

# Een geprogrammeerde instructie voor het verspaningsmodel van Merchant

**Citation for published version (APA):**

Net, van der, A. J. (1992). *Een geprogrammeerde instructie voor het verspaningsmodel van Merchant*. (TH Eindhoven. Afd. Werktuigbouwkunde, Vakgroep Produktietechnologie : WPB; Vol. WPA1281). Technische Universiteit Eindhoven.

**Document status and date:**

Gepubliceerd: 01/01/1992

**Document Version:**

Uitgevers PDF, ook bekend als Version of Record

**Please check the document version of this publication:**

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

**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
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

[www.tue.nl/taverne](http://www.tue.nl/taverne)

**Take down policy**

If you believe that this document breaches copyright please contact us at:

[openaccess@tue.nl](mailto:openaccess@tue.nl)

providing details and we will investigate your claim.

Verslag van een onderzoeksopdracht.  
Een geprogrammeerde instructie  
voor het verspaningsmodel van  
Merchant.

A.J. van der Net,

April 1992

WPA 1281

## Inhoud.

Opzet van het programma . . . . .	1
Overzicht van de schermen . . . . .	3
De programmakop . . . . .	8
De procedures . . . . .	10
Het hoofdprogramma . . . . .	54
Toelichting op het programma . . . . .	55
Literatuur . . . . .	58

- Bijlagen:
- Overzichtstabel.
  - 2 floppydisks (5¼").

## Opzet van het programma.

### Inleiding:

Als onderdeel van het eerstejaars-onderwijs wordt het verspaningsmodel van Merchant behandeld. De diverse fasen in de modelvorming blijken voor de studenten vaak moeilijk te begrijpen. Ook het overzien van het gehele model lukt in een aantal gevallen niet.

### Opdracht:

De opdracht is als volgt geformuleerd:

- \* Het schrijven van een softwarepakket dat een geprogrammeerde instructie voor het verspaningsmodel van Merchant bevat. Dit pakket moet kunnen draaien op een standaard PC-AT en geschikt zijn voor gebruik door eerstejaars studenten.
- \* Er moet aanvullende informatie gegeven worden bij de behandeling van de diverse fasen van het model (kinematische opzet, krachten, vermogen etc.) en bij het gebruik van het model als geheel (verspaanbaarheid van materiaal).
- \* Bij het pakket moet een korte gebruikershandleiding worden geschreven.
- \* De informatie moet zodanig schriftelijk worden vastgelegd, dat studenten die niet de beschikking hebben over een PC toch over de informatie kunnen beschikken.

### Werkwijze:

Vanwege de uitgebreide grafische mogelijkheden is ervoor gekozen dit programma in Turbo Pascal, versie 5.5 te schrijven.

De basis voor het programma wordt gevormd door het eerstejaars dictaat dat tot nu toe is gebruikt (literatuur [5]).

De verschillende onderwerpen die in dit dictaat behandeld worden zijn ingedeeld in bij elkaar behorende 'blokken' van informatie.

Een informatieblok wordt ondergebracht in een procedure. Het hoofdprogramma maakt het mogelijk om met behulp van een aantal eenvoudige handelingen deze informatieblokken op te roepen.

Alvorens met het schrijven van de procedures is begonnen is eerst in een schema aangegeven de volgorde van de blokken en hun onderlingen samenhang. Daarna worden de afzonderlijke informatieblokken met behulp van de literatuur uitgewerkt. Alle blokken behandelen een bepaald aspect van het model. Wanneer alle blokken zijn doorlopen is het model stap voor stap verklaard. Om het model duidelijker te maken is ervoor gekozen ook enige animaties aan het programma toe te voegen, waardoor het moeilijk voor te stellen begrip 'afschuiving' sneller duidelijk wordt. De juiste volgorde van het doorlopen van alle blokken wordt door het programma aangegeven. Er kan echter van afgeweken worden, zodat reeds behandelde onderwerpen eenvoudig nog eens opnieuw bekeken kunnen worden.

Om het overzicht te kunnen bewaren is het aan te bevelen het schema samen met het programma te gebruiken. De uiteindelijke uitvoering van het schema is bijgevoegd in bijlage 1.

# Overzicht van de schermen.

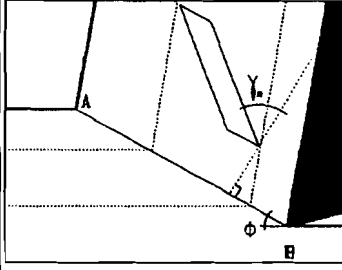
Hierna volgt een overzicht van alle schermen die het programma produceert. Dit overzicht is ook als hand-out bij het programma verstrekt met plaatjes van 9 bij 6 centimeter.

**TECHNISCHE UNIVERSITEIT EINDHOVEN**  
**FACULTEIT WERKTUIGBOUWKUNDE, VAKGROEP WPA**  
**PRODUKTIE-TECHNOLOGIE**  
**Het verspaningsmodel van Merchant**

Dit softwarepakket bevat een computergeprogrammeerde instructie voor het verspaningsmodel van Merchant. Hierbij wordt informatie gegeven voor het model en theorie die eraan ten grondslag ligt. Het programma wordt gebruikt als simulatie op het vak "Productietechnologie 1" in het eerste jaar.  
 Het programma is geschied in het kader van een onderzoeksopdracht aan de Faculteit Werktuigbouwkunde van de TU Eindhoven.

Voor meer informatie: prof. dr. Ir. A.B.H. van der Wolf  
 A.J. van der Mat  
 Eindhoven, April 1992

Alt-N = help  
 ← = volgend scherm  
 → = vorige scherm  
 t u  
 80:18



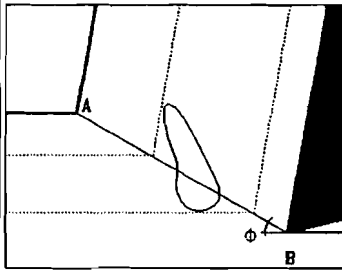
Het verspaningsproces kan worden voorgesteld zoals hier weergegeven.  $\phi$  is de afschuifhoek. Er treedt een hoekverdraaiing  $\gamma$  op.

Het proces kan worden gedemonstreerd door middel van een animatie.

Alt-N = help  
 ← = de de animatie te stem  
 → = volgend scherm  
 ← = vorige scherm  
 t = simulatie met behulp van cirkel  
 i = informatie over de secundaire deformatiezone  
 t u  
 80:20

Het fysieke verspaningsproces is moeilijk waar te nemen. Met zgn. quick-stop devices is het mogelijk om het proces stil te zetten, waarna de structuur onderzocht kan worden. Er is dan te zien dat het gehele vervormingsproces plaats vindt in een smal gebied rondom de lijn AB (fig.1). Onderzoek heeft aangetoond dat verspanen in hoofdzaak een proces van plastische deformatie van het materiaal is. Spanmateriaal bestaat uit sterk plastisch vervormd werktuigmateriaal. De regelmatige roosterstructuur vertoont een verschuiving van vele malen de roosterafstand onder een hoek met de bewegingsrichting van de belt, nl. gebied AB. Dit gebied heet de "primaire deformatiezone" of "afschuifzone". De lijnen evenwijdig aan het oorspronkelijke oppervlak in fig. 1 zijn kunstmatig aangebracht.

Alt-N = help  
 druk op → om het resultaat van een quick-stop test te zien (fig. 1)  
 t u  
 80:19



De vorm of grootte van het materiaaldeel is niet van belang. Een cirkel is vanwege zijn zijdelingse symmetrie zeer geschikt voor het vinden van de richting van de grootste rek. Een animatie geeft weer hoe het afschuifmechanisme bij een cirkel verloopt.

Alt-N = help  
 t u  
 80:21

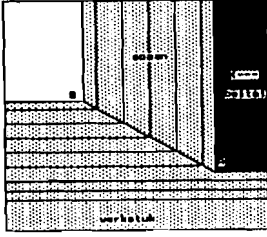
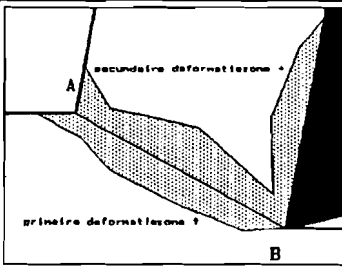


fig. 1  
 Resultaat van een quick-stop test.

Alt-N = help  
 → = volgend scherm  
 ← = vorige scherm  
 ← = terug naar startscherm  
 t u  
 80:19



Gebied AB heet de primaire deformatiezone. Er is ook een secundaire deformatiezone. Dit is het contactgebied tussen span en spanvlak. De deformatie die hier optreedt is het gevolg van de wrijving tussen span en spanvlak.

Alt-N = help  
 → = vorige scherm  
 t u  
 80:22

Om te komen tot een voorlopig model van het proces is het van belang dat afschuiven niet oneindig snel kan gebeuren. Dit betekent dat de primaire deformatiezone AB dus geen lijn kan zijn, maar dat de zone een bepaalde dikte moet hebben. Deze zone wordt begrensd door twee vlakken: het afschuifvlak en het stationaire vlak. Het afschuifvlak beweegt onder invloed van de afschuifkracht met een bepaalde snelheid t.o.v. het stationaire vlak. De richting wordt bepaald door de afschuifhoek. Er geldt:  $\tan \gamma = (V_s \cdot \Delta t) / \Delta c$ . In de tijd  $\Delta t$  gaat PQRS over in P'Q'R'S.

- = volgend scherm
- = vorige scherm
- = info over uniforme deformatie

Alt-H = help  
 tU  
 20123

Het tot nu toe beschreven proces verloopt schijnbaar continu. Het is echter essentieel het proces te beschrijven als opeenvolging van zich op identieke wijze herhalende deelprocessen:

- het vormen van een deformatiezone
- het afschuifproces in die zone
- het stationair worden van het afschuifvlak, zodra de bettel de deformatiezone heeft doorlopen.

Als het deformatieproces is voltooid is de deformatiezone overgegaan in een zgn spaanlamel.

Dit model van opeenvolging van deelprocessen resulterend in spaanlamellen berust op waarnemingen!

Het model wordt verder verduidelijkt op het volgende scherm.

- = vorig scherm
- = volgend scherm

Alt-H = help  
 tU  
 20122

Bij de animatie is getoond hoe een volume-element van vorm verandert. De oppervlakte is echter gelijk gebleven. Afschuiving gaat namelijk niet gepaard met volumeverandering als loodrecht op het vlak van tekering geen rek optreedt. Er is sprake van zgn. uniforme deformatie als de deformatie onafhankelijk is van de y-waarde (zie tekening).

Uniforme deformatie gaat dus gepaard met een lineaire verdeling van de verplaatsingssnelheid van materiaaldeel(tjes)  $v(y)$ . De deformatiesnelheid  $\dot{\gamma}_a$  is gedefinieerd als  $\tan \gamma_a = \dot{\gamma} / \Delta t$  en is over de hele deformatiezone constant:  $\tan \gamma_a = (V_s \cdot \Delta t) / \Delta c \Rightarrow \dot{\gamma}_a = V_s / \Delta c$

- = vorige scherm

Alt-H = help  
 tU  
 20125

De aansenschakeling van lamellen vormt de spaan, die met snelheid  $V_c$  langs het spaanvlak beweegt. Bij concentratie van de deformatie op het afschuifvlak kan men het proces zien als verschuiving van metalen plaatjes t.o.v. het afschuifvlak. De lamellenstructuur valt bij bekijken van de spaan meteen op. De dikte van de lamellen hangt af van het materiaal en de bewerkingscondities.

- = vorig scherm
- = informatie over geometrische relaties
- = informatie over krachten
- = informatie over temperatuur en versnemen
- = informatie over deformatierelaties

Alt-H = help  
 tU  
 20126

Het is nu mogelijk een eerste model van het verspaningsproces op te stellen. Neem aan dat een deformatiezone van bepaalde dikte ( $\Delta c$ ) ontstaat. De bettel beweegt met (snij)snelheid  $V$ . Het spaanvlak (het vlak waarop de gevormde spaan afsloopt) dwingt het afschuifvlak met een snelheid  $V_s$  te bewegen t.o.v. het stationaire vlak. De richting van de beweging wordt gegeven door de afschuifhoek  $\phi$ :

$$\sin \phi = \Delta c / BB'$$

- = volgend scherm
- = vorige scherm
- = meer uitleg over het proces

Alt-H = help  
 tU  
 20124

Uit de meetkunde van het verspaningsmodel kunnen we de zgn kinematische relaties halen. Het is mogelijk de snelheden  $V_s$  en  $V_c$  uit te drukken in de snelheid  $V$  en de hoeken  $\phi$  en  $\gamma$ :

$$V_c = V \cdot \frac{\sin \phi}{\cos(\phi - \gamma)}$$

$$V_s = V \cdot \frac{\cos \phi}{\cos(\phi - \gamma)}$$

Alt-H = help  
 tU  
 20127

Uit de geometrie kunnen we bovendien afleiden dat

$$\Delta c = V_s \sin \phi \cdot \Delta t$$

Uit de uniforme afschuiving in een deformatiezone met deze dikte volgde dat voor de deformatiegraad geldt

$$\tan \gamma_s = \frac{V_s}{\Delta c} \cdot \Delta t$$

Met de zojuist gevonden relaties voor  $V_s$  en  $\Delta c$  wordt dit

$$\tan \gamma_s = \frac{\cos \phi}{\sin \phi \cdot \cos(\phi - \gamma)}$$

Alt-H = help  
 tG  
 20127

- \* = terug naar het lamellenmodel
- = informatie over verspanbaarheid
- = informatie over ontbinden van de kracht
- = informatie over temperatuur en vermogen
- = informatie over deformatierelaties

Dit is een temperatuurverdeling die typisch is bij het verspanen. Opvallend is de hoge temperatuur op enige afstand van de hoofdanlijkant. Deze veroorzaakt een slijtage die "kolkafslage" of "kratervorming" heet.

Alt-H = help  
 tG  
 20128

druk op + voor verdere informatie over specifieke verspaningsarbeid.

Alle vervormingsprocessen kunnen worden beschreven vanuit het begrip specifieke arbeid. Dit is de arbeid die per volume-eenheid nodig is om een zekere vervorming te bereiken. De arbeid bestaat uit specifieke vervormingsarbeid en specifieke wrijvingsarbeid. De specifieke vervormingsarbeid is de arbeid, nodig om het materiaal te vervormen en is dus een soort materiaalkonstante. De vervorming vindt plaats in de primaire deformatiezone. De specifieke wrijvingsarbeid heeft te maken met de wrijving in de secundaire deformatiezone tussen materiaal en gereedschap en is dus een soort procesconstante.

Bijna het totale toegevoegde vermogen wordt omgezet in warmte. De verdeling van de afvoer tussen spaan, beitel en werkstuk hangt af van de snij snelheid en is op het volgende scherm geschetst.

Alt-H = help  
 tG  
 20128

druk op + om een grafiek te zien van de verdeling van de warmteafvoer

Zoals eerder getoond: de som van specifieke vervormingsarbeid en specifieke wrijvingsarbeid wordt de specifieke verspaningsarbeid genoemd. Deze specifieke verspaningsarbeid  $E_{sp}$  kan als volgt geschreven worden:

$$E_{sp} = \frac{F_v \cdot V}{V \cdot A} = \frac{F_v \cdot V_s}{V \cdot A} + \frac{F_f \cdot V_c}{V \cdot A}$$

Hierin de relaties voor  $V_s$  en  $V_c$  uit de geometrie invullen geeft

$$E_{sp} = \frac{F_v \cos \gamma}{A \cos(\phi - \gamma)} + \frac{F_f \sin \phi}{A \cos(\phi - \gamma)}$$

Alt-H = help  
 tG  
 20129

- \* = terug naar het lamellenmodel
- = informatie over verspanbaarheid
- = informatie over ontbinden van de kracht
- = informatie over deformatierelaties
- = informatie over geometrie

De verdeling van de warmte-afvoer :

Deze verdeling geeft bij lage snij snelheid een gereedschapstemperatuur die ongeveer 80% hoger is dan de temperatuur van de spaan. Dit is 140% bij hoge snelheid, omdat daarbij het vermogen toeneemt, waarbij de toevoer van warmte en het volume van de spaan evenredig vergroten. Het beitelvolume blijft echter gelijk.

Alt-H = help  
 tG  
 20129

druk op + om een temperatuurverdeling te zien

De verhouding tussen de gevormde en de onvervormde spaandooersnede  $A_c/A$  heet de "spaanstuik"  $\lambda_c$ . Als loodrecht op het vlak van tekening geen rek optreedt is dit gelijk aan  $hc/h$ . Uit de voorwaarde voor constante materiaalstroom volgt  $V \cdot h = V_c \cdot hc$  dus  $hc/h = V/V_c = \lambda_c$ . Als we de uit de geometrie gevonden relaties voor  $V$  en  $V_s$  invullen:

$$\lambda_c = \cos(\phi - \gamma) / \sin \phi$$

De spaanstuik is een meetbare grootte waaruit  $\phi$  is te halen.

Alt-H = help  
 tG  
 20131

- \* = terug naar het lamellenmodel
- = informatie over verspanbaarheid
- = informatie over ontbinden van de kracht
- = informatie over temperatuur en vermogen
- = informatie over geometrie



De algemene snijkracht  $F$  is op meer manieren te ontbinden. De eerste manier is ontbinding in een hoofdsnijkracht  $F_v$  en een aanzetkracht  $F_y$ . Deze werken in de richting van resp. de hoofd- en de aanzetbeweging.

Alt-H = help  
 20131

- \* = terug naar levelmodel
- \* = een andere ontbinding van de kracht
- ! = informatie over geometrische relaties
- ! = informatie over temperatuur en verspanen
- ← = informatie over deformatierelaties

Er zijn nu meerdere relaties gevonden. Ten eerste:

$$\tan \gamma_s = \frac{\cos \phi}{\sin \phi \cdot \cos(\phi - \gamma)}$$

uit de geometrie.

Voor kleine spanhoek  $\gamma$  is dit gelijk aan  $\frac{2}{\sin(2\phi)}$

Dit geeft als verband tussen deformatiegraad en afschuifhoek: ( $\phi$  van 0 tot 45 gr.)

De deformatie neemt dus toe bij kleinere afschuifhoek.

Alt-H = help  
 20132

druk op + voor uitleg

Een tweede manier van ontbinden van de algemene snijkracht is langs en loodrecht op het spanvlak van het gereedschap. Als de secundaire deformatiezone is bepaald door wrijving, dan is  $\tan \beta = F_I / F_{In}$  een maat voor de wrijving.  $\tan \beta$  heet dan de "schijnbare wrijvingscoëfficiënt".

Alt-H = help  
 20133

- \* = terug naar vorige ontbinding van de kracht
- \* = een andere ontbinding van de kracht
- ! = informatie over geometrische relaties
- ! = informatie over temperatuur en verspanen
- ← = informatie over deformatierelaties

De tweede relatie betreft de spanstuk:

$$\lambda = \frac{\cos(\phi - \gamma)}{\sin \phi}$$

(uit de deformatierelaties)

Als de spanstuk groot is wordt een materiaal slechter bewerkbaar geacht. Dit blijkt dus gepaard te gaan met een kleine afschuifhoek.

Alt-H = help  
 20133

druk op + voor uitleg

De derde manier om de snijkracht  $F$  te ontbinden is in de kracht  $F_s$  in het afschuifvlak en de kracht  $F_{sn}$  loodrecht daarop. De kracht veroorzaakt de afschuiving en is dan ook al eerder genoemd in het voorlopige model. Met afschuifsmaltheid  $V_s$  bepaalt  $F_s$  het vermogen dat de primaire deformatie "verbruikt".

Alt-H = help  
 20134

- \* = terug naar de vorige ontbinding van de kracht
- \* = informatie over verspanbaarheid
- ! = informatie over geometrische relaties
- ! = informatie over temperatuur en verspanen
- ← = informatie over deformatierelaties

Een derde relatie betreft de wrijving.

Er blijkt een relatie te bestaan tussen afschuifhoek, wrijvingshoek en spanhoek:

Verspanen wordt moeilijker, de afschuifhoek wordt kleiner als de wrijvingshoek groter is (bij vaste spanhoek), dus bij een relatief grote waarde van de wrijvingskracht.

Alt-H = help  
 20134

druk op + voor uitleg

Over de verspanbaarheid van materiaal is dus het volgende te zeggen:


Een materiaal is moeilijker te verspanen als de afschuifhoek  $\phi$  kleiner is. Er geldt dan:

- \* De spanstuik is groter
- \* De deformatiegraad is groter
- \* De wrijving is groter

Spanstuik en deformatiegraad blijken bovendien te worden beïnvloed door de beitelgeometrie (spaanhoek).

---


\* = terug naar het levelmodel  
 \* = einde programma  
 ? = informatie meer ontbinden van de kracht  
 \* = informatie meer deformatiesnelheid  
 ← = informatie meer geometrie


Alt-H = help  
  
 20194

Dit is het einde van het programma.  
 Als U nog niet alle schermen hebt doorlopen kunt U nog terug naar het eerste scherm van de Informatie over verspanbaarheid.

---

\* = terug naar verspanbaarheid  
 \* = stoppen

Alt-H = help  
  
 20194



programma gestopt. druk een toets om terug te gaan naar DOS

## De programmakop.

Het programma begint met de naamgeving en de declaratie van alle globale variabelen. De globale variabelen kunnen in het gehele programma worden gebruikt, in tegenstelling tot de lokale variabelen, die alleen zijn te gebruiken in het programmagedeelte waarin ze zijn gedeclareerd. Op de declaratie van variabelen wordt ingegaan in de literatuur (zie [1] en [3]). In de kop van het programma staat bovendien het zogenaamde 'USES' statement. Dit statement roept units aan, waarin procedures en functies staan, die standaard in Turbo-Pascal zijn ingebouwd. In dit programma wordt gebruik gemaakt van drie units:

- \* De Crt unit, waarin zich functies bevinden voor de controle van toetsenbord en scherm, speciale toetsenbordfuncties en geluidseffecten.
- \* De Graph unit, waarin zich een krachtig pakket bevindt van grafische mogelijkheden, zoals het tekenen van lijnen, kleurinstellingen, vulpatronen en speciale tekstfuncties. Van deze unit is in dit programma zeer veelvuldig gebruik gemaakt.
- \* De Dos unit, waarin zich functies uit het DOS besturingssysteem bevinden zoals tijd, datum informatie over directories en het uitvoeren van programma's binnen DOS.

In het vervolg zullen alle delen uit het programma die zijn opgenomen in dit verslag in een klein lettertype worden afgedrukt.

De kop van het programma ziet er dan als volgt uit:

```
PROGRAM Verspaningsmodel;  
  USES Crt, Graph, Dos;  
  CONST fi=0.463647609; x0=10; y0=200; l0=40; x00=50; y00=191;  
        l00=27; Memory=100; Windows=4;  
  TYPE Str80=STRING[80];  
        ColorList=ARRAY [1..Windows] OF INTEGER;
```

```
VAR Colors: ColorList;
    MaxColors: WORD;
    Xmax, Ymax, ViewXmax, ViewYmax, X1, X2, Y1, Y2, CurrentLine,
    ColorCount, IncrementCount, DeltaX1, DeltaY1, DeltaX2,
    DeltaY2, BackColor, GraphDriver, GraphMode, MaxDelta,
    StartX, StartY, TestX, TestY, c1, c2, c3, c4: INTEGER;
    Lline: ARRAY [1..Memory] OF RECORD
        LX1,LY1: INTEGER; LX2,LY2: INTEGER; LColor : ColorList; END;
    a, b, c, d, e, f, g, h, i, j, k, l, w, x, y, MaxX,
    MaxY, Error, GMode, GDriver, kaart, TekstSoort, pt: INTEGER;
    spoints : array[1..7] of Pointtype;
    cpoints : array[1..21] of Pointtype;
    Stoppen: BOOLEAN;
    Keytype: BYTE;
    Verder, Terug: Str80;
    ch: CHAR;
```

## De procedures.

Voorafgaand aan het eigenlijke programma worden alle procedures gedefinieerd. In het programma kan dan de naam van een procedure worden aangeroepen, waarna de procedure zijn functie vervult. Na het doorlopen van de procedure wordt teruggekeerd naar de plaats van aanroep. Het is mogelijk om in een procedure een voorafgaande procedure aan te roepen. Een procedure die pas later in het programma wordt gedefinieerd is echter niet oproepbaar.

Nu volgt een korte bespreking van elk van de procedures, voorzover deze van belang zijn voor de uitvoering van het programma. Eventueel gebruikte lijnen {-----} zijn voor het aanbrengen van een scheiding tussen de procedures. Deze lijnen zijn ook in het eigenlijke programma gebruikt en zijn slecht ter verduidelijking. Vanwege de { } tekens zal de Pascal compiler ze negeren.

De procedure 'Beep' produceert een kort geluidsignaal. Deze procedure wordt in het verloop van het programma meerdere keren gebruikt, bijvoorbeeld als een verkeerde toets wordt ingedrukt of als een menu wordt afgesloten. De gebruikte statements komen uit de unit Crt.

```
PROCEDURE Beep;  
  BEGIN  
    Sound(300); Delay(200); NoSound  
  END;
```

De nu volgende procedures zijn gebruikt voor de grafische effecten op het laatste scherm. Het idee voor deze effecten is overgenomen uit een voorbeeld dat is bijgeleverd bij de TurboPascal documentatie. Deze procedures zijn niet van belang voor de uitvoering van het programma en zullen dan ook niet verder worden besproken. De gebruikte commando's komen uit de unit Graph.

```

PROCEDURE TextPort;
BEGIN
  SetViewPort(0,0, Xmax, Ymax, ClipOn);
END;

{-----}

```

```

PROCEDURE FullPort;
BEGIN
  SetViewPort(0, 0, Xmax, Ymax-30, ClipOn);
  SetColor(MaxColors);
END;

{-----}

```

```

PROCEDURE Init;
BEGIN
  CurrentLine:=1;
  ColorCount:=0;
  IncrementCount:=0;
  MaxColors:=GetMaxColor; BackColor:=0;
  Xmax:=GetMaxX; Ymax:=GetMaxY;
  ViewXmax:=Xmax-2; ViewYmax:=Ymax-2;
  StartX:=Xmax div 2; StartY:=Ymax div 2;
  FOR I:=1 TO Memory DO WITH Lline[I] DO BEGIN
    LX1:=StartX; LX2:=StartX;
    LY1:=StartY; LY2:=StartY;
  END;
  X1:=StartX;
  X2:=StartX;
  Y1:=StartY;
  Y2:=StartY;
END;

{-----}

```

```

PROCEDURE AdjustX(var X,DeltaX: integer);
BEGIN
  TestX:=X+DeltaX;
  IF (TestX<1) OR (TestX>ViewXmax) THEN
  BEGIN
    TestX:=X;
    DeltaX:=-DeltaX;
  END;
  X:=TestX;
END;

```

{-----}

```
PROCEDURE AdjustY(var Y,DeltaY: integer);
BEGIN
  TestY:=Y+DeltaY;
  IF (TestY<1) OR (TestY>ViewYmax) THEN
    BEGIN
      TestY:=Y;
      DeltaY:=-DeltaY;
    END;
  Y:=TestY;
END;
```

{-----}

```
PROCEDURE SelectNewColors;
BEGIN
  Colors[1]:=Random(MaxColors)+1;
  Colors[2]:=Random(MaxColors)+1;
  Colors[3]:=Random(MaxColors)+1;
  Colors[4]:=Random(MaxColors)+1;
  ColorCount:=3*(1+Random(5));
END;
```

{-----}

```
PROCEDURE SelectNewDeltaValues;
BEGIN
  DeltaX1:=Random(MaxDelta)-(MaxDelta Div 2);
  DeltaX2:=Random(MaxDelta)-(MaxDelta Div 2);
  DeltaY1:=Random(MaxDelta)-(MaxDelta Div 2);
  DeltaY2:=Random(MaxDelta)-(MaxDelta Div 2);
  IncrementCount:=2*(1+Random(4));
END;
```

{-----}

```
PROCEDURE SaveCurrentLine(CurrentColors: ColorList);
BEGIN
  WITH Lline[CurrentLine] DO
    BEGIN
      LX1:=X1;
      LY1:=Y1;
      LX2:=X2;
      LY2:=Y2;
```

```

    LColor:=CurrentColors;
  END;
END;

```

```
{-----}
```

```

PROCEDURE Draw(x1,y1,x2,y2,color:word);
BEGIN
  SetColor(color);
  Graph.Line(x1,y1,x2,y2);
END;

```

```
{-----}
```

```

PROCEDURE Regenerate;
BEGIN
  FullPort;
  FOR I:=1 TO Memory DO WITH Lline[I] DO
  BEGIN
    Draw(LX1,LY1,LX2,LY2,LColor[1]);
    Draw(ViewXmax-LX1,LY1,ViewXmax-LX2,LY2,LColor[2]);
    Draw(LX1,ViewYmax-LY1,LX2,ViewYmax-LY2,LColor[3]);
    Draw(ViewXmax-LX1,ViewYmax-LY1,ViewXmax-LX2,ViewYmax-LY2,LColor[4]);
  END;
END;

```

```
{-----}
```

```

PROCEDURE Updateline;
BEGIN
  Inc(CurrentLine);
  IF CurrentLine > Memory THEN CurrentLine:=1;
  Dec(ColorCount);
  Dec(IncrementCount);
END;

```

```
{-----}
```

```

PROCEDURE DrawCurrentLine;
BEGIN
  c1:=Colors[1];
  c2:=Colors[2];
  c3:=Colors[3];
  c4:=Colors[4];
  IF MaxColors=1 THEN
  BEGIN

```



```

    c2:=c1; c3:=c1; c4:=c1;
  END;
  Draw(X1,Y1,X2,Y2,c1);
  Draw(ViewXmax-X1,Y1,ViewXmax-X2,Y2,c2);
  Draw(X1,ViewYmax-Y1,X2,ViewYmax-Y2,c3);
  IF MaxColors=3 THEN c4:=Random(3)+1;
  Draw(ViewXmax-X1,ViewYmax-Y1,ViewXmax-X2,ViewYmax-Y2,c4);
  SaveCurrentLine(Colors);
END;

```

```
{-----}
```

```

PROCEDURE EraseCurrentLine;
BEGIN
  WITH Lline[CurrentLine] DO
  BEGIN
    Draw(LX1,LY1,LX2,LY2,0);
    Draw(ViewXmax-LX1,LY1,ViewXmax-LX2,LY2,0);
    Draw(LX1,ViewYmax-LY1,LX2,ViewYmax-LY2,0);
    Draw(ViewXmax-LX1,ViewYmax-LY1,ViewXmax-LX2,ViewYmax-LY2,0);
  END;
END;

```

```
{-----}
```

```

PROCEDURE DoArt;
BEGIN
  SelectNewColors;
  REPEAT
    EraseCurrentLine;
    IF ColorCount=0 THEN SelectNewColors;
    IF IncrementCount=0 THEN SelectNewDeltaValues;
    AdjustX(X1,DeltaX1); AdjustX(X2,DeltaX2);
    AdjustY(Y1,DeltaY1); AdjustY(Y2,DeltaY2);
    IF Random(5)=3 THEN
      BEGIN
        x1:=(x1+x2) div 2;
        y2:=(y1+y2) div 2;
      END;
    DrawCurrentLine;
    Updateline;
  UNTIL KeyPressed;
END;

```

De procedure 'ClearKb' is bedoeld om de buffer van het toetsenbord leeg te maken voordat tijdens het verloop van het programma een toets moet worden ingedrukt. Op deze manier wordt bereikt dat toetsen die te vroeg worden ingedrukt geen effect hebben. De commando's 'KeyPressed' en 'ReadKey' komen uit de unit Crt.

```
PROCEDURE ClearKb;  
  BEGIN  
    WHILE KeyPressed DO ch:=ReadKey;  
  END;
```

De grafische mogelijkheden van de computer en van TurboPascal zijn niet zonder meer te gebruiken. Het is noodzakelijk om de computer eerst in de grafische mode te schakelen (zie literatuur [4]). Dit gebeurt met de procedure 'StartGrafisch'.

Tijdens het schrijven van het programma is gebleken dat de voor de tekeningen en tekst benodigde ruimte vrij groot is. Daarom is het minimum aantal pixels voor dit programma op 640x400 gesteld. Van de veel gebruikte grafische kaarten vallen hieronder de VGA -, de superVGA - en de Att grafische kaart.

TurboPascal kan meestal de grafische kaart detecteren, maar in bepaalde gevallen zal toch de verkeerde kaart worden gekozen, meestal CGA. Daarom is aan het begin van het programma, vóór het omschakelen naar de grafische mode, een keuzemogelijkheid aangebracht.

Voor het gebruik van de grafische kaart is een file nodig. Deze file wordt ingelezen als het commando 'InitGraph' wordt gegeven. Hierbij moeten de grafische kaart, de mode van de resolutie en de drive worden opgegeven (dat '' is opgegeven wil zeggen dat de files gezocht worden op de drive van waaruit het programma is gestart). Er wordt hier ook de mogelijkheid geboden vroegtijdig te stoppen met het programma. Na dit commando wordt een contrôle uitgevoerd op het resultaat van de initialisatie, waarbij ook de oorzaak van een eventuele storing zal worden gemeld (een lijst hiervan is te vinden in literatuur [2]). Tenslotte wordt nog aan enige variabelen een waarde gegeven. Deze variabelen zijn nodig voor het positioneren van onder andere het kader, dat bij verschillende grafische kaarten toch op dezelfde plaats moet staan, en voor de standaard kleur- en tekstinstellingen. De belangrijkste commando's uit deze procedure komen uit de unit Graph.

```

PROCEDURE StartGrafisch;
BEGIN
  WriteLn('Deze versie is geschikt voor XT/AT computers met een beeldschermresolutie');
  WriteLn('van minimaal 640x400 pixels, in kleur of monochroom'); WriteLn;
  WriteLn('Welke grafische kaart gebruikt de computer?');
  WriteLn;
  WriteLn('Type 0, 1, 2, 3 of 4, gevolgd door <return>'); WriteLn;
  WriteLn('0 = stoppen met dit programma;');
  WriteLn('1 = (super)VGA met kleurenmonitor;');
  WriteLn('2 = VGA met monochroommonitor;');
  WriteLn('3 = Att400 (Olivetti monochroom)');
  WriteLn('4 = laat de computer zelf zoeken'); WriteLn;
  WriteLn('let op: optie 4 kan een foutief resultaat geven!');
  WriteLn; Write('Uw keuze: '); Read(kaart);
  CASE kaart OF 0: Halt;
    1: BEGIN GDriver:=VGA; GMode:=VGAHi END;
    2: BEGIN GDriver:=VGA; GMode:=VGAHi END;
    3: BEGIN GDriver:=Att400; GMode:=Att400Hi END;
    4: GDriver:=Detect
    ELSE GDriver:=Detect
  END {CASE};
  ClrScr;
  WriteLn('Initialiseren van de graphics...');
  InitGraph(GDriver, GMode, '');
  Error:=GraphResult;
  IF Error < 0 THEN
    BEGIN
      ClrScr; Beep;
      WriteLn('Fout in initialisatie van de graphics');
      WriteLn('Omschrijving: '+GraphErrorMsg(Error));
      WriteLn('Programma gestopt, terug naar DOS...');
      Delay(5000); Halt
    END;
  MaxX:=GetMaxX;
  MaxY:=GetMaxY;
  IF (kaart=1) OR (kaart=2) THEN BEGIN pt:=42; MaxDelta:=16 END;
  IF (kaart=3) THEN BEGIN pt:=35; MaxDelta:=7 END;
  IF (kaart=4) THEN BEGIN pt:=round((MaxY/400)*40-6); MaxDelta:=16 END;
  SetBkColor(Blue); TekstSoort:=1;
END;

```

De procedure 'Afschuifomgeving' tekent een figuur in een kader op het scherm. Deze figuur is in een aparte procedure ondergebracht omdat deze figuur meerdere keren wordt gebruikt, onder andere bij de twee simulaties en het scherm over de secundaire deformatie. De commando's komen uit de unit Graph.

```
PROCEDURE AfschuifOmgeving;
BEGIN
  Rectangle(10,10,400,277);
  SetLineStyle(SolidLn,0,ThickWidth);
  Line(10,120,90,120); Line(90,120,112,10); Line(330,240,400,240);
  Line(330,240,376,10); Line(330,240,400,225);
  SetFillStyle(InterLeaveFill,White); FloodFill(399,11,White);
  SetLineStyle(SolidLn,0,NormWidth); Line(90,120,330,240);
END;
```

In de grafische mode van TurboPascal is het mogelijk om teksten op het scherm te zetten in verschillende lettertypen. De procedure hiervoor is echter nogal onhandig in het gebruik (zie literatuur [4]). Daarom is een procedure geschreven waardoor het mogelijk is om via regels en kolommen tekst te plaatsen. Omdat het nogal bewerkelijk is om steeds te tekstsoort te veranderen wanneer meerdere soorten tekst in één scherm worden gebruikt, is voor de verschillende soorten tekst een eigen procedure gemaakt: 'PutStr' voor kleine tekst in menu's etc. en 'Tekst' voor een grotere tekstsoort. De grotere tekstsoort heeft een font-file nodig die op de diskette staat. Het programma controleert of een disk met deze file aanwezig is en geeft zonodig een foutmelding. Voordat tekst wordt geplaatst wordt een balk met de kleur van de achtergrond ter breedte van de tekst geplaatst. De commando's komen uit de unit Graph, en worden verder behandeld in literatuur [4].

```
PROCEDURE PutStr(txt: Str80; r,k: BYTE);
BEGIN
  SetTextStyle(0,0,1); {Bitmapped}
  SetTextJustify(0,2);
  SetFillStyle(1,0);
  x:=(k-1)*8; y:=(r-1)*10;
  Bar(x,y,x+TextWidth(txt)-1,y+TextHeight(txt));
  OutTextXY(x,y,txt);
END;
```

```
{-----}
```

```
PROCEDURE Tekst(txt: Str80; r,k: BYTE);
BEGIN
  SetTextStyle(TekstSoort,0,1);
  Error:=Graphresult;
  IF Error<0 THEN
    BEGIN
      Beep; Beep; Beep;
      PutStr('Fout in files voor lettertype !!!!',2,2);
      PutStr('Omschrijving: '+GraphErrorMsg(Error),3,2);
      PutStr('Controleer of Uw drive gereed is om te lezen en druk op een toets.',4,2);
      ch:=ReadKey;
      SetTextStyle(TekstSoort,0,1);
    END;
  SetTextJustify(0,2);
  SetFillStyle(1,0);
  x:=(k-1)*8; y:=((r-1)*22)-8;
  Bar(x,y,x+TextWidth(txt)-1,y+TextHeight(txt));
  OutTextXY(x,y,txt);
END;
```

Tijdens het verloop van het programma kunnen de achtergrondkleur en het type van de tekst worden gewijzigd. Hiervoor dient de procedure 'Instellingen'. Deze procedure plaatst een keuzemenu op het scherm en verandert de kleur en de tekst aan de hand van de ingevoerde keuzes. Zoals al eerder is gezegd kan een procedure niet worden gebruikt als deze ná een andere is gedefinieerd. Daarom komt deze procedure in het eigenlijke programma tussen de twee voorgaande tekstprocedures in. Reden hiervoor is dat de procedure 'PutStr' wordt gebruikt in 'Instellingen' en dat in 'Tekst' bepaalde keuzes uit 'Instellingen' worden gebruikt !

De commando's 'GetMem', 'GetImage' en 'PutImage' slaan het gedeelte van het scherm waar het menu overheen wordt gezet op, en plaatsen het terug, nadat het menu wordt afgesloten, omdat anders slechts een zwart vlak zou overblijven. De commando's komen voornamelijk uit de unit Graph (meer uitleg in literatuur [4]).

```
PROCEDURE Instellingen;
VAR keuze: CHAR;
    MenuPtr: pointer;
BEGIN
  GetMem(MenuPtr,ImageSize(140,40,460,260));
```

```
GetImage(140,40,460,260,MenuPtr^);
SetFillStyle(SolidFill,Black); Bar(140,40,460,260);
SetFillStyle(InterLeaveFill,Yellow); Bar(170,70,450,250);
SetFillStyle(SolidFill,Black); Bar(160,60,440,240);
SetColor(Yellow); Rectangle(162,62,438,238);
PutStr(' INSTELLINGEN ',7,30);
PutStr('Type 1 voor lettertype',9,23);
PutStr('Triplex.',10,23);
PutStr('Type 2 voor lettertype',12,23);
PutStr('Sans Serif.',13,23);
PutStr('Type 3 voor achtergrond',15,23);
PutStr('blauw.',16,23);
PutStr('Type 4 voor achtergrond',18,23);
PutStr('zwart',19,23);
PutStr('einde menu: type 5',21,23);
ClearKb;
WHILE NOT (keuze='5') DO
  BEGIN
    keuze:=ReadKey;
    IF (keuze='1') THEN TekstSoort:=1;
    IF (keuze='2') THEN TekstSoort:=3;
    IF (keuze='3') THEN SetBkColor(Blue);
    IF (keuze='4') THEN SetBkColor(Black);
    PutStr('uw keuze: '+keuze,23,23);
    Delay(500); PutStr('          ',23,23);
  END;
SetColor(White);
ClearKb;
PutImage(140,40,MenuPtr^,CopyPut);
END;
```

Tijdens het verloop van het programma wordt rechtsonder in de hoek van het scherm de tijd geplaatst. Deze tijd wordt met het commando 'GetTime' uit de unit Dos ingelezen vanuit het besturingssysteem (literatuur [2]). Als het programma met behulp van de autoexecute-file op de disk wordt gestart (zie 'toelichting op het programma verspaningsmodel') is het mogelijk dat geen tijd kan worden ingelezen. Er zal dan geen tijd worden aangegeven omdat anders de tijd met 0:00 zou beginnen om vervolgens op te lopen (0:01, 0:02 etc.). Bij de meeste computers, zeker AT's, zal er geen probleem optreden.

```

PROCEDURE Tijd;
VAR Hour, Minute, Second, Sec100: WORD;
    Ur, Minuut: STRING;
BEGIN
    GetTime(Hour,Minute,Second,Sec100);
    Str(Hour,Uur); Str(Minute,Minuut);
    SetColor(LightGray);
    IF (Minute<10) THEN Minuut:='0'+Minuut;
    IF (Uur<>'0') THEN PutStr(' '+Uur+' '+Minuut+' ',pt+4,72);
    {geen tijd gevonden-->alle variabelen 0-->geen tijd in beeld}
    SetColor(White);
END;

```

Elk scherm wordt omgeven door hetzelfde kader. Dit kader bevat een menubalk, waarin de bedieningsinstructies, de tijd en het TUE logo zullen worden geplaatst. De kleur van dit kader wijkt af van de kleur van de overige tekst (grijs in plaats van wit). Voordat het kader wordt geplaatst, wordt eerst het scherm leeggemaakt, omdat anders in de grafische mode de opeenvolgende schermen over elkaar heen geplaatst worden. De procedure 'Kaders' wordt bij elke procedure die tekst of figuren op het scherm plaatst als eerste aangeroepen. In deze procedure zijn commando's gebruikt uit de unit Graph.

```

PROCEDURE Kaders;
BEGIN
    ClearDevice;
    SetColor(LightGray);
    b:=ROUND((MaxY/400)*65);
    SetLineStyle(SolidLn,0,NormWidth);
    Rectangle(0,0,MaxX,MaxY);
    Rectangle(3,3,MaxX-2,MaxY-2);
    Rectangle(7,MaxY-b,MaxX-7,MaxY-7);
    SetTextStyle(0,0,2); SetTextJustify(0,2);
    OutTextXY(MaxX-65,MaxY-50,'tu');
    Line(MaxX-33,MaxY-45,MaxX-21,MaxY-45);
    Arc(MaxX-35,MaxY-46,0,163,14); Arc(MaxX-35,MaxY-46,255,335,14);
    PutStr(' Alt-H = help ',pt-1,65);
    Tijd; SetColor(White);
END;

```

De teksten die op het scherm moeten verschijnen kunnen worden geplaatst met de al eerder behandelde procedures. In de tekenset van deze fonts zitten echter geen griekse letters. Deze moeten daarom apart worden gemaakt met behulp van de standaard grafische elementen: lijnen, cirkels en bogen. De eerste twee griekse letters zijn zodanig gemaakt, dat ze met regels en kolommen worden geplaatst. Bij de andere letters is dit niet nauwkeurig genoeg (beperkt aantal plaatsen waar een letter kan staan) en moet de letter met pixelcoördinaten worden geplaatst.

```
PROCEDURE SchrijfFi(r,k: BYTE);
BEGIN
  Circle((k-1)*8,((r-1)*22+2),6);
  Line((k-1)*8,((r-1)*22)-6,(k-1)*8,((r-1)*22)+10)
END;
```

{-----}

```
PROCEDURE SchrijfDelta(r,k: BYTE);
BEGIN
  Line((k-1)*8+3,(r-1)*22+8,(k-1)*8+13,(r-1)*22+8);
  Line((k-1)*8+3,(r-1)*22+8,(k-1)*8+8,(r-1)*22);
  Line((k-1)*8+8,(r-1)*22,(k-1)*8+13,(r-1)*22+8);
END;
```

{-----}

```
PROCEDURE SchrijfGamma(r,k: INTEGER);
BEGIN
  Arc(r,k,344,40,30); Arc(r+57,k,140,196,30)
END;
```

{-----}

```
PROCEDURE SchrijfGammajc(r,k: INTEGER);
BEGIN
  Arc(r,k,344,40,16); Arc(r+30,k,140,196,16)
END;
```

{-----}

```
PROCEDURE SchrijfBeta(r,k: INTEGER);
BEGIN
  Arc(r+4,k,270,180,4); Arc(r+4,k+7,180,90,4);
```



```
Line(r,k,r,k+18);
END;
```

```
{-----}
```

```
PROCEDURE SchrijfLabda(r,k: INTEGER);
BEGIN
  Line(r,k+2,r+8,k+20); Line(r,k+20,r+4,k+10);
END;
```

De procedure 'Stop' beëindigt het programma. Voordat het programma definitief stopt wordt nog een scherm geproduceerd waarop bewegende gekleurde lijnen te zien zijn (de procedures die hiervoor gemaakt zijn, zijn al eerder genoemd, maar niet verder behandeld). Na een druk op een toets wordt teruggekeerd naar DOS. Terugkeren naar het programma is niet mogelijk. De procedure is aan te roepen met Alt-Q (dit wordt verderop in het verslag behandeld).

```
PROCEDURE Stop;
BEGIN
  SetBkColor(Black); ClearDevice;
  Init; TextPort;
  PutStr('programma gestopt, druk een toets om terug te gaan naar DOS',pt+5,10);
  FullPort; DoArt;
  CloseGraph; Halt
END;
```

Het programma bevat een help-menu, dat een kort overzicht geeft van de betekenis van de verschillende mogelijkheden en aangeeft welke toetsencombinaties daarvoor gebruikt kunnen worden. Ook hier wordt het scherm hersteld nadat het menu is afgesloten. In deze procedure zijn commando's gebruikt uit Graph.

```
PROCEDURE Menu;
VAR MenuPtr: pointer;
BEGIN
  GetMem(MenuPtr,ImageSize(140,40,460,260));
  GetImage(140,40,460,260,MenuPtr^);
  SetFillStyle(SolidFill,Black); Bar(140,40,460,260);
  SetFillStyle(InterLeaveFill,Yellow); Bar(170,70,450,250);
```

```

SetFillStyle(SolidFill,Black); Bar(160,60,440,240);
SetColor(Yellow); Rectangle(162,62,438,238);
PutStr(' HELPMENU ',7,34);
PutStr('Alt-Q      = stoppen',9,23);
PutStr('Alt-H      = dit menu',10,23);
PutStr('Alt-I      = scherm instellen',11,23);
PutStr('Alt-Shift-F9 = (deel van) scherm',12,23);
PutStr('          scherm naar .wpg',13,23);
PutStr('          file',14,23);
PutStr('Shift-PRTSC = scherm uitprinten',15,23);
PutStr('Veranderen van scherm: cursor-',17,23);
PutStr('toetsen en <return> zoals onder',18,23);
PutStr('in de menubalk aangegeven.',19,23);
PutStr('Alle toetsen zijn buiten werking',20,23);
PutStr('als er geen tekst in de menubalk',21,23);
PutStr('staat !!',22,23);
PutStr('Druk op een toets...',23,35);
WHILE NOT KeyPressed DO
  BEGIN
    {wacht}
  END;
SetColor(White);
ClearKb; Tijd;
PutImage(140,40,MenuPtr^,CopyPut);
END;

```

De besturing van het programma gebeurt met de cursortoetsen en met een beperkt aantal toetsencombinaties. Deze toetsencombinaties kunnen gedefinieerd worden met commando's uit de unit Crt. De procedure 'GetKey' kijkt of er een toets is ingedrukt, en zo ja, welke. Als deze toets een betekenis heeft voor het programma wordt de bijbehorende actie uitgevoerd, anders is een geluidssignaal te horen. Elke bestaande toetsencombinatie heeft een nummer. Door afhankelijk van dit nummer een actie uit te voeren kan dus aan een toets een functie worden toegekend.

Alle toetsen die geen speciale betekenis hebben zijn dus uitgeschakeld, behalve de reset door Ctrl-Alt-Del en de door graphics geactiveerde Shift-Prtsc.

De nummers zijn als volgt:

- |   |                |
|---|----------------|
| * #72, #75, #77 en #80 zijn cursortoetsen | * #23 is Alt-I |
| * #13 is de return-toets                  | * #35 is Alt-H |
| * #16 is Alt-Q                            |                |

De conditie 'ch OF #0' controleert of de ingedrukte toets uit de ASCII-set komt (ch <> #0) of uit de extended key codes (de functietoetsen en cursor-toetsen (ch = #0)). Voor een meer uitvoerige beschrijving wordt verwezen naar literatuur [2].

Deze procedure is de basis voor de besturing van het het programma.

```
PROCEDURE GetKey(VAR Keytype:BYTE);
BEGIN
  Keytype:=0; Tijd;
  REPEAT
    ch:=ReadKey;
    CASE ch OF #0: BEGIN
      ch:=ReadKey;
      CASE ch OF #72: Keytype:=1;
        #75: Keytype:=2;
        #77: Keytype:=3;
        #80: Keytype:=4;
        #16: Stop;
        #35: Menu;
        #23: Instellingen;
      END {CASE}
    END;
    ELSE IF ch=#13 THEN Keytype:=5
  END {CASE};
  IF Keytype=0 THEN Beep
  UNTIL Keytype<>0
END;
```

De procedures die hierna volgen zullen elk één scherm of een serie bij elkaar behorende schermen (bv. QuickStop en Temperatuur) produceren met tekst en een afbeelding. Deze procedures maken allemaal gebruik van dezelfde elementen zoals tekst, lijnen, rechthoeken, arceringen en dergelijke. Verdere uitleg is bij de meeste procedures niet nodig. Uitzondering vormen de procedures met bewegende beelden. Daar zal meer uitleg worden gegeven.

Procedure 'StartScherm' produceert het eerste scherm.

```

PROCEDURE Startscherm;
BEGIN
  Kaders;
  Tekst('      TECHNISCHE UNIVERSITEIT EINDHOVEN ',2,3);
  Tekst('      FACULTEIT WERKTUIGBOUWKUNDE, VAKGROEP WPA ',3,3);
  Tekst('      PRODUKTIE TECHNOLOGIE ',5,3);
  Tekst('      Het verspaningsmodel van Merchant ',7,3);
  PutStr('Dit softwarepakket bevat een geprogrammeerde instructie voor het ',20,8);
  PutStr('verspaningsmodel van Merchant. Hierbij wordt informatie gegeven ',21,8);
  PutStr('over het model en theorie die eraan ten grondslag ligt. Het ',22,8);
  PutStr('programma wordt gebruikt als aanvulling op het vak "Productie-',23,8);
  PutStr('technologie 1" in het eerste jaar.',24,8);
  PutStr('Het programma is gemaakt in het kader van een onderzoeksopdracht ',25,8);
  PutStr('aan de faculteit Werktuigbouwkunde van de TU Eindhoven.',26,8);
  PutStr('Voor meer informatie: prof.dr.ir. A.C.H. van der Wolf ',30,8);
  PutStr('      A.J. van der Net ',31,8);
  PutStr('Eindhoven, April 1992 ',pt-3,55);
  PutStr('Verder,pt,3);
  ClearKb;
  REPEAT
    GetKey(Keytype);
    IF Keytype=3 THEN i:=1 ELSE Beep
  UNTIL Keytype=3;
  Tijd;
END;

```

Procedure 'Quickstop' produceert twee schermen. Tussendoor moet een toets worden ingedrukt.

```

PROCEDURE Quickstop;
BEGIN
  Kaders;
  Tekst('Het fysische verspaningsproces is moeilijk waar te nemen.',2,3);
  Tekst('Met zgn. quick-stop devices is het mogelijk om het proces ',3,3);
  Tekst('stil te zetten, waarna de structuur onderzocht kan worden.',4,3);
  Tekst('Er is dan te zien dat het gehele vervormingsproces plaats ',5,3);
  Tekst('vindt in een smal gebied rondom de lijn AB (fig.1). ',6,3);
  Tekst('Onderzoek heeft aangetoond dat verspanen in hoofdzaak een ',7,3);
  Tekst('proces van plastische deformatie van het materiaal is.',8,3);
  Tekst('Spaanmateriaal bestaat uit sterk plastisch vervormd ',9,3);
  Tekst('werkstukmateriaal. De regelmatige roosterstructuur vertoont ',10,3);

```

```

Tekst('een verschuiving van vele malen de roosterafstand onder een',11,3);
Tekst('hoek met de bewegingsrichting van de beitel, nl. gebied AB.',12,3);
Tekst('Dit gebied heet de "primaire deformatiezone" of "afschuifzone".',13,3);
Tekst('De lijnen evenwijdig aan het oorspronkelijke oppervlak',14,3);
Tekst('in fig. 1 zijn kunstmatig aangebracht.',15,3);
PutStr('druk op '+#26+' om het resultaat van een',pt+1,6);
PutStr('quick-stop test te zien (fig. 1)',pt+2,6);
ClearKb; Tijd;
GetKey(KeyType);
Kaders;
SetFillStyle(10,LightGray);Bar(45,45,345,285);
SetFillStyle(9,White);Bar(285,45,345,225);
SetFillStyle(1,Black);Bar(45,45,135,150);
SetColor(White);
Rectangle(45,45,135,150);Rectangle(45,45,150,158);
Rectangle(45,45,180,173);Rectangle(45,45,210,188);
Rectangle(45,45,240,203);Rectangle(45,45,270,218);
Rectangle(45,45,345,285);Rectangle(285,45,345,225);
Line(45,233,345,233);Line(45,243,345,243);
Line(45,253,345,253);Line(135,150,285,225);
PutStr('<---',13,39);PutStr('beitel',15,38);
PutStr('spaan',11,25);PutStr('werkstuk',28,20);
PutStr('B',15,16);PutStr('A',22,37);Tekst('fig. 1',5,47);
Tekst('Resultaat van een',6,47);
Tekst('quick-stop test.',7,47);
PutStr(Verder,pt,3);
PutStr(Terug,pt+1,3);
PutStr('#17+#196+#217+' terug naar startscherm',pt+2,3);
ClearKb; Tijd;
REPEAT
  GetKey(Keytype);
  CASE Keytype of 2: QuickStop;
                 3: i:=2;
                 5: i:=0;
                 ELSE Beep
                 END {CASE};
UNTIL (keytype=2) OR (keytype=3) OR (keytype=5);
END;

```

Bij de procedure 'AfschuifSimulatie' is enige uitleg noodzakelijk. De basis van deze procedure is een normaal scherm met tekst en een figuur. In de figuur is echter een bewegend deel te zien. Dit deel moet vervormen volgens de theorie van het afschuiven (zie o.a. literatuur [5]).

Dit is bereikt door een figuur te definiëren die bestaat uit zes punten, die een polygoon vormen. Voor het tekenen van de polygoon bestaat in TurboPascal een standaardprocedure uit de unit Grap, nl. 'Drawpoly'. Het maken van een simulatie is dus teruggebracht tot het bepalen van de plaats van de zes punten in het verloop van de tijd. De uitgangssituatie is te beschrijven met geometrische formules bij een bekende afschuifhoek. Alle punten kunnen nu horizontaal worden opgeschoven totdat de voorste punten de afschuifzone bereiken. De voorste twee punten gaan daarna omhoog, de achterste twee blijven horizontaal bewegen, totdat ook deze de afschuifzone bereiken. De vereiste vervorming wordt bereikt door de beweging van de twee middelste punten, die op de twee zijden van het vierkant liggen.

Deze twee punten bewegen met de andere punten mee, behalve wanneer de voorste punten zich achter, en de achterste punten zich vóór de afschuifzone bevinden. Wanneer dit zo is verplaatsen deze punten zich langs de lijn die de afschuifzone vormt. Wanneer nu steeds de nieuwe punten worden berekend, de oude figuur wordt weggehaald en een nieuwe geplaatst, wordt de indruk gewekt dat het vierkant afschuift. In deze cyclus is een vertragingstijd ingebouwd om te voorkomen dat de simulatie te snel gaat.

De procedure is gesplitst in twee delen; de beweging zelf is ingebouwd in de procedure 'Beweeg', die in de procedure 'AfschuifSimulatie' kan worden opgeroepen. Deze opzet voorkomt dat steeds de gehele simulatie ongevraagd wordt getoond.

```

PROCEDURE Beweeg;
BEGIN
  a:=x0+round(10*cos(fi)); b:=y0+round(10*sin(fi)); c:=x0; d:=y0;
  e:=a+round(10*cos(pi/2-fi)); f:=b-round(10*sin(pi/2-fi));
  g:=c+round(10*cos(pi/2-fi)); h:=d-round(10*sin(pi/2-fi));
  i:=e; j:=f; k:=g; l:=h;
  WHILE h > 15 DO
    BEGIN
      IF a < -150+round(b*(cos(fi)/sin(fi))) THEN a:=a+5
        ELSE BEGIN a:=a+1; b:=b-5 END;
      IF c < -150+round(d*(cos(fi)/sin(fi))) THEN c:=c+5
        ELSE BEGIN c:=c+1; d:=d-5 END;
      IF e < -150+round(f*(cos(fi)/sin(fi))) THEN BEGIN e:=e+5; i:=i+5 END
        ELSE BEGIN e:=e+1; f:=f-5 END;
      IF g < -150+round(h*(cos(fi)/sin(fi))) THEN BEGIN g:=g+5; k:=k+5 END
        ELSE BEGIN g:=g+1; h:=h-5 END;
    
```

```

IF (a < -150+round(b*(cos(fi)/sin(fi)))) AND
  (e > -150+round(f*(cos(fi)/sin(fi))))
THEN BEGIN i:=i+4; j:=j+2; k:=k+4; l:=l+2;
  setcolor(white); line(90,120,330,240) END;
IF (a >= -150+round(b*(cos(fi)/sin(fi)))) AND
  (e >= -150+round(f*(cos(fi)/sin(fi))))
THEN BEGIN i:=i+1; j:=j-5; k:=k+1; l:=l-5;
  setcolor(white); line(90,120,330,240) END;
spoints[1].x:=c;spoints[1].y:=d;
spoints[2].x:=k;spoints[2].y:=l;
spoints[3].x:=g;spoints[3].y:=h;
spoints[4].x:=e;spoints[4].y:=f;
spoints[5].x:=i;spoints[5].y:=j;
spoints[6].x:=a;spoints[6].y:=b;
spoints[7].x:=c;spoints[7].y:=d;
SetColor(Yellow);DrawPoly(7,spoints);
Delay(250);
SetColor(Black);DrawPoly(7,spoints);
END;
SetColor(Yellow);Drawpoly(7,spoints);SetColor(White);
SchrijfGamma(263,102);PutStr('s',11,38);
SetLineStyle(DottedLn,0,NormWidth);Line(a,b,a+60,b-90);
Line(a,b,a-32,b+48);Arc(a,b,50,115,50);
SetLineStyle(SolidLn,0,NormWidth);
Line(a-27,b+40,a-20,b+44);Line(a-20,b+44,a-27,b+53);
ClearKb;Tijd;
GetKey(Keytype);
SetColor(Black);Drawpoly(7,spoints);
SchrijfGamma(263,102);PutStr('s',11,38);
SetLineStyle(DottedLn,0,NormWidth);Line(a,b,a+60,b-90);
Line(a,b,a-32,b+48);Arc(a,b,50,115,50);
SetLineStyle(SolidLn,0,NormWidth);
Line(a-27,b+40,a-20,b+44);Line(a-20,b+44,a-27,b+53);
SetColor(White);
CASE Keytype OF 2: i:=1;
  3: i:=5;
  1: i:=3;
  4: i:=4;
  5: Beweeg;
  ELSE Beep;
  END {CASE};
END;

```

```

{-----}
PROCEDURE AfschuifSimulatie;
BEGIN
  Kaders; Afschuifomgeving;
  Tekst('A',6,13); Tekst('B',13,42); SchrijfFi(12,37);
  Line(330,240,300,240);Arc(330,240,130,190,25);
  SetLineStyle(DottedLn,0,NormWidth);
  Line(10,162,173,162);Line(10,220,288,220);
  Line(176,162,206,10);Line(288,220,329,10);
  SetLineStyle(SolidLn,0,NormWidth);
  Tekst('Het verspaningsproces',2,52);
  Tekst('kan worden voorgesteld',3,52);
  Tekst('zoals hier weergegeven.',4,52);
  Tekst(' is de afschuifhoek.',5,52);SchrijfFi(5,53);
  Tekst('Er treedt een hoekver-',6,52);
  Tekst('draaiing op.',7,52);
  SchrijfGamma(467,143); PutStr('s',15,64);
  Tekst('Het proces kan worden',9,52);
  Tekst('gedemonstreerd door',10,52);
  Tekst('middel van een',11,52);
  Tekst('animatie.',12,52);
  Tijd;
  SetLineStyle(solidln,0,normwidth);
  ClearKb; Tijd;
  PutStr(#17+#196+#217+' om de animatie te zien',pt,3);
  PutStr(Verder,pt+1,3); PutStr(Terug,pt+2,3);
  PutStr(#24+' = simulatie met behulp van cirkel',pt+3,3);
  PutStr(#25+' = informatie over de secundaire deformatiezone',pt+4,3);
  GetKey(Keytype);
  CASE Keytype OF 2: i:=1;
                 3: i:=5;
                 1: i:=3;
                 4: i:=4;
                 5: Beweeg;
                 ELSE Beep;
                 END {CASE};
END;

```

Ook bij de procedure 'CirkelSimulatie' is enige uitleg noodzakelijk. De basis van deze procedure is dezelfde als bij de AfschuifSimulatie. Het bewegende deel is nu echter een cirkel. Ook dit deel moet vervormen volgens de theorie van het afschuiven (zie o.a. literatuur [5]). Dit is bereikt door een figuur te definiëren die bestaat uit 21 punten, die een polygoon vormen.



De werkwijze, zoals gevolgd bij het vierkant werkt hier niet, omdat punten alleen door een rechte lijn verbonden kunnen worden. Er zijn dus meer punten nodig. De 21 met rechte lijnen verbonden punten geven samen een goede benadering van een cirkel. Voor het tekenen van de figuur is weer 'Drawpoly' gebruikt. Het bepalen van de plaats van de 21 punten in het verloop van de tijd is gedaan met geometrische formules bij een bekende afschuifhoek. Alle punten kunnen nu horizontaal worden opgeschoven totdat de voorste punten de afschuifzône bereiken. De punten die de afschuifzône passeren gaan daarna omhoog, de rest blijft horizontaal bewegen, totdat ook deze punten de afschuifzone bereiken. Wanneer nu steeds de nieuwe punten worden berekend, de oude figuur wordt weggehaald en een nieuwe geplaatst, wordt de indruk gewekt dat de cirkel afschuift. Ook in deze cyclus is een vertragingstijd ingebouwd om te voorkomen dat de simulatie te snel gaat. Omdat deze simulatie via een 'zijsprong' moet worden aangeroepen en nooit ongevraagd zal worden getoond is deze niet in twee delen gesplitst.

```

PROCEDURE CirkelSimulatie;
BEGIN
  Kaders;
  Tekst('A',6,13); Tekst('B',13,42); SchrijfFi(12,37);
  AfschuifOmgeving;
  Line(330,240,300,240);Arc(330,240,130,190,25);
  SetLineStyle(DottedLn,0,NormWidth);
  Line(10,162,173,162);Line(10,220,288,220);
  Line(177,162,207,10);Line(292,220,333,10);
  SetLineStyle(SolidLn,0,NormWidth);
  Tekst('De vorm of grootte van',2,52);
  Tekst('het materiaaldeel is',3,52);
  Tekst('niet van belang.',4,52);
  Tekst('Een cirkel is vanwege',5,52);
  Tekst('zijn alzijdige symme-',6,52);
  Tekst('trie zeer geschikt',7,52);
  Tekst('voor het vinden van',8,52);
  Tekst('de richting van de',9,52);
  Tekst('grootste rek.',10,52);
  Tekst('Een animatie geeft',11,52);
  Tekst('weer hoe het afschuif-',12,52);
  Tekst('mechanisme bij een',13,52);
  Tekst('cirkel verloopt.',14,52);
  SetLineStyle(SolidLn,0,normwidth);
  Tijd;

```

```

FOR w:=1 TO 21 DO
  BEGIN
    cpoints[w].x:=x00+round(100*cos((pi/10)*(w-1)));
    cpoints[w].y:=y00+round(100*sin((pi/10)*(w-1)));
  END;
WHILE (cpoints[16].y) > 18 DO
  BEGIN
    FOR w:=1 TO 21 DO
      BEGIN
        IF cpoints[w].x < -150+round(cpoints[w].y*(cos(fi)/sin(fi))) THEN
          cpoints[w].x:=cpoints[w].x+5
        ELSE BEGIN
          cpoints[w].x:=cpoints[w].x+1;
          cpoints[w].y:=cpoints[w].y-5;
          SetColor(White); Line(90,120,330,240)
        END;
      END;
    END;
    SetColor(Yellow);DrawPoly(21,cpoints);
    Delay(300);
    SetColor(Black);DrawPoly(21,cpoints);
  END;
SetColor(Yellow);DrawPoly(21,cpoints); SetColor(White);
ClearKb;Tijd;
PutStr(Terug,pt,3);
PutStr(#17+#196+#217+' om dit scherm nog eens te zien',pt+1,3);
REPEAT
  GetKey(Keytype);
  CASE Keytype OF 2:i:=2;
    5:i:=3
    ELSE Beep
  END{CASE};
UNTIL (Keytype=2) OR (Keytype=5);
END;

```

De procedure 'Secundair' geeft informatie over de secundaire deformatiezône. Dit scherm bevat alleen een figuur en tekst.

```

PROCEDURE Secundair;
BEGIN
  Kaders;
  Tekst('A',5,11); Tekst('B',13,40);
  Afschuifomgeving;
  MoveTo(90,120);LineTo(100,70);LineTo(130,115);

```

```

LineTo(230,135);LineTo(315,205);LineTo(312,125);
LineTo(345,50);LineTo(374,20);LineTo(330,240);
LineTo(280,242);LineTo(205,215);LineTo(130,180);
LineTo(95,145);LineTo(45,120);LineTo(90,120);
SetFillStyle(WideDotFill,White);
FloodFill(325,235,White);FloodFill(205,210,White);
PutStr('primaire deformatiezone '+#24,24,5);
PutStr('secundaire deformatiezone '+#26,8,15);
Tekst('Gebied AB heet de pri-',2,52);
Tekst('maire deformatiezone.',3,52);
Tekst('Er is ook een secun-',4,52);
Tekst('daire deformatiezone',5,52);
Tekst('Dit is het contactge-',6,52);
Tekst('bied tussen spaan en',7,52);
Tekst('spaanvlak. De defor-',8,52);
Tekst('matie die hier optreedt',9,52);
Tekst('is het gevolg van de',10,52);
Tekst('wrijving tussen spaan',11,52);
Tekst('en spaanvlak.',12,52);
PutStr(Terug,pt,3);
ClearKb;Tijd;
REPEAT
  GetKey(Keytype);
  IF Keytype=2 THEN i:=2 ELSE Beep;
UNTIL (Keytype=2);
END;

```

De procedure 'Afschuifvlak' behandelt het eerste deel van de modelvorming van het verspaningsproces. Dit wordt gedaan met een eenvoudige simulatie, die laat zien hoe een materiaaldeel afschuift, als het boven- en ondervlak een verschillende snelheid hebben. De simulatie is gemaakt door steeds een nieuwe lijn te tekenen en de oude te overschrijven met de achtergrondkleur. De rest van het scherm wordt gemaakt met reeds behandelde standaardfuncties uit de unit Graph en met procedures zoals kaders, tekst etc.

```

PROCEDURE Afschuifvlak;
BEGIN
  Kaders;
  Rectangle(10,10,300,232);
  Tekst('Om te komen tot een voorlopig',2,41);
  Tekst('model van het proces is het van',3,41);
  Tekst('belang dat afschuiven niet on-',4,41);

```

```

Tekst('eindig snel kan gebeuren. Dit',5,41);
Tekst('betekent dat de primaire defor-',6,41);
Tekst('matiezone AB dus geen lijn kan',7,41);
Tekst('zijn, maar dat de zone een be-',8,41);
Tekst('paalde dikte moet hebben. Deze',9,41);
Tekst('zone wordt begrensd door twee',10,41);
Tekst('vlakken: het afschuifvlak en het',11,41);
Tekst('stationaire vlak. Het afschuifvlak beweegt onder invloed van de',12,3);
Tekst('afschuifkracht met een bepaalde snelheid t.o.v. het stationaire',13,3);
Tekst('vlak. De richting wordt bepaald door de afschuifhoek. Er geldt:',14,3);
Tekst('tan = (Vs . t)/ c. In de tijd t gaat PQRS over in P"Q"RS.',15,3);
SchrijfDelta(15,17); SchrijfDelta(15,22);
SchrijfGamma(30,320); SchrijfDelta(15,42);
PutStr('s',33,9);
Delay(500);
Tijd;
Line(60,40,280,40); Line(60,199,280,199);
Line(80,40,80,199); Line(260,40,260,199);
PutStr('afschuifvlak',6,13);PutStr('stationaire vlak',20,13);
PutStr('//////////',21,10);
Tekst('P" P Q" Q',2,5);
Tekst(' R S',11,5);
Tekst('Vs',2,17); PutStr(#27,3,16);
FOR w:=1 TO 45 DO
  BEGIN
    SetColor(Black);
    Line(80-w+1,41,80,199); Line(260-w+1,41,260,199);
    Line(60-w+1,40,280-w+1,40);
    SetColor(White);
    Line(80-w,41,80,199); Line(260-w,41,260,199);
    Line(60-w,40,280-w,40);
    Delay(150);
  END;
SetLineStyle(DottedLn,0,NormWidth);
Line(80,40,80,199); Arc(80,199,80,112,55);
Line(260,41,260,199); Line(240,40,280,40);
SetLineStyle(SolidLn,0,NormWidth);
SchrijfGamma(70,160); PutStr('s',16,15);
Line(27,40,27,199); Line(27,40,33,50); Line(27,40,22,50);
Line(27,199,33,189); Line(27,199,22,189);
Tekst(' ',6,3); SchrijfDelta(6,3); Tekst('c',6,5);
PutStr(Verder,pt,3);
PutStr(Terug,pt+1,3);
PutStr(#25+' = info over uniforme deformatie',pt+2,3);
ClearKb; Tijd;

```

```

REPEAT
  GetKey(Keytype);
  CASE Keytype OF 2: i:=2;
                 3: i:=7;
                 4: i:=6;
                 ELSE Beep;
                 END {CASE};
  UNTIL (Keytype=2) OR (Keytype=3) OR (Keytype=4);
END;

```

Procedure 'Uniform' behandelt de uniforme deformatie. Ook in deze procedure zijn geen bijzondere commando's gebruikt.

```

PROCEDURE Uniform;
BEGIN
  Kaders;
  Rectangle(440,90,630,300);
  Line(580,123,580,262); Line(620,123,620,262); Line(620,123,615,133);
  Line(620,123,625,133); Line(620,262,615,252); Line(620,262,625,252);
  Tekst(' c ',9,76); SchrijfDelta(9,76);
  Tekst('Vs',6,57); Tekst('V(y)',10,61);
  Tekst('y',10,74);
  PutStr(#24,19,74); PutStr(#27,14,61); PutStr(#27,16,63);
  PutStr(#27,18,65); PutStr(#27,20,67); PutStr(#27,22,69);
  PutStr(#27,24,71); PutStr(#27,13,60);
  Line(469,123,580,123); Line(485,133,580,133); Line(501,153,580,153);
  Line(517,173,580,173); Line(533,193,580,193); Line(549,213,580,213);
  Line(565,233,580,233);
  SetLineStyle(DottedLn,0,NormWidth);
  Line(469,123,580,262); Line(580,123,620,123); Line(580,262,620,262);
  SetLineStyle(SolidLn,0,NormWidth);
  PutStr('afschuifvlak',11,62); PutStr('stationaire vlak',28,60);
  Tekst('Bij de animatie is getoond hoe een volume-element van vorm',2,3);
  Tekst('verandert. De oppervlakte is echter gelijk gebleven. Afschuiving',3,3);
  Tekst('gaat namelijk niet gepaard met volumeverandering als loodrecht',4,3);
  Tekst('op het vlak van tekening geen rek optreedt.',5,3);
  Tekst('Er is sprake van zgn. uniforme deformatie',6,3);
  Tekst('als de deformatie onafhankelijk is van de',7,3);
  Tekst('y-waarde (zie tekening).',8,3);
  Tekst('Uniforme deformatie gaat dus gepaard met',9,3);
  Tekst('een lineaire verdeling van de verplaat-',10,3);
  Tekst('singssnelheid van materiaaldeeltjes v(y).',11,3);
  Tekst('De deformatiesnelheid is gedefinieerd',12,3);

```

```

Tekst('als: tan / t',13,3);SchrijfDelta(13,17);
Tekst('en is over de hele deformatiezone constant.',14,3);
SchrijfGamma(70,275); SchrijfGamma(199,255);
PutStr('s',26,30); PutStr('s',28,14);
PutPixel(227,235,White); PutPixel(228,235,White);
PutPixel(227,236,White); PutPixel(228,236,White);
Tekst('tan = (Vs . t)/ c',15,3);
SchrijfDelta(15,19); SchrijfDelta(15,24);
SchrijfGamma(30,320); PutStr('s',33,9);
SchrijfGamma(230,320); PutStr('s',33,34);
PutPixel(257,300,White); PutPixel(258,300,White);
PutPixel(257,301,White); PutPixel(258,301,White);
Tekst('= Vs/ c',15,35); SchrijfDelta(15,41);
PutStr(#26,32,30); PutStr(Terug,pt,3);
ClearKb; Tijd;
REPEAT
  GetKey(Keytype);
  IF Keytype=2 THEN i:=5 ELSE Beep;
UNTIL (Keytype=2);
END;

```

De procedure 'EersteModel' behandelt een eerste modellering van het verspaningsmodel.

```

PROCEDURE EersteModel;
BEGIN
  Kaders;
  Tekst('Het is nu mogelijk een eerste model van het verspaningsproces',2,3);
  Tekst('op te stellen. Neem aan dat een deformatiezone van bepaalde',3,3);
  Tekst('dikte ( c) ontstaat. De beitel beweegt met (snij)snelheid V.',4,3);
  SchrijfDelta(4,10);
  Tekst('Het spaanvlak (het vlak waarop de gevormde spaan afloopt)',5,3);
  Tekst('dwingt het afschuifvlak',6,3);
  Tekst('met een snelheid Vs te',7,3);
  Tekst('bewegen t.o.v. het sta-',8,3);
  Tekst('tionaire vlak. De rich-',9,3);
  Tekst('ting van de beweging',10,3);
  Tekst('wordt gegeven door de',11,3);
  Tekst('afschuifhoek :',12,3);SchrijfFi(12,19);
  Tekst('sin = c/BB"',14,5);SchrijfDelta(14,13);
  SchrijfFi(14,10);Rectangle(245,110,610,325);
  Line(255,315,610,315); Line(500,315,550,110); Line(500,315,350,110);
  Line(400,315,450,110); Line(400,315,250,110);

```

```

SetFillStyle(CloseDotFill,White); FloodFill(550,300,White);
SetLineStyle(DashedLn,0,NormWidth);
  Line(400,315,450,110); Line(400,315,250,110);
SetFillStyle(WideDotFill,White); FloodFill(255,115,White);
SetLineStyle(DottedLn,0,NormWidth);
  Line(537,165,390,165); Line(390,165,353,315);
SetLineStyle(SolidLn,0,ThickWidth);
  Line(500,315,353,315); Line(500,315,537,165); Line(500,315,390,165);
  Line(353,315,360,320); Line(353,315,360,310); Line(537,165,530,174);
  Line(537,165,540,175); Line(390,165,390,174); Line(390,165,400,169);
PutStr('beitel',30,70); PutStr('#27+' stationair vlak',13,34);
PutStr('#27+' afschuifvlak',14,48); PutStr('spaanvlak '+#26,13,58);
PutStr('defor-',17,39); PutStr('matie-',18,40); PutStr('zone',19,41);
Tekst('V',15,42); Tekst('Vs',8,46); Tekst('Vc',8,69);
Tekst(' B"',15,49); Tekst(' B ',15,62);
SetLineStyle(SolidLn,0,NormWidth); SchrijfFi(14,59);
  Arc(500,315,105,190,30); Line(420,180,300,250);
  Line(406,188,412,177); Line(406,188,420,188);
  Line(336,229,323,229); Line(336,229,330,239);
Tekst(' c',12,38); SchrijfDelta(12,37);
PutStr(Verder,pt,3); PutStr(Terug,pt+1,3);
PutStr('#24+' = meer uitleg over het proces',pt+2,3);
ClearKb; Tijd;
REPEAT
  GetKey(Keytype);
  CASE Keytype OF 2: i:=5;
    3: i:=9;
    1: i:=8;
  ELSE Beep;
  END {CASE};
UNTIL (Keytype=2) OR (Keytype=3) OR (Keytype=1);
END;

```

De procedure 'DeelProcessen' behandelt de verdeling van het proces in een aantal stappen. Deze procedure bevat alleen tekst.

```

PROCEDURE DeelProcessen;
BEGIN
  Kaders;
  Tekst('Het tot nu toe beschreven proces verloopt schijnbaar continu.',2,3);
  Tekst('Het is echter essentieel het proces te beschrijven als opeen-',3,3);
  Tekst('volging van zich op identieke wijze herhalende deelprocessen:',4,3);
  Tekst('- het vormen van een deformatiezone',6,3);

```

```

Tekst('- het afschuifproces in die zone',7,3);
Tekst('- het stationair worden van het afschuifvlak, zodra de beitel de',8,3);
Tekst(' deformatiezone heeft doorlopen.',9,3);
Tekst('Als het deformatieproces is voltooid is de deformatiezone over-',11,3);
Tekst('gegaan in een zgn spaanlamel.',12,3);
Tekst('Dit model van opeenvolging van deelprocessen resulterend in',13,3);
Tekst('spaanlamellen berust op waarnemingen !',14,3);
Tekst('Het model wordt verder verduidelijkt op het volgende scherm.',15,3);
PutStr(Terug,pt,3); PutStr(Verder,pt+1,3);
ClearKb; Tijd;
REPEAT
  GetKey(Keytype);
  CASE Keytype OF 2: i:=7;
                 3: i:=9;
                 ELSE Beep;
                 END {CASE};
UNTIL (Keytype=2) OR (Keytype=3);
END;

```

De procedure 'LamellenModel' beschrijft de modellering van het proces als een beweging van spaanlamellen en laat dit zien in een figuur.

```

PROCEDURE LamellenModel;
BEGIN
  Kaders;
  AfschuifOmgeving;
  Line(330,240,72,110); Line(332,230,74,100); Line(334,220,76,90);
  Line(336,210,78,80); Line(338,200,80,70); Line(340,190,100,70);
  Line(312,230,332,230); Line(314,220,334,220); Line(316,210,336,210);
  Line(318,200,338,200); Line(320,190,340,190); Line(72,110,92,110);
  Line(74,100,94,100); Line(76,90,96,90); Line(78,80,98,80);
  Line(80,70,100,70);
  SetLineStyle(DottedLn,0,NormWidth);
  Line(70,120,310,240); Line(10,240,400,240); Line(210,10,210,276);
  Line(50,100,222,100); Line(222,100,370,180);
  SetLineStyle(SolidLn,0,ThickWidth);
  Line(210,180,370,180); Line(210,180,222,100); Line(210,180,50,100);
  Line(370,180,360,175); Line(370,180,360,185); Line(222,100,226,111);
  Line(222,100,216,109); Line(50,100,62,100); Line(50,100,56,109);
  SetLineStyle(SolidLn,0,NormWidth);
  Tekst('Vc',5,29); Tekst('V',9,48); Tekst('Vs',5,4);
  PutStr('werkstuk',27,15); PutStr('spaan',4,30);
  Tekst('De aaneenschakeling',2,52);

```



```

Tekst('van lamellen vormt de',3,52);
Tekst('spaan, die met snelheid',4,52);
Tekst('Vc langs het spaanvlak',5,52);
Tekst('beweegt. Bij concentra-',6,52);
Tekst('tie van de deformatie',7,52);
Tekst('op het afschuifvlak',8,52);
Tekst('kan men het proces',9,52);
Tekst('zien als verschuiving',10,52);
Tekst('van metalen plaatjes',11,52);
Tekst('t.o.v. het afschuifvlak.',12,52);
Tekst('De lamellenstructuur',13,52);
Tekst('valt bij bekijken van de spaan meteen op. De dikte van de',14,3);
Tekst('lamellen hangt af van het materiaal en de bewerkingscondities.',15,3);
PutStr(Terug,pt,3);
PutStr('#26+' = informatie over geometrische relaties',pt+1,3);
PutStr('#24+' = informatie over krachten',pt+2,3);
PutStr('#25+' = informatie over temperatuur en vermogen',pt+3,3);
PutStr('#17+#196+#217+' informatie over deformatierelaties',pt+4,3);
ClearKb; Tijd;
GetKey(Keytype);
CASE Keytype OF 2: i:=7;
                3: i:=11;
                1: i:=101;
                4: i:=13;
                5: i:=12;
                ELSE Beep
                END {CASE};
END;
```

Hierna volgen drie procedures, die elk een bepaalde ontbinding van de kracht laten zien. Deze drie zijn uitgevoerd als afzonderlijke procedures, omdat dan tussen de schermen heen en weer geschakeld kan worden. De procedures bevatten elk een figuur met daarbij verklarende tekst.

```

PROCEDURE Krachten;
BEGIN
  Kaders;
  {omgeving};
  Rectangle(10,10,400,330);
  SetLineStyle(SolidLn,0,ThickWidth);
  Line(10,40,90,40); Line(90,40,96,10);
  Line(330,160,400,160); Line(330,160,376,10);
  Line(330,160,400,145);
```

```

SetFillStyle(InterLeaveFill,White);
FloodFill(399,11,White);
SetLineStyle(SolidLn,0,NormWidth);
Line(90,40,330,160);
{einde omgeving};
SetLineStyle(SolidLn,0,ThickWidth);
Line(330,160,90,270); Line(330,160,330,270); Line(330,160,90,160);
Line(330,270,335,260); Line(330,270,325,260); Line(90,160,100,155);
Line(90,160,100,165); Line(90,270,104,270); Line(90,270,98,260);
SetLineStyle(DottedLn,0,Normwidth);
Line(90,160,90,270); Line(90,270,330,270);
SetLineStyle(SolidLn,0,NormWidth);
Tekst('F',13,9);Tekst('Fv',8,9);Tekst('Ff',13,44);
Tekst('De algemene snijkraft',2,52);
Tekst('F is op meer manieren',3,52);
Tekst('te ontbinden. De eerste',4,52);
Tekst('manier is ontbinding',5,52);
Tekst('in een hoofdsnijkraft',6,52);
Tekst('Fv en een aanzetkraft',7,52);
Tekst('Ff. Deze werken in de',8,52);
Tekst('richting van resp. de',9,52);
Tekst('hoofd- en de aanzetbe-',10,52);
Tekst('weging.',11,52);
PutStr(#27+ ' = terug naar lamellenmodel',pt,3);
PutStr(#26+ ' = een andere ontbinding van de kraft',pt+1,3);
PutStr(#24+ ' = informatie over geometrische relaties',pt+2,3);
PutStr(#25+ ' = informatie over temperatuur en vermogen',pt+3,3);
PutStr(#17+#196+#217+ ' informatie over deformatierelaties',pt+4,3);
ClearKb; Tijd;
GetKey(Keytype);
CASE Keytype OF 1: i:=11;
                2: i:=9;
                3: i:=102;
                4: i:=13;
                5: i:=12;
                ELSE Beep;
                END {CASE};
END;

{-----}

```

```

PROCEDURE Krachten2;
BEGIN
  Kaders;
  {omgeving};
  Rectangle(10,10,400,330);
  SetLineStyle(SolidLn,0,ThickWidth);
  Line(10,40,90,40); Line(90,40,96,10);
  Line(330,160,400,160); Line(330,160,376,10);
  Line(330,160,400,145);
  SetFillStyle(InterLeaveFill,White);
  FloodFill(399,11,White);
  SetLineStyle(SolidLn,0,NormWidth);
  Line(90,40,330,160);
  {einde omgeving};
  SetLineStyle(SolidLn,0,ThickWidth);
  Line(330,160,90,270); Line(330,160,280,310); Line(330,160,135,115);
  Line(90,270,104,270); Line(90,270,98,260); Line(280,310,290,302);
  Line(280,310,277,298); Line(135,115,149,112); Line(135,115,145,124);
  SetLineStyle(DottedLn,0,Normwidth);
  Line(90,270,280,310); Line(90,270,135,115);
  SetLineStyle(SolidLn,0,NormWidth);
  Tekst('F',13,10); Tekst('F n',6,13); Tekst('F',15,38);
  SchrijfGammatje(94,119);SchrijfGammatje(297,318);
  SchrijfBeta(288,159);
  Tekst('Een tweede manier van',2,52);
  Tekst('ontbinden van de alge-',3,52);
  Tekst('mene snijkraft is',4,52);
  Tekst('langs en loodrecht',5,52);
  Tekst('op het spaanvlak van',6,52);
  Tekst('het gereedschap. Als',7,52);
  Tekst('de secundaire defor-',8,52);
  Tekst('matiezone is bepaald',9,52);
  Tekst('door wrijving, dan is',10,52);
  Tekst('tan = F /F n',11,52);SchrijfBeta(453,220);
  SchrijfGammatje(493,228);SchrijfGammatje(533,228);
  Tekst('een maat voor de wrij-',12,52);
  Tekst('ving. tan heet dan',13,52);SchrijfBeta(497,263);
  Tekst('de "schij~bare wrij-',14,52);
  Tekst('vingscoefficient".',15,52);
  PutStr(#27+' = terug naar vorige ontbinding van de kracht',pt,3);
  PutStr(#26+' = een andere ontbinding van de kracht',pt+1,3);
  PutStr(#24+' = informatie over geometrische relaties',pt+2,3);
  PutStr(#25+' = informatie over temperatuur en vermogen',pt+3,3);
  PutStr(#17+#196+#217+' informatie over deformatierelatyes',pt+4,3);
  ClearKb; Tijd;

```

```

GetKey(Keytype);
  CASE Keytype OF 1: i:=11;
                  2: i:=101;
                  3: i:=103;
                  4: i:=13;
                  5: i:=12;
                  ELSE Beep;
                  END {CASE};
END;

```

```
{-----}
```

PROCEDURE Krachten3;

BEGIN

Kaders;

{omgeving}

Rectangle(10,10,400,330);

SetLineStyle(SolidLn,0,ThickWidth);

Line(10,40,90,40); Line(90,40,96,10);

Line(330,160,400,160); Line(330,160,376,10);

Line(330,160,400,145);

SetFillStyle(InterLeaveFill,White);

FloodFill(399,11,White);

SetLineStyle(SolidLn,0,NormWidth);

Line(90,40,330,160);

{einde omgeving};

SetLineStyle(SolidLn,0,ThickWidth);

Line(330,160,114,259); Line(330,160,220,320); Line(330,160,218,104);

Line(114,259,129,259); Line(114,259,122,249); Line(218,104,230,104);

Line(218,104,225,114); Line(220,320,234,314); Line(220,320,221,307);

SetLineStyle(DottedLn,0,NormWidth);

Line(114,259,218,104); Line(114,259,220,320);

SetLineStyle(SolidLn,0,NormWidth);

Tekst('Fs',6,23); Tekst('F',12,14); Tekst('Fsn',15,31);

Tekst('De derde manier om de',2,52);

Tekst('snijkraft F te ont-',3,52);

Tekst('binden is in de kracht',4,52);

Tekst('Fs in het afschuifvlak',5,52);

Tekst('en de kracht Fsn lood-',6,52);

Tekst('recht daarop. De kracht',7,52);

Tekst('veroorzaakt de afschui-',8,52);

Tekst('ving en is dan ook al',9,52);

Tekst('eerder genoemd in het',10,52);

Tekst('voorlopige model.',11,52);

Tekst('Met afschuifsnelheid Vs',12,52);

```

Tekst('bepaalt Fs het vermo-',13,52);
Tekst('gen dat de primaire',14,52);
Tekst('deformatie "verbruikt".',15,52);]
PutStr(#27+' = terug naar de vorige ontbinding van de kracht',pt,3);
PutStr(#26+' = informatie over verspaanbaarheid',pt+1,3);
PutStr(#24+' = informatie over geometrische relaties',pt+2,3);
PutStr(#25+' = informatie over temperatuur en vermogen',pt+3,3);
PutStr(#17+#196+#217+' informatie over deformatierelaties',pt+4,3);
ClearKb; Tijd;
  GetKey(Keytype);
    CASE Keytype OF 1: i:=11;
      2: i:=102;
      3: i:=14;
      4: i:=13;
      5: i:=12;
      ELSE Beep;
    END {CASE};
END;

```

De procedure 'Geometrie' behandelt de meetkunde van het verspaningsmodel en geeft enige belangrijke relaties aan, die daaruit zijn af te leiden. Deze procedure bestaat uit twee schermen. Tussen deze twee schermen moet een toets worden ingedrukt.

```

PROCEDURE Geometrie;
BEGIN
  Kaders;
  Tekst('Uit de meetkunde van het ver-',2,40);
  Tekst('spaningsmodel kunnen we de',3,40);
  Tekst('zgn kinematische relaties halen.',4,40);
  Tekst('Het is mogelijk de snelheden Vs',5,40);
  Tekst('en Vc uit te drukken in de snel-',6,40);
  Tekst('heid V en de hoeken en :',7,40);
  SchrijfFi(7,66);SchrijfGamma(550,150);
  Tekst('sin',9,52);SchrijfFi(9,57);
  Tekst('Vc = V. ----- en',10,40);
  Tekst('cos( - )',11,50);SchrijfFi(11,56);SchrijfGamma(450,230);
  Tekst('cos',13,52);SchrijfFi(13,57);
  Tekst('Vs = V. -----',14,40);
  Tekst('cos( - )',15,50);SchrijfFi(15,56);SchrijfGamma(450,315);
  Rectangle(10,10,300,300);
  Line(100,100,190,250);Arc(210,150,30,110,100);SchrijfGamma(230,50);
  Line(175,210,178,215);Line(178,215,172,218);Line(18,95,143,34);

```

```

Line(90,65,70,65);Line(85,57,75,73);Line(18,95,28,95);
Line(143,34,133,34);Line(18,95,23,87);Line(143,34,138,42);
SetLineStyle(SolidLn,0,ThickWidth);
Line(10,100,100,100);Line(100,100,165,10);Line(190,250,299,85);
Line(190,250,300,250);Line(190,250,300,235);
Line(190,250,130,150);Line(190,250,65,250);
Line(65,250,75,245);Line(65,250,75,255);
Line(255,150,257,160);Line(255,150,244,155);
Line(130,150,129,162);Line(130,150,143,157);
Tekst('Vs',8,20);Tekst('Vc',8,34);Tekst('V',13,7);
SetLineStyle(DottedLn,0,NormWidth);
Line(10,250,190,250);Line(14,90,110,250);Line(139,30,233,186);
Line(190,250,190,20);Line(233,184,169,212);Line(130,150,255,150);
Line(130,150,65,250);Line(100,100,77,60);Line(233,187,155,187);
Line(153,187,110,250);
SetLineStyle(SolidLn,0,NormWidth);
Line(110,250,110,295);Line(190,250,190,295);
Line(110,290,190,290);Line(110,290,120,285);
Line(110,290,120,295);Line(190,290,180,295);
Line(190,290,180,285);Line(235,186,293,215);
Line(190,250,250,280);Line(250,280,293,215);
Line(293,215,293,227);Line(250,280,250,268);
Line(293,215,280,222);Line(250,280,263,274);
Tekst('V. t',14,17);SchrijfDelta(14,19);
SetColor(black);
Line(240,251,292,251);
SetColor(White);
Tekst('Vc. t',12,31);SchrijfDelta(12,34);
Arc(190,250,100,190,20);SchrijfFi(11,21);
Arc(235,183,194,253,30);SchrijfFi(11,30);
Tekst('-',11,31);SchrijfGamma(226,230);
Tekst('c',4,6); Tekst('c',3,13);
SchrijfDelta(4,4); SchrijfDelta(3,11);
ClearKb; Tijd;
PutStr('druk op '+#26+' voor vervolg',pt+2,5);
GetKey(KeyType);
Kaders;
Tekst('Uit de geometrie kunnen we bovendien afleiden dat:',2,3);
Tekst('c = V.sin . t',4,5);
SchrijfDelta(4,3); SchrijfDelta(4,19); SchrijfFi(4,16);
Tekst('Uit de uniforme afschuiving in een deformatiezone met',6,3);
Tekst('deze dikte volgde dat voor de deformatiegraad geldt:',7,3);
Tekst('Vs',8,15);
Tekst('tan = ---- . t',9,3); SchrijfDelta(9,22);
Tekst('c',10,17); SchrijfDelta(10,15);

```

```

SchrijfGamma(25,190); PutStr('s',20,9);
Tekst('Met de zojuist gevonden relaties voor Vs en c wordt dit:',12,3);
SchrijfDelta(12,54);
Tekst('          cos',13,3); SchrijfFi(13,28);
Tekst('tan = -----',14,3);
Tekst('      sin .cos ( - )',15,3);
SchrijfFi(15,20); SchrijfFi(15,29);
SchrijfGamma(240,320);
SchrijfGamma(25,300);
PutStr('s',31,9);
PutStr('#27+' = terug naar het lamellenmodel',pt,3);
PutStr('#26+' = informatie over verspaanbaarheid',pt+1,3);
PutStr('#24+' = informatie over ontbinden van de kracht',pt+2,3);
PutStr('#25+' = informatie over temperatuur en vermogen',pt+3,3);
PutStr('#17+#196+#217+' informatie over deformatierelaties',pt+4,3);
ClearKb; Tijd;
GetKey(Keytype);
  CASE Keytype OF 1: i:=101;
                  2: i:=9;
                  3: i:=14;
                  4: i:=13;
                  5: i:=12;
                  ELSE Beep;
                  END {CASE};
END;
END;

```

De procedure 'Deformatie' behandelt enige definities en relaties die af te leiden zijn uit grootheden vóór en na het deformereren.

```

PROCEDURE Deformatie;
BEGIN
  Kaders;
  Rectangle(10,10,270,295);
  Line(110,210,200,260); Line(30,190,30,280); Line(125,65,245,97);
  Line(127,70,140,69); Line(130,62,140,69); Line(245,92,232,94);
  Line(232,92,241,101); Line(30,210,35,200); Line(30,210,25,200);
  Line(30,260,35,270); Line(30,260,25,270); Line(148,155,152,134);
  Line(152,134,155,144); Line(152,134,146,142);Line(60,230,80,230);
  Line(80,230,70,225); Line(80,230,70,235);
  Arc(205,220,60,110,40); SchrijfGamma(210,195);
  Arc(200,260,140,190,40); SchrijfFi(12,19);
  SetLineStyle(SolidLn,0,ThickWidth);

```

```

Line(10,210,110,210); Line(110,210,125,135); Line(200,260,270,260);
Line(200,260,270,250); Line(200,260,250,10);
Arc(-60,135,0,41,185); Arc(10,135,0,34,215);
SetLineStyle(CenterLn,0,NormWidth);
Line(10,260,200,260); Line(200,260,200,180); Line(125,135,140,60);
SetLineStyle(SolidLn,0,NormWidth);
Tekst('h',12,5); Tekst('hc',4,23);
Tekst('Vc',6,20); Tekst('V',11,12);
Tekst('De verhouding tussen de vervormde',2,36);
Tekst('en de onvervormde spaandoorsnede',3,36);
Tekst('Ac/A heet de "spaanstuik" .',4,36);
SchrijfLabda(525,55); PutStr('c',8,68);
Tekst('Als loodrecht op het vlak van te-',5,36);
Tekst('kening geen rek optreedt is dit ge-',6,36);
Tekst('lijk aan hc/h. Uit de voorwaarde',7,36);
Tekst('voor constante materiaalstroom volgt',8,36);
Tekst('V.h = Vc.hc dus hc/h = V/Vc = .',9,36);
SchrijfLabda(580,165); PutStr('c',19,75);
Tekst('Als we de uit de geometrie gevonden',10,36);
Tekst('relaties voor V en Vs invullen:',11,36);
Tekst('= cos( - )/sin',13,39);
SchrijfFi(13,47); SchrijfGamma(380,275);
SchrijfFi(13,61); SchrijfLabda(282,255);
PutStr('c',28,38);
Tekst('De spaanstuik is een meetbare grootheid waaruit is te halen.',15,3);
SchrijfFi(15,60);
PutStr('#27+' = terug naar het lamellenmodel',pt,3);
PutStr('#26+' = informatie over verspaanbaarheid',pt+1,3);
PutStr('#24+' = informatie over ontbinden van de kracht',pt+2,3);
PutStr('#25+' = informatie over temperatuur en vermogen',pt+3,3);
PutStr('#17+#196+#217+' informatie over geometrie',pt+4,3);
ClearKb; Tijd;
GetKey(Keytype);
CASE Keytype OF 1: i:=101;
                2: i:=9;
                3: i:=14;
                4: i:=13;
                5: i:=11;
                ELSE Beep;
                END {CASE};
END;
```



Procedure 'Temperatuur' behandelt alle aspecten, die te maken hebben met temperatuur, arbeid, vermogen en hun relatie met de vervormingsprocessen. Deze procedure bestaat uit een tekstschermb, een scherm met een grafiek over verdeling van de warmte, een scherm met een temperatuurverdeling en een scherm met enkele aanvullende formules.

Het scherm met de temperatuurverdeling behoeft nog enige uitleg. Er is hier weer gebruik gemaakt van polygonen. De punten van deze polygonen zijn gedefinieerd en vormen een aantal isothermen. Met behulp van de functie 'Fillpoly' uit de unit Graph kunnen deze polygonen ingekleurd worden. De procedure is zo gemaakt, dat wanneer aan het begin van het programma is opgegeven dat een monochroom monitor wordt gebruikt, er gearceerd zal worden in plaats van ingekleurd. De isothermen zijn als locale variabelen gedeclareerd.

PROCEDURE Temperatuur;

VAR iso400, iso500, iso600, iso650, iso700 : array[1..13] of PointType;

BEGIN

Kaders;

Tekst('Alle vervormingsprocessen kunnen worden beschreven vanuit het',2,2);

Tekst('begrip specifieke arbeid. Dit is de arbeid die per volume-eenheid',3,2);

Tekst('nodig is om een zekere vervorming te bereiken. De arbeid bestaat',4,2);

Tekst('uit specifieke vervormingsarbeid en specifieke wrijvingsarbeid.',5,2);

Tekst('De specifieke vervormingsarbeid is de arbeid, nodig om het mate-',6,2);

Tekst('riaal te vervormen en is dus een soort materiaalkonstante. De',7,2);

Tekst('vervorming vindt plaats in de primaire deformatiezone. De speci-',8,2);

Tekst('fieke wrijvingsarbeid heeft te maken met de wrijving in de se-',9,2);

Tekst('cundaire deformatiezone tussen materiaal en gereedschap en is',10,2);

Tekst('dus een soort procesconstante.',11,2);

Tekst('Bijna het totale toegevoegde vermogen wordt omgezet in warmte.',12,2);

Tekst('De verdeling van de afvoer tussen spaan, beitel en werkstuk',13,2);

Tekst('hangt af van de snijsnelheid en is op het volgende scherm',14,2);

Tekst('geschetst.',15,2);

PutStr('druk op '+#26+' om een grafiek te zien',pt+1,6);

PutStr('van de verdeling van de warmteafvoer',pt+2,6);

ClearKb; Tijd;

GetKey(KeyType);

Kaders;

Tekst('De verdeling van de warmte-afvoer:',2,3);

SetLineStyle(SolidLn,0,ThickWidth);

Line(120,75,120,275); Line(120,275,350,275);

SetLineStyle(DottedLn,0,NormWidth);

Line(120,75,350,75); Line(350,75,350,275);

```

SetLineStyle(SolidLn,0,NormWidth);
IF (kaart=3) THEN BEGIN Arc(380,460,95,130,400);
    Arc(305,382,90,122,340);
    Line(305,85,345,85) END
    ELSE BEGIN Arc(380,503,95,130,400);
    Arc(305,425,90,122,340);
    Line(305,85,345,85) END;
PutStr('% van de',10,3); PutStr('totale',11,3);
PutStr('warmte',12,3); PutStr('100',8,12);
PutStr('80',12,13); PutStr('60',16,13);
PutStr('40',20,13); PutStr('20',24,13);
PutStr('0',28,14);
PutStr('0.0 0.3 0.6 0.9',29,16);
PutStr('snijsnelheid V [m/s]',32,20);
PutStr('warmte in werk-',9,17); PutStr('stuk',10,17);
PutStr('warmte in',11,27); PutStr('gereedschap',12,23);
PutStr('warmte in spaan',18,27);
Tekst('Deze verdeling geeft bij la-',3,47);
Tekst('ge snijsnelheid een gereed-',4,47);
Tekst('schapstemperatuur die on-',5,47);
Tekst('geveer 60% hoger is dan de',6,47);
Tekst('temperatuur van de spaan.',7,47);
Tekst('Dit is 140% bij hoge snel-',8,47);
Tekst('heid, omdat daarbij het',9,47);
Tekst('vermogen toeneemt',10,47);
Tekst('waarbij de toevoer van',11,47);
Tekst('warmte en het volume van',12,47);
Tekst('de spaan evenredig ver-',13,47);
Tekst('groten. Het beitelvolume',14,47);
Tekst('blijft echter gelijk.',15,47);
PutStr('druk op '+#26+' om een temperatuurverdeling te zien',pt+1,6);
ClearKb; Tijd;
GetKey(KeyType);
Kaders;
IF (kaart>1) THEN BEGIN SetFillStyle(EmptyFill,Black) END
    ELSE BEGIN SetFillStyle(InterLeaveFill,Yellow) END;
Bar(10,10,350,300); Rectangle(10,10,350,300);
{isothermen}
iso600[1].x:=200; iso600[1].y:=250;
iso600[2].x:=206; iso600[2].y:=220;
iso600[3].x:=200; iso600[3].y:=140;
iso600[4].x:=245; iso600[4].y:=40;
iso600[5].x:=320; iso600[5].y:=80;
iso600[6].x:=345; iso600[6].y:=160;
iso600[7].x:=335; iso600[7].y:=230;

```

```

iso600[8].x:=200; iso600[8].y:=250;
IF (kaart>1) THEN BEGIN SetFillStyle(LtBkSlashFill,LightRed) END
    ELSE BEGIN SetFillStyle(InterLeaveFill,LightRed) END;
bar(360,50,375,65); Rectangle(360,50,375,65); FillPoly(8,iso600);
iso650[1].x:=208; iso650[1].y:=210;
iso650[2].x:=204; iso650[2].y:=138;
iso650[3].x:=234; iso650[3].y:=85;
iso650[4].x:=280; iso650[4].y:=95;
iso650[5].x:=310; iso650[5].y:=150;
iso650[6].x:=290; iso650[6].y:=238;
iso650[7].x:=250; iso650[7].y:=244;
iso650[8].x:=243; iso650[8].y:=220;
iso650[9].x:=208; iso650[9].y:=210;
IF (kaart>1) THEN BEGIN SetFillStyle(InterLeaveFill,Magenta) END
    ELSE BEGIN SetFillStyle(InterLeaveFill,Magenta) END;
bar(360,30,375,45); Rectangle(360,30,375,45); FillPoly(9,iso650);
iso700[1].x:=225; iso700[1].y:=125;
iso700[2].x:=240; iso700[2].y:=120;
iso700[3].x:=270; iso700[3].y:=124;
iso700[4].x:=280; iso700[4].y:=150;
iso700[5].x:=260; iso700[5].y:=172;
iso700[6].x:=225; iso700[6].y:=178;
iso700[7].x:=215; iso700[7].y:=173;
iso700[8].x:=210; iso700[8].y:=140;
iso700[9].x:=225; iso700[9].y:=125;
IF (kaart>1) THEN BEGIN SetFillStyle(HatchFill,Red) END
    ELSE BEGIN SetFillStyle(InterLeaveFill,Red) END;
bar(360,10,375,25); Rectangle(360,10,375,25); FillPoly(9,iso700);
iso500[1].x:=200; iso500[1].y:=250;
iso500[2].x:=185; iso500[2].y:=130;
iso500[3].x:=205; iso500[3].y:=10;
iso500[4].x:=240; iso500[4].y:=10;
iso500[5].x:=250; iso500[5].y:=10;
iso500[6].x:=245; iso500[6].y:=40;
iso500[7].x:=200; iso500[7].y:=140;
iso500[8].x:=206; iso500[8].y:=220;
iso500[9].x:=230; iso500[9].y:=230;
iso500[10].x:=240; iso500[10].y:=242;
iso500[11].x:=200; iso500[11].y:=250;
IF (kaart>1) THEN BEGIN SetFillStyle(SolidFill,black); FloodFill(205,249,White) END
    ELSE BEGIN SetFillStyle(InterLeaveFill, Yellow) END;
Bar(360,70,375,85); Rectangle(360,70,375,85); FillPoly(11,iso500);
iso400[1].x:=173; iso400[1].y:=10;
iso400[2].x:=100; iso400[2].y:=10;
iso400[3].x:=60; iso400[3].y:=162;

```

```

iso400[4].x:=80; iso400[4].y:=210;
iso400[5].x:=110; iso400[5].y:=235;
iso400[6].x:=185; iso400[6].y:=260;
iso400[7].x:=220; iso400[7].y:=260;
iso400[8].x:=300; iso400[8].y:=250;
iso400[9].x:=200; iso400[9].y:=250;
iso400[10].x:=185; iso400[10].y:=245;
iso400[11].x:=180; iso400[11].y:=205;
iso400[12].x:=170; iso400[12].y:=130;
iso400[13].x:=173; iso400[13].y:=10;
IF (kaart>1) THEN BEGIN SetFillStyle(CloseDotFill,LightBlue) END
    ELSE BEGIN SetFillStyle(InterLeaveFill,LightBlue) END;
Bar(360,110,375,125); Rectangle(360,110,375,125); FillPoly(13,iso400);
SetLineStyle(SolidLn,0,ThickWidth);
    Line(10,200,50,200); Line(50,200,100,10);
    Line(200,250,350,250); Line(200,250,350,230);
    Line(200,250,250,10);
SetLineStyle(SolidLn,0,NormWidth);
    Line(350,230,350,250);
SetFillStyle(EmptyFill,black);
FloodFill(11,11,White); FloodFill(348,248,White);
Line(173,10,205,10); Line(185,245,200,250);
IF (kaart>1) THEN BEGIN SetFillStyle(LineFill,Green) END
    ELSE BEGIN SetFillStyle(InterLeaveFill,Green) END;
Bar(360,90,375,105); Rectangle(360,90,375,105); FloodFill(174,11,White);
Line(350,250,350,300); Line(350,300,10,300); Line(10,300,10,200);
IF (kaart>1) THEN BEGIN SetFillStyle(XHatchFill,White) END
    ELSE BEGIN SetFillStyle(InterLeaveFill,White) END;
Bar(360,130,375,145); Rectangle(360,130,375,145); FloodFill(11,299,White);
Line(50,200,200,250);
PutStr('= > 700',2,49);
PutStr('= 700 - 650',4,49);
PutStr('= 650 - 600',6,49);
PutStr('= 600 - 500  temperaturen in',8,49);
PutStr('= 500 - 400  graden Celsius',10,49);
PutStr('= 400 - 300',12,49);
PutStr('= < 300',14,49);
Tekst('Dit is een temperatuurver-',8,46);
Tekst('deling die typisch is bij',9,46);
Tekst('verspanen. Opvallend is de',10,46);
Tekst('hoge temperatuur op enige',11,46);
Tekst('afstand van de hoofdsnij-',12,46);
Tekst('kant. Deze veroorzaakt',13,46);
Tekst('een slijtage die "kolkslijtage"',14,46);
Tekst('of "kratervorming" heet.',15,46);

```

```

PutStr('druk op '+#26+' voor verdere informatie over',pt+1,6);
PutStr('specifieke verspaningsarbeid.',pt+2,6);
ClearKb; Tijd;
GetKey(KeyType);
Kaders;
Tekst('Zoals getoond: de som van de specifieke vervormingsarbeid',2,3);
Tekst('en de specifieke wrijvingsarbeid wordt de specifieke verspaningsarbeid',3,3);
Tekst('genoemd. Deze specifieke verspaningsarbeid Esp kan als volgt',4,3);
Tekst('geschreven worden:',5,3);
Tekst('Esp = Fv.V = Fs.Vs + F .Vc',7,5);
Tekst('      V.A      V.A      V.A',8,5);
SchrijfGammatje(227,141);
Line(90,147,125,147); Line(155,147,200,147); Line(230,147,285,147);
Tekst('Als we hierin de uit de geometrie gevonden relaties voor Vs en',10,3);
Tekst('Vc invullen wordt dit:',11,3);
Tekst('Fs.cos      + F .sin',13,13);
Tekst('Esp = ',14,5);
Tekst('      A.cos( - )      A.cos( - )',15,6);
SchrijfGamma(130,275); SchrijfGammatje(255,280);
SchrijfFi(13,42);      SchrijfFi(15,20);
SchrijfFi(15,39);      SchrijfGamma(160,315);
SchrijfGamma(315,315);
Line(90,290,360,290); Tekst('      ',14,27);
PutStr('#27+' = terug naar het lamellenmodel',pt,3);
PutStr('#26+' = informatie over verspaanbaarheid',pt+1,3);
PutStr('#24+' = informatie over ontbinden van de kracht',pt+2,3);
PutStr('#25+' = informatie over deformatierelaties',pt+3,3);
PutStr('#17+#196+#217+' informatie over geometrie',pt+4,3);
ClearKb; Tijd;
GetKey(Keytype);
CASE Keytype OF 1: i:=101;
                2: i:=9;
                3: i:=14;
                4: i:=12;
                5: i:=11;
                ELSE Beep;
                END {CASE};
END;
```

De procedure 'Verspaanbaarheid' behandelt verschillende aspecten van het begrip verspaanbaarheid van een materiaal. Er wordt een relatie gelegd tussen de tot nu toe behandelde parameters van het model en de verspaanbaarheid.

Dit gebeurt in vier schermen; drie schermen behandelen de relatie tussen een bepaalde grootte en de verspaanbaarheid, het vierde geeft een conclusie. Tussen deze schermen moet steeds een toets worden ingedrukt.

PROCEDURE Verspaanbaarheid;

BEGIN;

Kaders;

Tekst('Er zijn nu meerdere relaties gevonden. Ten eerste:',2,3);

Tekst('cos',4,3); SchrijfFi(4,28);

Tekst('tan = ----- uit de geometrie.',5,3);

Tekst('sin .cos ( - )',6,3);

SchrijfFi(6,20); SchrijfFi(6,29);

SchrijfGamma(240,120); SchrijfGamma(25,100);

PutStr('s',11,9);

Tekst('Voor kleine spaanhoek is dit gelijk aan -----',8,3);

Tekst('sin(2 )',9,54); Tekst('2',7,57);

SchrijfFi(9,61); SchrijfGamma(205,165);

Tekst('Dit geeft als verband',11,5);

Tekst('tussen deformatiegraad',12,5);

Tekst('en afschuifhoek:',13,5);

Tekst('( van 0 tot 45 gr.)',14,5);

SchrijfFi(14,7); SchrijfFi(15,54);

Tekst('tan',10,36); SchrijfGamma(290,205);

Line(300,210,300,300); Line(300,300,430,300);

MoveTo(310,210); LineTo(315,230); LineTo(330,260);

LineTo(350,280); LineTo(370,285); LineTo(390,290);

LineTo(430,292);

Tekst('De deformatie',11,60); Tekst('necmt dus toe',12,60);

Tekst('bij kleinere',13,60); Tekst('afschuifhoek.',14,60);

ClearKb; Tijd;

PutStr('druk op '+#26+' voor vervolg',pt+2,5);

GetKey(KeyType);

Kaders;

Tekst('De tweede relatie betreft de spaanstuik:',2,3);

Tekst('cos( - )',4,10); SchrijfFi(4,15);

Tekst(' = ----- (uit de deformatierelaties)',5,3);

SchrijfGamma(125,75); SchrijfLabda(23,80);

Tekst('sin',6,13); SchrijfFi(6,18);

Tekst('Als de spaanstuik groot is wordt een materiaal slechter',8,3);

Tekst('bewerkbaar geacht. Dit blijkt dus gepaard te gaan met een',9,3);

Tekst('kleine afschuifhoek.',10,3);

ClearKb; Tijd;

PutStr('druk op '+#26+' voor vervolg',pt+2,5);

GetKey(KeyType);

Kaders;

Tekst('Een derde relatie betreft de wrijving:',2,3);

```

Tekst('Er blijkt een relatie te bestaan tussen afschuifhoek,',4,3);
Tekst('wrijvingshoek en spaanhoek:',5,3);
Line(100,110,100,230); Line(100,230,400,230); Line(120,120,395,200);
SchrijfFi(7,8); Tekst('-',11,57);
SchrijfBeta(430,220); SchrijfGamma(445,232);
PutStr('45',12,11); PutStr('0',25,13); PutStr('45',25,47);
Tekst('Verspanen wordt moeilijker, de afschuifhoek wordt kleiner,',13,3);
Tekst('als de wrijvingshoek groter is (bij vaste spaanhoek), dus',14,3);
Tekst('bij een relatief grote waarde van de wrijvingskracht.',15,3);
ClearKb; Tijd;
PutStr('druk op '+#26+' voor vervolg',pt+2,5);
GetKey(KeyType);
Kaders;
Tekst('Over de verspaanbaarheid van materiaal is dus het volgende',2,3);
Tekst('te zeggen:',3,3);
Tekst('Een materiaal is moeilijker te verspanen als de afschuifhoek',5,3);
Tekst(' kleiner is. Er geldt dan:',6,3); SchrijfFi(6,4);
Tekst('* De spaanstuik is groter',8,4);
Tekst('* De deformatiegraad is groter',10,4);
Tekst('* De wrijving is groter',12,4);
Tekst('Spaanstuik en deformatiegraad blijken bovendien te worden',14,3);
Tekst('beïnvloed door de beitelgeometrie (spaanhoek).',15,3);
PutStr('#27+' = terug naar het lamellenmodel',pt,3);
PutStr('#26+' = einde programma',pt+1,3);
PutStr('#24+' = informatie over ontbinden van de kracht',pt+2,3);
PutStr('#25+' = informatie over deformatierelaties',pt+3,3);
PutStr('#17+#196+#217+' informatie over geometrie',pt+4,3);
ClearKb; Tijd;
GetKey(Keytype);
CASE Keytype OF 1: i:=101;
                2: i:=9;
                3: i:=15;
                4: i:=12;
                5: i:=11;
                ELSE Beep;
                END {CASE};
END;

```

De laatste procedure sluit het programma af nadat alle schermen zijn doorlopen. Het is in deze procedure nog mogelijk om terug te keren naar het programma. Ook na deze procedure zal nog het scherm met grafische effecten verschijnen, omdat procedure 'Stop' nog wordt aangeroepen (zie hoofdprogramma).

```
PROCEDURE Einde;
BEGIN
  Kaders;
  Tekst('Dit is het einde van het programma.',3,3);
  Tekst('Als U nog niet alle schermen hebt doorlopen kunt U nog',4,3);
  Tekst('terug naar het scherm van de informatie over',5,3);
  Tekst('verspaanbaarheid',6,3);
  PutStr('#27+' = terug naar verspaanbaarheid',pt,3);
  PutStr('#26+' = stoppen',pt+1,3);
  ClearKb; Tijd;
  GetKey(Keytype);
  CASE Keytype OF 2: i:=14;
                 3: Stoppen:=TRUE;
                 ELSE Beep;
                 END {CASE};

  ClearKb;
END;
```



## Het hoofdprogramma.

Het hoofdprogramma is zeer eenvoudig van opzet. Het bestaat uit een oneindige lus, die slechts onderbroken wordt als de Boolean 'Stoppen' TRUE wordt. Dit kan alleen gebeuren als de procedure wordt aangeroepen die het programma beëindigt. Afhankelijk van de waarde van parameter 'i' wordt een bepaalde procedure aangeroepen. De waarde van de parameter wordt aan het eind van elke procedure bepaald aan de hand van de ingedrukte toetsen. Het programma start op met de reeds besproken initialisatie-procedure 'StartGrafisch' en eindigt met de procedure 'Stop'.

```
BEGIN
  ClrScr;
  StartGrafisch;
  Terug :=#27+' = vorig scherm';
  Verder:=#26+' = volgend scherm';
  i:=0; Stoppen:=FALSE;
  WHILE (Stoppen=FALSE) DO
  BEGIN
    CASE i OF 0: Startscherm;
      1: Quickstop;
      2: AfschuifSimulatie;
      3: CirkelSimulatie;
      4: Secundair;
      5: Afschuifvlak;
      6: Uniform;
      7: EersteModel;
      8: DeelProcessen;
      9: LamellenModel;
    101: Krachten;
    102: Krachten2;
    103: Krachten3;
    11: Geometrie;
    12: Deformatie;
    13: Temperatuur;
    14: Verspaanbaarheid;
    15: Einde;
  END {CASE};
  END {WHILE...DO};
  Stop;
END.
```

# Toelichting op het programma

Deze toelichting beschrijft het gebruik van het programma. De toelichting is ook als hand-out bij het programma verstrekt.

## 1 Starten van het programma

### 1.1 Starten als de computer nog niet is ingeschakeld:

De bijgeleverde diskette is "bootable", dat wil zeggen: bevat de benodigde DOS 5.0 systeemfiles. Om het programma te starten kan de diskette in de drive worden gedaan, waarna de computer wordt aangezet. Een zg. autoexec.bat file zorgt voor uitvoering van het programma.

### 1.1 Starten als de computer al is ingeschakeld:

Om het programma te starten dient de diskette in de drive te worden geplaatst. De directory moet worden veranderd naar de drive waarin de floppy is geplaatst (meestal A of B). Intypen van het commando "start" zorgt daarna voor het starten van het programma. Ook is het mogelijk de computer te resetten (start via autoexec.bat). Het commando "C:\a:start" of "C:\b:start" zal een foutmelding opleveren, omdat de files uit de batchfile dan op de C-drive gezocht worden.

## 2 Toelichting bij de opstartprocedure

Als het programma wordt gestart verschijnt worden de volgende commando's uitgevoerd:

- \* Starten van het "graphics" commando, waardoor uitprinten van het scherm met <Shift><PRTSC> mogelijk wordt tijdens draaien van het programma.
- \* Starten van het programma "grab"; een programma behorende bij WordPerfect en DrawPerfect. Dit programma is een zg. "schermkopieprogramma" en maakt het mogelijk om tijdens het draaien

van het programma delen van het getoonde scherm te bewaren als .wpg-file. Deze file kan worden ingelezen in WordPerfect en DrawPerfect. Met de toetsencombinatie <Alt><Shift><F9> kan deze optie worden opgeroepen. Er verschijnt dan een kader in beeld dat met de cursortoetsen op de juiste plaats gezet kan worden. Met de Shift-toets en de cursortoetsen kan het kader van grootte worden veranderd. Na een druk op de <enter> toets wordt het aangewezen gedeelte opgeslagen in een file "grab.wpg" op de diskette waarop ook het programma staat. Als de grabber meermalen wordt aangeroepen, worden de files "grab1.wpg", "grab2.wpg" etc. genoemd. Er kunnen nog maximaal 6 á 7 van deze files op de diskette bij. Eventueel dus bij gebruik van deze optie even de diskette verwisselen. Het schermkopieprogramma werkt in tegenstelling tot de andere toetsen ook tijdens de simulaties. De simulatie wordt dan tijdelijk onderbroken en voortgezet nadat de .wpg-file is weggeschreven.

- \* Starten van het eigenlijke programma waarin het verspaningsmodel van Merchant wordt behandeld.
- \* Na beëindigen van het programma wordt het schermkopieprogramma uit het geheugen verwijderd ("graphics" kan niet worden verwijderd).

Na beëindiging van het programma verschijnt opnieuw de DOS-prompt A:> of B:>

### **3 Files op de bijgeleverde diskette**

- command.com: systeemfile van DOS 5.0
- grab.com: schermkopieprogramma van WordPerfect/DrawPerfect
- graphics.com: DOS 5.0 file voor uitprinten van een grafisch scherm
- graphics.pro: instellingen van graphics.com
- autoexec.bat: voor starten computer vanuit diskette
- start.bat: voor starten van het programma met het commando "start"

- verspmod.exe: vanuit turbo-Pascal gecompileerde program-  
mafile van het verspaningsmodel van Mer-  
chant
- sans.chr en  
trip.chr: turbo-Pascal files die de lettertypen bevatten  
die het programma gebruikt
- att.bgi en  
egavga.bgi: turbo-Pascal files voor het aansturen van het  
scherm bij respectievelijk een ATT400 en een  
(super)VGA grafische kaart

Aangezien de sans.chr en trip.chr file in het verloop van het programma meerdere malen worden ingelezen en bovendien de de graphics- en grab-files eventueel gebruikt worden, is het niet mogelijk de diskette tijdens het verloop van het programma (bij gebruik van menu's of wisselen van scherm) uit de drive te halen. Er zal een foutmelding gegeven worden als de diskette niet gelezen kan worden. De diskette moet dan weer in de drive worden gedaan. Het is wel mogelijk even de diskette te verwisselen als het programma "grab" is opgeroepen. Een kort overzicht van bovenstaande mogelijkheden wordt in het programma gegeven na het intoetsen van <Alt><h>.

N.B.: In het programma wordt gebruik gemaakt van de klok van het besturingssysteem. Er zijn computers waar de klok niet gebruikt kan worden als er gestart wordt met behulp van de autoexec.bat file in drive A of drive B.

In het programma zal dan geen tijd verschijnen. Als geen tijd verschijnt is het beter op te starten met "start".

## Literatuur.

- [1] Turbo Pascal User's Guide, version 5.0/5.5, Borland International Inc., 1988.
- [2] Turbo Pascal Reference Guide, version 5.0/5.5, Borland International Inc., 1988.
- [3] Cursus Pascal, prof.dr. A van der Sluis, drs. C.A.C. Görts, Academic service, 's-Gravenhage 1981.
- [4] Power graphics using Turbo Pascal, Keith Weiskamp, Loren Heiny, Namir Shammas, John Wiley & Sons Inc., New York 1989.
- [5] Oriëntatie Produktietechniek A en B, Verspanende bewerkingen, verspanende werktuigen, prof.dr.ir. A.C.H. van der Wolf, ir. J.A.W. Hijink, collegedictaat Technische Universiteit Eindhoven, nummer 4510, zesde druk, Eindhoven 1987.
- [6] Manufacturing Engineering and Technology, Serope Kalpakjian, Addison Wesley publishing company, Illinois 1989.
- [7] The machining of metals, E.J.A. Armarego, R.H. Brown, Prentice Hall Inc., New Jersey 1969.

# BIJLAGE 1.

overzichtstabel.

