompuls opgewekt die door de miniatuurpatroon wordt gestuurd.

Er wordt een theoretische afleiding gegeven van de stroom door de miniatuurpatroon en de bijbehorende waarde van de  $I^2t$ .

In hoofdstuk 2 wordt een meetmethode besproken, waarmee het mogelijk wordt om een indicatie te krijgen van de gemiddelde draadtemperatuur van de smeltpatroon.

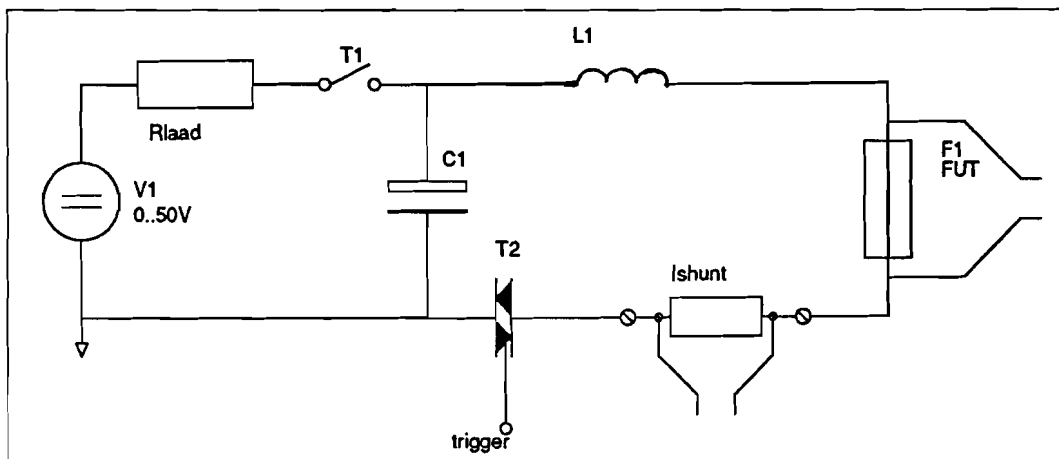
De opbouw van het meetsysteem wordt beschreven en er volgt een foutenbeschouwing.

Hoofdstuk 3 tenslotte beschrijft software die is ontwikkeld voor het omzetten van meetgegevens afkomstig van de registratie apparatuur uit de meetopstelling van hoofdstuk 2. De omzetting maakt verdere verwerking op de PC mogelijk.

Bij het in gebruik nemen van een nieuwe digitizer voor de meting is het noodzakelijk gebleken om deze software te ontwikkelen. De software is universeel van opzet, zodat deze als standaard conversie-software bij de digitizer kan dienen.

## 7. Het pulscircuit

*Dit hoofdstuk beschrijft het elektrisch circuit voor het opwekken van de stroompuls door de smeltpatroon.*



Figuur 7.1: Het pulscircuit voor het opwekken van de teststroom.

Het circuit zoals getekend in figuur 7.1 bestaat uit een seriekring van C, L en smeltpatroon weerstand (R) die door een halfgeleider-schakelaar T2 kan worden gesloten. Dit circuit wordt gebruikt voor het opwekken van een pulsvormige teststroom, die door de smeltpatroon wordt geleid.

Het pulsgedrag wordt bepaald door de componentwaarden van L, C en R en de beginspanning over de condensator. Een en ander zal worden afgeleid in paragraaf 7.1.

In het kort zal de werking van het circuit worden uitgelegd.

Het testcircuit bestaat uit twee delen:

- Het laadcircuit voor het opladen van de condensator. Door het sluiten van schakelaar T1 (T2 is open) wordt condensator C1 via de laadweerstand R1 opgeladen tot een spanning nagenoeg gelijk aan gelijkspanningsbron V1. De waarde van V1 is instelbaar. Na opladen van de condensator tot de eindspanning (na ongeveer  $5 \cdot R_1 C$  seconden) wordt schakelaar T1 weer geopend.
- Het pulscircuit voor het genereren van de stroompuls. Wanneer na het openen van T1 thyristorschakelaar T2 op de gate wordt aangestuurd komt deze in geleiding en daarmee wordt de LCR-kring gesloten. De condensator met beginspanning V1 ontladtd zich over de serieschakeling van spoel L1 en de smeltpatroon met weerstand R.

De componenten L en C hebben een vaste waarde. De R wordt bepaald door de weerstand van de smeltpatroon (vermeerderd met extra weerstand van het circuit, capaciteit en spoel).

## 7.1 Theoretische beschouwing

Bij smeltpatronen speelt de grootte  $I^2t$  een belangrijke rol. Ieder type smeltpatroon is ontworpen voor een maximaal toelaatbare waarde van  $I^2t$ . Kleinere waarden van  $I^2t$  leveren een minder grote belasting voor de patroon op. De  $I^2t$  is gedefinieerd als:

$$I^2t = \int_0^t i^2(t) dt$$

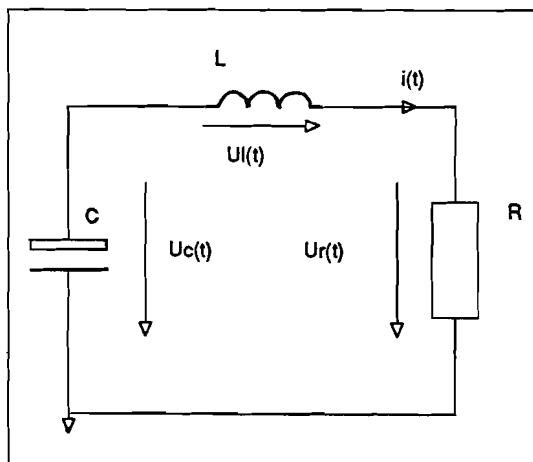
Merk hierbij op dat bij aanname van constante weerstand van de patroon, de  $I^2t$  gebaseerd is op de (vermogens)-inhoud van het signaal en niet afhangt van de vorm ( $di/dt$ ). Omdat de  $I^2t$  dus niet afhangt van de vorm van het signaal, is het eenvoudiger om de waarde met een of ander pulscircuit te reproduceren. De manier waarop de puls wordt gemaakt is van minder groot belang.

Omdat bij het pulscircuit de laadspanning kan worden gevarieerd, kan een relatie worden afgeleid tussen de laadspanning van de condensator ( $U_o$ ) en de waarde van  $I^2t$ . Op die manier kan bij iedere gewenste waarde van  $I^2t$  de bijbehorende laadspanning worden gevonden.

### 7.1.1 Stroom $i(t)$

Allereerst wordt een wiskundige uitdrukking gezocht voor het verloop van de stroom, als zou er sprake zijn van een eenmalig sluiten van een ideale schakelaar T2. De invloed van een thyristor als schakelaar wordt later nog besproken.

Daar er sprake is van een L en C (met voorspanning) ontstaat een slinger-circuit, dat door weerstand R wordt gedempt.



Figuur 7.2: Model LCR-circuit met daarin componentwaarden zoals deze in het pulscircuit zijn gebruikt.

Voor het model uit figuur 7.2 geldt de volgende vergelijking:

$$U_C(t) = U_L(t) + U_R(t) \quad (7.1)$$

Met de randvoorwaarde:  $U_C(0) = U_o$

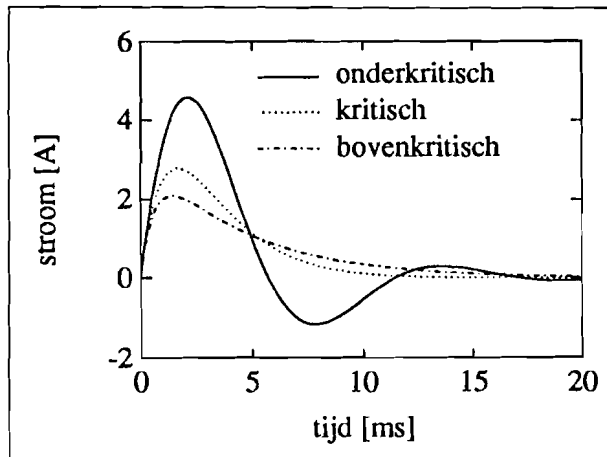
Door hanteren van de componentvergelijkingen voor spanning en stroom met de bijbehorende tekenafspraken, volgt uit formule 7.1:

$$\begin{aligned}
 U_C(t) &= L \frac{di(t)}{dt} + Ri(t) \Leftrightarrow \\
 \frac{dU_C(t)}{dt} &= L \frac{d^2i(t)}{dt^2} + R \frac{di(t)}{dt} \Leftrightarrow \\
 C \frac{dU_C(t)}{dt} &= -i_C(t) = -i(t) = LC \frac{d^2i(t)}{dt^2} + RC \frac{di(t)}{dt}
 \end{aligned}$$

zodat voor de stroom de volgende tweede-orde differentiaal-vergelijking ontstaat:

$$LC \frac{d^2i(t)}{dt^2} + RC \frac{di(t)}{dt} + i(t) = 0$$

Afhankelijk van de waarde van de weerstandcomponent R in het circuit treedt er een bepaalde vorm van demping op. Hierbij kunnen een viertal verschillende vormen worden onderscheiden (zie figuur 7.3):



Figuur 7.3: Verloop van  $i(t)$  voor verschillende waarden van circuitweerstand R.

1. Er is geen sprake van demping indien  $R = 0$ . Het circuit is in trilling met hoeksnelheid  $\omega = \frac{1}{\sqrt{LC}}$ ;
2. Er is sprake van kritische demping indien  $R = R_k = 2\sqrt{\frac{L}{C}}$ ;
3. De kring is onderkritisch gedempt indien  $R < R_k$ ;
4. Wanneer  $R > R_k$  is de kring overkritisch gedempt.

Het oplossen van deze DV gaat volgens de regels der wiskunde en levert uiteindelijk de volgende uitdrukking voor de stroom (beschouw  $R \leq 2\sqrt{\frac{L}{C}}$ ):

$$i(t) = e^{-\frac{R}{2L}t} [E \cos(\omega t) + F \sin(\omega t)] \quad \text{met:} \quad \omega = \sqrt{\frac{1}{LC} - \left(\frac{R}{2L}\right)^2}$$

Na toepassen van de geldende randvoorwaarden, volgen de constanten E en F. Er geldt:

- Door de eigenschap dat een stroom door een spoel een continu verloop moet hebben, zal na het sluiten van de schakelaar de stroom in eerste instantie nul blijven, dus  $i(0) = 0$ . Dit levert  $E = 0$ .
- Omdat  $i(0) = 0$  volgt dat  $U_R(0) = 0$  en uit formule 7.1 volgt dan dat  $U_L(0) = U_C(0)$ , dus  $U_C(0) = U_o = L \frac{di}{dt}$ . Dit levert  $F = \frac{U_o}{\omega L}$ .

Voor de circuitstroom  $i(t)$  volgt:

$$i(t) = \frac{U_o}{\omega L} e^{-\frac{R}{2L}t} \cdot \sin(\omega t) = A e^{Bt} \cdot \sin(\omega t) \quad (7.2)$$

Door differentiëren van vergelijking 7.2 kan het tijdstip worden berekend waarop de stroom zijn eerste maximum bereikt. Dit levert:

$$t_{max,I} = \frac{1}{\omega} \arctan\left(\frac{2\omega L}{R}\right) \quad (7.3)$$

De eerste nuldoorgang van de stroom volgt uit nulstellen van formule 7.2 en geeft:

$$t_{nul,I} = \frac{\pi}{\omega} \quad (7.4)$$

### 7.1.2 Condensatorspanning $U_c(t)$

Nu de uitdrukking van de stroom bekend is, kan uit  $U_c(t) = -\frac{1}{C} \int_0^t i(t) dt$  het verloop van de condensatorspanning worden bepaald. Er volgt<sup>1</sup>:

$$U_c(t) = \frac{A}{C(B^2 + \omega^2)} \cdot e^{Bt} (\omega \cos(\omega t) - B \sin(\omega t)) + \text{constante} \quad (7.5)$$

Uit de randvoorwaarde  $U_c(0) = U_o$  kan de constante worden bepaald. Na herschikken volgt:

$$\text{constante} = U_o - \frac{A\omega}{C(B^2 + \omega^2)}$$

zodat voor de condensatorspanning volgt:

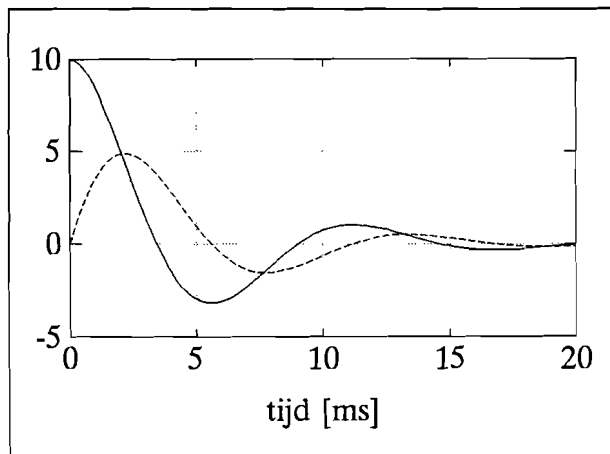
$$U_c(t) = \frac{A}{C(B^2 + \omega^2)} [e^{Bt} (\omega \cos(\omega t) - B \sin(\omega t)) - \omega] + U_o \quad (7.6)$$

$$= \frac{A}{C} \left[ \omega + \frac{e^{Bt}}{\sqrt{B^2 + \omega^2}} \cos\left(\omega t + \arctan\left(\frac{B}{\omega}\right)\right) \right] - U_o \quad (7.7)$$

<sup>1</sup>Maak hierbij gebruik van

$$\int e^{ax} \sin(bx) dx = \frac{e^{ax}}{a^2 + b^2} (a \sin(bx) - b \cos(bx)) = -\frac{e^{ax}}{\sqrt{a^2 + b^2}} \cos\left(bx + \arctan\left(-\frac{a}{b}\right)\right)$$

$$\text{want} \quad a \sin(cx) + b \cos(cx) = \sqrt{a^2 + b^2} \cos\left(cx - \arctan\left(\frac{a}{b}\right)\right)$$



Figuur 7.4: Verloop van stroom  $i(t)$  en spanning  $U_c(t)$  na het sluiten van schakelaar T2.

Een verloop van de stroom en spanning is weergegeven in figuur 7.4. Merk hierbij op dat bij de afleiding is uitgegaan van constante waarde van  $R$ ,  $L$  en  $C$ . In praktijk bevat de  $R$  een in waarde variërend deel van de smeltpatroon, zodat mogelijk het slingergedrag van de kring wordt beïnvloed.

Dit moet per situatie worden bekeken opdat de pulsform niet al te zeer afwijkt.

Tot slot de invloed van thyristor T2 op het slingergedrag van het LRC circuit. Uit formule 7.2 volgt een wisselstroom, die door het diode karakter van de thyristor maar in één richting zal worden geleid. De thyristor zal bij nuldoorgang van de stroom opengaan, waardoor het circuit wordt onderbroken.

In praktijk betekent dit dat alleen gedurende  $0 < t < t_{nul,I}$  een stroom zal vloeien (eerste piek). Er volgt slechts een omslingering, waarbij de spanning op de condensator wordt omgepoold.

**NB:** De condensator uit het pulscircuit is van elektrolytisch type en heeft daarom een polariteit. Voordeel van dit type is de grote capaciteit-volume verhouding, die veel groter is dan normaal.

De polariteit staat het eigenlijk niet toe om een dergelijke capaciteit om te polen, iets wat in dit circuit wel gebeurt. Dit kan beschadiging van het elektrolyt tot gevolg hebben.

Omdat hier slechts sprake is van korte stroompulsen met kleine waarden van  $I^2t$ , zullen niet direct problemen optreden. Dit probleem moet in geval van opschalen van het circuit echter serieus worden beschouwd, omdat het kan leiden tot een defect in de capaciteit en in het ergste geval tot ontploffing ervan.

Het gebruik van bipolaire elektrolyten biedt misschien uitkomst.

Een mogelijke beveiliging kan worden gevonden door het plaatsen van een diode antiparallel over de capaciteit. Deze diode komt in geleiding op het moment dat de spanning over de capaciteit gaat ompolen. De spanning blijft in dat geval ongeveer 0,7V. Hierbij moet men zich realiseren dat vanaf dat moment geen sprake meer is van een LCR-circuit, maar van een LR-circuit, zodat de stroom exponentieel gaat afnemen (de vorm van de puls verandert).

Een en ander is niet verder uitgewerkt.



## 7.2 Berekening $I^2t$

Met de afgeleide formules uit de vorige paragraaf kan nu een theoretische afleiding worden gegeven van de  $I^2t$ -waarde van de stroompuls, als functie van de componentwaarden uit het pulscircuit. Hierbij wordt uitgegaan van constante waarde van C, L en R.

Er geldt:

$$I^2t = \int_0^t i^2(t) dt \quad \text{met} \quad i(t) = Ae^{Bt} \sin(\omega t)$$

$$\text{met} \quad A = \frac{U_o}{\omega L}, \quad B = -\frac{R}{2L} \quad \text{en} \quad \omega = \sqrt{\frac{1}{LC} - \left(\frac{R}{2L}\right)^2}$$

Door de thyristorwerking zal de pulsvormige stroom gedurende  $t_{nul,I}$  seconden aanwezig zijn, zodat voor de bovengrens van de integraal  $t_{nul,I}$  moet worden ingevuld.

Uitwerken van  $I^2t$  volgens de regels der wiskunde geeft<sup>2</sup>:

$$\begin{aligned} I^2t &= \frac{A^2}{2} \int_0^{t_{nul,I}} e^{-2Bt} (1 - \cos(2\omega t)) dt \\ &= -\frac{A^2}{4B} (1 - e^{2Bt_{nul,I}}) - \frac{A^2}{4(B^2 + \omega^2)} [-B + e^{2Bt_{nul,I}} (\omega \sin(2\omega t_{nul,I}) + B \cos(2\omega t_{nul,I}))] \end{aligned}$$

Met  $t_{nul,I} = \frac{\pi}{\omega}$  volgt:

$$I^2t = \frac{A^2}{4B} (e^{2B\frac{\pi}{\omega}} - 1) \left( \frac{\omega^2}{B^2 + \omega^2} \right) \quad (7.8)$$

Uitdrukking 7.8 bevat alleen bekende coëfficiënten en kan worden opgevat als functie van de laadspanning  $U_o$ :

$$I^2t = \text{constante} \cdot U_o^2$$

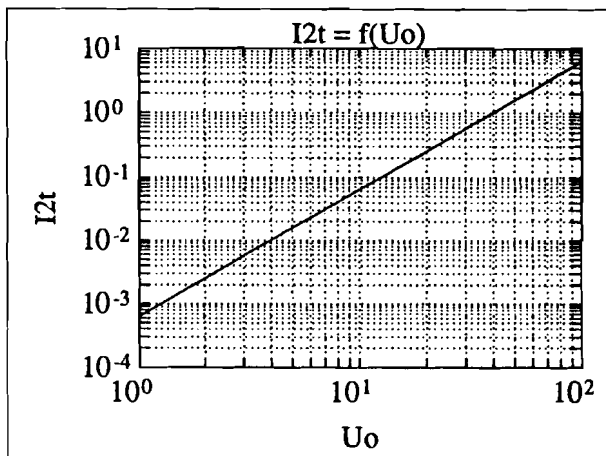
Wanneer deze laatste uitdrukking dubbellogaritmisch in een grafiek wordt weergegeven ontstaat een rechte lijn, waaruit eenvoudig de benodigde  $U_o$  kan worden gevonden bij iedere gewenste  $I^2t$ . Een voorbeeld is gegeven in grafiek 7.5.

In de praktijk kan een dergelijke  $I^2t$  curve worden geconstrueerd uit de meetwaarden van een enkele metingen bij een waarde van  $U_o$ . Hierbij worden L, C en R tijdens de meting constant verondersteld. De constante is dan bekend en de grafiek kan worden getekend.

Het is uiteraard beter om een aantal metingen bij verschillende  $U_o$  te doen ter controle.

<sup>2</sup>met gebruikmaking van

$$\int_0^t e^{ax} \cos(bx) dx = \frac{e^{at} [b \sin(bt) + a \cos(bt)] - a}{a^2 + b^2}$$



Figuur 7.5: Grafiek van  $I^2t$  als functie van  $U_0$ .

## 8. Het meetcircuit

*Dit hoofdstuk beschrijft het meetcircuit dat wordt gebruikt voor het bepalen van de gemiddelde draadtemperatuur. Het meetprincipe wordt besproken en er wordt aangegeven waarop gelet moet worden bij de realisering ervan.*

Om aan het pulscircuit, zoals beschreven in de vorige paragrafen, metingen te verrichten, moet een meetcircuit worden samengesteld. De met het meetsysteem verkregen resultaten kunnen vervolgens worden gebruikt voor verdere analyse.

### 8.1 Het meetprincipe

Het doel van de meting is bekend: *het verkrijgen van inzicht in het verloop van temperatuur van de smeltdraad in een miniatuur glaspatroon bij een pulsformige stroom.*

Materialen, zoals nikkel, bezitten temperatuur afhankelijke eigenschappen. Zo neemt de elektrische weerstand toe bij toenemende temperatuur (bijlage B.1 van deel 3). Met een meting van de draadweerstand kan een uitspraak worden gedaan over de draadtemperatuur.

Hierbij moet worden opgemerkt dat het alleen mogelijk is om te praten over een gemiddelde draadweerstand en dus ook een gemiddelde draadtemperatuur. Er kan alleen worden gemeten aan de uiteinden van de draad (op de caps).

Er bestaan (compacte) meetsystemen voor het nauwkeurig meten van weerstand, maar het toepassen ervan is hier niet mogelijk:

- het meetstysteem wordt volgens een vierpunts methode op de weerstand geschakeld en bij opgedrukte meetstroom wordt de spanning over de weerstand gemeten. Hieruit is de weerstand te bepalen. Tijdens de weerstandsmeting mag geen extra (puls)-stroom door de weerstand vloeien;
- door het koppelen van weerstandsmeting aan het pulscircuit kunnen problemen ontstaan met aarding en ongedefinieerde stroompaden. Dit kan in het ergste geval leiden tot een defect in de meetapparatuur;
- het meetsysteem is bedoeld voor het meten van DC-weerstand en daarom niet in staat om snelle transiënten te volgen.

Hoewel het direct toepassen van een weerstand meetsysteem niet mogelijk is, blijft het principe dat door een dergelijk systeem wordt gebruikt, goed bruikbaar.

Dit leidt tot de volgende meetmethode die zal worden gehanteerd:

Door het meten van de spanning over de smeltdraad en de stroom door de draad tijdens de stroompuls, kan uit het quotiënt de gemiddelde draadweerstand in de tijd worden bepaald. Uit de gemiddelde draadweerstand kan de gemiddelde temperatuur worden bepaald:

$$V, I \rightarrow R_{gem} \rightarrow T_{gem}$$

NB: Het is dus met deze meetmethode niet mogelijk om het temperatuurprofiel van de draad in axiale richting te meten. In hoofdstuk 3 wordt een theoretisch simulatiemodel uitgewerkt waarmee het wel mogelijk is om, uitgaande van de fysische eigenschappen van het materiaal, bij opgegeven stroompuls het temperatuurprofiel te bepalen.

## 8.2 Het circuit

Het meetsysteem moet de volgende eigenschappen bezitten:

- Het meetcircuit mag bij koppeling aan het pulscircuit de werking ervan niet beïnvloeden;
- Het systeem moet beschikken over een tweetal signaalingangen, voor het meten van de spanning en stroom;
- Omdat sprake is van een eenmalig snel verschijnsel (stroompuls) dienen de gemeten signalen te worden opgeslagen. Op een later tijdstip kan uit de signalen omrekening naar weerstand worden gepleegd. Dit noodzaakt het gebruik van een digitale oscilloscoop (digitizer);
- Bij het koppelen van de meetsignalen aan de digitizer is een aanpassing in de vorm van versterking vereist.

Een en ander heeft geleid tot een meetcircuit zoals is afgebeeld in figuur G.1 op bladzijde 132. De figuur toont het pulscircuit waarop het meetcircuit is aangesloten. Er is gebruik gemaakt van een tweetal differentiaalversterkers, gekoppeld aan een digitizer.

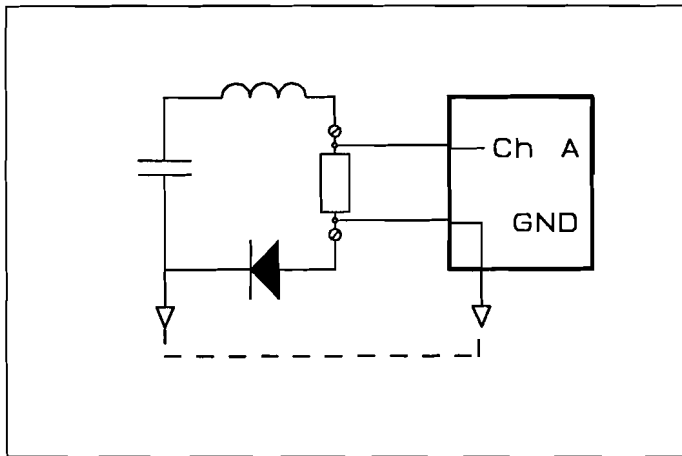
In de volgende paragrafen worden de diverse componenten apart besproken.

### 8.2.1 De differentiaalversterker

Voor het meten van de signaalspanningen voor spanning en stroom is een tweekanaals storage-oscilloscoop (digitizer) voldoende. Het aansluiten ervan vormt echter een probleem. Dit heeft te maken met het introduceren van een extra meetaarde. Een oscilloscoop meet altijd ten opzichte van deze aarde.

Deze aarde is in verband met de veiligheid ook doorverbonden met de randaarde van de oscilloscoopstekker. Via de aarde van het elektriciteitsnet kan deze alsnog verbonden zijn met de aarde van het pulscircuit.

Met het aansluiten van de oscilloscoop meetingang wordt het pulscircuit op een tweede plaats aan aarde gelegd, hetgeen problemen oplevert (sluiting). Deze situatie is in figuur 8.1 getekend. De aardlus zorgt ervoor dat de thyristor en stroomshunt worden kortgesloten, waardoor het circuitgedrag wordt beïnvloed.



Figuur 8.1: Sluiting (aardlus) door direct aansluiten oscilloscoop.

Dit probleem komt veelvuldig voor in de meettechniek en heeft geleid tot het introduceren van het begrip **differentiaalmeting** (verschilmeting), waarbij niet langer wordt gemeten tussen signaal en aarde, maar tussen signaal en een referentie niveau.

Omdat het referentieniveau niet verbonden is met aarde, wordt geen sluiting meer veroorzaakt.

De uitgang van de versterker levert een signaalspanning t.o.v. een referentie (nul), die zondermeer kan worden aangesloten op de nul van de scoop. Deze nul heeft nu geen invloed meer op de meetschakeling.

Door het aanbrengen van een bepaalde versterker combinatie, kan het meetsignaal tot een gewenst uitgangssignaal worden versterkt, zodat ook zwakke meetsignalen betrouwbaar kunnen worden gemeten.

Speciaal voor het differentiaal meten van signalen bestaan in de literatuur verschillende elektrische schakelingen, allen met eenzelfde basisontwerp. Voor professionele toepassingen bestaan complete differentiaal-units met compleet (galvanisch) gescheiden in- en uitgang.

Voor toepassing in dit ontwerp volstaat het versterkerontwerp zoals afgebeeld in figuur F.1 (bladzijde 130). Dit is afkomstig uit de *Product Applications* van Texas Instruments [17], bladzijde 3-5. Het toont één van de twee gebruikte differentiaalversterkers.

Voor een uiteenzetting van de werking, kan de schakeling het best worden opgedeeld in een drietal trappen:

**Input protectie** Door het toepassen van diodes D1...D4, worden signaalspanningen op +IN en -IN boven +15V en onder -15V afgekapt. Met twee hoogohmige weerstanden naar aarde (dit is scoop-aarde) worden +IN en -IN ten opzichte van aarde gebalanceerd. Deze trap versterkt  $\frac{R_{3a}}{R_{3a}+R_{1a}}$  maal.

Hieruit blijkt dat bij dit veelvuldig gebruikt type differentiaalingang nog steeds sprake is van hoogohmige verbinding naar aarde. Het gebruik van weerstanden  $R_{3a}$  en  $R_{3b}$  is noodzakelijk om de ingangen van de OPAMP's gebalanceerd te houden. De praktijk heeft uitgewezen dat de schakeling naar behoren functioneert;

Bij dit ontwerp wordt het signaal van +IN (kern) en -IN (mantel) over een enkele coax kabel geleid.

**Differentiaaltrap** Deze trap bestaat uit een drietal operationele versterkers (OPAMP's). Dit deel van de schakeling vormt de eigenlijke differentiaal versterker en heeft de volgende

overdrachts formule:

$$V_{out} = \left(2 \frac{R_3}{R_a} + 1\right) \cdot \frac{R_5}{R_4} \cdot dV_{in}$$

Bij vaste waarde van  $R_3$ ,  $R_4$  en  $R_5$  is de versterking te regelen met  $R_a$ .

**Offset regeling** In deze laatste trap wordt met een gestabiliseerde spanning de offset spanning in de uitgang beïnvloed. Het instellen ervan gebeurt met potmeter  $R_{10}$ . Door gebruik van een tweetal zenerdioden  $Z_1$  en  $Z_2$  wordt de spanning op het lopercontact van  $R_{10}$  niet beïnvloed door een rimpel op de voedingsspanning.

De uA741 opamp heeft in deze schakeling een spanningsversterking van één en vormt een buffer tussen de potentiometer en de laatste opamp. Omdat de ingangsimpedantie van de 741 zelf erg hoog is, wordt de potentiometer minimaal belast en blijft de spanning constant.

De uitgang heeft een weerstand van  $50\Omega$  in serie. Omdat de uitgangsimpedantie van de opamp erg laag is, is de totale uitgangsimpedantie van de differentiaalversterker gelijk aan  $50\Omega$ .

Deze waarde van weerstand komt overeen met de karakteristieke impedantie van de coax meetkabel, zodat een goede aanpassing bestaat tussen opamp en kabel en teruggekaatste signalen niet terug de versterker inkomen. Voor een juiste afsluiting, moet ook bij de scoop de kabel karakteristiek worden afgesloten met  $50\Omega$ . In dat geval ontstaat een theoretisch perfecte afsluiting.

In de praktijk is gebleken dat het gebruikte type OPAMP TL084 niet in staat is om in een dergelijke situatie goed te functioneren. Door het afsluiten met de  $50\Omega$  wordt de totale belasting van de versterker namelijk slechts  $100\Omega$ , wat erg laag is. Het gevolg is dan ook dat de opamp te veel stroom moet leveren en reeds bij kleine ingangssignalen niet meer functioneert (vastloopt). Wanneer de afsluitweerstand aan de oscilloscoopzijde wordt verwijderd, gedraagt de versterkerschakeling zich normaal en kan goed worden gebruikt. Uiteraard zullen in dit geval voor hoge frequenties in het signaal problemen ontstaan (reflecties in de kabel). Voor het laagfrequent pulssignaal blijkt dit geen problemen op te leveren.

In bijlage G staat aangegeven, hoe de versterker op het pulscircuit wordt aangesloten.

### Dimensionering componenten

Nu de signaaloverdracht van pulscircuit naar digitizer bekend is, kan voor beide kanalen een versterkingsformule worden opgeschreven.

Voor de stroom wordt dit:

$$\begin{aligned} V_{out,I} &= \left(2 \frac{R_3}{R_a} + 1\right) \cdot \frac{R_5}{R_4} \cdot dV_{in} \\ &= \left(2 \frac{R_3}{R_a} + 1\right) \cdot \frac{R_5}{R_4} \cdot \frac{R_2}{R_0 + R_2} \cdot (V_{IN+} - V_{IN-}) \\ &= \left(2 \frac{R_3}{R_a} + 1\right) \cdot \frac{R_5}{R_4} \cdot \frac{R_2}{R_0 + R_2} \cdot V_{shunt} \\ &= \left(2 \frac{R_3}{R_a} + 1\right) \cdot \frac{R_5}{R_4} \cdot \frac{R_2}{R_0 + R_2} \cdot (I_{shunt} \cdot R_{shunt}) \end{aligned}$$

In het pulscircuit wordt een weerstandshunt met constante waarde opgenomen (zie paragraaf 8.2.2).

Voor de spanning wordt dit:

$$\begin{aligned} V_{out,V} &= \left(2 \frac{R_3}{R_a} + 1\right) \cdot \frac{R_5}{R_4} \cdot dV_{in} \\ &= \left(2 \frac{R_3}{R_a} + 1\right) \cdot \frac{R_5}{R_4} \cdot \frac{R_2}{R_0 + R_2} \cdot (V_{IN+} - V_{IN-}) \\ &= \left(2 \frac{R_3}{R_a} + 1\right) \cdot \frac{R_5}{R_4} \cdot \frac{R_2}{R_0 + R_2} \cdot V_{patroon} \end{aligned}$$

Voor het ontwerp zijn een aantal standaard componentwaarden aangenomen, die een goed functioneren van de versterker verzekeren:  $R1a = R1b = R3a = R3b$ ,  $R2a = R2b = R3 = R4 = R5 = 10k\Omega$ .

Dit geeft:

$$V_{out,I} = \left(\frac{10k\Omega}{R_{a,I}} + 1\right) \cdot (I_{shunt} \cdot R_{shunt}) = A_I \cdot I_{shunt} \cdot V_{shunt} \quad (8.1)$$

$$V_{out,V} = \left(\frac{10k\Omega}{R_{a,V}} + 1\right) \cdot V_{patroon} = A_V \cdot V_{patroon} \quad (8.2)$$

Afhankelijk van de gewenste signaalniveau's kunnen  $R_{a,I}$  en  $R_{a,V}$  worden bepaald. Uiteraard dient rekening gehouden te worden met het feit dat dit een theoretische afleiding is. In de praktijk zal de werkelijke versterking per kanaal tijdens het ijken van de versterker moeten worden bepaald.

### IJken van de versterker

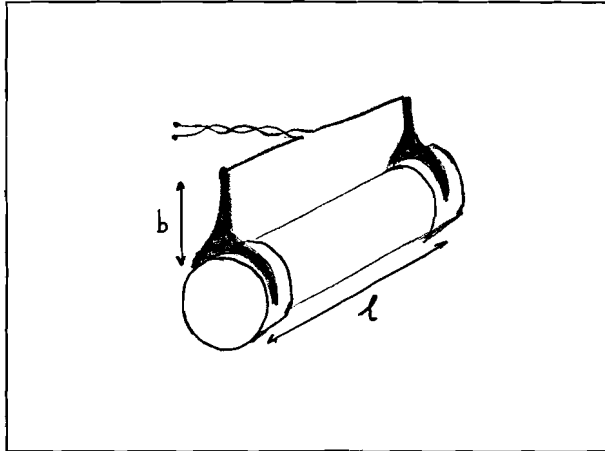
Het ijken van een versterkertrap bestaat minimaal uit het doen van een tweetal metingen.

1. het bepalen van de werkelijke versterking met behulp van een testsignaal;
2. het wegregelen van offset in het uitgangssignaal.

Bij het bepalen van de versterking wordt een sinusvormig signaal met bekende amplitude aangeboden tussen de differentiaalingangen en wordt aan de uitgang de amplitude gemeten. Het quotiënt levert de spanningsversterking. Voor de controle op fase draaiing in de versterker kunnen in- en uitgangssignaal tegelijkertijd op de oscilloscoop worden afgebeeld en kunnen de nuldoorgangen worden vergeleken.

Dit kan herhaald worden voor diverse frequenties, waarbij uiteraard voor alle frequenties die in het werkelijke signaal voorkomen eenzelfde versterkingsfactor vereist is (bepaling frequentie overdracht).

Het wegregelen van de DC-offsetspanning aan de uitgang van de versterker gebeurt bij kortgesloten ingangen. Deze ingangen kunnen het best verbonden worden met aarde van de versterker uitgang. Omdat de versterker een signaal met amplitude nul krijgt aangeboden, moet ook de uitgang amplitude nul hebben. Indien de uitgang een DC-signaal vertoont, kan dit worden weggedraaid met de offsetregeling, zoals deze in de versterker is ingebouwd.



Figuur 8.2: Aanbrengen van de meetklemmen voor het meten van de spanningsval over de patroon.

### 8.2.2 De differentiaalingangen

Voor het meten van de spanning over de smeltdraad van de patroon worden twee klemmen over de caps van de patroon gezet, zoals aangegeven in figuur 8.2. De klemmen zijn speciaal ontworpen voor bevestiging op de caps en garanderen een goed contactoppervlak. Door toepassen van dezelfde materiaaleigenschappen van cap en klem kan ook de invloed van thermospanning worden verwaarloosd.

Omdat bij het aanbrengen van de meetklemmen sprake is van een gesloten lus met oppervlak  $b \cdot l$ , kan nog worden onderzocht in hoeverre inductie ten gevolge van de pulsstroom door de patroon het meetsignaal van spanning kan beïnvloeden.

Uitgaande van het genoemde oppervlak zoals aangegeven in figuur 8.2 kan de volgende afleiding worden gedaan voor de inductiespanning:

$$U_{ind} = -\frac{\partial \Phi}{\partial t}$$

$$\text{met: } \Phi = \int_{r=r_d}^b B(r) \cdot l dr = \int_{r=r_d}^b \frac{l \mu i}{2\pi r} dr = \frac{l \mu i}{2\pi} \cdot \ln\left(\frac{b}{r_d}\right)$$

$$\text{zodat: } U_{ind} = -\frac{l \mu}{2\pi} \cdot \ln\left(\frac{b}{r_d}\right) \cdot \frac{di}{dt}$$

Door invullen van de diverse grootheden kan de invloed worden bepaald. Deze invloed kan drastisch worden verminderd door het oppervlak van de lus te verkleinen (meetdraden twisten, gebruik coax kabel).

Voor het meten van de stroom door de patroon wordt gebruik gemaakt van een weerstandshunt, in serie opgenomen met de patroon.

Een (serie)-shunt kenmerkt zich door het hebben van een constant weerstandsgedrag, meestal met een lage waarde. Door het meten van de spanning over de shuntweerstand is de grootte van de stroom bekend.

Bij de constructie van de shunt is ook aandacht besteed aan de problemen die bij de spanningsmeting zijn besproken. De shunt is daarom ook voorzien van een coaxiaal aansluiting.



## 8.3 De digitizer

De versterkeruitgangen worden aangesloten op een geheugen-oscilloscoop (digitizer). De digitizer is in staat om met een instelbare samplefrequentie de twee versterkersignalen te meten en op te slaan in geheugen. De in het geheugen opgeslagen samples kunnen worden omgezet voor verdere bewerking op een personal computer.

Bij de keuze van de digitizer speelt het aantal kwantiseringsbits een belangrijke rol. Dit bepaalt immers het aantal niveaus waarin het analoge ingangssignaal wordt opgedeeld.

De consequenties van het gebruik van een digitizer in de eerste plaats en het verschillende aantal kwantiseringsbits in de tweede plaats zullen in de volgende paragrafen worden uitgewerkt.

Het zal blijken dat men ook bij het gebruik van een digitizer weloverwogen te werk moeten gaan.

### 8.3.1 Kwantisering

Uit testmetingen, waarbij de smeltpatroon is vervangen door een weerstand met constante waarde volgt na deling van de signalen van spanning door stroom voor de berekende weerstanden geen constante waarde.

Soms ook aan het begin, maar vooral aan het einde van stroompuls treden grote schommelingen op in de berekende weerstandswaarden, die zoals zal blijken zijn toe te schrijven aan het effect van kwantiseren.

Dit effect zal aan de hand van een computer simulatie worden verklaard.

Voor de simulatie wordt een bekende stroomvorm (gedempte sinus) gedefinieerd en wordt de daarbij behorende spanningsvorm berekend, uitgaande van een weerstand met constante waarde.

Het verloop van stroom, spanning en weerstand staan in de linker afbeeldingen van figuur 8.3. De signalen hebben een continu karakter en vormen de ingangssignalen van een digitizer, die deze signalen zal bemonsteren (kwantiseren). Om de invloed van dit kwantiseren duidelijk te maken wordt gewerkt met de eigenschappen van twee verschillende digitizers van het merk VUKO, respectievelijk Bakker.

De VUKO-digitizer heeft een resolutie van 8-bits, wat betekent dat per ingesteld spanningsbereik van de oscilloscoop steeds  $2^8$  verschillende spanningsniveaus worden onderscheiden. Hiervan is dus automatisch een gevoeligheid te koppelen: het kleinste spanningsverschil dat de digitale uitgang een enkele waarde doet veranderen (kwantiseringsstap,  $\Delta V_{diskr}$ ).

$$\Delta V_{diskr} = \frac{\frac{V}{div} \cdot \text{aantal divisies}}{\text{aantal kwantiseringsniveaus}} \left[ \frac{V}{bit} \right] \quad (8.3)$$

De Bakker-digitizer is een 12-bits uitvoering en onderscheid dus een veel groter aantal ingangsniveaus ( $2^{12}$ ).

In de volgende tabel wordt een en ander weergegeven voor twee verschillende waarden van kwantiseringsbits:  $n = 8$  bij de VUKO-scoop en  $n = 12$  bij de Bakker-scoop. Hier dient bij opgemerkt te worden, dat de Bakker-scoop evenals de VUKO-scoop in de beeldopbouw 8 verticale divisies kent.

| V/Div | Div | $\Delta V_{diskr} (n = 8)$             | $\Delta V_{diskr} (n = 12)$                |
|-------|-----|----------------------------------------|--------------------------------------------|
| 0.01  | 8   | $\frac{8 \cdot 0.01}{2^8} = 313 \mu V$ | $\frac{8 \cdot 0.01}{2^{12}} = 19.6 \mu V$ |
| 0.5   | 8   | $\frac{8 \cdot 0.5}{2^8} = 15.6 mV$    | $\frac{8 \cdot 0.5}{2^{12}} = 977 \mu V$   |

Hieruit kunnen de volgende conclusies worden getrokken:

- Het aantal bits (de resolutie) is direkt gekoppeld aan de grootte van de kwantiseringsstap. Bij toenemend aantal bits wordt deze stap steeds kleiner.
- Bij het vergroten van het ingangsbereik van de spanning (b.v.  $0.01 \frac{V}{div} \Rightarrow 0.5 \frac{V}{div}$ ) wordt de kwantiseringsstap groter.

Bij een vast aantal bits moet een compromis tussen maximaal gewenst ingangssignaal en kleinste kwantiseringsniveau worden gevonden. Een toenemende signaalamplitude noodzaakt een groter ingangsbereik en verkleint de gevoeligheid. Een te klein ingangssignaal heeft mogelijk weer gevolgen storing op het signaal tot gevolg.

Het kwantiseren is in Matlab gesimuleerd. In de bovenste twee grafieken van figuur 8.3 is het effect van een 8-bit digitizer zichtbaar. Links staat de grafiek van gekwantiseerde spanning en stroom en rechts de hieruit berekende weerstand ( $R = \frac{V}{I}$ ).

Dit is herhaald voor een 12-bit digitizer en is afgebeeld in het onderste deel van grafiek 8.3.

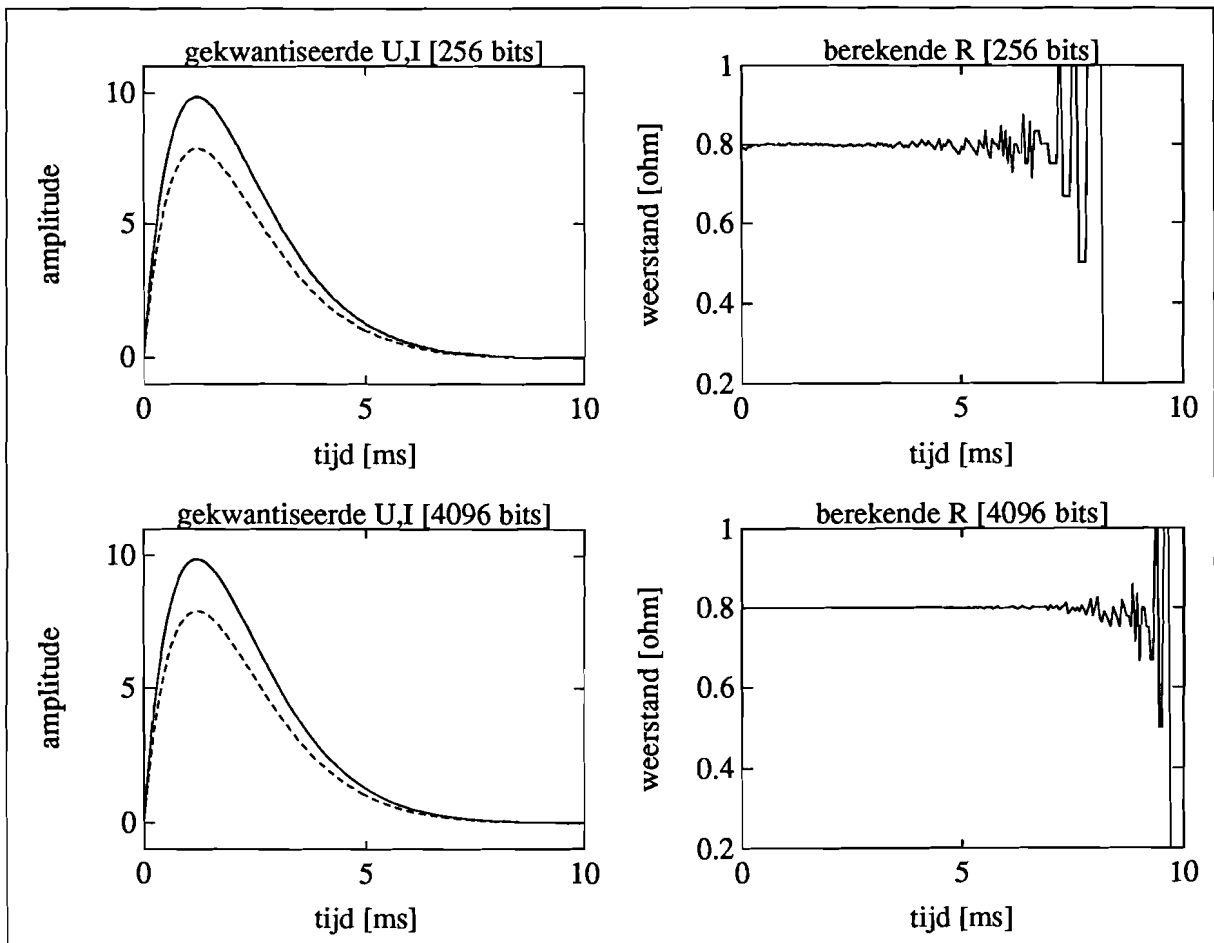
Uit het verschil in verloop van beide weerstandscurven kan het volgende worden geconcludeerd:

- De gekwantiseerde signalen van V en I voor n=8 respectievelijk n=12 tonen alleen in de staart kleine verschillen;
- Het werken met een hoger aantal kwantiseringsniveaus (van n=8 naar n=12) levert over een bredere periode de juiste weerstand op. Kleinere signalen (kop en staart) ondervinden het effect van kwantiseringsfouten;
- Ook de 12-bits kwantisering faalt, als de signalen te klein gaan worden.

### 8.3.2 Signaal optimalisering

Omdat het uiteraard de bedoeling is om zo nauwkeurig mogelijk de weerstand te berekenen, kan worden gezocht naar (extra) aanpassingen, die ook de nauwkeurigheid vergroten, dus de relatieve fout verkleinen.

Voor de relatieve fout kan een volgende afleiding worden gegeven. Hierbij concentreren we ons alleen op de meest belangrijke foutenbronnen, ontstaan bij het kwantiseren.



Figuur 8.3: Verschil in digitaliseren met  $n=8$  en  $n=12$ .

De weerstand wordt berekend uit

$$R_k = R + \Delta R = \frac{V_k}{I_k}$$

met:

$$V_k = V + \Delta V$$

$$I_k = I + \Delta I$$

Het verschil tussen het meetsignaal  $V$  en het uitgangssignaal  $V_k$  wordt aangeduid met  $\Delta V$  en ontstaat ten gevolge van kwantiseringsfouten van de digitizer. De grootte van deze fout hangt af van de resolutie van de digitizer, uitgedrukt in het aantal bits. Voor zowel het spanning- als het stroomsignaal (wordt door spanningssignaal gerepresenteerd) geldt een maximale fout:

$$\Delta V = \frac{1}{2} V_{diskr}$$

Voor de relatieve fout in de berekende weerstand volgt nu bij verwaarlozen van de hogere orde termen de volgende afleiding:

$$R(1 \pm \frac{\Delta R}{R}) = \frac{V}{I} \cdot \frac{(1 \pm \frac{\Delta V}{V})}{(1 \pm \frac{\Delta I}{I})} \Leftrightarrow (1 \pm \frac{\Delta R}{R}) = \frac{(1 \pm \frac{\Delta V}{V})(1 \mp \frac{\Delta I}{I})}{(1 - (\frac{\Delta I}{I})^2)} = 1 \pm \frac{\Delta I}{I} \pm \frac{\Delta V}{V}$$

Dit leidt tot de uiteindelijke uitdrukking:

$$\left| \frac{\Delta R}{R} \right| \leq \left| \frac{\Delta V}{V} \right| + \left| \frac{\Delta I}{I} \right| = \left| \frac{\frac{1}{2} V_{diskr,V}}{V_{out,V}} \right| + \left| \frac{\frac{1}{2} V_{diskr,I}}{V_{out,I}} \right| \quad (8.4)$$

De relatieve fout van het quotiënt is de som van de relatieve fouten in teller en noemer.

Formule 8.4 toont aan, welke factoren van invloed zijn op de juiste bepaling van de weerstand.

- Door het gebruiken van een hogere resolutie digitizer ontstaan bij eenzelfde signaalbereik kleinere waarden voor  $\Delta I$  en  $\Delta V$ , met gevolg een kleinere fout in  $R$ . Voor kleine waarden van  $V$  en  $I$  gaat dit echter ook weer fout. Een en ander is reeds eerder met de simulatie in Matlab aangetoond;
- De fout kan volgens de formule ook worden verkleind, door de signalen  $V$  en  $I$  te vergroten. Dat klopt, zolang dit geen invloed heeft op de waarden van  $\Delta V$  en  $\Delta I$ . In de praktijk betekent dit het vergroten van het signaal tot maximale uitslag op het scherm, zonder over te schakelen op een ongevoeliger bereik! Een ongevoeliger bereik levert immers een grotere kwantiseringstap en dus ook weer een grotere  $\Delta V$ .
- Een mogelijkheid tot het 'kunstmatig' verhogen van het stroom- (en spanningssignaal) door middel van een offset kan eveneens worden beschouwd.

### 8.3.3 Toepassen van offset stroom

In deze paragraaf wordt aan de hand van praktijkwaarden nader onderzocht of het mogelijk is om de meetnauwkeurigheid te vergroten, door toepassen van een extra constante meetstroom door de patroon.

Door het toepassen van de offsetstroom blijven de meetsignalen van spanning en stroom immers ook voor en na de puls ongelijk aan nul, waardoor het mogelijk blijft om de weerstand te berekenen uit het quotiënt van spanning en stroom.

Daarbij moet men zich wel realiseren dat de extra stroom kan zorgen voor een verwarming van de smeltdraad, waardoor ook de weerstand van de fuse wordt veranderd. Dit moet worden voorkomen.

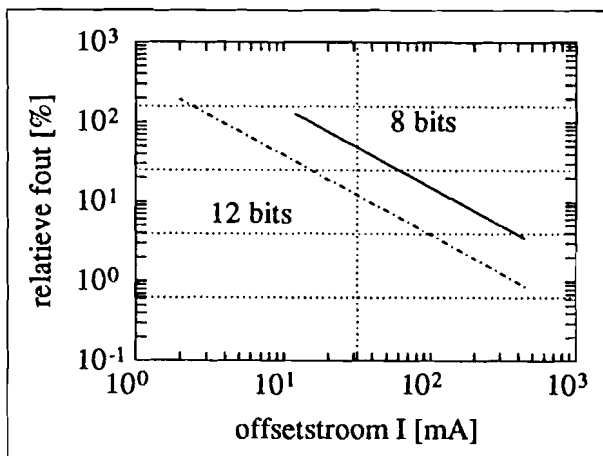
De volgende afleiding is gebaseerd op formule 8.4 van de vorige paragraaf en de benodigde getalgegevens zijn afkomstig van een praktijkmeting aan een 315mA miniatuurpatroon (type Littelfuse 217.315) bij een condensator laadspanning  $U_c$  van 10V. De meting is gedaan met een 8-bit digitizer.

| grootheid                | symbool     | waarde              |
|--------------------------|-------------|---------------------|
| aantal digitizer bits    | $n$         | 8                   |
| spanningsversterking     | $A_V$       | $1,074 \frac{V}{V}$ |
| stroomversterking        | $A_I$       | $25,80 \frac{A}{V}$ |
| shuntweerstand           | $R_{shunt}$ | $19,4 m\Omega$      |
| gemiddelde fuseweerstand | $R_{fuse}$  | $800 m\Omega$       |
| gevoeligheid V-kanaal    | $V_{Vsens}$ | $15,6 mV$           |
| gevoeligheid I-kanaal    | $V_{Isens}$ | $6,25 mV$           |

Worden de getallen in formule 8.4 ingevuld, dan volgt voor deze situatie:

$$\begin{aligned}
 \left| \frac{\Delta R}{R} \right| &\leq \left| \frac{\frac{1}{2} \cdot 15,6 \cdot 10^{-3}}{I \cdot 0,8 \cdot 1,074} \right| + \left| \frac{\frac{1}{2} \cdot 6,25 \cdot 10^{-3}}{I \cdot 19,4 \cdot 10^{-3} \cdot 25,80} \right| \\
 &\leq \left| \frac{9,078 \cdot 10^{-3}}{I} \right| + \left| \frac{6,244 \cdot 10^{-3}}{I} \right| \\
 &\leq \left| \frac{15,32}{I [mA]} \right|
 \end{aligned} \tag{8.5}$$

Relatie 8.5 is in grafiek 8.4 getekend.



Figuur 8.4: Grafiek van de relatie tussen stroom  $I$  door de fuse en de relatieve fout in berekende weerstand  $R$ .

Merk hierbij nogmaals op dat bij deze afleiding alleen rekening gehouden is met fouten ten gevolge van kwantiseren.

De afleiding kan ook worden uitgevoerd indien gebruik gemaakt wordt van een 12-bit digitizer (Bakker systeem).

Dit systeem heeft een ingangsbereik  $-5V \leq V_{in} \leq +5V$  met een 12-bit digitizer resolutie. Het systeem kent geen aparte instelling voor  $\frac{V}{div}$  zodat voor alle (vier) kanalen een vaste gevoeligheid geldt:

$$V_{sens} = \frac{V_{max} - V_{min}}{2^{12}} = \frac{5 - (-5)}{4096} = 2,44 mV$$

Wordt in eerste instantie de versterking van de differentiaal versterkers niet aangepast, dan volgt voor de relatieve fout in weerstand:

$$\begin{aligned} \left| \frac{\Delta R}{R} \right| &\leq \left| \frac{\frac{1}{2} \cdot 2,44 \cdot 10^{-3}}{I \cdot 800 \cdot 10^{-3} \cdot 1,074} \right| + \left| \frac{\frac{1}{2} \cdot 2,44 \cdot 10^{-3}}{I \cdot 19,4 \cdot 10^{-3} \cdot 25,8} \right| \leq \left| \frac{1,42 \cdot 10^{-3}}{I} \right| + \left| \frac{2,44 \cdot 10^{-3}}{I} \right| \\ &\leq \left| \frac{3,86}{I[mA]} \right| \end{aligned} \quad (8.6)$$

Ook formule 8.6 is in figuur 8.4 uitgezet.

Opgemerkt dient te worden dat bij gebruik van de Bakker scoop de versterking van zowel spanning- als stroomsignaal nog vergroot kan worden, zodat de relatieve fout nog kleiner wordt.

Vergelijkt men het verkregen antwoord uit formule 8.5 met formule 8.6 dan kunnen de volgende conclusies worden getrokken:

- De 12-bit digitizer levert bij dezelfde stroom een relatieve fout die ongeveer een orde 4 kleiner is dan die van de 8-bit digitizer;
- Bij beide digitizers gaat de relatieve fout voor kleine waarden van stroom naar oneindig. De stroom moet een bepaalde waarde niet onderschreiden. Dit noodzaakt het gebruik van een offsetstroom;
- Voor een relatieve fout van maximaal 10% moet bij de 8-bit digitizer een offsetstroom van 153,2mA worden aangeboden. Bij een offsetstroom van 315mA bedraagt deze fout nog 4,85%. Een dergelijk grote offsetstroom is uiteraard niet toelaatbaar in verband met de verstoring in het temperatuurgedrag;
- Bij de 12-bit digitizer kan worden volstaan met kleinere offsetstromen. Een stroom van 38,6mA offset levert een maximale relatieve fout van 10%. Bij 100mA offset is de fout 4,85%.

Men kan uiteraard stellen dat het gebruik van een 12-bit digitizer voorkeur verdient boven een 8-bit digitizer, maar dat voor beide systemen problemen ontstaan die inherent zijn aan het meetprincipe.

Door het toepassen van een offsetstroom kan de relatieve fout in de kritische gebieden worden verkleind, waarbij een te grote offsetstroom altijd moet worden vermeden, omdat anders door de temperatuur beïnvloeding een grote systematische fout wordt geïntroduceerd!

## 9. BACOS Software

*De handleiding is gebaseerd op BACOS versie 6.11. Het versienummer wordt bij het opstarten van BACOS vermeld. Voor afwijkende versienummers dient U contact op te nemen met dhr. L.A.H. Wilmes, EEG 0.17, tel. 4470.*

### 9.1 Beschrijving

Om meetdata, geregistreerd met de 12-bits digitizer van de firma Bakker, geschikt te maken voor verdere verwerking met andere software op een personal computer, is het noodzakelijk gebleken om software te ontwikkelen waarmee een conversie van data plaatsvindt.

De opgeslagen Bakker data bestaat uit tijd-data en meetwaarde-data van maximaal 4 kanalen, met ieder een maximale lengte van 65535 meetpunten.

De conversie is puur een formaat-conversie en bestaat uit het inlezen van de datafile van de digitizer en het wegschrijven in een gewenst uitvoerformaat.

Op dit ogenblik worden een tweetal uitvoerformaten ondersteund:

**ASCII formaat** Hierbij wordt de data in kolommen leesbare tekst gezet. Dit uitvoerformaat wordt door veel software pakketten voor de PC ondersteund;

**Matlab formaat** Hierbij wordt de data in een door Matlab<sup>1</sup> ondersteund formaat weggeschreven en kan daardoor direct in Matlab worden gebruikt.

De conversie-software is zodanig van opzet, dat op een later tijdstip alsnog nieuwe bestandsformaten kunnen worden bijgebouwd. De software is geschreven in *Turbo Pascal version 6.0*.

Voor een nadere beschrijving van de software wordt verwezen naar paragraaf 9.4. Een beschrijving van de formaten van BAKKER en Matlab wordt gegeven in bijlagen H en I.

### 9.2 Systeemeisen

Voor het gebruik van de software dient op de volgende punten te worden gelet:

- De software draait op PC, XT, AT en latere machine-typen, al dan niet uitgerust met een co-processor;
- Er is geen grafische kaart vereist;
- Conversie geschiedt van diskfile- naar diskfile zonder opslag in RAM geheugen. Er is daarom geen extra Extended- of Expanded geheugen vereist;

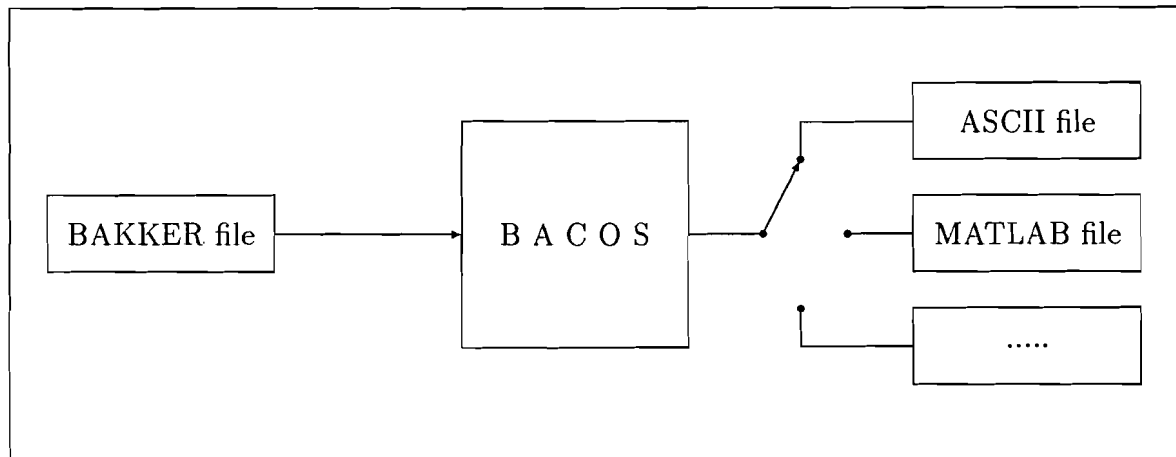
---

<sup>1</sup>Matlab is een programmeeromgeving waarbinnen standaard een groot aantal wiskundige bewerkingen mogelijk zijn. Ook voor dataverwerking biedt Matlab een groot aantal mogelijkheden. Het pakket is in de uitleen van het rekencentrum van de TUE.

- Omdat de geconverteerde files in sommige gevallen zeer groot kunnen worden, is het gebruik van harddisk in dergelijke gevallen noodzakelijk.

### 9.3 Gebruik software

De software heeft de naam **BACOS** gekregen wat staat voor **BAKker CO**nversie **S**oftware en bestaat uit een enkel, direct vanuit MSDOS uitvoerbaar programma.



Figuur 9.1: Functiediagram van het programma BACOS

Figuur 9.1 toont het functiediagram van BACOS: een invoerfile in Bakker-formaat wordt omgezet in een gewenste uitvoerfile. Daarbij kan tijdens de conversie worden aangegeven van welke kanalen informatie moet worden omgezet.

#### 9.3.1 Handleiding BACOS

In deze paragraaf zal aan de hand van een voorbeeld de bediening van het programma BACOS nader worden toegelicht. In het voorbeeld wordt een BAKKER-bestand DATA.001 omgezet in een Matlab-bestand PULS.MAT.

Het programma wordt opgestart door het intoetsen van:

(B) (A) (C) (O) (S) (↵)

Na de intromelding wordt gevraagd naar de naam van de invoerfile. Uiteraard mag behalve de naam ook het volledig pad worden ingevuld. Dit indien het bestand zich niet in de huidige directory bevindt.

Er wordt gecontroleerd of de file bestaat. Zoniet, dan wordt opnieuw gevraagd. Wordt geen naam opgegeven, dan wordt het programma gestopt.

Bestaat de file, dan wordt eerst gekeken of het werkelijk een Bakker datafile is. Stemt de identificatiecode in de file niet overeen met die van een Bakker datafile, dan volgt een melding en wordt programma uitvoer gestopt.



Voor het voorbeeld wordt ingetypt:

Vervolgens moet een naam voor de uitvoerfile worden gegeven. Ook hier kan, indien gewenst, een pad worden opgegeven. Het programma controleert of de genoemde file reeds bestaat en vraagt in dat geval om een bevestiging voor het overschrijven ervan.

Als voorbeeld volgt:

Opgemerkt dient te worden dat, in het geval van opslag voor Matlab formaat, een extensie 'MAT' wenselijk is. Dit wordt door BACOS niet automatisch gedaan om het mogelijk te maken alsnog een andere extensie te gebruiken.

Nu in- en uitvoerfile zijn opgegeven volgt de laatste stap, waarin wordt opgegeven in welk formaat de uitvoer beschikbaar zal komen en wat er wordt opgeslagen.

Daarvoor is het noodzakelijk om te weten wat er in de Bakkerfile is opgeslagen. Deze informatie is te vinden in de XHEADER (naamgeving is conform het Bakker-formaat) die door het intoetsen van

kan worden getoond. In de XHEADER staat te lezen hoeveel kanalen gebruikt zijn, welke tijdsresolutie is gebruikt, tijdstip van begin van de registratie, etcetera. Deze informatie dient als extra en wordt door BACOS gebruikt voor het genereren van de uitvoer file.

De keuze van uitvoerformaat wordt gedaan door het intoetsen van M of A. Voor dit voorbeeld geldt

Andere formaten worden in deze versie van BACOS niet ondersteund.

Zoals uit de XHEADER data reeds blijkt, bestaat de data uit tijdinformatie (tijdas) en kanaal-informatie (meetwaarden) van 1 to maximaal 4 kanalen. Omdat het niet altijd wenselijk is om al deze informatie te converteren, is in BACOS de mogelijkheid ingebouwd om de uitvoersignalen te selecteren.

BACOS zal van alle beschikbare kanalen gedetailleerde informatie verschaffen en vragen of het betreffende kanaal in de uitvoer moet worden opgenomen. Dit is eveneens het geval met de tijdinformatie.

Door het simpel antwoorden met

of

wordt een keuze gemaakt.

Na het beantwoorden van de laatste vraag start de conversie. Is de conversie voltooid, dan wordt het programma gestopt.

### 9.3.2 Specificatie van ASCII-uitvoerfile

ASCII uitvoer kan in programma's worden gebruikt, die de functie van ASCII-import ondersteunen. Alle data is opgeslagen in kolommen: Het aantal kolommen kan variëren van 1 to maximaal 5. De kolommen worden opgebouwd van links naar rechts met een selectie uit 'tijd - kanaal 1 - ... kanaal 4'. Boven iedere kolom wordt de naam van de kolom weergegeven en de bijbehorende eenheid, zoals in de invoerfile is opgeslagen.

Alle waarden  $x$  zijn afgedrukt in SINGLE formaat volgens WRITELN(WAARDE:10:4). Zie voor een overzicht van Turbo Pascal 6.0 formats, bijlage J op bladzijde 141.

### 9.3.3 Specificatie van Matlab-uitvoerfile

De Matlab uitvoerfile kan binnen Matlab direct met het commando

```
LOAD <FILENAAM>
```

worden ingelezen.

Met het commando

```
WHO
```

wordt aangegeven, welke variabelen zijn ingelezen.

Tijdinformatie wordt altijd opgeslagen in 3 variabelen:

**TB** De begintijd [s];

**TS** De stapgrootte [s];

**TE** De eindtijd [s].

Met het commando

```
T = TB:TS:TE;
```

wordt in Matlab de juiste tijdvektor aangemaakt. Op deze manier wordt de opslag van tijdinformatie in het Matlab-bestand op een efficiënte manier gerealiseerd.

Kanaalinformatie wordt opgeslagen in een variabelenaam, gelijk aan de naam van het kanaal op het BAKKER systeem.

Hierbij moet worden opgemerkt dat spaties in een naam altijd worden vervangen door een underscore-karakter (-). Zo wordt CH 1 binnen Matlab een variabele met naam CH\_1. Matlab kent geen variabelen met spaties.

## 9.4 Ontwerp software

De broncode van de BACOS software is geschreven in Turbo Pascal versie 6.0 en bestaat uit één kort hoofdprogramma BACOS.PAS en twee UNITS met de namen U\_BACOS.PAS en U\_BACOS1.PAS.

Alle functie- en procedure aanroepen vanuit het hoofdprogramma zijn opgeslagen in de UNIT U\_BACOS. Het hoofdprogramma zelf bevat geen functie- en procedure definities.

De UNIT U\_BACOS1.PAS wordt door het hoofdprogramma tijdens compileren in Pascal gekoppeld *en is bedoeld als opslag van nieuwe uitbreidingen op de huidige programmatuur*. Een nieuwe conversie mogelijkheid bijvoorbeeld zal hierin worden opgenomen. Hierdoor is het niet nodig om wijzigingen aan te brengen in U\_BACOS (alleen het hoofdprogramma verandert mee).

Hiermee wordt getracht om foutieve wijzigingen in de reeds bestaande software te voorkomen.

Voor deze opzet is gekozen na overleg met de beheerder van de software. Uiteraard dienen uitbreidingen op de software met hem te worden besproken en dient de broncode te worden voorzien van het nodige commentaar.

De beheerder van de software is : dhr. L.A.H. Wilmes, EEG 0.17, tel. 4470.

Bijlagen K, L en M op bladzijde 142 en verder, geeft een listing van de broncode.

## 9.5 Uitbreiding software

Het programma BACOS moet zoals het nu is gezien worden als een eerste opzet van conversie routines voor het BAKKER meetsysteem. Dat betekent dat op een aantal punten mogelijk uitbreidingen en verfijningen gewenst zijn. Hierbij valt te denken aan:

- Andere conversieslagen dan Matlab en ASCII;
- Mogelijkheid tot een selectie van een deel van de kanaal informatie;
- Mogelijkheid tot het gebruik van een middeling en/of data reductie;
- Verfijning van de user interface;

Om wildgroei van verschillende BACOS versies te vermijden, is het wenselijk om veranderingen en uitbreidingen met de beheerder van de software te overleggen.

## Deel III. Literatuur

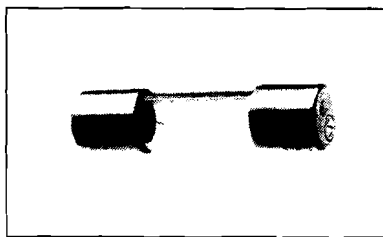
### 10. Literatuurstudie

*Dit deel bespreekt de resultaten van de literatuurstudie, die bij aanvang van het afstudeerwerk is verricht.*

#### 10.1 Doel literatuurstudie

Bij het opstellen van een theorie inzake de betrouwbaarheid van smeltpatronen onder normale bedrijfscondities, speelt de grootte van de temperatuur een belangrijke rol. Ten gevolge van veranderende temperatuur treden (blijvende) wijzigingen op in de materiaaleigenschappen, die leiden tot veroudering van de patroon.

Het is daarom wenselijk om bij een gegeven elektrisch belastingspatroon een idee te krijgen van de optredende temperatuurverdeling over de smeltdraad van de patroon. Dit houdt in dat de temperatuur van (een deel van) de draad gemeten moet gaan worden.



Figuur 10.1: Afbeelding van een miniatuur glaspatroon, vooral veel gebruikt in huishoudelijke apparatuur en elektronische schakelingen. Lengte 2cm, diameter 0,5cm. Nominaalstromen tot  $\pm 10A$ , 250V.

De betrouwbaarheidsstudie is in de vorm van een promotie onderzoek en beschouwt meerdere uitvoeringen van smeltpatronen (miniatuur smeltpatroon, substraatpatroon, ...). Op dit moment wordt onderzoek verricht aan miniatuur glaspatronen, die veelvuldig in elektrische (huishoudelijke) apparaten worden toegepast. Een afbeelding is gegeven in figuur 10.1.

Het is de bedoeling dat van de smeltdraad in de patroon, de temperatuur bepaald wordt. Nog nader te noemen omstandigheden maken echter dat deze temperatuurmeting geen standaard meting is.

Daarom is in de uitgevoerde literatuurstudie, die een verplicht onderdeel vormt van het afstuderen, nagegaan of voor deze speciale temperatuurmeting apparatuur en/of meetmethoden beschikbaar zijn.

Bij de literatuurstudie is het zoekprofiel gebaseerd op de volgende criteria:

- Omdat de draad zich in een glazen huls bevindt is het is een contactloze meting. Ook laten de afmetingen van de draad het niet toe om met contactsensoren te werken;
- Aangezien er sprake is van een temperatuurverdeling over de draad (in het midden wordt deze het warmst) moet het mogelijk zijn om de temperatuur te meten van een 'klein stukje' draad;
- De meting is bedoeld voor het registreren van het temperatuurverloop tijdens excitatie met een elektrische teststroom door de draad. Binnen een periode van ongeveer 6ms wordt de draad opgewarmd van kamertemperatuur tot zo'n 500°C en koelt daarna weer af. Een snelle responsie van het meetsysteem is daarom vereist;
- Tijdens de stroom-excitatie is de draad aan beweging onderhevig (trilling). De draad moet steeds binnen het beeld van de temperatuur-opnemer blijven;
- Het relatief lage temperatuurgebied van 20 tot 500°C stelt mogelijk extra eisen aan koeling van de temperatuur-opnemer, om deze op een voldoende lage temperatuur te brengen.

## 10.2 Conclusies literatuurstudie

De literatuurstudie heeft inclusief de verslaglegging ruim zeven weken van intensief onderzoek gevergd en geleid tot de volgende conclusies:

- Uit de literatuur volgt geen standaard oplossing in de vorm van een zelf te bouwen systeem. Ook binnen de TUE houdt men zich niet bezig met dit specifieke onderwerp;
- Voor contactloze metingen met snelle reponsietijden zijn HgCdTe of pyro-elektrische detectoren noodzakelijk. Prijzen hiervan variëren van fl. 5.000,- tot fl. 10.000,- (denk hierbij ook aan noodzakelijke koeling en choppers). Dergelijke detectoren zijn slechts verkrijgbaar bij een beperkt aantal, gespecialiseerde bedrijven;
- De afmetingen van de zekeringdraad maken het gebruik van een lenzensysteem noodzakelijk. Hierbij moeten speciale lenzen worden gebruikt om IR-straling door te laten. Deze lenzensystemen kosten ongeveer fl. 20.000,-;
- Bij zelfbouw van een detectorsysteem dient naast de sensor zelf ook aandacht besteed te worden aan benodigde sturingshardware. Denk hierbij aan versterkers, choppers, koeling, triggering en signaalverwerking. Een en ander moet zorg dragen voor een stabiel en betrouwbaar systeem, waarmee reproduceerbare metingen kunnen worden verricht;
- Er zal moeten worden onderzocht in welke mate de geijkte detector nog goed functioneert bij het meten van de draad temperatuur (de draad is aan beweging onderhevig). Eventueel dienen correctiemethoden te worden gevonden. Dit vereist een goede kennis op het gebied van temperatuurstraling;
- De responsietijd van de meeste standaard meetsystemen is te groot ( $\geq 10$ ms), zodat de snelle temperatuurgradiënten niet kunnen worden geregistreerd;

- Er zijn enkele commerciële systemen op de markt, waarmee de draadtemperatuur direct kan worden gemeten. Door de grote flexibiliteit van deze apparatuur, kan naast temperatuur ook de positie worden gemeten. Dergelijke systemen zijn echter niet goedkoop. Basissystemen kosten ongeveer fl. 200.000,-.

In overleg met de firma Optilas (Hr. Gritters) uit Alphen ad Rijn is een offerte gemaakt van een meetsysteem van Inframetrics, type 600XL. Er is foldermateriaal beschikbaar bij X.Z. Meng.

Voor een eventuele toekomstige financiering is een subsidie-aanvraag ingediend.

Naar aanleiding van deze conclusies is besloten om niet langer stil te staan bij methoden voor het rechtstreeks bepalen van de temperatuur. In deel 2 van dit afstudeerverslag wordt een alternatieve methode besproken.

Voor het volledige verslag van de literatuurstudie, inclusief de zeer uitgebreide literatuurlijst en foldermateriaal dient contact opgenomen te worden met de afstudeerbegeleider. Hiervan bestaat een apart verslag.

De literatuurlijst bevat een groot aantal publicaties op het gebied van temperatuurmeting. Er worden vergeleken gemaakt tussen de diverse methoden, specificaties besproken, etcetera.

## 10.3 Definitieve literatuurlijst

Tot slot van dit deel volgt een definitieve literatuurlijst, zoals deze voor het bibliotheekpracticum is samengesteld. De lijst is ontstaan door het hanteren van extra selectie criteria, toegepast op de complete literatuurlijst. Zo is gekeken naar het jaar van publikatie en de beschikbaarheid van het artikel op de TUE.

De lijst bevat literatuur op het gebied van contactloze temperatuurmeting zoals voor dit project wenselijk is. Zoals reeds uit de conclusies bleek, geeft geen van de artikelen een kant en klare oplossing.

Met de verzamelde artikelen kan echter wel een goed beeld worden verkregen van de laatste ontwikkelingen op dit gebied. Dit is mogelijk interessant voor latere experimenten, waarbij temperatuur gemeten moet worden bij andere soorten smeltpatronen.

De lijst bevat boektitels uit het boekzoeksysteem VUBIS en artikelen uit de Electric and Electronics Abstracts (EEA) en de Computer and Control Abstracts (CCA).

De specifieke nummering komt overeen met die in de complete literatuurlijst.

### 10.3.1 Bronnen VUBIS

**B1** William L. Wolfe, George J. Zissis (EL:GGJ78INF), *The Infrared handbook*. (Library of Congress Catalog Card Number: 77-90786). United States Government Printing Office, 1978.

**B2** Thomas R. Harrison (NT:GCE60HAR), *Radiation pyrometry and its underlying principles of radiant heat transfer*. (Library of Congress Catalog Card Number: 60-10294). New York - London, John Wiley & Sons Inc., 1960.

- B3** C. Lee (NT:GCE88NON), *Noncontact temperature measurement, proceedings of a workshop held at NASA Headquarters Washington, DC, April 30 - May 1, 1987*. (NASA publication number 2503). Washington, NASA, 1988.
- B4** R.D. Hudson jr., J. Wordsworth Hudson (EL:GGJ75INF), *Infrared detectors; part 2 of Benchmark Papers in optics*. Stroudsburg, Pennsylvania: Halsted Press, 1975.
- B5** J.C. Richmond, D.P. DeWitt (NT:GCE85APP), *Applications of radiation thermometry. ASTM special publication 895. A symposium sponsored by the American Society for Testing and Materials (ASTM) Committee E-20 on temperature measurement in cooperation with the National Bureau of Standards Gaithersburg, MD, May 1984*. Ann Arbor, MI: January 1986.
- B6** J. Scholz, T. Ricolfi (EL:DTE89SEN), *Sensors, a comprehensive survey; volume 4: thermal sensors*. Weinheim (Federal Republic of Germany), VCH Verlagsgesellschaft GmbH, D-6940, 1990.
- B7** D. Kuhn (CM:APG67KUE), *Ein Farbpyrometer zur Messung rasch veränderlicher Oberflächentemperaturen*. Fakultät für Maschinenwesen der Technischen Hochschule Hannover, 1967. Bamberger Fotodruck, Bamberg (1967).
- B8** V. Tank (CM:APG88TAN), *Infrarot-Temperaturmessung mit selbsttätiger Berücksichtigung des Emissionsgrads*. Fakultät für Elektrotechnik und Informationstechnik des Technischen Universität München, 1987.
- B9** Charles M. Herzfeld (CM:CGE62TEM), *Temperature: its measurement and control in science and industry: symposium : 4th, 1961, March 27-31, Columbus, Ohio*. Huntington, NY, Krieger, 1972 - 3dl.

### 10.3.2 Bronnen EEA en CCA

- C2** J. Dixon, *Radiation thermometry, advances*. Control and Instrumentation (UK), v18, n7, p45-47 (July 1986).
- C23** H. Molendijk, *Selection of temperature sensors*. Automatie (NL), v31, n10, p277-286, Oktober 1987.
- C28** s.n., *Pyrometers*. Control and Instrumentation (UK), v15, n10, p35,37,39-40,43, Oct 1983.
- C31** A. Krigman, *A guide to selecting non-contact temperature instrumentation*. Intech USA, v30, n6, p23-30 (June 1983).
- C43** C. Nugent, *Smart temperature sensors*. Electronics & Wireless World (UK), v95, n1636, p189-190 (Feb 1989).

- C44** C. Siskovic, *New developments expand use of fiber optic IR thermometers.*  
I&CS (USA), v62, n6, p37-39 (June 1989).
- C45** W. Labs, *Temperature monitoring and control; a technology update.*  
I&CS (USA), v62, n6, p25-31 (June 1989).
- C48** U. Kienitz, *Intelligent pyrometers.*  
Messen steuern regeln (East Germany), v32, n2, p63-66 (1989).
- C49** E. Kaiser, *Temperature sensors for process measurement.*  
Messen steuern regeln (East Germany), v32, n11, p482-485 (1989).
- C53** T. Kerlin, *Smart temperature measurement in the 1990's.*  
Control and Instrumentation (UK), v22, n7, p43-47 (July 1990).
- C54** T. Kerlin, *Temperature measurement in the 1990's.*  
Proceedings of the ISA'89.
- E18** s.n., *Monochromatic pyrometers to be chosen case by case.*  
Mesures (France), n613, p29-34 (20 nov 1989). (French).
- E20** J. Bladou, *Temperature sensors: from classic to modern.*  
Electronique Industrielle (France), n168, p44-48.
- E22** C. Lucas, *IR detection, some recent developments and future trends.*  
Sensors and Actuators A, Phys. (CH), vA25, n1-3, p147-154 (1991).
- E24** H. Schopf, *Integrated pyroelectric IR matrix arrays.*  
Sensors and Actuators A, Phys. (CH), vA26, n1-3, p401-405 (1991).
- E26** D. Peyton, *System-oriented survey of non-contact temperature measurement techniques for rapid thermal processing (in ULSI).*  
Proceedings SPIE - International Society of Optical Engineering (USA), v1393, p295-308.
- E28** G. Bartoslak, *Line scanning IR thermometers team up with powerful computers.*  
I&CS (USA), v64, n6, p47-49 (June 1991).
- E29** C. Warren, *Smart IR sensors improve product quality, process efficiencies.*  
I&CS (USA), v64, n6, p43-4 (June 1991).
- E36** R.W. Whatmore, *Pyroelectric ceramics and devices for thermal IR detection and imaging.*  
Ferroelectrics (UK), v118, n1-4, p241-259.
- E37** A. Mansingh, *Pyroelectric films for IR applications.*  
Indian Journal of Pure and applied physics (India), v29, n10, p657-664.
- E47** I. Pointer, *Temperature measurement.*  
Control and Instrumentation (UK), v18, n2, p32-37 (Feb 1986).



- E51** J.R. Saffell, *Accurate temperature measurement: recent advances*.  
Transducer Technology (UK), v9, n2, p13 (March 1986).
- E53** L.E. Ravich, *Optical fiber thermometry*.  
Laser Focus/Elektro-Optics (USA), v22, n7, p104-105,107,109-111,113-115 (July 1986).
- E54** T. Lorigoux, *The choice of IR radiation pyrometers*.  
Mesures (France), v51, n12, p59,61,63,65,68 (Oct 1986) French.
- E60** s.n., *Temperature measurement on and off the record*.  
Laboratry Equipment Digest (UK), v24, n11, p69-72 (Nov 1986).
- E71** B. Tinham, *Pyrometry advances expand applications*.  
Control and Instrumentation (UK), v19, n12, p45,47 (Dec 1987).
- E73** A.A. Turnbull, *High resolution 64-element pyroelectric linear array detector*.  
Proceedings SPIE - International Society of Optical Engineering (USA), v807, p92-97 (1987).
- E74** U. Baustian, *Improved IR temperature measurement by utilization of laser technology*.  
proceedings of SENSORS ECPO, p203-204.
- E75** J. Pook, *Choosing surface temperature sensors*.  
Control and Instrumentation (UK), v20, n4, p75-77 (Apr 1988).
- E81** D.M. Shcherbina, *Fast response pyrometer*.  
Measurement Techniques (UK), v22, n7, p824-827, (July 1980).
- E82** s.n., *IR pyrometer for continuous temperature measurement of moving wire or bar*.  
Mesures Regulation Automatisme (France), v47, n4, p96-97 (1982).
- E83** K. Herrmann, *Optische sensoren - stand und tendenzen*.  
Messen steuern regeln (East Germany), v32, n2, p496-500 (1989).
- E84** D. de Cogan, *Temperature measurement of thin films on substrates*.  
Proceedings of the IEE, v132, p1, n3, p143-146 (June 1985).
- E85** P.W. Webb, *Measurement of thermal transients in semiconductor power devices and circuits*.  
Proceedings of the IEE, v130, p1, n4, p153-159 (August 1983).

# Bijlagen van deel 1

## A. Literatuurlijst

- [1] Kenneth W. Awkward and Randall D. Lewis.  
*Model for determining thermal profiles of bond wires using PSPICE analysis.*  
7th IEEE-THERM Symposium, 1991, p86-90.
- [2] E. Hoefler en H. Nielingen; bewerkt door A. Derks en N. Bouwman.  
*SPICE.*  
Nijgh en van Ditmar Educatief, 2e druk, Den Haag, 1990. TUE-code:  
LLK90HOE bse.
- [3] P.W. Tuinenga.  
*A guide to circuit simulation and analysis using PSPICE.*  
ISBN 0-13-834609-0.
- [4] A. Goldsmith, T. Waterman, H. Hirschhorn.  
*Handbook of thermophysical properties of solid materials, vol. 1.*  
London, Pergamon 1961, revised edition. TUE-code: HDG61TAN bsw.
- [5] Y.S. Touloukian, R.W. Powlell, C.Y. Ho.  
*Thermal conductivity: metallic elements and alloys.*  
S.1, IFI/Plenum, 1970. Thermophysical properties of matter, vol. 1.  
TUE-code: GCA70TOU bsn.
- [6] Y.S. Touloukian, E.H. Buyo.  
*Specific heat: metallic elements and alloys.*  
S.1, IFI/Plenum, 1970. Thermophysical properties of matter, vol. 4.  
TUE-code: GCA70TOU bsn.
- [7] F. Kreith, M. Bohn.  
*Principles of heat transfer.*  
Harper & Row publishers inc, fourth edition, 1986.
- [8] J. Lienhard.  
*A heat transfer textbook.*  
Prentice-Hall inc., Englewood Cliffs, New Jersey 07632, second edition,  
1987. TUE-code: FMB87LIE bsw.

- [9] J. Holman.  
*Heat transfer.*  
McGraw-Hill book company, 1986, sixth edition. TUE-code: FMB86HOL bsn.
- [10] W. Elenbaas.  
*Dissipation of heat by free convection.*  
Philips Research report 3, p338-360, 450-465. 1948.
- [11] S. Churchill, H. Chu.  
*Correlating equations for laminar and turbulent free convection from a horizontal cylinder.*  
International Journal on Heat and mass Transfer, vol 18. p1049-1053, 1975.
- [12] T. Tsubouchi, H. Masuda.  
*Heat transfer by natural convection from horizontal cylinders at low Rayleigh numbers.*  
Report 190 of the Institute of High Speed Mechanics, Tohoku University, Japan. vol. 19, p205-219, 1967/1968.
- [13] R. Perry, C. Chilton.  
*Chemical Engineer's handbook.*  
McGraw-Hill, NY. 5th edition, section 10. 1973. TUE-code: JNA73CHE bsc.
- [14] M. van Ardenne.  
*Tabellen zur angewandten Physik, band II.*  
Deutscher verlag der Wissenschaften, Berlin, 1973. TUE-code: GAK73ARD bse.
- [15] J. Keown.  
*PSPICE and circuit analysis.*  
Merrill, New York: 1991. ISBN 0-02-946539-7. TUE-code: LLK91KEO bse.
- [16] W. Banzhof.  
*Computer aided circuit analysis using Spice.*  
Prentice-Hall, Englewood Cliffs. ISBN 0-13-162579-9. TUE-code: LLK89BAN.
- [17] s.n..  
*Linear and interface circuits. Product Applications, volume 1.*  
Texas Instruments databook: Opamps, Comparators and Timers, and Voltage Regulators, 1986. TUE-code: LMD85PIP bse.

## B. Materiaalgegevens Nikkel

In deze bijlage staan de karakteristieken van 99% zuiver Nikkel en gegevens van lucht en glas, zoals deze uit de literatuur zijn overgenomen. Voor zover deze gegevens niet beschikbaar waren bij de fabrikant van de smeltpatroon, zijn standaard tabellenboeken geraadpleegd.

In onderstaand overzicht is aangegeven uit welke referenties de data afkomstig is:

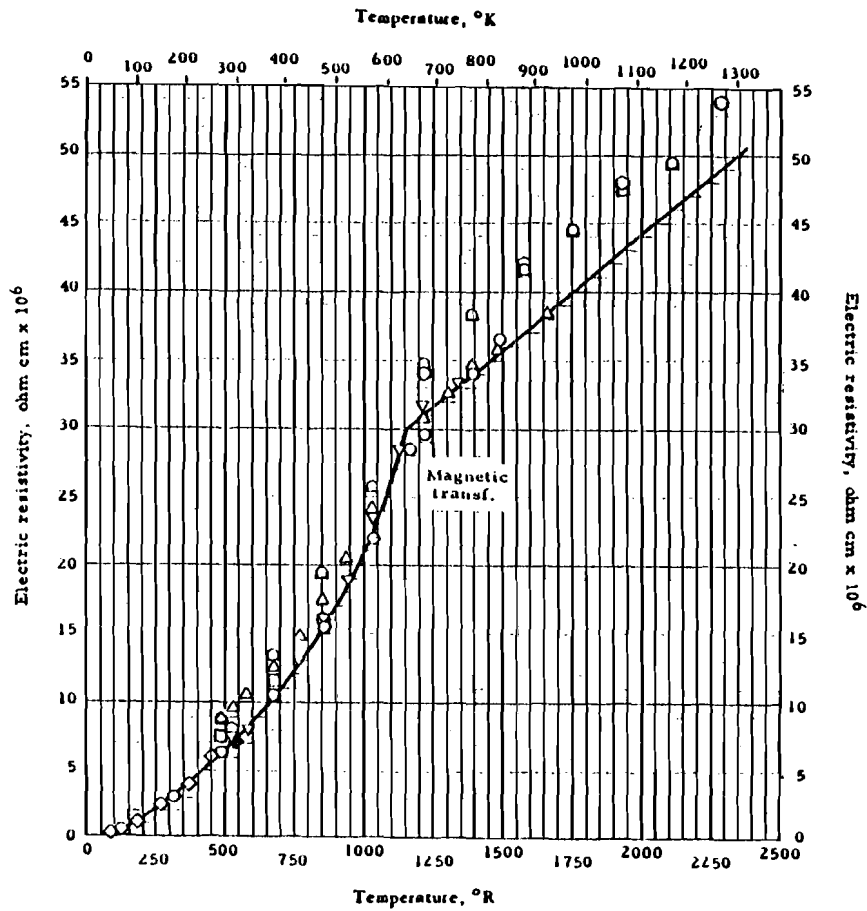
|                                              |      |
|----------------------------------------------|------|
| Soortelijke elektrische weerstand van Nikkel | [4]  |
| Soortelijke warmte van Nikkel                | [4]  |
| Thermische geleidbaarheid Nikkel             | [4]  |
| Eigenschappen van lucht                      | [9]  |
| Materiaalgegevens kwartsglas                 | [14] |

Bij de modellering zijn voor Nikkel de volgende constanten gebruikt:

|                       |                                                                        |
|-----------------------|------------------------------------------------------------------------|
| Soortelijke dichtheid | $8900 \frac{kg}{m^3}$                                                  |
| Smeltpunt             | $1728K = 1455^{\circ}C$                                                |
| Emissiviteit          | geoxideerd Nikkel: 0,41<br>glanzend Nikkel: 0,045<br>gemiddeld: 0,2275 |

Kenmerkend voor het materiaal nikkel (Ni) is het Curie-punt bij  $T=650K$ . Alle karakteristieken van materiaalgrootheden vormen vertonen in dat punt een knik. Tot deze temperatuur is nikkel ferromagnetisch, daarboven paramagnetisch.

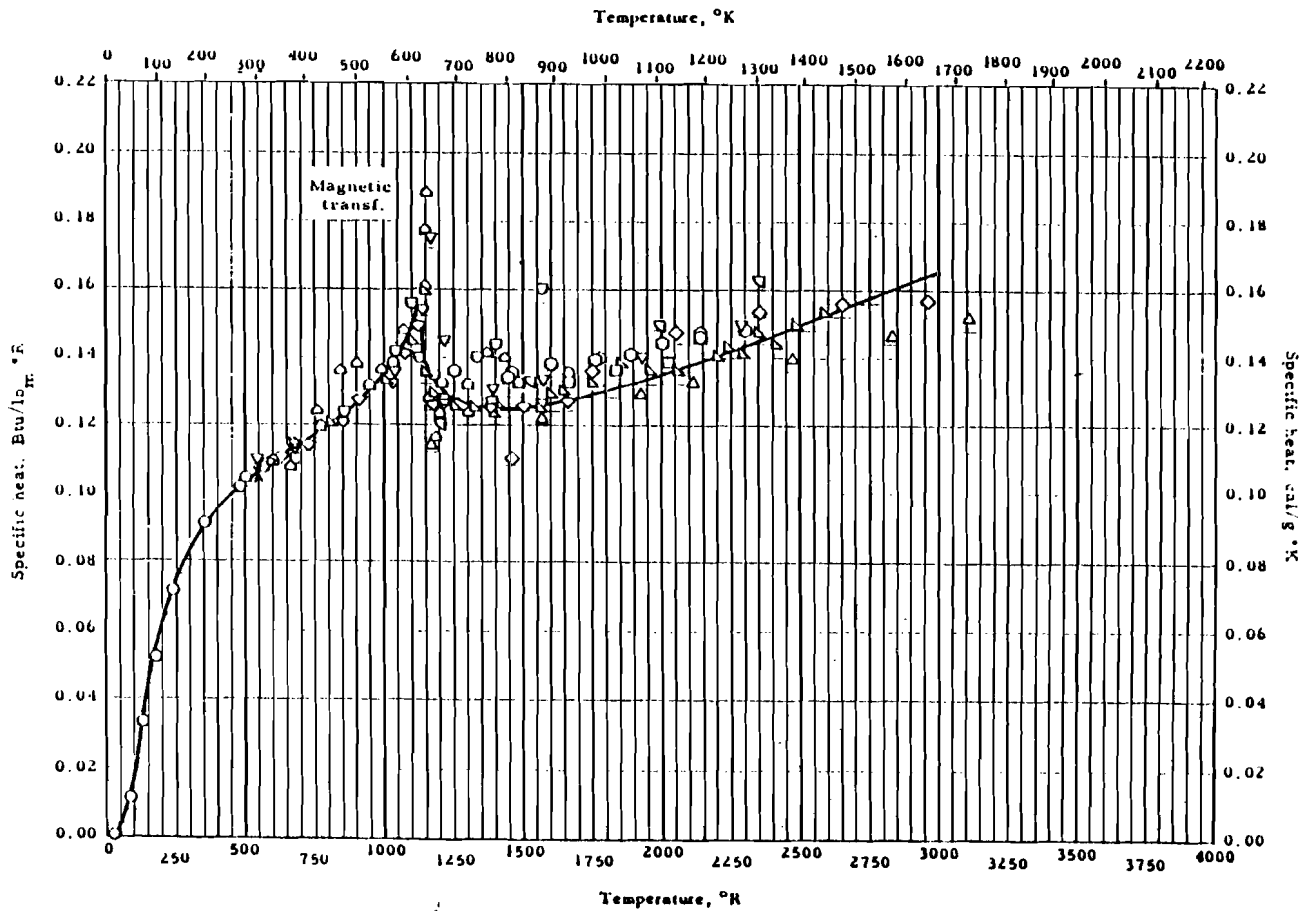
### B.1 Soortelijke elektrische weerstand van Nikkel $\rho$ .



| Soortelijke weerstand Ni $\rho$ |         |                                     |
|---------------------------------|---------|-------------------------------------|
| T [R]                           | T [K]   | $\rho(T)$ [ $\Omega m \cdot 10^8$ ] |
| 550                             | 305,556 | 7,8                                 |
| 600                             | 333,333 | 8,9                                 |
| 700                             | 388,889 | 11,0                                |
| 800                             | 444,444 | 13,8                                |
| 900                             | 500,000 | 17,0                                |
| 1000                            | 555,556 | 20,9                                |
| 1100                            | 611,111 | 26,0                                |

| Soortelijke weerstand Ni $\rho$ |          |                                     |
|---------------------------------|----------|-------------------------------------|
| T [R]                           | T [K]    | $\rho(T)$ [ $\Omega m \cdot 10^8$ ] |
| 1150                            | 638,889  | 30,0                                |
| 1200                            | 666,667  | 31,0                                |
| 1450                            | 805,556  | 35,0                                |
| 2000                            | 1111,111 | 44,0                                |
| 2350                            | 1305,556 | 50,0                                |
| 3200                            | 1777,778 | 64,6                                |

## B.2 Soortelijke warmte Nikkel $c$ .



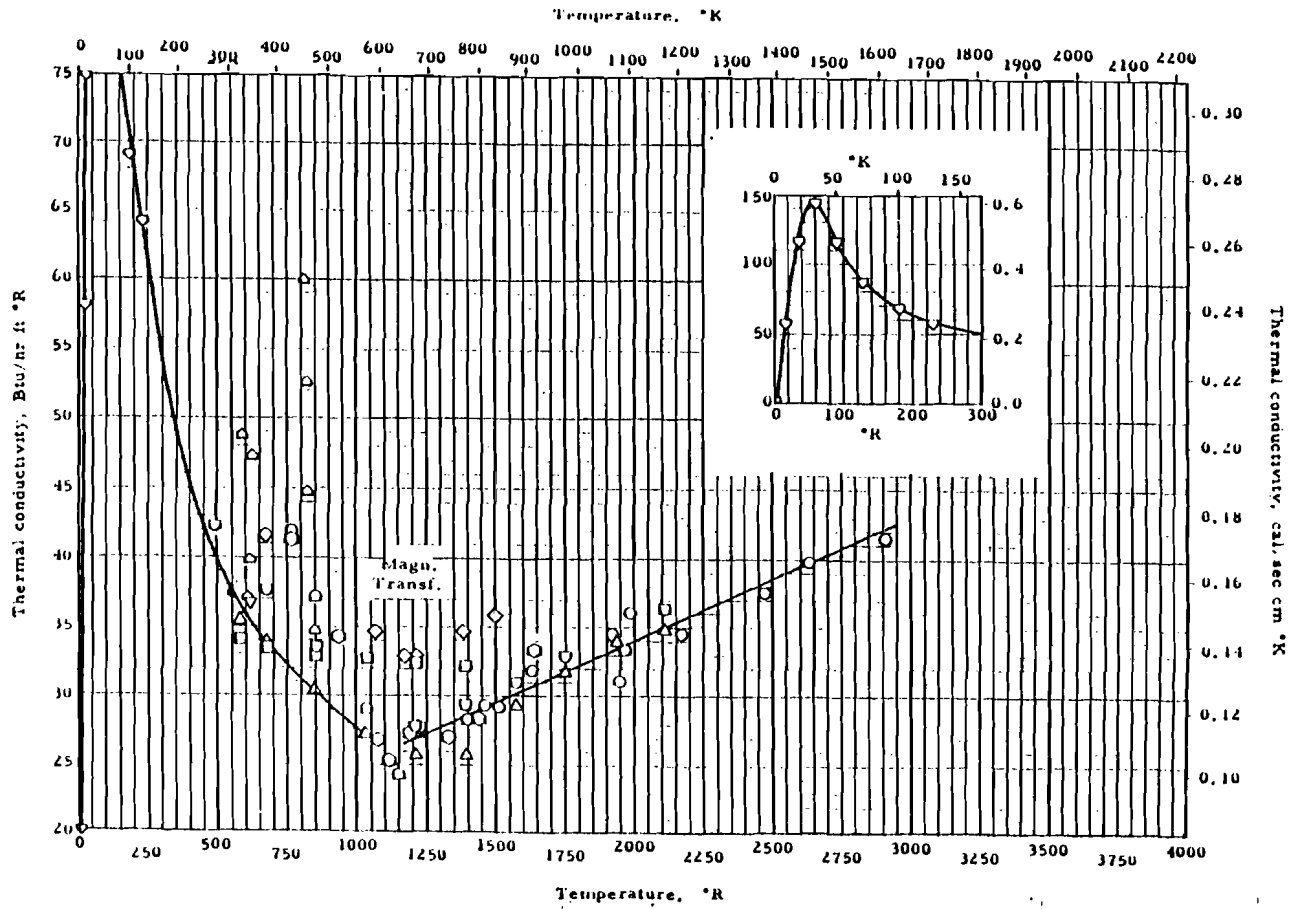
| Soortelijke warmte Ni $c$ |         |                                     |                                   |
|---------------------------|---------|-------------------------------------|-----------------------------------|
| T [R]                     | T [K]   | $c(T)$ [ $\frac{Btu}{lb \cdot R}$ ] | $c(T)$ [ $\frac{J}{kg \cdot K}$ ] |
| 550                       | 305,556 | 0,1064                              | 445,178                           |
| 650                       | 361,111 | 0,1120                              | 468,608                           |
| 700                       | 388,889 | 0,1152                              | 481,997                           |
| 750                       | 416,667 | 0,1168                              | 488,691                           |
| 800                       | 444,444 | 0,1200                              | 502,080                           |
| 850                       | 472,222 | 0,1228                              | 513,795                           |
| 900                       | 500,000 | 0,1268                              | 530,531                           |
| 950                       | 527,778 | 0,1304                              | 545,594                           |

| Soortelijke warmte Ni $c$ |         |                                     |                                   |
|---------------------------|---------|-------------------------------------|-----------------------------------|
| T [R]                     | T [K]   | $c(T)$ [ $\frac{Btu}{lb \cdot R}$ ] | $c(T)$ [ $\frac{J}{kg \cdot K}$ ] |
| 1000                      | 555,556 | 0,1360                              | 569,024                           |
| 1050                      | 583,333 | 0,1408                              | 589,107                           |
| 1100                      | 611,111 | 0,1480                              | 619,232                           |
| 1150                      | 638,889 | 0,1560                              | 652,704                           |
| 1200                      | 666,667 | 0,1320                              | 552,288                           |
| 1300                      | 722,222 | 0,1260                              | 527,184                           |
| 1400                      | 777,778 | 0,1240                              | 518,816                           |
| 1500                      | 833,333 | 0,1248                              | 522,163                           |

NB:  $1R = 1,8K$ ;  $1 \frac{Btu}{lb_m \cdot R} = 4184 \frac{J}{kg \cdot K}$  (zie volgend blad voor vervolg)

| Soortelijke warmte Ni $c$ |         |                               |                            |
|---------------------------|---------|-------------------------------|----------------------------|
| T [R]                     | T [K]   | $c(T)$ [ $\frac{Btu}{lb.R}$ ] | $c(T)$ [ $\frac{J}{kgK}$ ] |
| 1600                      | 888,889 | 0,1264                        | 528,858                    |
| 1700                      | 944,444 | 0,1280                        | 535,552                    |
| 1800                      | 1000,00 | 0,130                         | 543,920                    |
| 1900                      | 1055,56 | 0,132                         | 552,288                    |
| 2000                      | 1111,11 | 0,135                         | 565,677                    |
| 2100                      | 1166,67 | 0,137                         | 574,045                    |
| 2200                      | 1222,22 | 0,140                         | 585,760                    |
| 2300                      | 1277,78 | 0,144                         | 600,822                    |
| 2400                      | 1333,33 | 0,146                         | 609,190                    |
| 2500                      | 1388,89 | 0,149                         | 622,579                    |
| 2600                      | 1444,44 | 0,152                         | 637,642                    |
| 2700                      | 1500,00 | 0,156                         | 652,704                    |
| 2800                      | 1555,56 | 0,159                         | 666,093                    |
| 3450                      | 1916,67 | 0,180                         | 753,120                    |

### B.3 Thermische geleidbaarheid Nikkel $\lambda$ .



| Thermische geleiding Ni $\lambda$ |         |                                    |                             | Thermische geleiding Ni $\lambda$ |          |                                    |                             |
|-----------------------------------|---------|------------------------------------|-----------------------------|-----------------------------------|----------|------------------------------------|-----------------------------|
| T [R]                             | T [K]   | $\lambda(T) [\frac{Btu}{hr.ft.R}]$ | $\lambda(T) [\frac{W}{mK}]$ | T [R]                             | T [K]    | $\lambda(T) [\frac{Btu}{hr.ft.R}]$ | $\lambda(T) [\frac{W}{mK}]$ |
| 550                               | 305,556 | 37,8                               | 65,42                       | 1200                              | 666,667  | 27,0                               | 46,73                       |
| 600                               | 333,333 | 36,0                               | 62,31                       | 1550                              | 861,111  | 30,0                               | 51,92                       |
| 700                               | 388,889 | 33,0                               | 57,11                       | 2000                              | 1111,111 | 34,0                               | 58,84                       |
| 800                               | 444,444 | 31,0                               | 53,65                       | 2650                              | 1472,222 | 40,0                               | 69,23                       |
| 900                               | 500,000 | 29,2                               | 50,54                       | 2900                              | 1611,111 | 42,0                               | 72,69                       |
| 1000                              | 555,556 | 27,8                               | 48,11                       | 3300                              | 1833,333 | 46,0                               | 79,61                       |
| 1100                              | 611,111 | 26,0                               | 45,00                       |                                   |          |                                    |                             |

NB:  $1R = 1,8K$ ;  $1 \frac{Btu}{hr.ft.R} = 1,73073 \frac{W}{m.K}$



## B.4 Eigenschappen lucht.

| T<br>[K] | $\gamma$<br>[ $\frac{kg}{m^3}$ ] | c<br>[ $\frac{kJ}{kgK}$ ] | $\mu \times 10^5$<br>[ $\frac{kg}{ms}$ ] | $\nu \times 10^6$<br>[ $\frac{m^2}{s}$ ] | $\lambda(T)$<br>[ $\frac{W}{mK}$ ] | Pr<br>[-] |
|----------|----------------------------------|---------------------------|------------------------------------------|------------------------------------------|------------------------------------|-----------|
| 300      | 1,1774                           | 1,0057                    | 1,8462                                   | 15,69                                    | 0,02624                            | 0,708     |
| 350      | 0,9980                           | 1,0090                    | 2,075                                    | 20,76                                    | 0,03003                            | 0,697     |
| 400      | 0,8826                           | 1,0140                    | 2,286                                    | 25,90                                    | 0,03365                            | 0,689     |
| 450      | 0,7833                           | 1,0207                    | 2,484                                    | 31,71                                    | 0,03707                            | 0,683     |
| 500      | 0,7048                           | 1,0295                    | 2,671                                    | 37,90                                    | 0,04038                            | 0,680     |
| 550      | 0,6423                           | 1,0392                    | 2,848                                    | 44,34                                    | 0,04360                            | 0,680     |
| 600      | 0,5879                           | 1,0551                    | 3,018                                    | 51,34                                    | 0,04659                            | 0,680     |
| 650      | 0,5430                           | 1,0635                    | 3,177                                    | 58,51                                    | 0,04953                            | 0,682     |
| 700      | 0,5030                           | 1,0752                    | 3,332                                    | 66,25                                    | 0,05230                            | 0,684     |
| 750      | 0,4709                           | 1,0856                    | 3,481                                    | 73,91                                    | 0,05509                            | 0,686     |
| 800      | 0,4405                           | 1,0978                    | 3,625                                    | 82,29                                    | 0,05779                            | 0,689     |
| 850      | 0,4149                           | 1,1095                    | 3,765                                    | 90,75                                    | 0,06028                            | 0,692     |
| 900      | 0,3925                           | 1,1212                    | 3,899                                    | 99,3                                     | 0,06279                            | 0,696     |
| 950      | 0,3716                           | 1,1321                    | 4,023                                    | 108,2                                    | 0,06525                            | 0,699     |
| 1000     | 0,3524                           | 1,1417                    | 4,152                                    | 117,8                                    | 0,06752                            | 0,702     |
| 1100     | 0,3204                           | 1,160                     | 4,44                                     | 138,6                                    | 0,0732                             | 0,704     |
| 1200     | 0,2947                           | 1,179                     | 4,69                                     | 159,1                                    | 0,0782                             | 0,707     |
| 1300     | 0,2707                           | 1,197                     | 4,93                                     | 182,1                                    | 0,0837                             | 0,705     |
| 1400     | 0,2515                           | 1,214                     | 5,17                                     | 205,5                                    | 0,0891                             | 0,705     |
| 1500     | 0,2355                           | 1,230                     | 5,40                                     | 229,1                                    | 0,0946                             | 0,705     |
| 1600     | 0,2211                           | 1,248                     | 5,63                                     | 254,5                                    | 0,100                              | 0,705     |
| 1700     | 0,2082                           | 1,267                     | 5,85                                     | 280,5                                    | 0,105                              | 0,705     |
| 1800     | 0,1970                           | 1,287                     | 6,07                                     | 308,1                                    | 0,111                              | 0,704     |

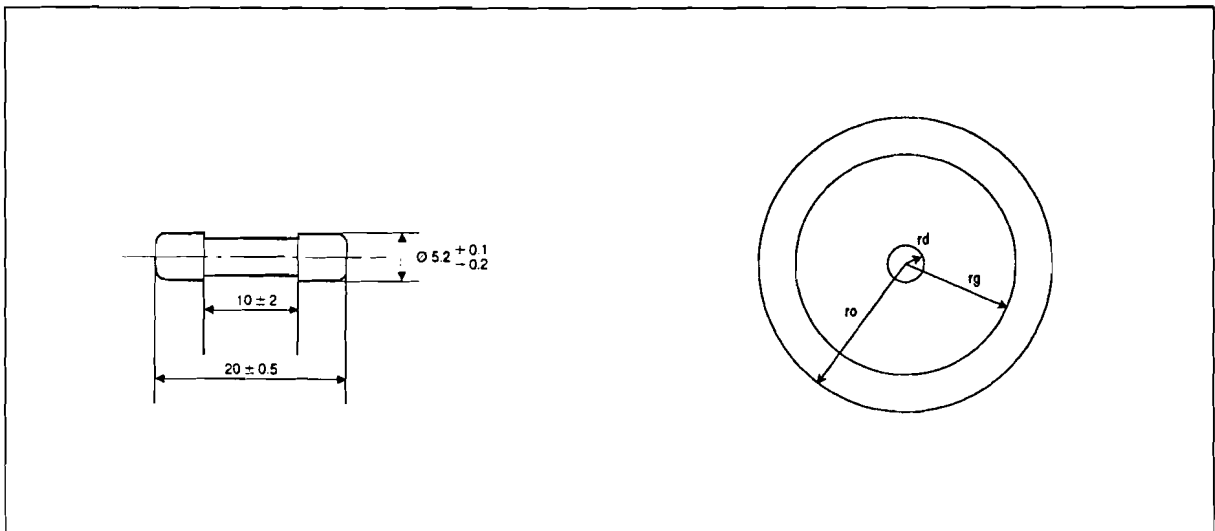
## B.5 Materiaalgegevens glas.

De getallen zijn gebaseerd op richtwaarden van kwartzglas.

|                           |                          |                  |      |
|---------------------------|--------------------------|------------------|------|
| dichtheid                 | $2000 < \gamma < 2200$   | $\frac{kg}{m^3}$ | [14] |
| thermische geleidbaarheid | $1.2 < \lambda < 2.7$    | $\frac{W}{mK}$   | [14] |
| soortelijke warmte        | $840 < c < 1130$         | $\frac{J}{kgK}$  | [14] |
| emissiviteit              | $0,85 < \epsilon < 0,95$ | -                | [9]  |

## C. Afmetingen patroon 217.315

Gegevens zijn afkomstig uit documentatie van Littelfuse Utrecht.



Figuur C.1: Afmetingen van miniatuurpatroon 217.315

|                        |       |                  |
|------------------------|-------|------------------|
| effectieve draadlengte | L     | 1,75 cm          |
| draadstraal            | $r_d$ | 25 $\mu\text{m}$ |
| binnenstraal glas      | $r_g$ | 1,5 mm           |
| buitenstraal glas      | $r_o$ | 2,25 mm          |

## D. PSPICE listing NI2.CIR

Deze bijlage bevat de PSPICE broncode voor het simuleren van het thermisch gedrag van een 217.315 type miniatuurpatroon.

Thermisch model van HALVE Nikkel fuse-draad (217.315)

```

* NI2.CIR
* J.P van Herel
* 11 augustus 1992

* ----- *
*           Algemene parameters voor de Nikkeldraad (217.315)           *
* ----- *

.PARAM n = 7           ; aantal segmenten van HELE draad

* ----- *
*           Opbouw helft van het n-segmenten model                       *
* ----- *

Xcap      1           Eindcap
Xblok1   1  2  3     21 22 23  999 Segment
Xblok2   3  4  5     41 42 43  999 Segment
Xblok3   5  6  7     61 62 63  999 Segment
Xblok4   7  8           81 82 83  999 Segmenthalf

Xstroom  999           Stroosignaal_puls

* ----- *
*           definities van blackboxes                                     *
* ----- *

* =S T R O O M S I G N A A L Idc===== *
*
*           1           V(1) is het signaal van i(t)
*           + |
*           ( ) Estroom
*           |
*           ^

.SUBCKT Stroosignaal_dc 1
Vdc      2  0  PWL ( 0,0 2ns,756E-3 1000s,756E-3)
Rdc      2  0  1MEG
Vstep    3  0  1V

```

```

Rstep      3  0  1MEG
Estroom    1  0  VALUE = { V(2) * V(3) }
Rstroom    1  0  1MEG
.ENDS Stroosignaal_dc

```

```
* =S T R O O M S I G N A A L Ipuls===== *
```

```

*
*      1          V(1) is het signaal van i(t)
*      + |
*      ( ) Estroom
*      |
*      ~

```

```

.PARAM frequentie      89.4772
.PARAM amplitude       8.0851
.PARAM demping         -204.5455

```

```

.SUBCKT Stroosignaal_puls 1
Vsendemp 2 0 SIN( 0V {amplitude} {frequentie} 0V {-demping} )
Rsendemp 2 0 1MEG
Vstep     3 0 PULSE( 1V 0V {1/(2*frequentie)-1E-6} 1E-6 1E-9 )
Rstep     3 0 1MEG
Estroom   1 0 VALUE = { V(2) * V(3) }
Rstroom   1 0 1MEG
.ENDS Stroosignaal_puls

```

```
* =S E G M E N T===== *
```

```

*
*      Rth_Ni          Rth_Ni
*      1 ---XXXX--- 2 ---XXXX--- 3          7 --<-- i(t)
*
*      |
*      .-----|-----|-----|-----|
*      |      |      |      |
*      |      |CNi X      X 1/2 Rconv
*      Iinj (^) === X Rrad 4-----|
*      |      |      X      X 1/2 Rconv |
*      |      |      |      |          === Clucht
*      |      |      |      |      5      |
*      |      |      |      |      |      |
*      |      | 1/2 Rth_glas X      |      |
*      |      |      |      |      |      |
*      |      |      |      |      6---|      |
*      |      |      |      |      |      |
*      |      | 1/2 Rth_glas X === Cglas |
*      |      |      |      |      |      |
*      ~      ~      ~      ~      ~      ~

```

```
.SUBCKT Segment 1 2 3 4 5 6 7
Xpower 2 7      Iinj      PARAMS: factor = 1
XCNi    2        Nicap     PARAMS: factor = 1
Xconv1  2 4 2    Rconvectie PARAMS: factor = 0.5
Xconv2  4 5 2    Rconvectie PARAMS: factor = 0.5
XClucht 4        Luchtcap  PARAMS: factor = 1
XRad    2 0      Rrad      PARAMS: factor = 1
XCglas  6        Glascap   PARAMS: factor = 1
XRglas1 5 6      Rth_glas_rad PARAMS: factor = 0.5
XRglas2 6 0      Rth_glas_rad PARAMS: factor = 0.5
Xgel_l  2 1      Rth_Ni_ax
Xgel_r  3 2      Rth_Ni_ax
.ENDS Segment
```

```
.SUBCKT Segmenthalf 1 2 4 5 6 7
Xpower 2 7      Iinj      PARAMS: factor = 0.5
XCNi    2        Nicap     PARAMS: factor = 0.5
Xconv1  2 4 2    Rconvectie PARAMS: factor = 1
Xconv2  4 5 2    Rconvectie PARAMS: factor = 1
XClucht 4        Luchtcap  PARAMS: factor = 0.5
XRad    2 0      Rrad      PARAMS: factor = 2
XCglas  6        Glascap   PARAMS: factor = 0.5
XRglas1 5 6      Rth_glas_rad PARAMS: factor = 1
XRglas2 6 0      Rth_glas_rad PARAMS: factor = 1
Xgel_l  2 1      Rth_Ni_ax
.ENDS Segmenthalf
```

```
* =E I N D C A P===== *
```

```
*
*      1
*      | +
* Vcap ( )
*      |
*      ^
```

```
.SUBCKT Eindcap 1
Vcap 1 0 0V
.ENDS Eindcap
```

```
* =C O N V E C T I E===== *
```

```
*
* 2 ----XXXX---- 1          3: meetklem voor draadtemperatuur
* (+) Rconv (-)
*
```

```
.PARAM Rconv_300 = 12216.3 ; Rconvectie bij 300K, n=7
```

```

.SUBCKT Rconvectie 2 1 3 PARAMS: factor = 1
R0      2 4 { factor * Rconv_300 * (n/7) }
Vstroom 4 5 0V
ERth    5 1 VALUE { factor * V(6)*I(Vstroom) * (n/7) }
EGraph  6 0 TABLE { V(3) } =
*   dT      dRconv
+   0        0.0
+   25      -1468.83
+   75      -2589.71      ; tabel geeft verandering in Rconv aan tov 300K
+   175     -3972.04
+   275     -4918.65      ; dT is temperatuurverschil met 300K
+   425     -5937.25      ; dRconv is weerstandsverschil met Rconv_300
+   625     -6861.17
+   925     -7773.54
+  1275     -8461.06
+  1475     -8758.52
Rbel    6 0 1MEG
.ENDS Rconvectie

* =T H E R M I S C H E   G E L E I D I N G   a x i a l===== *
*
*   2 -----XXXX----- 1
*   (+)      Rth_Ni_ax   (-)
*
.PARAM Rth_Ni_ax_300 = 9731.03; Rth_Ni_ax voor half segment bij n=7

.SUBCKT Rth_Ni_ax 2 1
R0      2 3 { Rth_Ni_ax_300 * (7/n) }
Vstroom 3 4 0V
ERth    4 1 VALUE { V(5)*I(Vstroom) * (7/n) }
EdeltaR 5 0 TABLE { V(2,1) } =
*   dT      dRth,ax
+   0.0      0.0
+   27.7778  486.552
+   83.3333  1415.42
+   138.889  2134.55
+   194.444  2865.99
+   250.000  3500.37      ; tabel geeft verandering in Rth_Ni_ax aan tov 300K
+   305.556  4416.39
+   361.111  3892.41      ; dT is temperatuurverschil met 300K
+   555.556  2530.07      ; dRth is weerstandsverschil met Rth_Ni_ax_300
+   805.556  1087.59
+  1166.67   -535.207
+  1305.56   -973.101
+  1527.78   -1734.66
Rbel    5 0 1MEG

```

```
.ENDS Rth_Ni_ax

* =E L E C T R I S C H   V E R M O G E N===== *
*
*      1  2          V(1) is temperatuursignaal van segment
*      |  |
* Iinj  (^)          V(2) is i(t) signaal
*      |
*      ^

.SUBCKT Iinj 1 2 PARAMS: factor=1
  Ginj  0 1  VALUE = { factor * pwr(V(2),2) * V(7)*1.2E-3 * (7/n) }
  Xrho  7 1 0  Rel_tabel
.ENDS Iinj

.SUBCKT Rel_tabel 1 2 3
  Erho_tabel 1 0 TABLE { V(2,3) } =
  * dT [K]  Rel [mohm]
  +  0.0    99.3127
  + 27.78   113.318
  + 83.33   140.056;      Tabel geeft de weerstand van Nikkel
  + 138.89  175.707;      zoals deze uit de materiaalgrafiek is genomen.
  + 194.44  216.451
  + 250.00  266.107;      dT is de temperatuurtoename tov 300K
  + 305.56  331.042
  + 333.33  381.972
  + 361.11  394.704
  + 500.00  445.634
  + 805.56  560.225
  +1000.00  636.620
  +1472.22  822.513
  Rrel  1 0 1MEG
.ENDS Rel_tabel

* =W A R M T E O P S L A G   Nikkel===== *
*
*  1 ----||---- 0
*  (+)  Cni  (-)

.PARAM NiCap_300  = 19.4488E-6 ; equivalent warmtecapaciteit bij 300K (n=7)

.SUBCKT Nicap 1 PARAMS: factor = 1
  Eref  2 0  VALUE = { V(1,0) }
  C0    2 3  1
  Vsens 3 0  0V
  C300  1 0  { factor * NiCap_300 * (7/n) }
```



```

Gout  1 0  VALUE = { factor * I(Vsens) * V(4)*1E-6 * (7/n) }
Etable 4 0  TABLE { V(1) } =
*  V(1)  dC [uF]
+  0.0  0.0
+  83.33  1.60855
+  166.667  2.99775
+  222.222  4.38695
+  305.556  7.60405
+  333.333  9.06637
+  361.111  4.67941
+  416.667  3.58268
+  472.222  3.21710
+  527.222  3.36333
+  694.444  4.31384
+  1027.78  7.16535
+  1611.11  13.4533
Rtable 4 0 1MEG
.ENDS Nicap

* =W A R M T E O P S L A G  lucht===== *
*
*  1 ----||---- 0
*  (+) Clucht (-)

.PARAM LuchtCap_1500 = 4.47916E-6 ; warmtecapaciteit bij 1500K, n=7

.SUBCKT Luchtcap 1 PARAMS: factor = 1
Eref  2 0  VALUE = { V(1,0) }
C0  2 3  1
Vsens 3 0  0V
C1500 1 0  { factor * LuchtCap_1500 * (7/n) }
Gout  1 0  VALUE = { factor * I(Vsens) * V(4)*1E-6 * (7/n) }
Etable 4 0  TABLE { V(1,0)-1500 } =
*  V(1)  dC [uF]
+  -1500  16.4400
+  -1450  13.3107
+  -1350  9.6455
+  -1200  6.4793
+  -1000  4.0641
+  -650  1.8669
+  -350  0.7718
+  0  0.0
Rtable 4 0 1MEG
.ENDS Luchtcap

* =S T R A L I N G S W E E R S T A N D===== *

```

```

*
*      2 ----XXXX---- 1
*      (+)   Rrad   (-)
*
.PARAM Rrad_300 = 1.828E6 ; stralingsweerstand bij 300K, n=7

.SUBCKT Rrad 2 1 PARAMS: factor = 1
R0      2 3 {factor * Rrad_300 * (n/7) }
Vstroom 3 4 0V
ERth    4 1 VALUE { factor * V(5)*1E3*I(Vstroom) * (n/7) }
EGraph  5 0 TABLE { V(2,1) } =
*   dT      dR_rad [kOhm]
+   0        0
+  100      -699.824
+  200      -1102.11
+  350      -1422.43
+  550      -1616.62
+  750      -1705.27
+ 1150      -1776.45
+ 1400      -1794.77
Rbel    5 0 1MEG
.ENDS Rrad

* =W A R M T E O P S L A G  glas===== *
*
*   1 ----||---- 0
*   (+)   Cglas  (-)
*
.PARAM GlasCap_300 = 49.39E-3;   warmtecapaciteit bij 300K, n=7

.SUBCKT Glascap 1 PARAMS: factor = 1
Eref    2 0 VALUE = { V(1,0) }
C0      2 3 1
Vsens   3 0 0V
C300    1 0 { factor * GlasCap_300 * (7/n) }
Gout    1 0 VALUE = { factor * I(Vsens) * V(4) * (7/n) }
Etable  4 0 TABLE { V(1,0) } =
*   V(1)  dC [uF]
+   0.0   0
+  1700   0
Rtable  4 0 1MEG
.ENDS Glascap

* =T H E R M I S C H E   G E L E I D I N G  glas radiaal===== *
*
*   2 ----XXXX---- 1

```

```

* (+) Rth_glas (-)
*
.PARAM Rth_glas_rad300 = 13.237 ; Rth van glas voor segment bij n=7

.SUBCKT Rth_glas_rad 2 1 PARAMS: factor = 1
R0      2 3 {factor * Rth_glas_rad300 * (n/7) }
Vstroom 3 4
ERth    4 1 VALUE { V(5)*I(Vstroom) * (n/7) * factor}
EGraph  5 0 TABLE { V(2,1) } =
*  dT    dRth_glas_rad
+   0      0
+  1700    0
Rbel    5 0 1MEG
.ENDS Rth_glas_rad

* =T O T A A L W E E R S T A N D   D R A A D===== *
*
* Berekenen van elektrische weerstand totale draad
*
* V(207) is maat equivalent van elektrische weerstand draad

XRsegment1  201    0    (2,1)  SegmentR
XRsegment2  202   201   (3,2)  SegmentR
XRsegment3  203   202   (4,3)  SegmentR
XRsegment4  204   203   (5,4)  SegmentR
XRsegment5  205   204   (6,5)  SegmentR
XRsegment6  206   205   (7,6)  SegmentR
XRsegment7  207   206   (8,7)  SegmentR
Rbelastin   207    0                1MEG

.SUBCKT SegmentR 1 2 3 4
EsegmentR   1 2    VALUE = { 2 * (V(6)+V(5))/2E3 * 2 }
XR_table_low 5 3 0  Rel_tabel
XR_table_hgh 6 4 0  Rel_tabel
.ENDS SegmentR

* ----- *
*                               Stuurcodes voor PSPICE simulatie                               *
* ----- *

.PROBE
.PRINT TRAN v(2) v(3) v(4) v(5) v(6) v(7) v(8)
.OPTIONS RELTOL=0.01 WIDTH=132
.TRAN 0.01ms 10ms 0ms 0.1ms
.END

```

## E. It-curven data

Onderstaande lijst bevat coördinaatpunten uit de It-curven (hoog en laag) voor een 217.315 miniatuurpatroon, zoals deze door Littelfuse Nederland beschikbaar gesteld zijn.

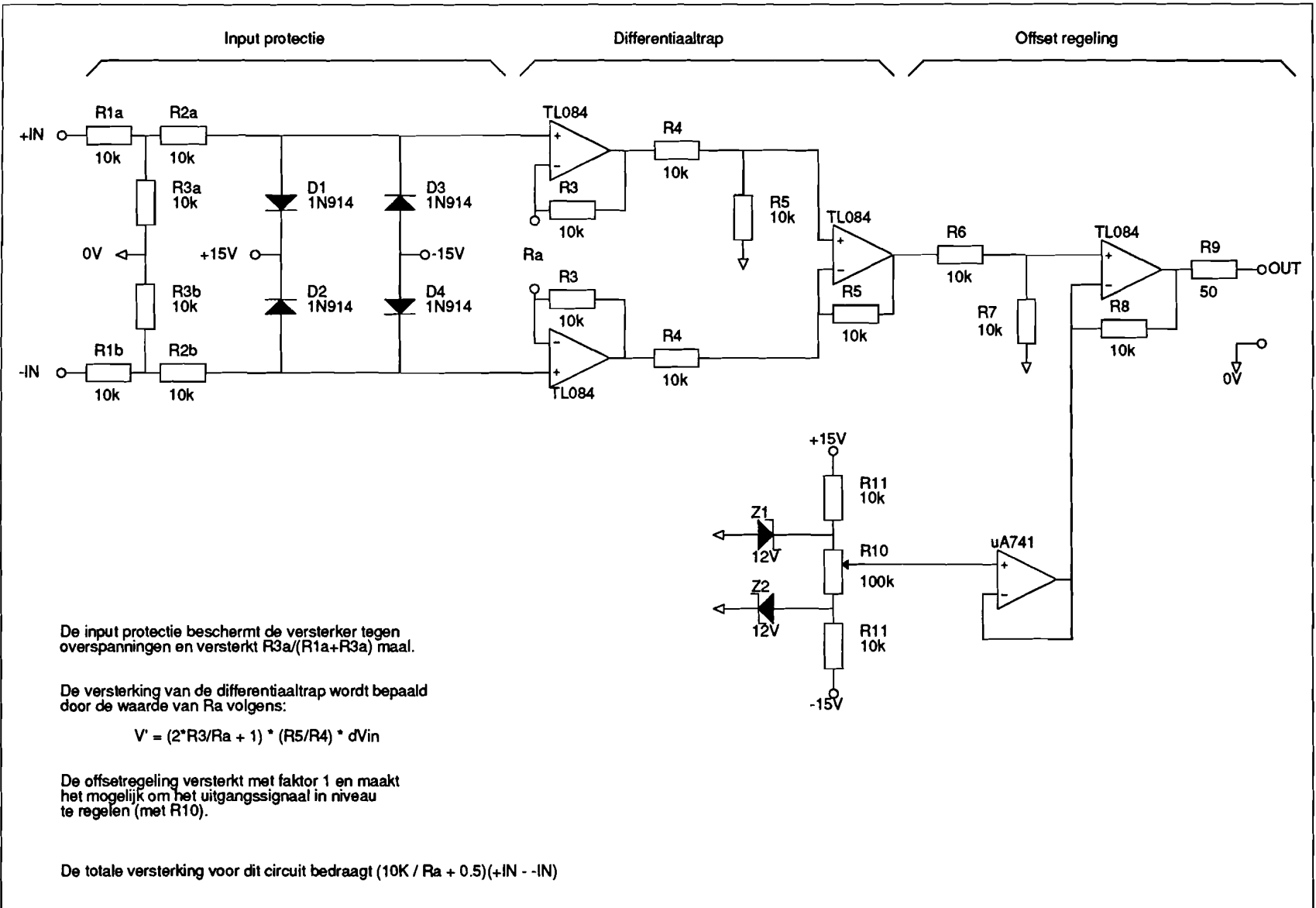
| $I / I_n$ | $t_L[ms]$ | $t_H[ms]$ |
|-----------|-----------|-----------|
| 1,7       | 43972     | $\infty$  |
| 1,8       | 9,673     | $\infty$  |
| 1,9       | 3,027     | $\infty$  |
| 2         | 1,223     | $\infty$  |
| 2,1       | 844       | 15395     |
| 2,2       | 595       | 4289      |
| 2,3       | 472       | 1920      |
| 2,4       | 333       | 1342      |
| 2,5       | 270       | 947       |
| 2,6       | 235       | 669       |
| 2,7       | 191       | 530       |
| 2,8       | 176       | 420       |
| 2,9       | 148       | 333       |
| 3         | 139       | 297       |
| 4         | 60        | 117       |
| 5         | 37        | 65        |
| 6         | 24        | 42        |
| 7         | 18        | 33        |
| 8         | 13        | 26        |
| 9         | 11        | 18        |
| 10        | 8         | 14        |

## Bijlagen van deel 2

### F. Schema differentiaalversterker

Op de volgende bladzijde staat het schema getekend van de differentiaalversterker, zoals deze in het meetcircuit is opgenomen.

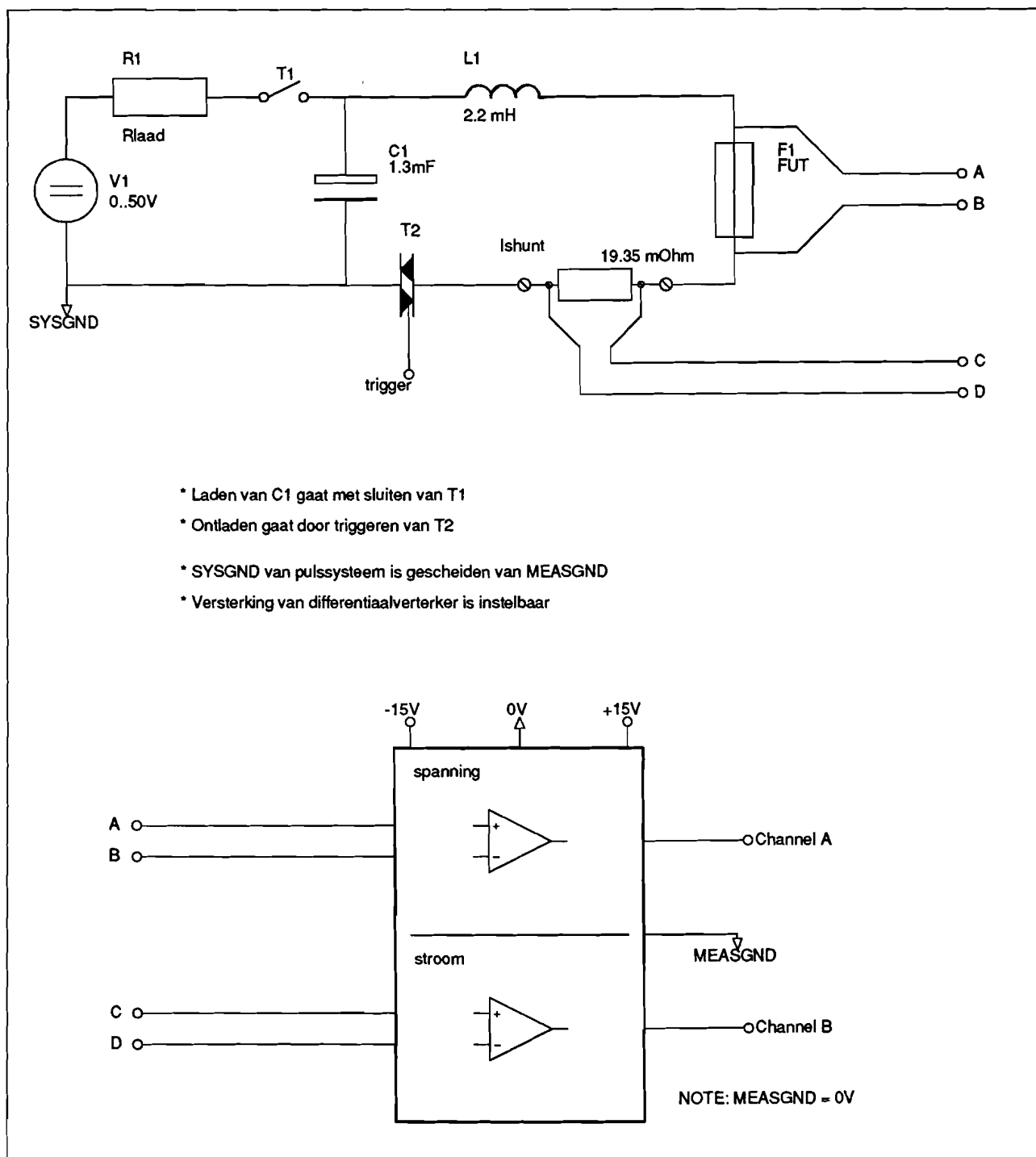
Voor een beschrijving wordt verwezen naar deel 2 van dit verslag.



Figuur F.1: Schets van één enkele differentiaalversterker.

## G. Koppelen meetcircuit

Op de volgende bladzijde staat getekend hoe het meetcircuit gekoppeld wordt aan het pulscircuit.



Figuur G.1: Schets van het pulscircuit met daaraan het meetcircuit.



## H. BAKKER formaat

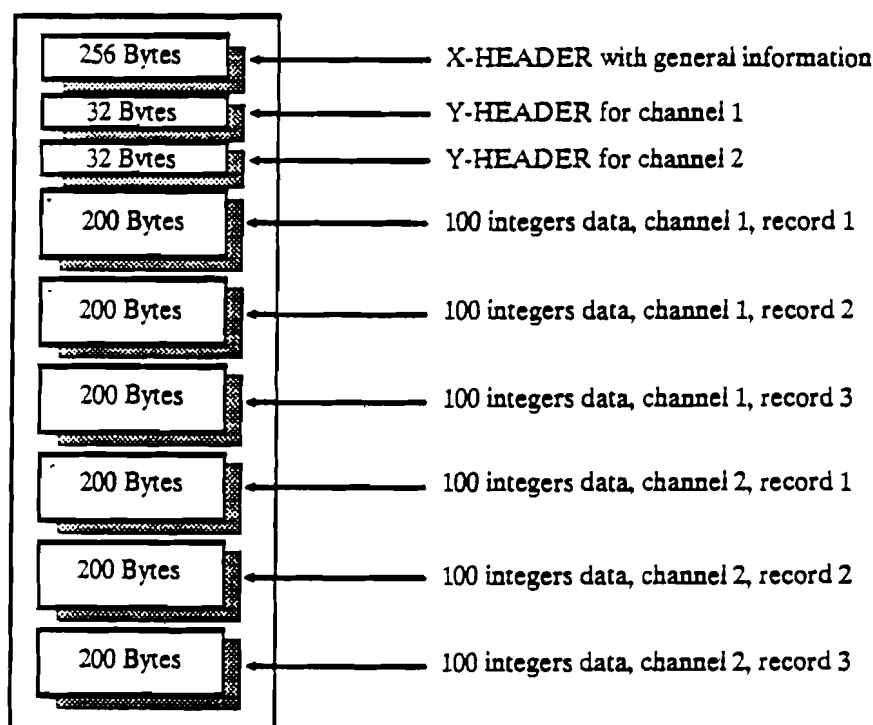
Overzicht van opslagformaat BAKKER datafiles. Het overzicht is afkomstig uit de handleiding van de BAKKER documentatie (B-1 t/m B4).

### Data file layout and size

A data file consists of an X-HEADER, followed by an Y-HEADER for each stored channel, followed by a number of data records. The data records can contain integers (16 bit) or IEEE-754 standard floating point numbers (32 bit).

Raw data from the analog input is always stored as integers. Processed data is always stored in floating point format for maximum precision and dynamic range.

As an example the layout of a file containing 2 channels with 3 records per channel, 100 (integer) samples stored per record, is depicted below:



The file size can be calculated from the following formula:

File size =  $256 + NCHAN \cdot 32 + NCHAN \cdot NREC \cdot NSAMP \cdot SAMPSIZE$  bytes

With:

- NCHAN = number of stored channels.
- NREC = number of records per channel.
- NSAMP = number of stored samples per record.
- SAMPSIZE = 2 for integers, 4 for IEEE floating point numbers.

**The header information**

The X-HEADER and Y-HEADER(s) contain information about the stored data. The X-HEADER is 256 bytes long. Each Y-HEADER is 32 bytes long.

**Variable types used**

INT: integer (16 bit);  
 LONG: long integer (32 bit);  
 FLOAT: single precision IEEE-754 floating point number (32 bit);  
 DOUBLE: double precision IEEE-754 floating point number (64 bit);  
 CHAR[n]: a string of n-1 characters (each character is one byte) terminated by 0;

These variable types are the same as used by the 8087 co-processor in an IBM personal computer or compatible. Storage of a multibyte variable is in the order least significant byte first to most significant byte last (Intel format).

**Note**

The BASIC language as implemented on IBM personal computers and compatibles use a special internal floating point format which is incompatible with the IEEE-754 floating point standard. Therefore BASIC is not recommended for processing the datafiles. High level language compilers like "Lattice C" conform however to the IEEE-754 floating point format.

**The layout of the X-HEADER.**

The X-HEADER starts at file location 0 (offset = 0).

| Byte #   | Format    | Contents        |
|----------|-----------|-----------------|
| 0 - 1    | (INT)     | HEADER-VERSION  |
| 2 - 3    | (INT)     | NCHAN           |
| 4 - 5    | (INT)     | NREC            |
| 6 - 9    | (LONG)    | LENGTH          |
| 10 - 13  | (LONG)    | NSKIP           |
| 14 - 17  | (LONG)    | NSAMP           |
| 18 - 19  | (INT)     | SAMPTYPE        |
| 20 - 27  | (CHAR[8]) | XUNITS          |
| 28 - 35  | (DOUBLE)  | X0              |
| 36 - 43  | (DOUBLE)  | XSTEP           |
| 44 - 49  | (CHAR[6]) | DATE-INFO       |
| 50 - 255 |           | <i>Reserved</i> |

**The layout of the Y-HEADER.**

The Y-HEADER of channel "i" starts at file location  $256 + (i-1)*32$

| Byte # | Format    | Contents        |
|--------|-----------|-----------------|
| 0-6    | (CHAR[7]) | NAME            |
| 7-13   | (CHAR[7]) | YUNITS          |
| 14-17  | (FLOAT)   | Y0              |
| 18-21  | (FLOAT)   | YSTEP           |
| 22-25  | (FLOAT)   | MAX             |
| 26-29  | (FLOAT)   | MIN             |
| 30-31  |           | <i>Reserved</i> |

Explanation of header entries.

The various entries in the headers fully describe the data contained in the rest of the file. The meaning of each entry will now be explained in more detail.

X-HEADER entries

|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| HEADER-VERSION | This <i>integer</i> must be 1. Otherwise the file is not a 2570 datafile.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| NCHAN          | This <i>integer</i> defines the number of stored channels in the file.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| NREC           | This <i>integer</i> defines the number of stored records per channel in the file.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| LENGTH         | This <i>long integer</i> represents the original record length. When for instance from a 4K recording only samples 101 to 300 were stored, this value would be 4096 (= 4K).                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| NSKIP          | This <i>long integer</i> is set to the number of samples skipped in the original record during data storage. When for instance from a 4K recording only samples 101 to 300 were stored, this value would be 100 indicating that the first 100 samples were skipped during storage.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| NSAMP          | This <i>long integer</i> determines the number of samples stored in each record in the file. When for instance from a 4K recording only samples 101 to 300 were stored, this value would be 200.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| SAMPTYPE       | This <i>integer</i> defines the type of samples stored in the records. <ul style="list-style-type: none"> <li>• SAMPTYPE = 2 : The data type of each sample is 16 bit integer.</li> <li>• SAMPTYPE = 4 : The data type of each sample is IEEE-754 single precision floating point number (32 bit).</li> <li>• SAMPTYPE = 5 : This indicates that complex data is stored in the file. In this case the number of channels is always 2. Channel 1 contains the real information, channel 2 the imaginary information. The data type of the samples is again single precision floating point.</li> <li>• SAMPTYPE = 6 : This also indicates the presence of complex data in the file. However, channel 1 now contains the magnitude, channel 2 the phase in radians of the complex data. Data type of the samples is again single precision floating point.</li> </ul> |
| XUNITS         | This <i>character string</i> contains the x-units information, for instance "seconds" or "Hz".                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| X0             | This <i>double precision floating point</i> value is the X-value (for instance time relative to the trigger point) corresponding with the first sample in the original (!) record from which the data has been stored. So the X-value of the first sample in the <i>stored</i> record is:<br>$X1\text{-value} = X0 + NSKIP * XSTEP.$                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| XSTEP          | This <i>double precision floating point</i> value defines the difference in X-value between two consecutive samples. With XSTEP, X0 and NSKIP the X-value of sample <i>i</i> in the stored record can be calculated by the formula:<br>$Xi\text{-value} = X0 + (NSKIP + i - 1) * XSTEP$                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| DATE-INFO      | This " <i>character string</i> " contains the date and time at which the recording was finished. The precision of this time is 1 second.<br>The info consists of 6 characters (NOT 5) which must be treated as 6 unsigned bytes (range 0-255). The numbers stored in these bytes have the following meaning:<br>Byte 1 <b>hours</b> in the range 0 to 23<br>Byte 2 <b>minutes</b> in the range 0 to 59<br>Byte 3 <b>seconds</b> in the range 0 to 59<br>Byte 4 <b>years</b> since 1980<br>Byte 5 <b>month</b> in the range 1 to 12<br>Byte 6 <b>day</b> in the range 1 to 31                                                                                                                                                                                                                                                                                        |

**Y-HEADER entries**

---

|               |                                                                                                                                                                                                                                                                                                                                             |
|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>NAME</b>   | This <i>character string</i> contains the channel name of the corresponding channel number. The character string is terminated by a 0.                                                                                                                                                                                                      |
| <b>YUNITS</b> | This <i>character string</i> contains the technical units of the data of the corresponding channel number. The character string is terminated by a 0.                                                                                                                                                                                       |
| <b>Y0</b>     | This <i>single precision floating point</i> value is used for scaling integer data samples. For floating point data samples this value is always zero.                                                                                                                                                                                      |
| <b>YSTEP</b>  | This <i>single precision floating point</i> value is used for scaling both integer and floating point data samples for the corresponding channel. The Y-value (Yv) in technical units can be calculated from each data sample-value (Dv) in the records by the formula:<br>$Y_v = Y_0 + YSTEP * D_v$                                        |
| <b>MAX</b>    | This <i>single precision floating point</i> value contains the maximum data sample value which can be encountered in the stored record(s) for this channel. For instance this value is 32768.0 for integer data. MAX and MIN can be used for scaling plots of the data without the need to scan the data for the maximum and minimum first. |
| <b>MIN</b>    | This <i>single precision floating point</i> value contains the minimum data sample value which can be encountered in the record(s) for this channel. For instance this value is -32768.0 for integer data.                                                                                                                                  |

# I. MATLAB formaat

Overzicht van opslagformaat Matlab datafiles. het overzicht is afkomstig uit de handleiding van de Matlab (v3.5) software (3-119 t/m 3-122).

## Description:

`save` and `load` are the MATLAB commands to store and retrieve variables on disk. They can also import and export ASCII data files.

### MAT-files

MAT-files are full-precision binary MATLAB format files created by the `save` command and readable by the `load` command.

`save temp` stores all variables in a MAT-file named `temp.mat`.

`save temp X` saves only variable `X`.

`save temp X Y Z` saves `X`, `Y`, and `Z`.

`save`, by itself, saves everything in a file named `matlab.mat`.

The `load` command is the inverse of `save`.

`load temp` retrieves all variables from the MAT-file named `temp.mat`.

`load`, by itself, loads the variables from the file `matlab.mat`.

Double-precision MAT-files can be transferred between PC, VAX, Sun, Macintosh, and other implementations of MATLAB. MAT-files contain a signature that indicates the machine that wrote the file. The `load` command checks the signature and performs conversion, if necessary. Full numeric precision is retained when moving between IEEE arithmetic machines, such as the PC, Sun, and Macintosh. Precision is lost when files are exchanged with the VAX, however.

We recommend a public-domain program called *Kermit* for transferring M-files and MAT-files between computers.

### ASCII file import/export

`save temp.dat X /ascii` or `save temp.dat X -ascii` exports variable `X` in 8-digit ASCII format to the file named `temp.dat`.

`save temp.dat X /ascii /double` exports variable `X` in 16-digit ASCII form.

load temp.dat imports the ASCII flat file temp.dat, putting the result into a variable named temp. If the data are stored in ASCII form, with fixed-length rows terminated with newlines (carriage returns), and with spaces separating the numbers, then the file is called a *flat file*. (ASCII flat files can be edited using a normal text editor.) The load command directly imports flat files, putting the result into a variable with the same name as the file, with the extension stripped.

### Algorithm:

The save command saves MATLAB variables on disk in a specially structured file we call a MAT-file, so-called because the filename ends with “.mat”. It is possible to read and write MAT-files from your own programs, provided you use the special file structure.

A MAT-file may contain one or more variables. The variables are written sequentially on disk, with the bytes conceptually forming a continuous stream. Each variable starts with a fixed-length 20-byte header that contains information describing certain attributes of the variable. The 20-byte header consists of 5 four-byte long integers (words):

type      Type flag. Word 1 contains an integer whose decimal digits encode the variable type. If the integer is represented as *MOPT* where *M* is the thousands digit, *O* is the hundreds digit, *P* is the tens digit, and *T* is the ones digit, then:

*M* indicates the numeric format of binary numbers on the machine that wrote the file. Use this table to determine the number to use for your machine:

| MACHINE ID     |   |
|----------------|---|
| PC             | 0 |
| Other Intel    | 0 |
| Sun            | 1 |
| Apollo         | 1 |
| Macintosh      | 1 |
| Other Motorola | 1 |
| VAX D-float    | 2 |
| VAX G-float    | 3 |

*O* is normally 0, which means the data are stored in a *column-wise* orientation (varies fastest down a column). If *O* = 1, the data are transposed and stored in a *row-wise* orientation (varies fastest across a row).

$P$  is normally 0, which means the data are stored on disk in double-precision (8 bytes/element). If  $P = 1$ , the data are stored in single precision (4 bytes/element).  $P = 2$  is signed 32-bit integer data,  $P = 3$  is 16-bit signed integers, and  $P = 4$  is unsigned 16-bit integers.

$T$  is normally 0, indicating that the data that follow describe a matrix. If  $T = 1$ , the variable is a text variable. This means that the numbers in the variable are floating point numbers between 0 and 255 representing the ASCII code of characters.

For PCs, `type` is usually 0000, or 0, which indicates PC double precision matrix data stored by columns. Note that  $P \neq 0$  and  $O = 1$  are not produced by the `save` command, but they could be generated outside of MATLAB (to save file space) and accepted by `load`.

- `mrows` Row dimension. Word 2 contains an integer with the row dimension of the variable.
- `ncols` Column dimension. Word 3 contains an integer with the column dimension of the variable.
- `imagf` Imaginary flag. Word 4 is an integer that is either 0 or 1. If 1, then the variable has an imaginary part. If 0, there is only real data.
- `namlen` Name length. Word 5 contains an integer with the length of the variable name plus 1.

Immediately following the fixed length header is data that has a length dependent on the variables in the fixed length header:

- `name` Variable name. The name consists of `namlen` ASCII bytes, the last one of which must be a NUL character (encoded as 0).
- `real` Real part of the matrix. The real data consists of `mrow * ncol` double precision (8-byte) floating point numbers. Matrices are stored column-wise, first the first column, then the second column, etc., unless otherwise specified by `type` in the fixed header.
- `imag` Imaginary part of the matrix, if any. If the imaginary flag `imagf` is nonzero, the imaginary part of a matrix is here. It is stored in the same way as real data.

The structure is repeated for each variable stored in the file.

Here is some C language code that writes a single matrix on disk:

```
typedef struct {
    long type; /* type */
    long mrows; /* row dimension */
    long ncols; /* column dimension */
    long imagf; /* flag indicating imag part */
    long namlen; /* name length (including NULL) */
} Fmatrix;

char *pname; /* pointer to matrix name */
double *pr; /* pointer to real data */
double *pi; /* pointer to imag data */
FILE *fp;
Fmatrix x;
int mn;

fwrite(&x, sizeof(Fmatrix), 1, fp);
fwrite(pname, sizeof(char), x.namlen, fp);
mn = x.mrows * x.ncols;
fwrite(pr, sizeof(double), mn, fp);
if (x.imagf)
    fwrite(pi, sizeof(double), mn, fp);
```

The Fortran and C equivalents of this code are available in the MATLAB TOOLBOX. Perhaps you'll find code for other languages there too.

### Limitations:

There are some rules that MATLAB uses when it looks for data files. If the filename does not contain an extension beginning with period ".", MATLAB appends the extension .mat before opening the file. If the filename contains an extension, i.e., it has a period in it, MATLAB will open the file with the complete name. This means that files without an extension cannot be saved or loaded, because MATLAB will append the .mat before trying to open them.

On VAX/VMS and MS-DOS computers, you can load extensionless files by putting a period at the end of the filename, for example, "load temp.". This does not work on Unix systems because Unix does not require filename extensions.



## J. Pascal TYPEN overzicht

Onderstaande lijst geeft een overzicht van de beschikbare TYPEN in Turbo Pascal versie 6.0. Alle in het programma BACOS gebruikte typen zijn hiervan afgeleid.

| INTEGER types |                          |              |
|---------------|--------------------------|--------------|
| typenaam      | range                    | woordbreedte |
| Shortint      | -128..+127               | 8 bits       |
| Integer       | -32768..+32767           | 16 bits      |
| Longint       | -2147483648..+2147483647 | 32 bits      |
| Byte          | 0..+255                  | 8 bits       |
| Word          | 0..+65535                | 16 bits      |

| REAL types |                                                     |              |
|------------|-----------------------------------------------------|--------------|
| typenaam   | range                                               | woordbreedte |
| Real       | $+2,9 \cdot 10^{-39} \dots +1,7 \cdot 10^{+38}$     | 6 bytes      |
| Single     | $+1,5 \cdot 10^{-45} \dots +3,4 \cdot 10^{+38}$     | 4 bytes      |
| Double     | $+5,0 \cdot 10^{-324} \dots +3,7 \cdot 10^{+308}$   | 8 bytes      |
| Extended   | $+3,4 \cdot 10^{-4932} \dots +1,1 \cdot 10^{+4932}$ | 10 bytes     |
| Comp       | $-9,2 \cdot 10^{+18} \dots +9,2 \cdot 10^{+18}$     | 8 bytes      |

## K. Listings BACOS

Listing van hoofdprogramma BACOS.PAS.

```
PROGRAM BACOS;
```

```
{Compiler directives}
```

```
  {$R+} {Use range checking}
```

```
  {$N+} {Use real-type calculation facility supported by Co-processor}
```

```
  {$E+} {Emulate use of Co-processor if it is not available}
```

```
{* * * * *
*
*   <B> <A> k k e r   <C> <O> n v e r s i o n   <S> o f t w a r e
*
*
*           BAKKER BINARY DATA -> ASCII DATA
*
*                   -> MATLAB DATA
*
*
*                   J.P van Herel
*
*
* source code written in Turbo Pascal 6.0
* * * * *
*
* Info: L.A.H. Wilmes EEG 0.17, telefoon 4470
*
*   University of Technology
*
*   E Faculty, Group EG
*
* * * * * }
```

```
USES U_BACOS, U_BACOS1, CRT;
```

```
CONST ProgramVersion = '6.11'; { June 1992 }
```

```
VAR InputFilename,
    OutputFilename   : FilenameType;
    Xheader          : XheaderType;
    YArray           : YArrayType;
    ResultCode       : INTEGER;
    Answer           : CHAR;
    Teller           : BYTE;
    WriteChannelText : STRING[8];
    StoreChannel     : StoreChannelType;
    StoreFormat      : CHAR;
```

```
BEGIN
  IntroTekst(ProgramVersion);

  Get_InputFilename(InputFilename, ResultCode);
  IF (ResultCode = -1) THEN BEGIN
      WRITE(#10, 'No filename specified. ');
      WRITELN('Program aborted. ');
      HALT(1);
  END;
  WRITELN('Input will be read from ', InputFilename, '.', #10, #10);

  Ask('Store data in Matlab or ASCII format ?', 'MA', Answer);
  StoreFormat := Answer;

  Get_OutputFilename(OutputFilename, StoreFormat, ResultCode);
  IF (ResultCode = -1) THEN BEGIN
      WRITE(#10, 'No filename specified. ');
      WRITELN('Program aborted. ');
      HALT(1);
  END;
  WRITELN('Output will be written to ', OutputFilename, '.');

  MessageToScreen('File information');
  Read-Headers(InputFilename, Xheader, YArray, ResultCode);
  IF (ResultCode = -1) THEN
  BEGIN
      WRITELN('Datafile is not a known BAKKER format!');
      WRITELN('Program aborted. ');
      HALT(2);
  END;

  Ask('Want an overview of XHeader DATA ?', 'YN', Answer);
  IF Answer = 'Y' THEN Show_XHeaderInfo(XHeader);

  Ask_ChannelsToStore(StoreChannel, XHeader, YArray, StoreFormat, ResultCode);
  IF ResultCode = 1 THEN
  BEGIN
      WRITELN(#10, #10, 'You did not select any Y-data. No file written...');
  END
  ELSE
  BEGIN
      MessageToScreen('Disk storage');
      CASE StoreFormat OF
          'M' : WriteToMatlabfile(InputFilename, OutputFilename, XHeader, YArray,
                                  StoreChannel, ResultCode);
          'A' : WriteToASCIIfile(InputFilename, OutputFilename, XHeader, YArray,
```

```
                                StoreChannel, ResultCode);
END;
IF ResultCode = -1 THEN
BEGIN
    WRITELN('Error. DOS is not able to free enough memory for me..');
    WRITELN('- Please free up some memory: get rid of resident programs');
    WRITELN('- Or decrease Bufferlength value in procedure WriteToMatlab');
    WRITELN(' and recompile BACOS.');
```

```
END
ELSE
    WRITELN('Data written succesfully...');
END;
MessageToScreen('Program finished...');
END.
```

## L. Listings BACOS

Listing van UNIT met naam U\_BACOS.PAS.

```
UNIT U_BACOS;
```

```
{Compiler directives}
```

```
  {$R+} {Use range checking}
```

```
  {$N+} {Use real-type calculation facility supported by Co-processor}
```

```
  {$E+} {Emulate use of Co-processor if it is not available}
```

```
{ - - - - - }
{                                     }
{                                     }
{ - - - - - }
{
```

```
    J.P. van Herel [311890]
```

```
    laatste wijzigingen: 26 mei 1992
```

Deze UNIT bevat routines voor het inlezen van data files zoals deze door het BAKKER-systeem 2570 worden geproduceerd.

Iedere procedure is voorzien van beknopt commentaar.

Voor een beschrijving van het gebruik van BACOS wordt verwezen naar het afstudeerverslag van J.P. van Herel.

```
}
{ - - - - - }
{                                     }
{                                     }
{ - - - - - }
{
```

```
INTERFACE
```

```
USES CRT;
```

```
TYPE String7      = STRING[7];
   String12     = STRING[12];
   FilenameType  = STRING;
   DateString    = STRING[18];
   DateArray     = ARRAY[0..5] OF CHAR;
   BArr12       = ARRAY[0..12] OF CHAR;
   WriteChannelType = ARRAY[1..4] OF BOOLEAN;
   XheaderType  = RECORD
                       Header_Version,
                       Nchan,
```

```

                                NRec          : INTEGER;
                                Length,
                                Nskip,
                                Nsamp         : LONGINT;
                                Samptype     : INTEGER;
                                XUnits       : String7;
                                X0,
                                Xstep        : DOUBLE;
                                Date_Info    : DateArray;
                                END;
YHeaderType = RECORD
    Name,
    Yunits      : String[6];
    Y0,
    Ystep,
    Max,
    Min         : SINGLE;
    END;
StoreChannelType = ARRAY[0..4] OF BOOLEAN;
YArrayType       = ARRAY[1..4] OF YheaderType;

VAR YInt  : INTEGER;
    YReal : SINGLE;

{- -}
{           PROCEDURE- AND FUNCTIONDECLARATIONS           }
{- -}

PROCEDURE Get_InputFileName(VAR Filename: FilenameType;
                           VAR ResultCode: INTEGER);

PROCEDURE Get_OutputFileName(VAR Filename: FilenameType;
                             StoreFormat: CHAR;
                             VAR ResultCode: INTEGER);

PROCEDURE Get_Filename(VAR Filename: FilenameType;
                      VAR ResultCode: INTEGER);

PROCEDURE Read-Headers(InputFilename: FilenameType;
                      VAR Xheader: XHeaderType;
                      VAR YArray: YArrayType;
                      VAR ResultCode: INTEGER);

PROCEDURE WriteToASCIIFile(InputFilename, OutputFilename: FileNameType;
                          XHeader: XHeaderType;
                          YArray: YArrayType);

```

```

Storechannel: StoreChannelType;
VAR ResultCode: INTEGER);

PROCEDURE WriteToMatlabFile(InputFilename, OutputFilename: FileNameType;
    XHeader: XHeaderType;
    YArray: YArrayType;
    Storechannel: StoreChannelType;
    VAR ResultCode: INTEGER);

PROCEDURE Ask_ChannelsToStore(VAR StoreChannel: StoreChannelType;
    XHeader: XHeaderType;
    Yarray: YarrayType;
    StoreFormat : CHAR;
    VAR ResultCode: INTEGER);

PROCEDURE Show_XHeaderInfo(XHeader: XHeaderType);

PROCEDURE Show_YheaderInfo(XHeader: XHeaderType;
    Yarray: YarrayType;
    YChannel: INTEGER);

PROCEDURE MessageToScreen(Screentext: STRING);

PROCEDURE Introtekst(ProgramVersion:STRING);

PROCEDURE Ask(AskString, AnswerSet: STRING;
    VAR Answer: CHAR);

FUNCTION CDate(DateData: DateArray): DateString;

{- - - - -}
{          PROCEDURE- AND FUCTIONDEFINITIONS          }
{- - - - -}
IMPLEMENTATION

PROCEDURE Get_InputFilename;
{ = = = = = }
Doel: Verkrijgen van naam van BAKKER datafile. Ook wordt gecontroleerd
      of de file bestaat.
In   : -
Uit  : Filename   : naam van de invoerfile die is opgegeven
      ResultCode  : = 0 : filenaam opgegeven en file bestaat
                  =-1 : Geen filenaam opgegeven
= = = = = }
BEGIN
    MessageToScreen('Specify the name of the BAKKER input file');

```

```

REPEAT
  WRITELN('You can include path and extension of input filename.');
```

Get\_Filename(Filename, ResultCode);

```

CASE ResultCode OF
  0 : {Filename ok};
  -1 : {No filename chosen} ;
  3 : WRITELN(#10,'Sorry. Path seems to be wrong.');
```

103 : WRITELN(#10,'Sorry. I can''t find ',Filename,'.')

```

ELSE WRITELN(#10,'Sorry. Error ', ResultCode, ' occurred.');
```

END;

```

UNTIL (ResultCode <= 0);
END;
```

PROCEDURE Get\_OutputFilename;

```

{ = = = = = }
Doel: Verkrijgen van filenaam voor de uitvoerfile. Ook wordt gecontroleerd
      of de file bestaat. Indien de file reeds bestaat wordt deze (indien
      toegestaan) overschreven.
In   : StoreFormat: gekozen opslag formaat
Uit  : Filename   : naam van de uitvoerfile die is opgegeven
      ResultCode  : = 0 : filenaam opgegeven en file bestaat niet of mag
                    worden overschreven
                    =-1 : Geen filenaam opgegeven
= = = = = }
```

```

VAR Answer : CHAR;
BEGIN
  MessageToScreen('Specify the name of the outputfile');
  WRITELN('Please include path and extension if needed!');
```

CASE StoreFormat OF

```

  'M': WRITELN('Normally .MAT is used as extension for Matlab format.',#10);
  'A': WRITELN('Normally .TXT is used as extension for ASCII format.',#10);
END;
```

REPEAT

```

  Get_Filename(Filename, ResultCode);
  CASE ResultCode OF
    103 : {Filename ok};
    -1  : {No file chosen};
    3   : WRITELN(#10,'Sorry. Path seems to be wrong.');
```

0 : BEGIN

```

    WRITELN(#10,'WARNING. ',Filename, ' exists!!',#10);
    ASK('Want to overwrite file','YN', Answer);
    IF (Answer='Y') THEN ResultCode := 103;
  END
  ELSE WRITELN(#10,'Sorry. Error ', ResultCode, ' occurred.');
```

END;

```

UNTIL (ResultCode = -1) OR (ResultCode = 103);
```



```

END;

PROCEDURE Get_FileName;
{ = = = = = }
Doel: Opvragen filenaam en controle of file bestaat
In   : -
Uit  : Filenaam   : Naam van de te openen file
       ResultCode : -1  : geen filenaam opgegeven
       <> 0: anders
= = = = = }
VAR TempFile: TEXT;
BEGIN
  WRITE('Enter filename (RETURN to stop): ');
  READLN(Filenaam);
  IF LENGTH(Filenaam)=0 THEN ResultCode := -1
  ELSE
  BEGIN
    {$I-}
    ASSIGN(TempFile, Filenaam);
    RESET(TempFile);
    CLOSE(TempFile);
    {$I+}
    ResultCode := IOResult;
  END;
END;

FUNCTION CStr (ZeroStr : Pointer) : String12;
{ = = = = = }
Doel: Converteren van BAKKER string data in Pascal STRING
In   : ZeroStr : Pointer naar begin van de BAKKER String. De string eindigt
       met #00.
Uit  : CStr    : De Pascal string
= = = = = }
VAR Loop      : Byte;
    BAPtr     : ^Barr12;
BEGIN
  BAPtr := @ZeroStr^;
  CStr := '';
  FOR Loop := 0 TO 12 DO
  BEGIN
    IF BAPtr^ [Loop] = #0 THEN BEGIN
      CStr [0] := Chr(Loop);
      Exit;
    END;
    CStr [Loop+1] := BAPtr^ [Loop];
  END;
END;

```

```

END;

FUNCTION CDate;
{ = = = = =
Doel: Omzetten van gecodeerde BAKKER datum-tijd-informatie in leesbare tekst
In  : DateData : De gecodeerde string
Uit : CDate    : De leesbare string met datum en tijd
= = = = = }
CONST Month : ARRAY[1..12] OF STRING[3]
      = ('jan','feb','mar','apr','may','jun','jul',
        'aug','sep','oct','nov','dec');
VAR Hour, Minute, Second, Day, Year: STRING[2];
BEGIN
  STR(ORD(DateData[0]),Hour);
  STR(ORD(DateData[1]),Minute);
  STR(ORD(DateData[2]),Second);
  STR(ORD(DateData[3])+80:2,Year);
  STR(ORD(DateData[5]),Day);
  Hour := COPY('0',1,2-LENGTH(Hour)) + Hour;
  Minute := COPY('0',1,2-LENGTH(Minute)) + Minute;
  Second := COPY('0',1,2-LENGTH(Second)) + Second;
  Year := COPY('0',1,2-LENGTH(Year)) + Year;
  Day := COPY('0',1,2-LENGTH(Day)) + Day;
  CDate := Day + '/' + Month[ORD(DateData[4])] + '/' + Year + ' ' +
          Hour + ':' + Minute + ':' + Second;
END;

PROCEDURE Read_Headers;
{ = = = = =
Doel: Het inlezen van de Header-informatie van X en Y-kanalen. Per file is
er een enkele Xheader (met informatie over sampletijden, het aantal
samples, het aantal Y-kanalen, etc. Voor ieder opgeslagen Y-kanaal is
er een aparte YHeader.
In  : filename : naam van de file die moet worden gebruikt voor het lezen
Uit : XHeader  : recordstructuur met hierin alle X-data
      YHeaderArray : array met de Yheaders (maximaal 4)
      ResultCode  : -1 : geen BAKKER datafile
                  <> 0 : anders
= = = = = }
VAR InputFile : FILE;
    YTeller   : WORD;
BEGIN
  ResultCode := 0;
  ASSIGN(InputFile, InputFilename);
  RESET(InputFile,1);
  BLOCKREAD(InputFile, Xheader, SIZEOF(Xheader));

```

```

IF XHeader.Header_Version <> 1 THEN
  ResultCode := -1
ELSE
  BEGIN
    XHeader.xunits := CStr (@XHeader.xunits);
    FOR Yteller := 1 TO XHeader.NCHAN DO
      BEGIN
        SEEK(InputFile,256 + (Yteller-1)*32);
        BLOCKREAD(InputFile, YArray[Yteller], SIZEOF(YArray[Yteller]));
        YArray[yteller].name := CStr(@YArray[yteller].name);
        YArray[yteller].yunits := CStr(@YArray[yteller].yunits);
      END;
    CLOSE(InputFile);
  END;
END;

PROCEDURE WriteToASCIIFile;
{ = = = = = }
Doel: wegschrijven van BAKKER datafile in ASCII kolom formaat. Aangegeven
      kan worden welke data gewenst is. Boven iedere kolom wordt de naam
      van het betreffende kanaal weergegeven, inclusief de eenheid. Indien
      tijdinformatie gewenst is, wordt deze in de eerste kolom geplaatst.
In   : Inputfilename : naam van de BAKKER file die moet worden ingelesen
      Outputfilename: naam van de ASCII file die wordt aangemaakt
      XHeader       : recordstructuur met hierin alle X-data
      YHeaderArray  : array met de Yheaders (maximaal 4)
Uit  : ResultCode    : 0 : (niet gebruikt)
{ = = = = = }
VAR  OutputFile  : TEXT;
      InputFile   : FILE;
      Ychannel    : Word;
      YTeller2,
      YTeller2End,
      Seekptr     : LONGINT;
      T0, Tstep   : DOUBLE;
      XUnits      : STRING;
BEGIN
  Writeln(#10,'Creating ASCII file named ',OutputFilename,'.',#10);
  Assign(InputFile, InputFilename);
  Assign(OutputFile,OutputFilename);
  Reset(InputFile,1);
  Rewrite(OutputFile);
  ResultCode := 0;

  { write only the specified columnheaders }
  IF StoreChannel[0] THEN WRITE(OutputFile,'time':10,' ');

```

```

FOR Ychannel := 1 TO XHeader.NChan DO
  IF StoreChannel[Ychannel] THEN
    WRITE(OutputFile,Yarray[YChannel].Name:10,' ');
  WRITELN(OutputFile);
  XUnits := CONCAT(XHeader.XUnits,' *1E-3');
  IF StoreChannel[0] THEN WRITE(OutputFile,XUnits:10,' ');
  FOR Ychannel := 1 TO XHeader.NChan DO
    IF StoreChannel[Ychannel] THEN
      WRITE(OutputFile,Yarray[Ychannel].YUnits:10,' ');
    WRITELN(OutputFile);

  TO := XHeader.X0 + XHeader.NSkip*XHeader.XStep;
  Tstep := XHeader.XStep;
  YTeller2End := XHeader.NRec * XHeader.NSamp - 1;
  FOR YTeller2 := 0 TO YTeller2End DO
  BEGIN
    WRITE(YTeller2 / Yteller2End * 100:4:1,'% ',#13);
    IF StoreChannel[0] THEN
      WRITE(OutputFile,1000*(TO+Yteller2*Tstep):10:4,' ');
    FOR Ychannel := 0 TO XHeader.Nchan-1 DO
      IF Storechannel[Ychannel+1] THEN
        BEGIN
          IF (XHeader.Samptype = 2) THEN
            BEGIN
              Seekptr := 256+XHeader.NChan*32+YTeller2*2+
                Ychannel*XHeader.NRec*XHeader.NSamp*2;
              SEEK(InputFile, Seekptr);
              BLOCKREAD(InputFile,YInt,2);
              WRITE(OutputFile,
                YInt*Yarray[YChannel+1].Ystep +
                Yarray[Ychannel+1].Y0:10:4,' ');
            END
          ELSE { XHeader.Samptype = 4, 6, ... }
            BEGIN
              Seekptr := 256+XHeader.NChan*32+YTeller2*4+
                Ychannel*XHeader.NRec*XHeader.NSamp*4;
              SEEK(InputFile, Seekptr);
              BLOCKREAD(InputFile,YReal,4);
              WRITE(OutputFile,Yreal*Yarray[YChannel+1].Ystep:10:4,' ');
            END;
          END;
        WRITELN(OutputFile);
      END;
    CLOSE(InputFile);
    CLOSE(OutputFile);
  END;

```

```

PROCEDURE WriteTomatlabFile;
{ = = = = = }
Doel: wegschrijven van BAKKER data in een MATLAB formaat file (zie de hand-
      leiding van BACOS voor nadere informatie betreffende de opbouw).
      Er worden drie tijdvariabelen weggeschreven: TB (begintijd), TS (stap-
      tijd) en TE (eindtijd). De eenheid is seconden.
      Van de gewenste Y-kanalen wordt alle data weggeschreven.
In   : Inputfilename : naam van de BAKKER file die moet worden ingelesen
      Outputfilename: naam van de ASCII file die wordt aangemaakt
      XHeader       : recordstructuur met hierin alle X-data
      YHeaderArray  : array met de Yheaders (maximaal 4)
Uit  : ResultCode   :   -1 : niet voldoende werkgeheugen
      <> 0 : anders
= = = = = }
CONST BufferLength      = 3000; {Must be less than 8192 for Pascal}
TYPE MatlabHeaderType = RECORD
    type_M, type_O, type_P, type_T : BYTE;
    mrows, ncols, imagf, namelen   : LONGINT;
END;
PascalIntWorkBufType  = ARRAY[1..BufferLength] OF INTEGER;
PascalDoubWorkBufType = ARRAY[1..BufferLength] OF DOUBLE;
MatlabWorkBufType     = ARRAY[1..BufferLength] OF DOUBLE;
VAR MatlabHeader      : MatlabHeaderType;
    MatlabFile, InputFile : FILE;
    Channelname       : STRING;
    TB, TS, TE, YChanneldata : DOUBLE;
    YchannelSamples,
    YchannelByteLength,
    PascalByteLength,
    LeftOver, NumberOfPasses,
    PassCounter, SeekPTR,
    OffsetPTR        : LONGINT;
    Ychannel, Teller, Loop : INTEGER;
    ConversionFac    : SINGLE;
    PascalIntWorkBuf : ^PascalIntWorkBufType;
    PascalDoubWorkBuf : ^PascalDoubWorkBufType;
    MatlabWorkBuf    : ^MatlabWorkBufType;

PROCEDURE StoreTimeVar(MatlabHeader: MatlabHeaderType;
                      Timevarname: STRING; TimeValue: DOUBLE);
{ = = = = = }
{Doel: opslaan van enkele tijdvariabele (DOUBLE) in matlab formaat
In   : MatlabHeader : record volgens Matlab data formaat
      Timevarname  : naam van variabele
      Timevalue    : waarde van de tijdvariabele

```

```

    Uit : -
    = = = = = }
VAR Teller : INTEGER;
BEGIN
    MatlabHeader.Namelen := LENGTH(Timevarname);
    MatlabHeader.ncols   := 1;
    BLOCKWRITE(MatlabFile, MatlabHeader,20);
    FOR Teller := 1 TO MatlabHeader.Namelen DO
        BLOCKWRITE(MatlabFile, Timevarname[Teller], 1);
        BLOCKWRITE(MatlabFile, TimeValue,8);
    END;

BEGIN
    IF (MAXAVAIL < SIZEOF(PascalIntWorkBuf) +
        SIZEOF(PascalDoubWorkBuf)+
        SIZEOF(MatlabWorkBuf))
    THEN BEGIN
        ResultCode := -1;
        Exit;
        END
    ELSE ResultCode := 0;

    { create sufficient space on heap NOT on stack }
    NEW(PascalIntWorkBuf);
    NEW(PascalDoubWorkBuf);
    NEW(MatlabWorkBuf);

    { setup default values in Matlab record file }
    MatlabHeader.type_M := 0;  {ID number}
    MatlabHeader.type_0 := 0;  {column wise data storage}
    MatlabHeader.type_P := 0;  {double precision storage, 8 bytes}
    MatlabHeader.type_T := 0;  {data storage as matrix type, not text}
    MatlabHeader.imagf  := 0;  {no imaginary data}
    MatlabHeader.mrows  := 1;  {single column of data (vector)}

    WRITELN(#10,'Creating MATLAB file named ',OutputFilename,','.#10);
    ASSIGN(Matlabfile, OutputFilename);
    REWRITE(MatlabFile,1);

    { Store Time variables first (if wanted) }
    IF StoreChannel[0] THEN
    BEGIN
        WRITELN('Storing time data...');
        Channelname := 'TB'+ #0;
        TB := XHeader.X0 + XHeader.NSkip * XHeader.XStep;
        StoreTimeVar(MatlabHeader, Channelname, TB);
    END

```

```

Channelname := 'TS'+ #0;
TS := XHeader.XStep;
StoreTimeVar(MatlabHeader, Channelname, TS);
Channelname := 'TE'+ #0;
TE := TB + (XHeader.NRec * XHeader.NSamp-1) * TS;
StoreTimeVar(MatlabHeader, Channelname, TE);
END;

{ Now store the Y-channel data }
ASSIGN(InputFile, InputFilename);
RESET(InputFile,1);
FOR Ychannel := 1 TO 4 DO
IF Storechannel[Ychannel] THEN
BEGIN
WRITELN('Storing channel #',Ychannel,' data...');
Channelname := Yarray[Ychannel].Name + #0;
MatlabHeader.namelen := LENGTH(Channelname);
YchannelSamples := XHeader.NRec*XHeader.NSamp;
MatlabHeader.ncols := YchannelSamples;

{ first write matlabHeader }
BLOCKWRITE(MatlabFile, MatlabHeader,20);
{ write channelname }
FOR Teller := 1 TO LENGTH(ChannelName) DO
BEGIN
IF ChannelName[Teller] = ' ' THEN ChannelName[Teller] := '_';
BLOCKWRITE(MatlabFile, ChannelName[Teller], 1);
END;

{ buffertype depends on type of input data: INT or DOUBLE }
IF (XHeader.Samptype = 2) THEN
BEGIN
Ychannelbytelength := YchannelSamples * 2;
PascalBytelength := SIZEOF(PascalIntWorkBuf^);
END
ELSE
BEGIN
Ychannelbytelength := YchannelSamples * 8;
PascalBytelength := SIZEOF(PascalDoubWorkBuf^);
END;

NumberofPasses := TRUNC(YchannelbyteLength / PascalBytelength);
Leftover := Ychannelbytelength - NumberofPasses * PascalBytelength;
OffsetPTR := 256 + XHeader.NChan * 32 + (YChannel-1) * YchannelbyteLength;
ConversionFac := Yarray[YChannel].Ystep;

```

```

{ in case of RAW input data }
IF (XHeader.Samptype = 2) THEN
BEGIN
  FOR PassCounter := 1 TO NumberOfPasses DO
  BEGIN
    SeekPTR := OffsetPTR + (PassCounter - 1) * SIZEOF(PascalIntWorkBuf^);
    SEEK(InputFile, SeekPTR);
    BLOCKREAD(InputFile, PascalIntWorkBuf^, SIZEOF(PascalIntWorkBuf^));

    FOR Loop := 1 TO BufferLength DO
    MatlabWorkBuf^[Loop] :=
      Yarray[Ychannel].Y0 + PascalIntWorkBuf^[Loop] * ConversionFac;
    BLOCKWRITE(MatlabFile, MatlabWorkBuf^, SIZEOF(MatlabWorkBuf^));
  END;

  { last part }
  IF (LeftOver > 0) THEN
  BEGIN
    SeekPTR := OffsetPTR + NumberOfPasses * SIZEOF(PascalIntWorkBuf^);
    SEEK(InputFile, SeekPTR);
    BLOCKREAD(InputFile, PascalIntWorkBuf^, LeftOver);
    FOR Loop := 1 TO BufferLength DO
    MatlabWorkBuf^[Loop] :=
      Yarray[Ychannel].Y0 + PascalIntWorkBuf^[Loop] * ConversionFac;
    BLOCKWRITE(MatlabFile, MatlabWorkBuf^, LeftOver*4);
  END;
END
ELSE

{ in case of formatted input data (DOUBLE-type) }
BEGIN
  FOR PassCounter := 1 TO NumberOfPasses DO
  BEGIN
    SeekPTR := OffsetPTR + (PassCounter - 1) * SIZEOF(PascalDoubWorkBuf^);
    SEEK(InputFile, SeekPTR);
    BLOCKREAD(InputFile, PascalIntWorkBuf^, SIZEOF(PascalDoubWorkBuf^));

    FOR Loop := 1 TO BufferLength DO
    MatlabWorkBuf^[Loop] :=
      Yarray[Ychannel].Y0 + PascalIntWorkBuf^[Loop] * ConversionFac;
    BLOCKWRITE(MatlabFile, MatlabWorkBuf^, SIZEOF(MatlabWorkBuf^));
  END;
  SeekPTR := OffsetPTR + NumberOfPasses * SIZEOF(PascalIntWorkBuf^);
  SEEK(InputFile, SeekPTR);

  { last part }

```



```

IF (LeftOver > 0) THEN
BEGIN
    SeekPTR := OffsetPTR + NumberOfPasses * SIZEOF(PascalIntWorkBuf^);
    SEEK(InputFile, SeekPTR);
    BLOCKREAD(InputFile, PascalIntWorkBuf^, LeftOver);
    FOR Loop := 1 TO BufferLength DO
        MatlabWorkBuf^[Loop] :=
            Yarray[Ychannel].Y0 + PascalIntWorkBuf^[Loop] * ConversionFac;
    BLOCKWRITE(MatlabFile, MatlabWorkBuf^, LeftOver);
    END;
END;
END;
CLOSE(InputFile);
CLOSE(MatlabFile);

{ clean up old heap memory }
DISPOSE(MatlabWorkBuf);
DISPOSE(PascalDoubWorkBuf);
DISPOSE(PascalIntWorkBuf);
END;

PROCEDURE Ask_ChannelsToStore;
{ = = = = = }
Doel: Opvragen welke kanalen moeten worden opgeslagen. Tevens kan worden
aangegeven of ook tijdinformatie moet worden opgeslagen.
In :
Uit :
= = = = = }
CONST StoreChannelDefault : StoreChannelType =
(TRUE,TRUE,FALSE,FALSE,FALSE);
VAR Teller, Kolom : INTEGER;
    Answer : CHAR;
BEGIN
    StoreChannel := StoreChannelDefault;
    Kolom := 1;
    MessageToScreen('Please specify which data must be converted');
    IF (StoreFormat = 'M') THEN
    BEGIN
        WRITELN('For the MATLAB output:');
        WRITELN('The requested variables will be stored in Matlab-format ');
        WRITELN('and can be read in by the LOAD command.',#10);
    END
    ELSE
    BEGIN
        WRITELN('For the ASCII output:');
        WRITELN('Data is stored in columns.');
```

```

WRITELN('Time data (if included) will always be in the first column.');
```

```

WRITELN('The next columns will contain Y-data.',#10);
END;

IF (StoreFormat = 'M') THEN
BEGIN
WRITELN('Three time variables will be created for use in Matlab:');
WRITELN('TB : Begin time, TS : Step time and TE : End time.',#10);
Ask('Want to include this time data in the Matlab file ?', 'YN', Answer);
WRITE('Timedata will ');
IF Answer <> 'Y' THEN WRITE('not ');
WRITELN('be included in the file.');
```

```

END
ELSE
BEGIN
Ask('Must timedata be included (in first column)?', 'YN', Answer);
IF Answer = 'Y' THEN
BEGIN
WRITELN('Timedata is written in column ',Kolom,#10);
Kolom := Kolom + 1;
END
ELSE WRITELN('No timedata written!',#10);
END;
StoreChannel[0] := (Answer = 'Y');
```

```

FOR Teller := 1 TO Xheader.NChan DO
BEGIN
Show_YHeaderInfo(XHeader, Yarray, Teller);
IF (StoreFormat = 'A') THEN
WRITELN(#10,'This channel will be stored in column ',Kolom, '.',',#10);
Ask('Want to store this channel data ?', 'YN', Answer);
StoreChannel[Teller] := (Answer = 'Y');
Kolom := Kolom + ORD((Answer = 'Y'));
END;
IF Kolom = 1 THEN ResultCode := 1 ELSE ResultCode := 0;
END;

PROCEDURE Show_XHeaderInfo;
{ = = = = =
Doel: Geef in een overzicht de informatie uit de Xheader data op het scherm.
In : Xheader : record met hierin de Xheader informatie
Uit : -
= = = = = }
VAR TO,TS,TE : DOUBLE;
BEGIN
WITH XHeader DO
```

```

BEGIN
  MessageToScreen('Contents of the Xheader:');
  WRITELN;
  WRITELN('Header version           : ',Header_Version,'.');
  WRITELN('Date of recording              : ',Cdate(Date_Info),'.',#10);
  WRITELN('Name of X-units                 : ',XUnits,'.',#10);
  WRITELN('Number of Y-channels used       : ',NChan,'.');
  WRITELN('Number of records per channel   : ',NRec,'.');
  WRITELN('Number of samples per record    : ',NSamp,'.');
  WRITELN('This gives a total of          : ',NRec*NSamp,
    ' samples per channel.');
```

```

  WRITELN;
  WRITELN('The original recordlength was : ',Length,' samples.');
```

```

  WRITELN('While storing, the first      : ',Nskip,
    ' samples were skipped.',#10);
  WRITE ('The type of stored sample data : ');
  CASE SampType of
    2   : WRITELN('RAW data. (16 bit integer).',#10);
    4,5,6: WRITELN('Processed data (32 bit IEEE-754 single precision).',#10)
  END;
  TO := X0 + Nskip * XStep;
  TS := XStep;
  TE := TO + (NRec * NSamp - 1) * TS;
  WRITELN('Channel data starts at       : ', 1E3*TO:15:6,' ms.');
```

```

  WRITELN('The timestep dT is           : ', 1E3*TS:15:6,' ms.');
```

```

  WRITELN('End time is at                : ', 1E3*TE:15:6,' ms.',#10,#10);
END;
WRITE('Press any key when ready...',#13);
REPEAT UNTIL KEYPRESSED;
WRITE(' ':50);
END;

PROCEDURE Show_YheaderInfo;
{ = = = = = }
Doel: tonen van informatie uit de Yheader data (per kanaal).
In   : XHeader : record met XHeader informatie
      : YHeaderArray : array van records met per kanaal de YHeader informatie
Uit  : -
{ = = = = = }
BEGIN
  WITH Yarray[Ychannel] DO
  BEGIN
    WRITELN(#10,'Channel ',YChannel,': ');
    WRITELN('Name of channel is   : "',Name,'"');
    WRITELN('Units of channel       : ', YUnits,'.');
    IF XHeader.SampType = 2 THEN

```

```

        WRITELN('Offset Y-samples      : ',Y0:10:6,'. ');
        WRITELN('Scaling of Y-samples : ', YStep:10:6,'. ');
        WRITELN('Maximum sample value : ', Max:10:4,'. ');
        WRITELN('Minimum sample value : ', Min:10:4,'. ');
    END;
END;

PROCEDURE MessageToScreen;
{ = = = = =
Doel: afdrukken van een tekstmelding op het scherm
In  : ScreenText : de af te drukken tekst
Uit : -
= = = = = }
VAR Pos, Teller: INTEGER;
BEGIN
    WRITELN; WRITELN;
    Pos := (80 - LENGTH(ScreenText)-2) DIV 2;
    FOR Teller := 1 TO Pos-1 DO WRITE('-');
    TEXTBACKGROUND(CYAN); TEXTCOLOR(BLACK);
    WRITE(' ',ScreenText,' ');
    TEXTBACKGROUND(BLACK); TEXTCOLOR(WHITE);
    FOR Teller := 1 TO Pos-1 DO WRITE('-');
    WRITELN; WRITELN;
END;

PROCEDURE Introtekst;
{ = = = = =
Doel : geeft introtekst bij opstarten van BACOS
In   : -
Uit  : -
= = = = = }
BEGIN
    TEXTBACKGROUND(BLACK); TEXTCOLOR(WHITE);
    CLRSCR;
    MessageToScreen('- = - <B><A>kker <C><O>nversion <S>oftware - = -');
    WRITELN(' ':11,'A software utility for conversion of BAKKER 2570 datafiles');
    WRITELN(' ':20,'Version ',ProgramVersion,'J.P. van Herel':27,#10,#10);

    WRITELN(
        ' ':9,'This program converts datafiles from the BAKKER 2570 digitizer',
            #13,#10,
        ' ':9,'to other formats. Currently supported formats are:',#13,#10,#10,
        ' ':15,'          .1. ASCII column output',#13,#10,
        ' ':15,'          .2. Matlab output format',#13, #10,#10,#10,
        ' ':19,'Additional information is available at:',#13,#10,
        ' ':19,' L.A.H. Wilmes EEG 0.17, phone 4470');

```

```

END;

PROCEDURE Ask;
{ = = = = = }
Doel: verkrijgen van een antwoord op een vraag, waarbij het antwoord uit
      een op te geven lijst van letters moet komen
In   : AskString : de vraag
      AnswerSet  : de mogelijke antwoorden (letters)
Uit  : Answer    : het antwoord
= = = = = }
CONST Rotate : ARRAY[0..5] OF CHAR = (' ' , '-' , '=' , '+' , '=' , '-');
VAR Teller: INTEGER;
BEGIN
  TEXTCOLOR(YELLOW);
  WRITE(AskString,' [');
  FOR Teller := 1 TO LENGTH(AnswerSet)-1 DO WRITE(AnswerSet[Teller],',');
  WRITE(AnswerSet[Teller+1],'] : ');
  TEXTCOLOR(WHITE);
  REPEAT
    Teller := 0;
    REPEAT
      WRITE (#8,Rotate[Teller]);
      Teller := (Teller + 1) - 6*TRUNC(Teller/5);
      DELAY(50);
    UNTIL KEYPRESSED;
    Answer := UPCASE(READKEY);
  UNTIL POS(Answer,AnswerSet) <> 0;
  WRITELN(#8,Answer);
END;

END.

```

# M. Listings BACOS

Listing van UNIT met naam U\_BACOS.PAS.

```

UNIT U_BACOS1;

{Compiler directives}
  {$R+} {Use range checking}
  {$N+} {Use real-type calculation facility supported by Co-processor}
  {$E+} {Emulate use of Co-processor if it is not available}

{ - - - - - }
{              PROGRAM INFORMATION              }
{ - - - - - }
{
  In deze UNIT kunnen uitbreidingen worden aangebracht voor het BACOS pakket.
  Gelieve het bestand U_BACOS.PAS niet aan te passen.

  J.P. van Herel
}
{ - - - - - }
{              DEFINITION OF GLOBAL VARIABLES              }
{ - - - - - }
INTERFACE

{- - - - - }
{              PROCEDURE- AND FUNCTIONDECLARATIONS              }
{- - - - - }

{- - - - - }
{              PROCEDURE- AND FUCTIONDEFINITIONS              }
{- - - - - }
IMPLEMENTATION

END.
```