

## MASTER

### Aanpassingen aan de $^{124}\text{Xe}$ -bestralingsopstelling voor routinematige productie van $^{123}\text{I}$ op de THE

van den Burg, R.M.W.J.

*Award date:*  
1986

[Link to publication](#)

#### **Disclaimer**

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

#### **General rights**

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

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

#### **Take down policy**

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

Aanpassingen aan de  $^{124}\text{Xe}$ -bestra-  
lingsopstelling voor routinematige  
produktie van  $^{123}\text{I}$  op de THE

Appendices

R.M.W.J. van den Burg  
Technische Hogeschool Eindhoven  
Afdeling der Technische Natuurkunde  
Vakgroep Deeltjesfysica  
Onderwerpgroep Cyclotrontoepassingen  
Afstudeerhoogleraar Prof.dr.ir. H.L. Hagedoorn  
Begeleider ir. A.J. Witsenboer

mei 1986

Afstudeerverslag

## INHOUD

APPENDIX A	Toelichting bij het programma JOOD	A.1
	Listing van het programma JOOD	A.3
	Listing van het programma 'FUNC'	A.14
	Listing van het programma 'GRAF'	A.17
	Voorbeeld van een computeruitdraai	A.19
APPENDIX B.1	Toelichting bij het programma PROSIM	B.1
B.2	Listing van het programma PROSIM en overzicht van de gebruikte variabelen	B.2
	Listing van programma 'IO'	B.11
B.3	Berekening van de stochastische variabelen $x, y, \theta$ en $\phi$	B.19
B.3.1	Transformatie van stochastische variabelen	B.19
B.3.2	Transformatie naar normaal verdeelde stochastische variabelen	B.19
B.3.3	Bepalen van de verstrooihoeken $\theta$ en $\phi$	B.21
B.3.4	Het programma INTEGR	B.23
	Listing van het programma INTEGR	B.24
B.4	De stopping power van verrijkte gassen	B.27
B.5	Invloed van de parameters $N_{folie}$ en $N_{prot}$	B.29
B.6	Invloed van de divergentie van de bundel en de dichtheidsreductie	B.31
B.7	Vergelijking van de met het program- ma PROSIM berekende verstrooiing aan dun nikkel-folie met de door Marion en Zimmerman theoretisch afgeleide verstrooiing	B.32
APPENDIX C.1	Toelichting bij het programma PROGRA	C.1
C.2	Listing van het programma PROGRA en overzicht van de gebruikte variabelen	C.1
C.3	Exitatiefuncties	C.17
APPENDIX D	Fysische eigenschappen van xenon	D.1

APPENDIX A

Om de  $^{123}\text{I}$ -activiteit in de targethouder en in de koelvallen te berekenen, afhankelijk van de bestralingsparameters en de tijdstippen waarop het xenon-gas naar een ander vat wordt overgebracht, is een computerprogramma 'JOOD' geschreven. In deze appendix wordt een toelichting gegeven op dit programma.

Het programma JOOD is, net als alle andere programma's (die beschreven worden in appendix B en C) geschreven in FORTRAN. De listing van het programma JOOD die hieronder afgedrukt is is voorzien van enig commentaar.

De tijdstippen en de tijdsintervallen die in het programma gebruikt worden zijn in fig. A.1 weergegeven.

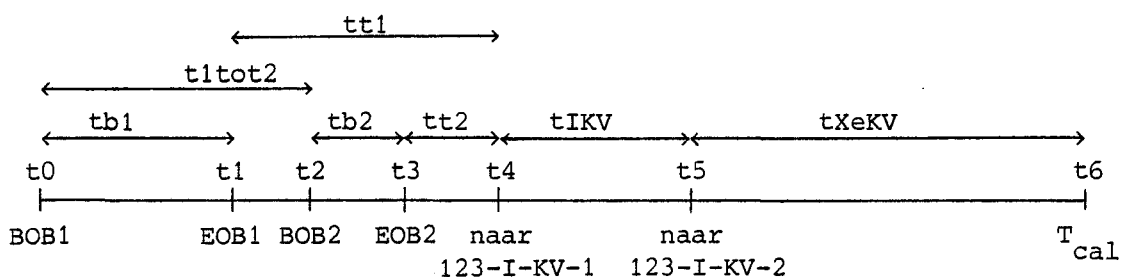


Fig. A.1 Gebruikte tijdstippen en tijdsintervallen in het programma JOOD.

Voor het aantal kernen van een bepaald isotoop op een bepaalde plaats en op een bepaald tijdstip worden variabelen gebruikt die bestaan uit 4 karakters, waarvan de eerste een N is. Het tweede karakter geeft aan welk isotoop het betreft: C= $^{123}\text{Cs}$ , X= $^{123}\text{Xe}$ , I= $^{123}\text{I}$ . Het derde karakter geeft de plaats aan: T = in targethouder, I=in 123-I-KV-1, X=in 123-I-KV-2. Het vierde karakter geeft het tijdstip aan: 1=EOB1, 2=BOB2, 3=EOB2, 4=tijdstip van terugwinnen in 123-I-KV-1, 5=tijdstip van terugwinnen in 123-I-KV-2, 6=calibratietijd.

Het hoofdprogramma JOOD dient gelinkt te worden met de 'programma's' FUNC en GRAF, die een aantal functies bevatten die in het programma JOOD gebruikt worden. Ook listings van de 'programma's' FUNC en GRAF zijn afgedrukt in deze appendix.

```
PROGRAM jood
-----
C
C Alle tijden in uren, alle activiteiten in Ci
C
C Alle variabelen die met N of 1 beginnen zijn Real's:
C IMPLICIT REAL*4(N,1)
C COMMON /B/11,12,13
C
C Declaratie van variabelen:
REAL*4 NCT1,NCT3,NCT4,NXT1,NXT3,NXT4,NIT1,NIT3,NIT4,NIT6,NXI4,
* NXI5,NII5,NII6,NXX5,NXX6,NIX6,t0,t1,t2,t3,t4,t5,tbl,
* tb2,tltot2,ttl,tt2,tXeKV,tIKV,t,ct,xt,it,xi,ii,xx,ix,
* tb2on,tb2bo,tb2oud,tIKVon,tIKVbo,Ngewen,Igem2,Ngew0,
* 11,12,13,sf,rj,P1Cs,P1Xe,P2Cs,P2Xe,ttloud,
* N(7,800),B,druk,temp,uAhl,uAh2,nauwk,
* tX,tI,NXII5,NIII5,NITT6,NIII6,NIXX6,NITOT,maxhoo,
* Igem20,maxho0,druk0,temp0,t10,t20,t30,t40,t50,t60
INTEGER*2 j,k,stap,device,x,regel,best2,best20,stap0,extra
LOGICAL*1 EOBber,vraag,vraagx,gevuld,stop,datum(30)
C l=ct, 2=xt, 3=it, 4=xi, 5=ii, 6=xx, 7=ix
C DATA 11,12,13/6.931,0.3332,0.0525/! vervalconstanten in (1/uur)
C nauwk=.000001
C
C decnaar60(t)=int(t)+(t-int(t))*.6! Omrekenen van 10- naar 60-tallig
C
C -----
C
C inlezen van gegevens
C
C
C
C best20=2
10 gevuld='N'
t60=34.
EOBber='N'
TYPE 1015,best20
ACCEPT 1101,best2
IF(best2.NE. 123) GOTO 11! 123 -> andere vervalconstanten
extra=1
TYPE 1015,best20
ACCEPT 1101,best2
vraagx='N'
TYPE*, 'wil je andere vervalconstanten invoeren (N)'
ACCEPT 1,vraagx
1 FORMAT(A1)
IF (vraagx.NE. 'J') GOTO 11
TYPE*, 'voer 11,12,13 in'
ACCEPT*,11,12,13
C
C
11 IF(best2.EQ. 0) best2=best20
TYPE 1001,druk0
ACCEPT 1100,druk
IF(druk.EQ. 0.) druk=druk0
TYPE 1002,temp0
ACCEPT 1100,temp
IF(temp.EQ. 0.) temp=temp0
TYPE 1003,t00
ACCEPT 1100,t0
```

```
IF(t0.EQ. 0.) t0=t00
TYPE 1004,t10
ACCEPT 1100,t1
IF(t1.EQ. 0.) t1=t10
TYPE 1005,uAh10
ACCEPT 1100,uAh1
IF(uAh1.EQ. 0.) uAh1=uAh10
uAh10=uAh1
t2=t1
t3=t1
uAh2=0
IF (best2.EQ.1) GOTO 20
TYPE 1006,t20
ACCEPT 1100,t2
IF(t2.EQ. 0.) t2=t20
TYPE 1007,t30
ACCEPT 1100,t3
IF(t3.EQ. 0.) t3=t30
TYPE 1008,uAh20
ACCEPT 1100,uAh2
IF(uAh2.EQ. 0.) uAh2=uAh20
20 TYPE 1009,t40
ACCEPT 1100,t4
IF(t4.EQ. 0.) t4=t40
TYPE 1010,t50
ACCEPT 1100,t5
IF(t5.EQ. 0.) t5=t50
TYPE 1011,t60
ACCEPT 1100,t6
IF(t6.EQ. 0.) t6=t60
1001 FORMAT(' Druk (in Bar overdruk)                (' ,F6.2, '))
1002 FORMAT(' Temperatuur van het koelwater         (' ,F6.2, '))
1003 FORMAT(' BOB 1                                  (' ,F6.2, '))
1004 FORMAT(' EOB 1                                  (' ,F6.2, '))
1005 FORMAT(' Aantal uAh bij de le bestraling       (' ,F6.2, '))
1006 FORMAT(' BOB 2                                  (' ,F6.2, '))
1007 FORMAT(' EOB 2                                  (' ,F6.2, '))
1008 FORMAT(' Aantal uAh bij tweede bestraling      (' ,F6.2, '))
1009 FORMAT(' naar 123-I-KV                          (' ,F6.2, '))
1010 FORMAT(' naar Xe-KV                             (' ,F6.2, '))
1011 FORMAT(' Tcal (normaal 34)                     (' ,F6.2, '))
1015 FORMAT(' Een of twee bestralingen ?          (' ,I6, '))
1016 FORMAT(' Stapgrootte                            (' ,I6, '))
1100 FORMAT(F8.0)
1101 FORMAT(I8)
C
best20=best2
druk0=druk
temp0=temp
t00=t0
t10=t1
t20=t2
t30=t3
t40=t4
```

```
t50=t5
t60=t6
uAh10=uAh1
uAh20=uAh2
TYPE*, 'OKE ? (J)'
ACCEPT 1, vraag
IF (vraag.EQ. 'N') GOTO 10
```

C  
C  
C  
C  
C

-----  
tijdsduren en productietempi berekenen  
-----

```
tbl =tijdverschil(t0,t1)! bestralingstijd 1
ttl =tijdverschil(t1,t4)! tijd in targetbuis 1
tltot2=tijdverschil(t0,t2)! tijd tussen BOB1 en BOB2
tb2 =tijdverschil(t2,t3)! bestralingstijd 2
tt2 =tijdverschil(t3,t4)! tijd in targetbuis 2
tIKV =tijdverschil(t4,t5)! tijd in I-KV
tXeKV =tijdverschil(t5,t6)! tijd in Xe-KV
P1Cs=.06666*uAh1*(druk+1)*(temp+273)/(tbl*2.48*290)! exp.bep.
P1Xe=.06666*uAh1*(druk+1)*(temp+273)/(tbl*2.48*290)
P2Cs=0.
P2Xe=0.
IF (best2.NE.1)
* P2Cs=.06666*uAh2*(druk+1)*(temp+273)/(tb2*2.48*290)
IF (best2.NE.1)
* P2Xe=.06666*uAh2*(druk+1)*(temp+273)/(tb2*2.48*290)
IF (extra.NE.1) GOTO 30
TYPE*, ' P1Cs:', P1Cs
TYPE*, ' P1Xe:', P1Xe
TYPE*, ' P2Cs:', P2Cs
TYPE*, ' P2Xe:', P2Xe
vraagx='N'
TYPE*, 'wil je de Productiesnelheden veranderen (N)'
ACCEPT 1, vraagx
IF (vraagx.NE. 'J') GOTO 30
TYPE*, 'voer verm.factoren in voor P1Cs,P2Cs,P1Xe,P2Xe'
ACCEPT*, flc, f2c, flx, f2x
P1Cs=P1Cs*flc
P2Cs=P2Cs*f2c
P1Xe=P1Xe*flx
P2Xe=P2Xe*f2x
```

-----  
P is tevens de evenwichtsactiviteit van 123-Cs resp. 123-Xe  
in C1. Om P in (1/s) te verkrijgen: vermenigv. met 3.7E10.  
-----

C  
C  
C  
C  
C  
C  
30

```
device=7
TYPE*, ' Op Scherm, Printer of Geen output ? (S)'
ACCEPT 1, vraag
IF (vraag .EQ. 'P') device=6
IF (vraag .EQ. 'G') GOTO 305
```

C



```
130 TYPE*, 'Voer datum in (eerst een spatie)'  
TYPE*  
CALL GTLIN(datum)  
  
C  
C  
C -----  
C invoergegevens printen/displayen  
C -----  
C  
135 WRITE (device,140) (datum(k),k=1,30)  
140 FORMAT (5X,30A1)  
WRITE(device,150) t0  
150 FORMAT('0 BOB 1 :',F6.2)  
WRITE(device,160) t1,decnaar60(tbl)  
160 FORMAT(' EOB 1 :',F6.2,' (',F5.2,' uur na BOB1)')  
IF (best2.NE.1) WRITE(device,170) t2,decnaar60(tltot2)  
170 FORMAT(' BOB 2 :',F6.2,' (',F5.2,' uur na BOB1)')  
IF (best2.NE.1) WRITE(device,180) t3,decnaar60(tltot2+tb2)  
180 FORMAT(' EOB 2 :',F6.2,' (',F5.2,' uur na BOB1)')  
181 FORMAT(' aantal uAh 2e bestr. :',F6.2,)  
WRITE(device,190) t4,decnaar60(tbl+t1l)  
190 FORMAT(' naar I-KV :',F6.2,' (',F5.2,' uur na BOB1)')  
WRITE(device,200) t5,decnaar60(tbl+t1l+tIKV)  
200 FORMAT(' naar Xe-KV :',F6.2,' (',F5.2,' uur na BOB1)')  
WRITE(device,210) 10.,decnaar60(tbl+t1l+tIKV+tXeKV)  
210 FORMAT(' Tcal :',F6.2,' (',F5.2,' uur na BOB1)')  
*)  
  
C  
WRITE(device,40) decnaar60(tbl)  
40 FORMAT(' bestralingsduur1 :',F6.2)  
IF (best2.NE.1) WRITE(device,50) decnaar60(tb2)  
IF (best2.NE.1) WRITE(device,60) decnaar60(tltot2-tbl)  
IF (best2.NE.1) WRITE(device,70) decnaar60(t1l),decnaar60(tt2)  
IF (best2.EQ.1) WRITE(device,80) decnaar60(t1l)  
50 FORMAT(' bestralingsduur2 :',F6.2)  
60 FORMAT(' tijd tussen 1 en 2 :',F6.2)  
70 FORMAT(' in targetbuis :',F6.2,' resp. :',F6.2)  
80 FORMAT(' in targetbuis :',F6.2)  
WRITE(device,90) decnaar60(tIKV)  
90 FORMAT(' in 123-I-KV :',F6.2)  
WRITE(device,100) decnaar60(tXeKV)  
100 FORMAT(' in Xe-KV :',F6.2)  
WRITE(device,220) uAh1  
220 FORMAT('/', ' Aantal uAh1 :',F7.3)  
IF (best2.NE.1) WRITE(device,230) uAh2  
230 FORMAT(' Aantal uAh2 :',F7.3)  
WRITE(device,240) uAh1/tbl  
240 FORMAT(' Igemidd.1 (uA) :',F7.3)  
IF (best2.NE.1) WRITE(device,250) uAh2/(tb2+1E-20)  
250 FORMAT(' Igemidd.2 (uA) :',F7.3)  
WRITE(device,260) druk  
260 FORMAT(' Druk in Bar overdruk :',F7.3)  
WRITE(device,270) temp  
270 FORMAT(' Temperatuur in gr. C :',F7.3)  
WRITE(device,280) P1Cs,P1Xe,P2Cs,P2Xe  
280 FORMAT(' P1Cs,P1Xe,P2Cs,P2Xe :',4F7.3)
```

```
WRITE(device,290)
290  FORMAT(/' Berekend met een opbrengst van 1.4 mCi/uAh bij
      * een bestraling om 0.00 uur met')
WRITE(device,300)
300  FORMAT(' p=2.48 bar, T=290K.')
```

C  
C  
C  
C  
C

```
-----
      matrixelementen berekenen
-----
```

```
305  NCT1=NCsbestralen(P1Cs,tb1)
      NCT3=NCsbestralen(P2Cs,tb2)
      NXT1=NXebestralen(P1Cs,P1Xe,tb1)
      NXT3=NXebestralen(P2Cs,P2Xe,tb2)
      NIT1=NIbestralen (P1Cs,P1Xe,tb1)
      NIT3=NIbestralen (P2Cs,P2Xe,tb2)
      NCT4=NCstargetbuis(NCT1,tt1)+NCstargetbuis(NCT3,tt2)
      NXT4=NXetargetbuis(NCT1,NXT1,tt1)+NXetargetbuis(NCT3,NXT3,tt2)
      NIT4=NITargetbuis (NCT1,NXT1,NIT1,tt1)
      * +NITargetbuis (NCT3,NXT3,NIT3,tt2)
      NXI4=NXT4
      NXI5=NXeIKV(NXI4,tIKV)
      NII5=NIIKV (NXI4,tIKV)
      NXX5=NXI5
      NIT6=NInaIKVintargetbuis(NIT4,tIKV+tXeKV)
      NII6=NInaXeKVinIKV(NII5,tXeKV)
      NXX6=NXeXeKV(NXX5,tXeKV)
      NIX6=NIXeKV(NXX5,tXeKV)
      NITOT6=(NIT6+NII6+NIX6)/100.
      IF ((EOBber.NE.'N') .AND. (EOBber.NE.'Z')) GOTO 437
```

C  
C  
C  
C  
C

```
-----
      berekende matrixelementen en activiteiten uitprinten/displayen
-----
```

```
307  WRITE(device,310)
310  FORMAT(/' Matrixelementen (in mCi) :'/)
      WRITE(device,320) NCT1*11*1000.,NCT3*11*1000.,NCT4*11*1000.
320  FORMAT(' ct1=',F9.3,' ct3=',F9.3,' ct4=',F9.3)
      WRITE(device,330) NXT1*12*1000.,NXT3*12*1000.,NXT4*12*1000.,
      * NXI5*12*1000.,NXX6*12*1000.
330  FORMAT(' xt1=',F9.3,' xt3=',F9.3,' xt4=',F9.3,' xi5=',F9.3,
      * ' xx6=',F9.3)
      WRITE(device,340) NIT1*13*1000.,NIT3*13*1000.,NIT4*13*1000.,
      * NIT4*13*1000.*exp(-13*tIKV),NII5*13*1000.
340  FORMAT(' it1=',F9.3,' it3=',F9.3,' it4=',F9.3,' it5=',F9.3,
      * ' i15=',F9.3)
      WRITE(device,350) NIT6*13*1000.,NII6*13*1000.,NIX6*13*1000.,
      * (NIT6+NII6+NIX6)*13*1000.
350  FORMAT(' it6=',F9.3,' i16=',F9.3,' ix6=',F9.3,' IT6=',F9.3)
360  WRITE(device,360) NIT6*13*1000.,NII6*13*1000.,NIX6*13*1000.
      FORMAT
      *(/' I-Act. in mCi op Tcal. (target,IKV,XeKV) resp. : ',
      *3F9.3)
```

```
WRITE(device,370) NIT6/NITOT6,NII6/NITOT6,NIX6/NITOT6
370 FORMAT
*(' Percentages op Tcal. (target,IKV,XeKV) resp. : ',
*3F9.3/)
C
C
C -----
C Nu volgt het berekenen van de activiteitsverdeling als functie
C van het tijdstip waarop teruggewonnen wordt in de Xe-KV.
C -----
C
WRITE(device,380)
WRITE(device,390)
WRITE(device,400)
380 FORMAT(' I-123-Activiteit (in mCi) als functie van')
390 FORMAT(' tijdstip van terugwinnen naar XeKV')
400 FORMAT(
* ' T naarXeKV Atarget % AIKV %
*AXeKV %')
DO 410 x=95,130,5
tI=tijdverschil(t4,decnaar60(x/10.))
tX=tijdverschil(decnaar60(x/10.),t6)
NXII5=NXeIKV(NXI4,tI)
NIII5=NIIKV (NXI4,tI)
NITT6=NInaIKVintargetbuis(NIT4,tI+tX)
NIII6=NInaXeKVinIKV(NIII5,tX)
NIXX6=NIXeKV(NXII5,tX)
NITOT=(NITT6+NIII6+NIXX6)/100.
WRITE(device,420) decnaar60(x/10.),NITT6*13*1000.,NITT6/NITOT,
* NIII6*13*1000.,NIII6/NITOT,NIXX6*13*1000.,NIXX6/NITOT
420 FORMAT(X,F10.2,3(F13.3,F7.2))
410 CONTINUE
C
WRITE(device,420)
WRITE(device,430) (NIT6+NII6+NIX6)*13*1000.
430 FORMAT(' Totale Activiteit op Tcal. :',F9.3,' mCi')
IF (device.EQ.6) CLOSE(UNIT=6)
IF (EOBber.EQ.'Z') GOTO 439
C
C
C -----
C EOB 2 berekenen
C -----
C
IF (best2.NE.2) GOTO 439
EOBber='N'
TYPE*, ' Wil je EOB 2 berekenen (N)'
ACCEPT 1,EOBber
IF (EOBber .NE. 'J') GOTO 439
TYPE 1012,Ngew0
1012 FORMAT(' Gewenste I-Activiteit in target+123-I-KV (mCi) (' ,
* F6.2,')')
ACCEPT 1100,Ngewen
IF(Ngewen.EQ. 0.) Ngewen=Ngew0
Ngew0=Ngewen
TYPE 1013,Igem20
1013 FORMAT(' Gemiddelde stroom (uA) (' ,F6.2,')')
```

```
ACCEPT 1100,Igem2
IF(Igem2.EQ. 0.) Igem2=Igem20
Igem20=Igem2
Ngewen=Ngewen/1000/13! naar aantal kernen omrekenen (13 in uren)
P2Cs=.06666*Igem2*(druk+1)*(temp+273)/(2.48*290)
P2Xe=.06666*Igem2*(druk+1)*(temp+273)/(2.48*290)
device=7
tb2oud=tb2
tIKVoud=tIKV
ttloud=ttl
tb2on=tb2oud-2.
tb2bo=tb2oud+4.
435 tb2=(tb2bo+tb2on)/2.
tIKV=tIKVoud+(tb2oud-tb2)
ttl=ttloud+(tb2-tb2oud)
GOTO 305! eigenlijk GOSUB 305, maar dat bestaat niet in FORTRAN

C
437 CONTINUE! doet niets
D TYPE*, 'tb2, activiteit :', decnaar60(tb2), (NIT6+NII6)*1000*13
IF ((NIT6+NII6).GT.Ngewen) tb2bo=tb2
IF ((NIT6+NII6).LE.Ngewen) tb2on=tb2
IF (ABS((NIT6+NII6-Ngewen)/(NIT6+NII6+Ngewen)).GT.nauwk) GOTO 435
EOBber='Z'! EOB 2 is nu berekend
t3=decnaar60(AMOD(tijdverschil(0.,t3)+tb2-tb2oud+24.,24.))
t4=decnaar60(AMOD(tijdverschil(0.,t4)+tb2-tb2oud+24.,24.))
uAh2=tb2*Igem2
ttl =tijdverschil(t1,t4)! tijd in targetbuis 1
tb2 =tijdverschil(t2,t3)! bestralingstijd 2
tt2 =tijdverschil(t3,t4)! tijd in targetbuis 2
tIKV =tijdverschil(t4,t5)! tijd in I-KV
t30=t3
t40=t4
uAh20=uAh2
TYPE*
TYPE 180, t3,decnaar60(tltot2+tb2)
TYPE 181, uAh2
device=7
TYPE*, ' Op Scherm, Printer of Geen output ? (S)'
ACCEPT 1,vraag
IF (vraag .EQ. 'P') device=6
IF (vraag .EQ. 'G') GOTO 305
GOTO 135

C
C
C
439 TYPE*, 'Wil je een lijst printen of een plaatje maken ? (N)'
ACCEPT 1,vraag
IF (vraag.NE.'J') GOTO 1000
IF (gevuld.EQ.'J') GOTO 440

C
C
C
vullen van array met N voor Cs in target(1), Xe in target(2),
I in target(3), Xe in IKV(4), I in IKV(5), Xe in XeKV(6), I in XeKV(7)
op tijdstippen n*3 minuten na BOB1, n = 1..800.
C
C
```

```
DO 450 j=1,800
t=j/20.
IF (t .LT. tbl)          GOTO 470
IF (t .LT. tltot2)      GOTO 480
IF (t .LT. tltot2+tb2)  GOTO 490
IF (t .LT. tbl+ttl)     GOTO 500
IF (t .LT. tbl+ttl+tIKV) GOTO 510
GOTO 460
470 N(1,j)=NCsbestralen(PlCs,t)*11
    N(2,j)=NXebestralen(PlCs,PlXe,t)*12
    N(3,j)=NIBestralen (PlCs,PlXe,t)*13
    N(4,j)=0.
    N(5,j)=0.
    N(6,j)=0.
    N(7,j)=0.
    GOTO 450
480 N(1,j)=NCstargetbuis(NCT1,          t-tbl)*11
    N(2,j)=NXetargetbuis(NCT1,NXT1,     t-tbl)*12
    N(3,j)=NITargetbuis (NCT1,NXT1,NIT1,t-tbl)*13
    N(4,j)=0.
    N(5,j)=0.
    N(6,j)=0.
    N(7,j)=0.
    GOTO 450
490 N(1,j)=(NCstargetbuis(NCT1,t-tbl)+NCsbestralen(P2Cs,t-tltot2))*11
    N(2,j)=(NXetargetbuis(NCT1,NXT1,     t-tbl)
    *      +NXebestralen(P2Cs,P2Xe,t-tltot2))*12
    N(3,j)=(NITargetbuis (NCT1,NXT1,NIT1,t-tbl)
    *      +NIBestralen (P2Cs,P2Xe,t-tltot2))*13
    N(4,j)=0.
    N(5,j)=0.
    N(6,j)=0.
    N(7,j)=0.
    GOTO 450
500 N(1,j)=(NCstargetbuis(NCT1,          t-tbl)
    *      +NCstargetbuis(NCT3,          t-tltot2-tb2))*11
    N(2,j)=(NXetargetbuis(NCT1,NXT1,     t-tbl)
    *      +NXetargetbuis(NCT3,NXT3,     t-tltot2-tb2))*12
    N(3,j)=(NITargetbuis (NCT1,NXT1,NIT1,t-tbl)
    *      +NITargetbuis (NCT3,NXT3,NIT3,t-tltot2-tb2))*13
    N(4,j)=0.
    N(5,j)=0.
    N(6,j)=0.
    N(7,j)=0.
    GOTO 450
510 N(1,j)=0.
    N(2,j)=0.
    N(6,j)=0.
    N(7,j)=0.
    N(3,j)=NInaIKVintargetbuis(NIT4,t-tbl-ttl)*13
    N(4,j)=NXeIKV          (NXI4,t-tbl-ttl)*12
    N(5,j)=NIIKV          (NXI4,t-tbl-ttl)*13
    GOTO 450
460 N(1,j)=0.
    N(2,j)=0.
```

```
N(4,j)=0.
N(3,j)=NInaIKVintargetbuis(NIT4,t-tbl-ttl      )*13
N(5,j)=NInaXeKVinIKV      (NII5,t-tbl-ttl-tIKV)*13
N(6,j)=NXeXeKV      (NXX5,t-tbl-ttl-tIKV)*12
N(7,j)=NIXeKV      (NXX5,t-tbl-ttl-tIKV)*13
450 CONTINUE
520 gevuld='J'
    vraag='N'
    TYPE*, 'Wil je een lijst printen ? (N)'
    ACCEPT 1,vraag
    IF (vraag .NE. 'J') GOTO 570

C
C
C
C
-----
lijst printen
-----

TYPE*, 'Voer stapgrootte in (1 = 3 min) '
TYPE 1016,stap0
ACCEPT 1101,stap
IF(stap.EQ. 0) stap=stap0
stap0=stap
WRITE(device,530)
530 FORMAT ('LACTIVITEITSVERDELING NA BOB 1'/)
WRITE(device,540)
540 FORMAT(
* ' Tijd    Cs-targ  Xe-targ  I-targ    Xe-IKV    I-IKV
*   Xe-XeKV  I-XeKV    I-Totaal'//)
DO 550 j=stap,800,stap
  WRITE (device,560)
  *      decnaar60(AMOD(tijdverschil(0.,t0)+j/20.,24.)),
  *      N(1,j),N(2,j),N(3,j),N(4,j),N(5,j),N(6,j),N(7,j),
  *      N(3,j)+N(5,j)+N(7,j)
560   FORMAT (X,F6.2,2X,8(F8.5,X))
550 CONTINUE
440 IF (device.EQ.6) CLOSE(UNIT=6)
570 TYPE*, 'Wil je een plaatje ? (N)'
    ACCEPT 1,vraag
    IF (vraag .NE. 'J') GOTO 1000

C
C
C
C
-----
plaatjes tekenen
-----

TYPE*, 'Voer stapgrootte in (1 = 3 min) '
ACCEPT*,stap
TYPE*, 'Wil je Cs+Xe+I of alleen I of een ALI-plaatje (C)'
ACCEPT 1,vraag
TYPE 1014,maxhoo
1014 FORMAT(' Maximale hoogte van het plaatje (in GBq resp ALI)
* bv 100 GBq of 150 ALI (' ,F5.1,')')
ACCEPT 1100,maxhoo
IF(maxhoo.EQ. 0.) maxhoo=maxhoo
maxhoo0=maxhoo

C
CALL omrand

C
```

```
DO 590 j=120,800,120
  CALL line( j/l., 0., j/l., 10.)
  CALL line( j/l.,479., j/l.,469.)
590 CONTINUE
C
DO 595 j=20,800,20
  CALL line( j/l., 0., j/l., 5.)
  CALL line( j/l.,479., j/l.,474.)
595 CONTINUE
C
DO 600 j=47,479,48
  CALL line( 1., j/l., 15., j/l.)
  IF (vraag .EQ. 'A') CALL line(800., j/l.,785., j/l. )
600 CONTINUE
C
IF (vraag .EQ. 'A') GOTO 605
DO 605 j=2,27,2
  CALL line(800.,j*480./27.,785.,j*480./27.)
605 CONTINUE
C
CALL line(tbl*20., 0.,tbl*20., 20.)
CALL line(tbl*20.,479.,tbl*20.,459.)
CALL line(tltot2*20., 0.,tltot2*20., 20.)
CALL line(tltot2*20.,479.,tltot2*20.,459.)
CALL line((tltot2+tb2)*20., 0.,(tltot2+tb2)*20., 20.)
CALL line((tltot2+tb2)*20.,479.,(tltot2+tb2)*20.,459.)
CALL line((tbl+ttl)*20. ,0.,(tbl+ttl)*20., 20.)
CALL line((tbl+ttl)*20.,479.,(tbl+ttl)*20.,459.)
CALL line((tbl+ttl+tIKV)*20., 0.,(tbl+ttl+tIKV)*20., 20.)
CALL line((tbl+ttl+tIKV)*20.,479.,(tbl+ttl+tIKV)*20.,459.)
CALL line((tbl+ttl+tIKV+tXeKV)*20.,479.,(tbl+ttl+tIKV+tXeKV)
*      *20.,459.)
C
regel=0
CALL text(500.,470.-regel*25.)
TYPE 140, (datum(k),k=1,30)
TYPE 610
610 FORMAT(X, ' ')
regel=regel+1
C
CALL text(500.,470.-regel*25.)
TYPE 620,t0,t1
620 FORMAT(' BOB1 ',F6.2,' EOB1 ',F6.2,' ')
regel=regel+1
C
IF (best2.EQ.1) GOTO 640
CALL text(500.,470.-regel*25.)
TYPE 630,t2,t3
630 FORMAT(' BOB2 ',F6.2,' EOB2 ',F6.2,' ')
regel=regel+1
C
640 CALL text(500.,470.-regel*25.)
TYPE 650,t4
650 FORMAT(' naar 123-I-KV-1 ',F6.2,' ')
regel=regel+1
C
```

```
CALL text(500.,470.-regel*25.)
TYPE 660,t5
660 FORMAT(' naar 123-I-KV-2 ',F6.2,'''')
regel=regel+1
C
IF(vraag .EQ. 'A') GOTO 690
CALL text(500.,470.-regel*25.)
TYPE 670,maxhoogte,maxhoogte/37.
670 FORMAT(' maximum ',F5.1,' GBq','F7.3,' Ci''')
C
GOTO 680
690 CALL text(500.,470.-regel*25.)
TYPE 700,maxhoogte
700 FORMAT(' maximum (in ALI)',F9.2,'''')
680 IF (vraag .EQ. 'I') GOTO 730
IF (vraag .EQ. 'A') GOTO 760
sf=479.*3.7E10/maxhoogte/l.E9
DO 720 j=1,800,stap
  rj=FLOAT(j)
  DO 710 k=1,7
    CALL plot(rj,N(k,j)*sf)
710 CONTINUE
  CALL plot(rj,(N(3,j)+N(5,j)+N(7,j))*sf)
720 CONTINUE
GOTO 780
730 sf=479.*3.7E10/maxhoogte/l.E9! I-plaatje
DO 750 j=1,800,stap
  rj=FLOAT(j)
  DO 740 k=3,7,2
    CALL plot(rj,N(k,j)*sf)
740 CONTINUE
  CALL plot(rj,(N(3,j)+N(5,j)+N(7,j))*sf)
750 CONTINUE
GOTO 780
C
760 sf=479./maxhoogte! ALI-plaatje
DO 770 j=1,800,stap
  rj=FLOAT(j)
  alics=N(1,j)*4.63! 1/(8E9/3.7E10)
  alixe=(N(2,j)+N(4,j)+N(6,j))*2.46667! 1/(2500*6E6/3.7E10)
  alii =(N(3,j)+N(5,j)+N(7,j))*185! 1/(2E8/3.7E10)
  CALL plot(rj,alics*sf)
  CALL plot(rj,alixe*sf)
  CALL plot(rj,alii *sf)
  CALL plot(rj,(alics+alixe+alii)*sf)
770 CONTINUE
C
780 ACCEPT 1,vraag
CALL REGex1
TYPE*, 'Nog een plaatje (J) ?'
ACCEPT 1,vraag
IF (vraag .NE. 'N') GOTO 580
1000 TYPE*, 'Nog een bestraling (J) ?'
vraag='N'
ACCEPT 1,vraag
IF (vraag .NE. 'N') GOTO 10
END
```



```
C ----- func ( ALLE TIJDEN IN UREN) -----
C
C Omrekenen naar een positief tijdverschil als begintijd < eindtijd
C Bovendien omrekenen van 60-tallig naar decimaal stelsel
C
REAL FUNCTION tijdverschil(begintijd,eindtijd)
  startdec=int(begintijd)+(begintijd-int(begintijd))/.6
  einddec =int(eindtijd) +(eindtijd -int(eindtijd)) /.6
  verschildec=einddec-startdec
  tijdverschil=(verschildec)
  if (verschildec .GE. 0) RETURN
  tijdverschil=(verschildec+24)
RETURN
END

C
C Berekenen van N voor Cs,Xe en I na bestraling van tb uur
C bij een productietempo PCs + PXe
C
REAL FUNCTION NCsbestralen(PCs,tb)
  REAL*4 11,12,13,PCs,tb
  COMMON/B/11,12,13
  NCsbestralen=(1.-exp(-11*tb))/11*PCs
RETURN
END

C
REAL FUNCTION NXebestralen(PCs,PXe,tb)
  REAL*4 11,12,13,PCs,PXe,tb
  COMMON/B/11,12,13
  NXebestralen=((1.-exp(-12*tb))/12+(exp(-12*tb)
* -exp(-11*tb))/(12-11))*PCs+
* (1.-exp(-12*tb))/12*PXe
RETURN
END

C
REAL FUNCTION NIBestralen(PCs,PXe,tb)
  REAL*4 11,12,13,PCs,PXe,tb
  COMMON/B/11,12,13
  NIBestralen=
* ((1.-exp(-13*tb))/13+(exp(-13*tb)-exp(-12*tb))/(13-12)
* +(exp(-13*tb)-exp(-11*tb))/(13-11)
* + (exp(-12*tb)-exp(-13*tb))/(13-12))*12/(12-11))*PCs
* +(1.-exp(-13*tb))/13+(exp(-13*tb)-exp(-12*tb))/(13-12))*PXe
RETURN
END

C
C N voor Cs,Xe en I als het gas tintarget uur in de targetbuis
C gezeten heeft.
C NCsEOB,NXeEOB,NIEOB zijn aantallen kernen aan het eind van een
C bestraling.
C
REAL FUNCTION NCstargetbuis(NCsEOB,tintarget)
  REAL*4 11,12,13,NCsEOB,tintarget
  COMMON/B/11,12,13
  NCstargetbuis=NCsEOB*exp(-11*tintarget)
RETURN
END
```

```
REAL FUNCTION NXetargetbuis(NCsEOB,NXeEOB,tintarget)
      REAL*4 11,12,13,NCsEOB,NXeEOB,tintarget
      COMMON/B/11,12,13
      NXetargetbuis=11*NCsEOB/(12-11)*(exp(-11*tintarget)-
*   exp(-12*tintarget))+NXeEOB*exp(-12*tintarget)
RETURN
END
```

C

```
REAL FUNCTION NItargetbuis(NCsEOB,NXeEOB,NIEOB,tintarget)
      REAL*4 11,12,13,NCsEOB,NXeEOB,NIEOB,tintarget
      COMMON/B/11,12,13
      NItargetbuis=((exp(-11*tintarget)-exp(-13*tintarget))/(13-11)
*   +(exp(-12*tintarget)-exp(-13*tintarget))/(12-13))*11*12*
*   NCsEOB/(12-11)
*   +12*NXeEOB/(13-12)*(exp(-12*tintarget)-
*   exp(-13*tintarget))+NIEOB*exp(-13*tintarget)
RETURN
END
```

C

C

N voor I in targetbuis, tnaIKV uur na terugwinnen in IKV.  
NIbegin is aantal I-kernen op tijdstip van terugwinnen.

C

C

```
REAL FUNCTION NIaIKVintargetbuis(NIbegin,tnaIKV)
      REAL*4 11,12,13,NIbegin,tnaIKV
      COMMON/B/11,12,13
      NIaIKVintargetbuis=NIbegin*exp(-13*tnaIKV)
RETURN
END
```

C

C

N voor Xe in IKV, tinIKV uur na terugwinnen in IKV.  
NXebeginIKV is aantal Xe-kernen op tijdstip van terugwinnen.

C

C

```
REAL FUNCTION NXeIKV(NXebeginIKV,tinIKV)
      REAL*4 11,12,13,NXebeginIKV,tinIKV
      COMMON/B/11,12,13
      NXeIKV=NXebeginIKV*exp(-12*tinIKV)
RETURN
END
```

C

C

N voor I in IKV, tinIKV uur na terugwinnen in IKV.  
NIbeginIKV is aantal I-kernen op tijdstip van terugwinnen.

C

C

```
REAL FUNCTION NIIKV(NXebeginIKV,tinIKV)
      REAL*4 11,12,13,NXebeginIKV,tinIKV
      COMMON/B/11,12,13
      NIIKV=12*NXebeginIKV*(exp(-12*tinIKV)-exp(-13*tinIKV))/
*   (13-12)
RETURN
END
```

C

C

N voor I in IKV, tnaXeKV uur na terugwinnen in XeKV.  
NIbegin is aantal I-kernen op tijdstip van terugwinnen.

C

```
REAL FUNCTION NInaXeKVinIKV(NIbegin,tnaXeKV)
      REAL*4 11,12,13,NIbegin,tnaXeKV
      COMMON/B/11,12,13
      NInaXeKVinIKV=NIbegin*exp(-13*tnaXeKV)
RETURN
END
```

C  
C  
C  
C

N voor Xe in XeKV, tinXeKV uur na terugwinnen in XeKV.  
NXebeginXeKV is aantal Xe-kernen op tijdstip van terugwinnen.

```
REAL FUNCTION NXeXeKV(NXebeginXeKV,tinXeKV)
      REAL*4 11,12,13,NXebeginXeKV,tinXeKV
      COMMON/B/11,12,13
      NXeXeKV=NXebeginXeKV*exp(-12*tinXeKV)
RETURN
END
```

C  
C  
C  
C

N voor I in XeKV, tinXeKV uur na terugwinnen in XeKV.  
NXebeginXeKV is aantal Xe-kernen op tijdstip van terugwinnen.

```
REAL FUNCTION NIXeKV(NXebeginXeKV,tinXeKV)
      REAL*4 11,12,13,NXebeginXeKV,tinXeKV
      COMMON/B/11,12,13
      NIXeKV=12*NXebeginXeKV*(exp(-12*tinXeKV)-exp(-13*tinXeKV))
      * / (13-12)
RETURN
END
```

```
C      ---- graf (grafische routines) ----
      SUBROUTINE plot(x,y)
      REAL*4 x,y
      TYPE 100,x,y
100    FORMAT(' PC',F5.1,',',F5.1,'JVLJ')
      END
      SUBROUTINE pos(x,y)
      REAL*4 x,y
      TYPE 100,x,y
100    FORMAT(' PC',F5.1,',',F5.1,'J')
      END
      SUBROUTINE line(x1,y1,x2,y2)
      REAL*4 x1,y1,x2,y2
      TYPE 100,x1,y1,x2,y2
100    FORMAT(' PC',F5.1,',',F5.1,'JVL',F5.1,',',F5.1,'J')
      END
      SUBROUTINE text(x,y)
      REAL*4 x,y
      TYPE 100,x,y
100    FORMAT(' PC',F5.1,',',F5.1,'JT"')
      END
      SUBROUTINE schaal(x1,y1,x2,y2)
      REAL*4 x1,y1,x2,y2
      TYPE 100,x1,y2,x2,y1
100    FORMAT(' S(AL',F5.1,',',F5.1,'JL',F5.1,',',F5.1,'J)')
      END
      SUBROUTINE clrscr
      TYPE 100
100    FORMAT(' S(E)')
      END
      SUBROUTINE REGent
      TYPE 100, 27
100    FORMAT(X,A1,'P0p')
      END
      SUBROUTINE REGexi
      TYPE 100, 27
100    FORMAT(X,A1,'[lp')
      END
      SUBROUTINE hrdcop
      TYPE 100, 27
100    FORMAT(X,A1,'[?oi')
      END
      SUBROUTINE frame
      CALL rand
      CALL line(1.,240.,800.,240.)
      CALL line(161.,0.,161.,479.)
      CALL line(321.,0.,321.,479.)
      CALL line(481.,0.,481.,479.)
      CALL line(641.,0.,641.,479.)

C
      DO 100 k=1,800,20
      fk=FLOAT(k)
      CALL line(fk,240.,fk,244.)
100    CONTINUE
      RETURN
      END
```

```
SUBROUTINE omrand
CALL REGent
CALL clrscr
CALL schaal(1.,0.,800.,479.)
CALL line(1.,0.,800.,0.)
CALL line(800.,0.,800.,479.)
CALL line(1.,0.,1.,479.)
CALL line(1.,479.,600.,479.)
RETURN
END
SUBROUTINE rand
CALL omrand
DO 100 k=1,800,20
fk=FLOAT(k)
CALL line(fk,0.,fk,4.)
CALL line(fk,479.,fk,475.)
100 CONTINUE
DO 200 k=24,479,24
fk=FLOAT(k)
CALL line(1.,fk,5.,fk)
CALL line(796.,fk,800.,fk)
200 CONTINUE
RETURN
END
SUBROUTINE rand2
CALL REGent
CALL clrscr
CALL schaal(1.,0.,800.,479.)
CALL line(1.,0.,800.,0.)
CALL line(800.,0.,800.,400.)
CALL line(1.,0.,1.,400.)
CALL line(1.,400.,800.,400.)
DO 100 k=1,800,40
fk=FLOAT(k)
CALL line(fk,0.,fk,5.)
CALL line(fk,394.,fk,400.)
100 CONTINUE
CALL line(401.,0.,401.,400.)
DO 200 k=40,400,40
fk=FLOAT(k)
CALL line(1.,fk,5.,fk)
CALL line(401.,fk,405.,fk)
CALL line(795.,fk,800.,fk)
200 CONTINUE
RETURN
END
```

14 APRIL 1986

BOB 1 : 16.45  
 EOB 1 : 20.05 ( 3.20 uur na BOB1)  
 BOB 2 : 22.55 ( 6.10 uur na BOB1)  
 EOB 2 : 3.18 (10.33 uur na BOB1)  
 naar I-KV : 3.36 (10.51 uur na BOB1)  
 naar Xe-KV : 10.40 (17.55 uur na BOB1)  
 Tcal : 10.00 (41.15 uur na BOB1)

bestralingsduur1 : 3.20  
 bestralingsduur2 : 4.23  
 tijd tussen 1 en 2 : 2.50  
 in targetbuis : 7.31 resp. : 0.18  
 in I23-I-KV : 7.04  
 in Xe-KV : 23.20

Aantal uAhl : 79.240  
 Aantal uAh2 : 112.000  
 Igemidd.1 (uA) : 23.772  
 Igemidd.2 (uA) : 25.551  
 Druk in Bar overdruk : 1.370  
 Temperatuur in gr. C : 18.500  
 P1Cs, P1Xe, P2Cs, P2Xe : 1.522 1.522 1.636 1.636

Berekend met een opbrengst van 1.4 mCi/uAh bij een bestraling om 0.00 uur met p=2.48 bar, T=290K.

Matrixelementen (in mCi) :

ct1= 1522.189 ct3= 1636.124 ct4= 204.545  
 xt1= 2016.427 xt3= 2493.534 xt4= 2491.819 x15= 236.549 xx6= 0.099  
 it1= 191.452 it3= 320.324 it4= 713.574 it5= 492.398 ii5= 277.353  
 it6= 144.646 ii6= 81.475 ix6= 12.978 IT6= 239.098

I-Act. in mCi op Tcal. (target, IKV, XeKV) resp. : 144.646 81.475 12.978

Percentages op Tcal. (target, IKV, XeKV) resp. : 60.496 34.076 5.428

I-123-Activiteit (in mCi) als functie van  
 tijdstip van terugwinnen naar XeKV

T naar XeKV	Atarget	%	AIKV	%	AXeKV	%
9.30	144.646	60.50	76.439	31.97	18.014	7.53
10.00	144.646	60.50	78.800	32.96	15.652	6.55
10.30	144.646	60.50	80.852	33.82	13.600	5.69
11.00	144.646	60.50	82.636	34.56	11.817	4.94
11.30	144.646	60.50	84.185	35.21	10.267	4.29
12.00	144.646	60.50	85.532	35.77	8.920	3.73
12.30	144.646	60.50	86.703	36.26	7.750	3.24
13.00	144.646	60.50	87.720	36.69	6.733	2.82

Totale Activiteit op Tcal. : 239.098 mCi

Voorbeeld van een computeruitdraai

## APPENDIX B

*In deze appendix worden een aantal aspecten van het programma PROSIM, waarmee de verstrooiing van protonen in een gastarget berekend kan worden, toegelicht.*

### B.1 Inleiding

De verstrooiing van protonen in een gastarget wordt berekend door de banen van een (groot) aantal protonen te simuleren via een z.g.n. Monte-Carlo methode. Er wordt steeds gesproken van protonenverstrooiing in een gastarget, maar in principe kan het programma gebruikt worden voor willekeurige deeltjes in een willekeurig target. Voorwaarden voor de geldigheid van de gebruikte theorie zijn (MAR 67):

- Het materiaal waarin de protonen verstrooid worden moet opgedeeld worden in een aantal folies met een zodanige dikte dat de parameter  $B$  (die ook afhangt van de energie van het proton, zie hoofdstuk 4) niet kleiner is dan 6 en niet groter is dan 12 (voor een xenon-target komt dit, bij een protonen-energie van 26 MeV, overeen met een dikte tussen 0,002 en 20 kg/m<sup>2</sup>).
- De energie van de protonen moet tussen 10 en 1000 MeV liggen.
- De deeltjes moeten een intrinsieke spin  $S=\frac{1}{2}$  hebben. Voor deeltjes met een andere spin moet de verstrooiingstheorie aangepast worden (de invloed van de spin is echter vrij klein, zie MAR 67).

De listing van het programma en een overzicht van de in dit programma gebruikte variabelen is afgedrukt in § B.2.

In § B.3 wordt beschreven hoe, uitgaande van een uniform verdeelde variabele, een variabele verkregen wordt met een willekeurige kansverdelingsfunctie, zoals noodzakelijk is bij het bepalen van de startpositie van het proton en de verstrooiingshoeken.

In § B.4 wordt de bepaling van de stopping power van het (verrijkte) targetmateriaal nader toegelicht.

In § B.5 wordt ingegaan op de invloed van de grootte van het aantal protonen dat gelanceerd wordt ( $N_{\text{prot}}$ ) en het aantal folies waarin het target verdeeld wordt ( $N_{\text{folie}}$ ).

De invloed van de divergentie van de protonen-bundel en de dichtheidsreductie van het target-gas wordt beschreven in § B.6.

## B.2 Listing van het programma PROSIM en overzicht van de gebruikte variabelen

Op de volgende pagina's is een listing van het programma PROSIM en een overzicht van de in dit programma gebruikte variabelen afgedrukt. De listing is van commentaar voorzien. De D(debug)-regels in het programma zijn toegevoegd om tijdens de uitvoering van het programma de waarden van de belangrijkste variabelen te kunnen zien.

In het programma worden subroutines uit het 'programma' IO gebruikt. Deze subroutines zijn in de listing van commentaar voorzien. De listing van het 'programma' IO is ook afgedrukt in deze appendix.

Alle FORTRAN-statements en STANDAARD-FORTRAN subroutines zijn met hoofdletters geschreven, alle zelf-gedefiniëerde variabelen en subroutines met kleine letters (uitgezonderd enkele variabelen een l bevatten, om verwarring tussen l en 1 te voorkomen).



PROGRAM prosim

C  
C  
C  
C  
C

-----  
declaraties  
-----

```

COMMON /F/Afol,Agas,Booran,dikfol,diverg,drho,Ebegin,Eeind,Ef,
*   finaam,fostap,hoek,j,lengas,Nfolie,Npass,Nprot,opbr,
*   Rafkap,Rdiafr,rest,rho0,rhofol,rin,ruit,sigger,sigmaE,
*   sigmax,Zfol,Zgas,Zproj
REAL*4 A(7,501),Afol,Agas,aintE,aintEg,b,bb,betakw,Booran,
*   c,cb,cbf,cbg,cosa,cosb,cosfi,cXckw,cXckwf,cXckwg
*   d,dE,dedx(27),dEdxfo,dikfol,diverg,dL,drho,dx,dy,
*   E,Ebegin,Eeind,Ef(10),Egem,epskw,expon,
*   fl,f2,factor,fal,fa2,fb1,fb2,fi,fk,gamma,h,hulp,
*   intpol,kortas,L,lengas,Lvolgf,Moc2,
*   pi,psi,pvkw,R,Rafkap(10),Rbegin,Rdiafr,Rellip,
*   rho,rhodL,rhofol,rhogas,rhor,rho0,rs,
*   sigger,sigmaE,sigmax,sina,theta,
*   tweepi,ul,u2,verh,wb(58),worB,worBkw,x,x1,x2,xa,
*   Xckw,xworB,y,y1,y2,ya,Zfol,Zgas,Zproj
INTEGER*2 al,a2,b1,b2,folnr,fostap,hk,hoek(100),
*   i,int5bb,intE,intEEg,intEgm,j,k,kweek,m,
*   Nfolie,Nlast,Npass(10),Nprot,Nstart,opbr(10,100),
*   rani,ranj,rest,rin(100),rseg,ruit(100),ur
LOGICAL*1 fldedx(14),flnaam(14),flRafk(14),wacht

```

C  
C

```

DATA Moc2,pi,rs/938.259,3.1415926536,5E-4/,

```

```

*   wb/1.995,2.061,2.123,2.182,2.239,2.294,2.346,2.398,2.447,
*   2.496,2.543,2.589,2.634,2.678,2.721,2.763,2.804,2.844,
*   2.884,2.923,2.962,2.999,3.037,3.073,3.109,3.145,3.180,
*   3.215,3.249,3.283,3.316,3.349,3.382,3.414,3.446,3.477,
*   3.509,3.539,3.570,3.600,3.630,3.660,3.689,3.718,3.747,
*   3.776,3.804,3.832,3.860,3.888,3.915,3.943,3.970,3.996,
*   4.023,4.049,4.076,4.102/

```

C  
C  
C  
C  
C  
C  
C

-----  
wb(i) bevat de waarde van SQRT(B) voor b=(i+12)/5  
ofwel de bij b behorende waarde van SQRT(B) bevindt  
zich tussen wb(INT(5\*b)-12) en wb(INT(5\*b)-11)  
-----

-----  
zelf-gedefinieerde functies  
-----

```

intpol(x,x1,x2,y1,y2)=y1+(x-x1)*(y2-y1)/(x2-x1)
rhor(r1,rhol,drhol,sigerl,Rafkl)=rhol+drhol*(1.-EXP(-(r1/
*   Rafkl/sigerl)**2/2.))
TAN(x)=x+x**3/3.
ARCTAN(x)=x-x**3/3.
tweepi=2.*pi

```

C  
C  
C  
C  
C

-----  
invoergegevens inlezen  
-----

```

TYPE*, 'Zproj'

```

```
ACCEPT*,Zproj
TYPE*, 'Zfolie'
ACCEPT*,Zfol
TYPE*, 'Afolie'
ACCEPT*,Afol
TYPE*, 'dichtheid folie in kg/m3'
ACCEPT*,rhofol
TYPE*, 'dikte folie in um'
ACCEPT*,hulp
dikfol=hulp*1E-6
TYPE*, 'dE/dx van folie (MeV/(kg/m2))'
ACCEPT*,dEdxfo
TYPE*, 'Zgas'
ACCEPT*,Zgas
TYPE*, 'Agas'
ACCEPT*,Agas
TYPE*, ' gemiddelde dichtheid gas (kg/m3)'
ACCEPT*,rhogas
TYPE*, 'rhomax/rhomin'
ACCEPT*,verh
TYPE*, 'sigma/Rafkap'
ACCEPT*,sigger
expon=1.-EXP(-1./sigger/sigger/2.)
drho=0.
rho0=rhogas
IF(verh.NE. 1.) drho=rhogas/(expon/(verh-1.))+1.-2*sigger*sigger
* *expon)
IF(verh.NE. 1.) rho0=drho*expon/(verh-1.)
TYPE*, 'rho0,deltarho',rho0,drho
TYPE*, 'lengte gastarget (m)'
ACCEPT*,lengas
TYPE*, 'in hoeveel folies moet het gas verdeeld worden (10-voud)?'
ACCEPT*,Nfolie
TYPE*, 'hoeveel protonen moeten gelanceerd worden (max 32768) ?'
ACCEPT*,Nprot
TYPE*, 'beginenergie protonen in MeV'
ACCEPT*,Ebegin
TYPE*, 'sigma E (MeV)'
ACCEPT*,sigmaE
TYPE*, 'sigma x (m)'
ACCEPT*,sigmax
TYPE*, 'hoever ligt het convergentiepunt voor het diafragma (m)'
ACCEPT*,hulp
diverg=1./hulp
TYPE*, 'voer kweekgetal in'
ACCEPT*,kweek
TYPE*, 'voer filenaam voor dE/dx in'
TYPE*
CALL GTLIN(fldedx)
TYPE*, 'voer filenaam voor afkap-straal in'
TYPE*
CALL GTLIN(flRafk)
TYPE*, 'voer filenaam voor opslag in'
TYPE*
CALL GTLIN(flnaam)
```

```
C -----
C   initialisatie 1
C -----
C
C   rani=kweek
C   ranj=kweek-1
C
C   DO 3 m=1,100
C   DO 2 k=1,10
C   opbr(k,m)=0
2   CONTINUE
C   rin(m)=0
C   ruit(m)=0
C   hoek(m)=0
3   CONTINUE
C   Nstart=1
C
C -----
C   Uitzoeken of opslag-file bestaat en compleet is
C -----
C
C   OPEN (UNIT=2,NAME=flnaam,TYPE='OLD',ERR=1000)
C   als ERR optreedt, dan normaal beginnen (file bestaat nog niet)
100  READ (2,100,ERR=30004) Nprot,Nfolie,Nlast,Booran,rest,fostap,Zproj
C   FORMAT (3I6,F8.0,2I6,F5.1)
C   IF (Nlast.GE.Nprot) GOTO 30000!           END, file al klaar
C   CALL leesin!                             data uit deze file inlezen
C   CLOSE (UNIT=2,DISPOSE='SAVE',ERR=30004)
C   Nstart=Nlast!                             verder met afgebroken file
1000 TYPE 40,Nstart,Nprot,Nfolie
C
C -----
C   initialisatie 2
C -----
C   dedx,Rafkap,en B-array inlezen
C -----
C
C   OPEN (UNIT=2,NAME=fldedx,TYPE='OLD',ERR=30001)
C   DO 4 k=1,27
C   READ (2,*,ERR=30001) dedx(k)
4   CONTINUE
C   CLOSE (UNIT=2,DISPOSE='SAVE',ERR=30001)
C
C -----
C   dedx(i) bevat de waarde voor dE/dx (in MeV/(kg/m2)) bij
C   E = 1 MeV
C -----
C
C   OPEN (UNIT=2,NAME=f1Rafk,TYPE='OLD',ERR=30003)
C   READ (2,*,ERR=30003) Rdiafr
C   DO 5 k=1,10
C   READ (2,*,ERR=30003) Rafkap(k)
5   CONTINUE
C   CLOSE (UNIT=2,DISPOSE='SAVE',ERR=30003)
C
C -----
```

```
C      Rafkap(1) bevat de waarde van de afkap-stralen voor 10 folies
C      -----
C
C      OPEN (UNIT=2,NAME='DK:BDATA.DAT',TYPE='OLD',ERR=30008)
C      DO 9 m=1,7
C      READ (2,10,ERR=30009) (A(m,k),k=1,500)
C      A(m,501)=8.
9      CONTINUE
10     CLOSE (UNIT=2,ERR=30010)
C      FORMAT(10F7.5)
C      -----
C      file BDATA.DAT ingelezen in array A(7,501)
C      -----
C
C      cbf=273.*(Zfol+1.)*(Zfol**(1./3.))*Zproj*Zproj/Afol
C      cbg=273.*(Zgas+1.)*(Zgas**(1./3.))*Zproj*Zproj/Agas
C      cXckwf=.01569*Zfol*(Zfol+1.)*Zproj*Zproj/Afol
C      cXckwg=.01569*Zgas*(Zgas+1.)*Zproj*Zproj/Agas
C      fostap=(Nfolie-1)/10+1
C      IF (fostap .EQ. 0) fostap=1
C      rest=INT(FLOAT(fostap+1)/2.+1.)
C      IF (Nfolie .LT. 40) rest=0
C
C      -----
C      Egem in 10 opbrengst-folies berekenen (Ef(i))
C      -----
C
C      i=1
C      dE=dEdxfo*rhofol*dikfol!          metaalfolie
C      E=Ebegin-dE
C      dL=lengas/10.
6      i=i+1
C      rhodL=rhor(.003,rho0,drho,sigger,Rafkap(1))*dL! neem rho op 3 mm
C                                                    ! afstand van de as
C
C      IF (i.EQ.2) rhodL=rhodL/2.
C      intE=INT(E)
C      aintE=AINT(E)
C      dE=intpol(E,aintE,aintE+1.,dedx(intE),dedx(intE+1))*rhodL
C      Egem=E-dE/2.! Egem grof benaderd
C      intEgm=INT(Egem)
C      aintEg=AINT(Egem)
C      dE=intpol(Egem,aintEg,aintEg+1.,dedx(intEgm),
C      *      dedx(intEgm+1))*rhodL! dE bij Egem in gasfolie
C      Ef(i-1)=E-dE/2.!          gemidd. energie in folie i
C      E=E-dE!          energie aan eind van folie i in MeV
C      IF (i.LT.11) GOTO 6
C
C      -----
C      hoofdlus (volgend proton)
C      -----
C
C      DO 32000 j=Nstart,Nprot,1
C
C      TYPE*
D      TYPE*, '-- proton nr. --',j
```

```
D      TYPE*
      IF (MOD(j,5000) .NE. 0) GOTO 12
      DO 11 k=1,kweek,10! randomgenerator een trap geven
      hulp=RAN(rani,ranj)
11     CONTINUE
      C
      C
      C      -----
      C      i,foinr,x,y,gamma,psi,E initialiseren
      C      -----
12     i=1! eerste folie (metaalfolie)
      folnr=1
      u1=RAN(rani,ranj)
      u2=RAN(rani,ranj)
      Rbegin=SQRT(-2.*ALOG(u1))*sigmax
      x=Rbegin*COS(tweepi*u2)
      y=Rbegin*SIN(tweepi*u2)! x en y Gaussisch verdeeld
      IF (Rbegin.GT.Rdiafr) GOTO 32000! proton buiten diafragma
      ur=INT(Rbegin/rs)+1! Rin turven
      IF (ur.LE.100) rin(ur)=rin(ur)+1
      gamma=x*diverg! divergentie is evenredig met afstand tot as
      psi=y*diverg
      C
      D      TYPE 13,u1,u2,sigmax,sigger,sigmaE
      u1=RAN(rani,ranj)
      u2=RAN(rani,ranj)
      E=Ebegin+SQRT(-2.*ALOG(u1))*COS(tweepi*u2)*sigmaE
      C
      D      TYPE 14,x,y,Rbegin,ur,rin(ur),gamma,psi
      C
      C      -----
      C      metaalfolie-parameters invoeren
      C      -----
      C
      L=dikfol
      cXckw=cXckwf
      cb=cbf
      C
      C
      C      -----
      C      Egem en E aan eind van folie i berekenen
      C      -----
31000  dL=L*SQRT(1.+(TAN(gamma))**2+(TAN(psi))**2)! doorlopen dikte
      IF (i.NE.1) GOTO 15
      rhodL=rhofol*dL
      dE=dEdxfo*rhodL
      GOTO 16! i=1 -> metaalfolie
15     rhodL=rhogas*dL
      IF (verh.EQ. 1.) GOTO 150
      rhodL=rhor(R,rho0,drho,sigger,Rafkap(folnr))*dL
150    intE=INT(E)
      aintE=AINT(E)
      dE=intpol(E,aintE,aintE+1.,dedx(intE),dedx(intE+1))*rhodL
      Egem=E-dE/2.! Egem grof benaderd
      intEgm=INT(Egem)
      aintEg=AINT(Egem)
```

```
dE=intpol(Egem,aintEg,aintEg+1.,dedx(intEgm),
*      dedx(intEgm+1))*rhodL! dE bij Egem in gasfolie
16 Egem=E-dE/2.!   gemidd. energie in folie i
E=E-dE!         energie aan eind van folie i in MeV
intE=INT(E)!    i=1 -> metaalfolie, i=2..Nfolie-2 -> gasfolie
aintE=AINT(E)
D  TYPE 17,dL,rhodL,E,dE,Egem
IF (E .LE. 0.) GOTO 34! proton gestopt

C
C
C  -----
C  theta en fi tgv het doorlopen van folie i bepalen
C  theta=Xc*worB*factor, waarbij factor een stoch. variabele is
C  fi=H(0,2*pi)
C  -----

betakw=1.-1./(1.+Egem/Moc2)**2
pvkw=(Egem*Egem+2.*Egem*Moc2)*betakw
Xckw=cXckw*rhodL/pvkw
bb=ALOG(cb*rhodL/betakw)-.1544
int5bb=INT(bb*5)
worB=intpol(bb,int5bb/5.,int5bb/5.+2,wb(int5bb-12),wb(int5bb-11))
worBkw=worB*worB
bl=INT(worBkw)-5
IF ((worBkw.GT. 6.) .AND. (worBkw.LT. 12.)) GOTO 18
Booran=Booran+1.! B out of Range -> extrapolatie ipv interpolatie
IF (worBkw.LT. 6.) bl=1
IF (worBkw.GT. 12.) bl=6
18 ul=RAN(rani,ranj)
al=INT(500.*ul)+1
a2=a1+1
b2=b1+1
fal=FLOAT(a1)
fa2=FLOAT(a2)
fbl=FLOAT(b1)
fb2=FLOAT(b2)
f1=intpol(500.*ul+1.,fal,fa2,A(b1,a1),A(b1,a2))
f2=intpol(500.*ul+1.,fal,fa2,A(b2,a1),A(b2,a2))
factor=intpol(worBkw-5.,fbl,fb2,f1,f2)

C
theta=SQRT(Xckw)*worB*factor
fi=RAN(rani,ranj)*tweepi
cosfi=COS(fi)

C
D  TYPE 19,a1,a2,b1,b2
D  TYPE 20,ul,fal,fa2,worBkw,fbl,fb2
D  TYPE 21,A(b1,a1),A(b1,a2),A(b2,a1),A(b2,a2),f1,f2
D  TYPE 22,factor,theta/factor
D  TYPE 23,Egem,dL,rhodL,cXckw
D  TYPE 24,betakw,pvkw,Xckw,bb
D  TYPE 25,E,dE/rhodL,dE
C
C  -----
C  over naar volgend folie (gas-folie)
C  -----

i=i+1
D  TYPE 26,i
```

L=lengas/Nfolie! dikte folie  
Lvolgf=L! afstand tot volgend folie  
IF (i.EQ.2 .OR. i.EQ.Nfolie+2) Lvolgf=L/2! eerste en laatste  
! folie halve afstand

C  
C  
C  
C  
C

-----  
x,y,gamma,psi berekenen ter plaatse van volgend folie  
-----

h=Lvolgf\*TAN(gamma)  
b=Lvolgf\*TAN(psi)  
IF (ABS(h).LT.1E-18) h=1E-18! dan geen division by zero  
c=SQRT(b\*b+h\*h)  
d=SQRT(c\*c+Lvolgf\*Lvolgf)  
cosb=Lvolgf/d  
sina=b/c  
cosa=h/c  
kortas=d\*TAN(theta)  
epskw =1.-cosb\*cosb  
Rellip=kortas/SQRT(1.-cosfi\*cosfi\*epskw)  
xa=Rellip\*cosfi  
ya=Rellip\*SIN(fi)  
dx=xa\*cosa-ya\*sina  
dy=xa\*sina+ya\*cosa  
x=x+h+dx  
y=y+b+dy  
R=SQRT(x\*x+y\*y)  
gamma=ARCTAN((h+dx)/Lvolgf)  
psi=ARCTAN((b+dy)/Lvolgf)  
D TYPE 27,kortas,epskw  
D TYPE 28,h,b,c,d,Lvolgf  
D TYPE 29,sina,cosa,cosb,Rellip  
D TYPE 30,dx,dy,xa,ya  
D TYPE 31,gamma,psi,theta,fi  
D TYPE 32,x,y  
D TYPE 36,E,i,fostap,rest  
IF (MOD(i,fostap) .NE. rest) GOTO 33!geen 'opbrengst-folie'  
IF (i .EQ. Nfolie+2) GOTO 33! geen 'opbrengst-folie'  
IF (R .GT. Rafkap(folnr)) GOTO 34! proton buiten target

D  
D  
D  
D  
D  
D  
D

-----  
opbrengst-folie : r en Npass turven  
-----

C  
C  
C  
C  
C

rseg=INT(R/rs)+1  
IF (rseg.LE.100) opbr(folnr,rseg)=opbr(folnr,rseg)+1  
Npass(folnr)=Npass(folnr)+1! aantal protonen door opbr.folie  
D TYPE 37,folnr,rseg,opbr(folnr,rseg)  
folnr=folnr+1

D

C  
C  
C

-----  
cXckw=cXckwg! gas-parameters invullen  
cb=cbg  
op=op+1! weer een 'proton-folie' gehad

33

C

IF (i.LT.Nfolie+2) GOTO 31000! volgend 'gas-folie'

C  
C  
C  
C  
C

-----  
turf R en hoek bij laatste folie  
-----

```
Eeind=E
ur=INT(R/rs)+1! Ruit turven
IF (ur.LE.100) ruit(ur)=ruit(ur)+1
hk=INT(SQRT(gamma*gamma+psi*psi)*100.)+1
IF (hk.LE.100) hoek(hk)=hoek(hk)+1
34 CONTINUE! doet niets, slechts label
D TYPE 38,Eeind,SQRT(gamma*gamma+psi*psi),R,hk,ur
D TYPE 39,Npass
IF (op .LT. 10000) GOTO 32000
CALL opslag! tussentijds opslaan na elke 10000 'proton-folies'
op=0
32000 CONTINUE! volgend proton (einde hoofdlus)
C
CALL opslag
GOTO 30000
13 FORMAT(' ul,u2,sigx,s/R,sigE ',5E12.5)
14 FORMAT(' x,y,Rbegin,ur,rin,gamma,psi ',3E12.5,2I4,2E12.5)
17 FORMAT(' dL,rhodL,E,dE,Egem ',5E12.5)
19 FORMAT(' al,a2,bl,b2 ',6I12)
20 FORMAT(' ul,fal,fa2,B,fb1,fb2',6E12.5)
21 FORMAT(' A11,A12,A21,A22,f1,f2',E11.5,5E12.5)
22 FORMAT(' factor,thetanor ',2E12.4)
23 FORMAT(' Egem,dL,rhodL,cXckw ',4E12.4)
24 FORMAT(' betakw,pvkw,Xckw,bb ',4E12.4)
25 FORMAT(' E,intpolE,dE ',3E12.4)
26 FORMAT(' ----- i= ',I7)
27 FORMAT(' kortas,epskw ',2E12.4)
28 FORMAT(' h,b,c,d,Lvolgf ',5E12.4)
29 FORMAT(' sina,cosa,cosb,Relli',4E12.4)
30 FORMAT(' dx,dy,xa,ya ',4E12.4)
31 FORMAT(' gamma,psi,theta,fi ',4E12.4)
32 FORMAT(' (x,y)',2E12.4)
36 FORMAT(' E,i,fostap,rest ',E12.5,3I12)
37 FORMAT(' folnr,rseg,opbr ',3I12)
38 FORMAT(' Eeind,hoek,R,hk,ur ',3E12.5,2I12)
39 FORMAT(' Npass ',10I6)
40 FORMAT(' Nstart,Nprot,Nfolie ',3I12)
30001 TYPE*, 'FOUT in dedx-file !'
30003 TYPE*, 'FOUT in Rafkap-file !'
30004 TYPE*, 'FOUT in opslag-file !'
30008 TYPE*, 'FOUT in OPEN !'
30009 TYPE*, 'FOUT in READ !'
30010 TYPE*, 'FOUT in CLOSE !'
30000 END
```



---- io (data in- en output) ----

SUBROUTINE opslag! opslaan op disk

C

```
COMMON /F/Afol,Agas,Booran,dikfol,diverg,drho,Ebegin,Eeind,Ef,  
* flnaam,fostap,hoek,j,lengas,Nfolie,Npass,Nprot,opbr,  
* Rafkap,Rdiafr,rest,rho0,rhofol,rin,ruit,sigger,sigmaE,  
* sigmaE,Zfol,Zgas,Zproj  
REAL*4 Afol,Agas,Booran,dikfol,diverg,drho,Ebegin,Eeind,Ef(10),  
* lengas,Rafkap(10),Rdiafr,rhofol,rho0,  
* sigmaE,sigmaE,sigger,versie,Zfol,Zgas,Zproj  
INTEGER*2 fostap,hoek(100),j,jj,k,m,Npass(10),Nprot,Nfolie,  
* opbr(10,100),rest,rin(100),ruit(100)  
LOGICAL*1 flnaam(14)
```

versie=6.1

jj=j

IF (jj.GT.Nprot) jj=jj-1

TYPE\*,jj

TYPE 50, (flnaam(k),k=1,14)

50

FORMAT(' filenaam: ',14A1)

OPEN (UNIT=2,NAME=flnaam,TYPE='NEW',ERR=1000)

WRITE (7,100) Nprot,Nfolie,jj,Booran,rest,fostap

WRITE (2,100,ERR=1001) Nprot,Nfolie,jj,Booran,rest,fostap,Zproj

WRITE (2,99) versie

WRITE (2,101) Zfol,Afol,rhofol,dikfol

WRITE (2,101) Zgas,Agas,rho0,lengas

WRITE (2,102) sigmaE,sigger,sigmaE,

\* Rdiafr,diverg,Ebegin,Eeind,drho

WRITE (2,103) (Rafkap(k),k=1,10)! alles in mm wegschrijven

WRITE (2,103) (Ef(k),k=1,10)

WRITE (2,105) (Npass(k),k=1,10)

DO 200 k=1,100

CALL IWRITE(2,ruit(k))

200

CONTINUE

DO 300 k=1,100

CALL IWRITE(2,hoek(k))

300

CONTINUE

DO 400 k=1,10

DO 500 m=1,100

CALL IWRITE(2,opbr(k,m))

500

CONTINUE

400

CONTINUE

DO 600 k=1,100

CALL IWRITE(2,rin(k))

600

CONTINUE

99

FORMAT (X,F3.1)

100

FORMAT (X,3I6,F8.0,2I6,F5.1)

101

FORMAT (X,4E12.5)

102

FORMAT (X,8E12.5)

103

FORMAT (X,10E12.5)

105

FORMAT (X,10I5)

106

FORMAT (X,10I4)

CLOSE (UNIT=2,DISPOSE='SAVE',ERR=1002)

RETURN

1000

TYPE\*, 'FOUT in open'

1001

TYPE\*, 'FOUT in write'

1002

TYPE\*, 'FOUT in close'

END

SUBROUTINE leesin

C

```
COMMON /F/Afol,Agas,Booran,dikfol,diverg,drho,Ebegin,Eeind,Ef,  
* flnaam,fostap,hoek,j,lengas,Nfolie,Npass,Nprot,opbr,  
* Rafkap,Rdiafr,rest,rho0,rhofol,rin,ruit,sigger,sigmaE,  
* sigmax,Zfol,Zgas,Zproj  
REAL*4 Afol,Agas,Booran,dikfol,diverg,drho,Ebegin,Eeind,Ef(10),  
* lengas,Rafkap(10),Rdiafr,rhofol,rho0,  
* sigmaE,sigmax,sigger,versie,Zfol,Zgas,Zproj  
INTEGER*2 fostap,hoek(100),j,jj,k,m,Npass(10),Nprot,Nfolie,  
* opbr(10,100),rest,rin(100),ruit(100)  
LOGICAL*1 flnaam(14)
```

C

```
READ (2,99) versie  
IF (versie .NE. 6.1) GOTO 1000  
READ (2,101) Zfol,Afol,rhofol,dikfol  
READ (2,101) Zgas,Agas,rho0,lengas  
READ (2,102) sigmax,sigger,sigmaE,  
* Rdiafr,diverg,Ebegin,Eeind,drho  
READ (2,103) (Rafkap(k),k=1,10)! alle lengtematen in mm opgeslagen  
READ (2,103) (Ef(k),k=1,10)  
READ (2,105) (Npass(k),k=1,10)  
DO 200 k=1,100  
200 READ (2,*) ruit(k)  
CONTINUE  
DO 300 k=1,100  
300 READ (2,*) hoek(k)  
CONTINUE  
DO 400 k=1,10  
DO 500 m=1,100  
500 READ (2,*) opbr(k,m)  
CONTINUE  
400 CONTINUE  
DO 600 k=1,100  
600 READ (2,*) rin(k)  
CONTINUE  
99 FORMAT (F3.1)  
101 FORMAT (4E12.5)  
102 FORMAT (8E12.5)  
103 FORMAT (10E12.5)  
105 FORMAT (10I5)  
106 FORMAT (10I4)  
RETURN  
1000 TYPE*, ' VERKEERDE VERSIE, geen 6.1 maar',versie  
CALL EXIT  
END
```

```

SUBROUTINE IWRITE(device,getal)
-----
C      schrijf getal in het kleinst mogelijke format naar device
C      -----
      INTEGER device,getal
      IF (getal.GE.10) GOTO 10
      WRITE (device,1) getal
      RETURN
10     IF (getal.GE.100) GOTO 100
      WRITE (device,2) getal
      RETURN
100    IF (getal.GE.1000) GOTO 1000
      WRITE (device,3) getal
      RETURN
1000   IF (getal.GE.10000) GOTO 10000
      WRITE (device,4) getal
      RETURN
10000  WRITE (device,5) getal
      RETURN
1      FORMAT (X,I1)
2      FORMAT (X,I2)
3      FORMAT (X,I3)
4      FORMAT (X,I4)
5      FORMAT (X,I5)
      END
```

Overzicht van de in het programma PROSIM gebruikte variabelen

A	array dat voor B=6,7,8,9,10,11 en 12 en voor x=0(0,002)1 de waarden van $F^{-1}$ bevat
Afol	massagetal van folie (kg/kmol)
Agas	massagetal van target (kg/kmol)
aintE	AINTE (MeV)
aintEg	AINTEg (MeV)
a1,a2	hulpvariabelen om $\theta$ te bepalen
b	b, zie fig 4.4 (m)
bb	b, benodigd op B te bepalen
betakw	$\beta^2$
Booran	B out of range (aantal malen de B buiten het interval [6,12] ligt)
b1,b2	hulpvariabelen om $\theta$ te bepalen
c	c, zie fig 4.4 (m)
cb	$273(Z+1)Z^4Z_{proj}^2/A$ (A=Agas of Afol)
cbf	cb voor folie
cbg	cb voor target
cosa	$\cos\alpha$ , zie fig. 4.4
cosb	$\cos\beta$ , zie fig. 4.4
cosfi	$\cos\phi$ , zie fig. 4.6
cXckw	$0,01569Z(Z+1)Z_{proj}^2/A$ (A=Agas of Afol)
cXckwf	cXckw voor folie
cXckwg	cXckw voor target
d	d, zie fig. 4.4 (m)
dE	energie-afname in 1 folie (MeV)
dEdx(i)	array-element dat de stopping power van het target-gas bevat bij i Mev ( $\text{MeV}/\text{kgm}^{-2}$ )
dEdxfo	stopping power van het folie bij de begin-energie van het proton ( $\text{MeV}/\text{kgm}^{-2}$ )
dikfol	dikte van het folie (m)
diverg	1/(afstand van convergentiepunt tot diafragma) ( $\text{m}^{-1}$ )
dL	doorlopen dikte in folie (m)
drho	dichtheidsafhankelijkheid van het target

dx            toename van proton-positie in x-richting t.g.v.  
verstrooiing (m)

dy            toename van proton-positie in y-richting t.g.v.  
verstrooiing (m)

E             energie van proton (MeV)

Ebegin       begin-energie van proton (MeV)

Eeind        eind-energie van proton (MeV)

Ef(i)        array-element dat energie van het proton in  
'opbrengs-folie' i bevat (MeV)

Egem        gemiddelde energie van proton in folie (MeV)

epskw        excentriciteit  $\varepsilon^2$

expon        hulpvariabele bij dichtheidsreductie

factor       stochastische hulpvariabele om  $\theta$  te bereken

fa1,fa1      FLOAT(a1), FLOAT(a2)

fb1,fb1      FLOAT(b1), FLOAT(b2)

fi            $\phi$ , zie fig. 4.3 en 4.6

fk            FLOAT(k)

fldEdx      array dat filenaam voor stopping power bevat

flnaam      array dat filenaam voor opslag-file bevat

flRafk      array dat filenaam voor file met waarden voor  
Rdiafr en Rafkap(i) bevat

folnr        'opbrengst-folie'-nummer

fostap      aantal folies tussen 2 'opbrengst-folies'

f1,f2        hulpvariabelen om  $\theta$  te berekenen

gamma        $\gamma$ , zie fig. 4.4

h            h, zie fig 4.4 (m)

hk           index van array hoek

hoek(hk)    array-element dat het aantal protonen bevat dat met  
de as van het target een hoek met divergentie  
tussen  $(hk-1)*10$  en  $hk*10$  mrad

hulp        hulpvariabele

i            folie-nr

intE        INT(E) (MeV)

intEgm      INT(Egem) (MeV)

intpol      functie die de geïnterpoleerde waarde van  $f(x)$   
berekend als  $x$ ,  $x_1$ ,  $y_1=f(x_1)$ ,  $x_2$  en  $y_2=f(x_2)$

	gegeven zijn.
int5bb	INT(5*b)
j	proton-nummer
k	hulpvariabele
kortas	korte as van ellips, zie fig. 4.6 (m)
kweek	kweekgetal voor random-generator
L	dikte van een folie (m)
lengas	lengte van het target (m)
lvolgf	afstand tot volgende folie (m)
m	algemene index
Moc2	rustenergie van proton (MeV)
Nfolie	aantal folies waarin het target verdeeld is
Nlast	hulpvariabele bij subroutine 'leesin'
Npass(i)	array-element dat het aantal protonen bevat dat het $i^e$ 'opbrengst-folie' passeert
Nprot	aantal protonen dat gelanceerd moet worden
Nstart	hulpvariabele bij subroutine 'leesin'
op	hulpvariabele om te bepalen of tussentijds opgeslagen moet worden
opbr(f,r)	array-element dat het aantal protonen bevat dat in folie f tussen $(r-1)*rs$ en $r*rs$ gepasseert is.
pi	$\pi$
psi	$Y$ , zie fig. 4.4 (radiaal)
pvkW	$(pv)^2$ ( $kg^2m^4s^{-4}$ )
r	afstand van proton tot de as van het target tijdens het doorlopen van het target (m)
Rafkap(i)	array-element dat de straal van het target ter plaatse van het $i^e$ 'opbrengst-folie' bevat
rani,ranj	hulpvariabelen voor random-generator
Rbegin	afstand van proton tot as van het target ter plaatse van het targetfolie (m)
Rdiafr	straal van het diafragma (m)
Rellip	zie fig. 4.6
rest	aantal folies voor het $1^e$ 'opbrengst-folie' + 1
rho	dichtheid ( $kgm^{-3}$ )
rhofol	dichtheid van het folie ( $kgm^{-3}$ )

rhogas	gemiddelde dichtheid van het target ( $\text{kgm}^{-3}$ )
rhoR	dichtheidsafhankelijkheid
rho0	dichtheid op as van het target ( $\text{kgm}^{-3}$ )
Rin	array-element dat het aantal protonen bevat dat tussen $(r-1)*rs$ en $r*rs$ het target binnentreedt
rs	stapgrootte bij turven van R in array opbr(folnr, rseg) (m)
rseg	index in array opbr(folnr,rseg), rin(rseg) en ruit(rseg)
Ruit	array-element dat het aantal protonen bevat dat tussen $(r-1)*rs$ en $r*rs$ het target aan het einde verlaat
sigger	$\sigma(\text{bundel})/\text{Rafkap}$ , nodig bij dichtheidsreductie
sigmaE	standaarddeviatie in begin-energie (MeV)
sigmax	standaarddeviatie in x- en y-positie van invallende bundel (m)
sina	$\sin\alpha$ , zie fig. 4.4
theta	verstrooiingshoek $\theta$ (radiaal)
tweepi	$2\pi$
ur	index
u1,u2	hulpvariabelen bij berekenen van random-getal
verh	verhouding tussen maximale en minimale dichtheid
wacht	hulpvariabele
wb(i)	array-element dat de waarde van $\sqrt{B}$ voor $b=(i+12)/5$ bevat
worB	$\sqrt{B}$
worBkw	B
x	x-positie van proton tijdens het doorlopen van het target (m)
xa	$x'$ , zie fig. 4.6 (m)
Xckw	$X_c^2$
xwworB	array-element dat de waarde van $X_c\sqrt{B}$ voor $b=(i+12)/5$ bevat
x1,x2	hulpvariabelen voor interpolatie
y	y-positie van proton tijdens het doorlopen van het target (m)

ya            y', zie fig. 4.6 (m)  
y1,y2        hulpvariabelen voor interpolatie  
Zfol         ladingsgetal van folie  
Zgas         ladingsgetal van target-materiaal  
Zproj        ladingsgetal van proton

Als input-parameters worden, tenzij anders vermeld, de volgende gegevens ingevoerd:

Zproj = 1  
Zfol = 42  
Afol = 95,94  
rhofo1 = 10200 (kg/m<sup>3</sup>)  
dikfo1 = 25 (μm)  
dEdxfo = 1,3 (MeV/kgm<sup>-1</sup>)  
Zgas = 54 (36 bij Kr)  
Agas = 124 (83,8 bij Kr)  
rhogas = 5,56\*druk(in bar) (kg/m<sup>3</sup>) (3,733\*druk bij Kr)  
verh = 1  
sigger = 1 (heeft geen invloed als verh=1)  
lengas = ... (m)  
Nfolie = 10  
Nprot = 32000  
Ebegin = 26 (MeV)  
sigmaE = 0 (MeV)  
sigmax = 0,0031 (m)  
diver = 10<sup>-6</sup>  
kweek = ...  
fldEdx = DEDXXE.124 (zie tabel B.2) (DEXX.KR bij Kr)  
flRafk = RAFKAP.GRO : Rdiafr=0,0065 (m), Rafkap(i)=0,05 (m)  
flnaam = ...



### B.3 Berekening van de stochastische variabelen $x, y, \theta$ en $\phi$

#### B.3.1 Transformatie van stochastische variabelen

In deze paragraaf wordt beschreven hoe een stochastische variabele met een uniforme kansverdelingsfunctie op het interval  $[0,1]$  getransformeerd kan worden naar een stochastische variabele met een willekeurige kansverdelingsfunctie (zie ook MIH 75).

Stel de stochastische variabele  $\underline{x}$  heeft verdelingsfunctie  $F_{\underline{x}}(x) (= P(\underline{x} < x))$ .  $F_{\underline{x}}(x)$  is een monotoon niet-dalende functie met functiewaarden tussen 0 en 1. Als  $F_{\underline{x}}(x)$  monotoon stijgend is, dan bestaan  $F_{\underline{x}}^{-1}(x)$ .

Definieer de stochastische variabele  $\underline{u}$  door  $\underline{u} = F_{\underline{x}}(x)$ ,  $\underline{u} \in [0,1]$ . Dan geldt:  $\underline{x} = F_{\underline{x}}^{-1}(u)$ .

$\underline{u}$  heeft als verdelingsfunctie  $F_{\underline{u}}(u)$ .

$$F_{\underline{u}}(u) = P(\underline{u} < u) = P(\underline{x} < F_{\underline{x}}^{-1}(u)) = F_{\underline{x}}(F_{\underline{x}}^{-1}(u)) = u$$

dus  $F_{\underline{u}}(u) = u$ , ofwel  $\underline{u}$  is uniform verdeeld ( $f_{\underline{u}}(u) = dF_{\underline{u}}(u)/du = 1$ ).

Om nu een stochastische variabele  $\underline{x}$ , met kansverdelingsfunctie  $f_{\underline{x}}(x)$  te genereren kunnen we uitgaan van een uniform verdeelde stochastische variabele  $\underline{u}$  en deze transformeren volgens  $\underline{x} = F_{\underline{x}}^{-1}(u)$ . Dan heeft  $\underline{x}$  als verdelingsfunctie  $F_{\underline{x}}(x)$ .

#### B.3.2 Transformatie naar normaal verdeelde stochastische variabelen

Het ingangprofiel van de protonenbundel heeft een Gaussisch profiel. De startpositie  $(x, y)$  van een proton wordt random gekozen. Hieronder wordt beschreven hoe  $x$  en  $y$  gekozen worden.

Stel  $x$  en  $y$  zijn onderling onafhankelijk en normaal verdeeld ( $N(0, d^2)$ ). Dan geldt:

$$f(x, y) = (1/2\pi d^2) \cdot \exp\{-(x^2 + y^2)/2d^2\}$$

Stel  $x = R \cos \theta$ ,  $y = R \sin \theta$ , dan geldt voor de kansverdelings-

functie  $g(r, \theta)$ :

$$g(r, \theta) = f(R \cos \theta, R \sin \theta) \cdot \begin{vmatrix} \partial x / \partial r & \partial x / \partial \theta \\ \partial y / \partial r & \partial y / \partial \theta \end{vmatrix}$$

$$\rightarrow g(r, \theta) = (1/2\pi\sigma^2) r \cdot \exp(-r^2/2\sigma^2), \quad r > 0, \quad 0 < \theta < 2\pi$$

De marginale kansverdelingsfuncties worden gegeven door:

$$h_\theta = \int g(r, \theta) dr = 1/2\pi, \quad 0 < \theta < 2\pi, \quad \text{en}$$

$$h_r = \int g(r, \theta) d\theta = (1/\sigma^2) r \cdot \exp(-r^2/2\sigma^2), \quad r > 0$$

$r$  en  $\theta$  zijn onderling onafhankelijk omdat  $g(r, \theta) = g_r(r) \cdot g_\theta(\theta)$ .  
Voor  $\sigma^2 = 1$  wordt de verdelingsfunctie  $H(r)$  gegeven door:

$$H_r(r) = 1 - \exp(-r^2/2).$$

De inverse functie  $H_r^{-1}(u)$  wordt gegeven door:

$$H_r^{-1}(u) = \sqrt{-2 \ln(1-u)},$$

immers  $1 - \exp(-(\sqrt{-2 \ln(1-u)})^2/2) = u$  ( $H_r(H_r^{-1}(u)) = u$ ).

Als we nu uitgaan van de uniform verdeelde stochastische variabelen  $u$ , en stellen  $r = H_r^{-1}(u)$ , dan heeft  $r$  als verdelingsfunctie  $H_r(r)$ .

Om nu standaard normaal verdeelde paren stochastische variabelen  $x$  en  $y$  te krijgen moeten we i.p.v.  $r = \sqrt{-2 \ln(1-u)}$ ,  $x = \sqrt{-2 \ln(1-u)} \cdot \cos(2\pi v)$  en  $y = \sqrt{-2 \ln(1-u)} \cdot \sin(2\pi v)$  nemen, waarbij  $v$  evenals  $u$  uniform verdeeld is.  $v$  neemt de rol van  $\theta$  over, die uniform verdeeld is op  $[0, 2\pi]$ .

Omdat  $1-u$  dezelfde verdeling heeft als  $u$  mogen we ook  $\sqrt{-2 \ln u}$  gebruiken.

Om van een standaard normaal verdeelde stochastische variabele  $x$  over te gaan naar een normaal verdeelde variabele  $z$  met  $Ez = \mu$  en  $\text{var}(z) = \sigma^2$  moeten we  $z = \sigma x + \mu$  kiezen.

### B.3.3 Bepalen van de verstrooihoek $\theta$ en $\phi$

In deze paragraaf wordt beschreven hoe een stochastische variabele  $x$ , die verdeeld is volgens kansverdelingsfunctie  $f(x)$  (zie § 4.2), bepaald wordt. Uitgaande van de stochas-

tische variabele  $x$  kan de verstrooihoek  $\theta$  berekend worden.

M.b.v. § B.3.2 wordt het probleem van het bepalen van  $x$  gereduceerd tot het bepalen van  $F_x^{-1}(u)$ .

Om  $F_x^{-1}(u)$  te bepalen is een computerprogramma 'INTEGR' geschreven (zie § B.3.4) :

Uitgaande van van de kansverdelingsfunctie  $f(x)$  wordt eerst de verdelingsfunctie  $F_x(x)$  berekend.

Er geldt  $F_\theta(\theta) = \int_0^\theta f(\theta') \sin(\theta') d\theta'$ . Voor kleine waarden van  $\theta$  mag dit vereenvoudigd worden tot  $F_\theta(\theta) = \int_0^\theta f(\theta') \theta' d\theta'$ .

Met  $x' = \theta' / X_c \sqrt{B}$  en  $x = \theta / X_c \sqrt{B}$  kan dit geschreven worden als  $F_x(x) = \int_0^x f(x') x' dx'$

Omdat  $f(x)$ , de kansverdelingsfunctie voor de genormeerde verstrooiingshoek, verwaarloosbaar is voor waarden van  $x > 8$  kan  $f(x)$  genormeerd worden door  $\int_0^8 f(x') x' dx'$  gelijk aan 1 te stellen.

Als  $F_x(x)$  berekend is wordt m.b.v. een iteratiemethode de inverse van  $F_x(x)$ ,  $F_x^{-1}(u)$  bepaald.

Omdat bij elke waarde van  $B$  een andere kansverdelingsfunctie bestaat (zie fig. 4.1 en tabel B.1) is voor een aantal waarden van  $B$  (6,7,8,9,10,11 en 12)  $F_x^{-1}(u)$  berekend. Het array  $A$  bevat voor deze 7 waarden van  $B$  de waarden van  $F_x^{-1}(u)$  voor  $u=0(0,002)1$ : het arrayelement  $A(i,j+1)$  bevat de waarde van  $F_x^{-1}(0,002j)$  bij  $B=i+5$ .

Om de verstrooihoek  $\theta$  te berekenen wordt eerst  $b$  bepaald. Het array  $wb(i)$  bevat de waarde van  $\sqrt{B}$  voor  $b=(i+12)/5$ , ofwel de bij  $b$  behorende waarde van  $\sqrt{B}$  bevindt zich tussen  $wb(\text{INT}(5b)-12)$  en  $wb(\text{INT}(5b)-11)$ . Vervolgens wordt random een getal  $u1$  gekozen, uniform verdeeld op  $[0,1]$ . De variabele  $x$ , met kansverdelingsfunctie  $f(x)$  wordt nu opgezocht in het array  $A$ :  $x=A(B-5,500.u1+1)$ .

Omdat  $B-5$  en  $500.u1+1$  geen integers zijn moet geïnterpoleerd worden. We definiëren  $b1=\text{INT}(B-5)$ ,  $a1=\text{INT}(500.u1+1)$ ,  $b2=b1+1$  en  $a2=a1+1$ . Eerst wordt lineair geïnterpoleerd tussen  $A(b1,a1)$  en  $A(b1,a2) \rightarrow f1$ , en tussen  $A(b2,a1)$  en  $A(b2,a2) \rightarrow f2$ , en vervolgens tussen  $f1$  en  $f2 \rightarrow x$ .

B	6	7	8	9	10	11	12
x							
0.0	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
0.2	.74850	.95058	.95208	.95321	.95409	.95473	.95537
0.4	.81017	.81821	.82232	.82616	.82916	.83134	.83351
0.6	.62535	.63731	.64601	.65259	.65772	.66146	.66520
0.8	.43939	.45363	.46402	.47192	.47811	.48264	.48716
1.0	.28491	.29793	.30752	.31486	.32063	.32488	.32913
1.2	.17437	.18337	.19077	.19616	.20045	.20363	.20681
1.4	.10393	.10911	.11304	.11612	.11859	.12045	.12231
1.6	.062258	.064103	.065557	.066729	.067692	.068435	.069178
1.8	.038158	.038067	.038046	.038059	.038088	.038127	.038165
2.0	.023871	.023029	.022403	.021920	.021538	.021255	.020972
2.2	.015071	.014158	.013446	.012878	.012413	.012058	.011702
2.4	.009557	.008826	.008237	.007754	.007353	.007040	.006726
2.6	.006465	.005894	.005425	.005038	.004709	.004451	.004193
2.8	.004382	.003960	.003611	.003323	.003077	.002884	.002690
3.0	.003096	.002772	.002508	.002291	.002106	.001961	.001816
3.2	.002271	.002016	.001810	.001644	.001502	.001393	.001283
3.4	.001714	.001510	.001348	.001219	.001108	.001024	.000940
3.6	.001320	.001156	.001027	.000926	.000839	.000774	.000708
3.8	.001031	.000900	.000797	.000718	.000648	.000589	.000529
4.0	.000816	.000710	.000628	.000565	.000510	.000464	.000418
4.2	.000654	.000568	.000502	.000452	.000407	.000369	.000330
4.4	.000530	.000460	.000406	.000366	.000329	.000297	.000264
4.6	.000434	.000377	.000333	.000300	.000269	.000241	.000213
4.8	.000359	.000312	.000275	.000249	.000223	.000199	.000174
5.0	.000300	.000260	.000230	.000208	.000186	.000165	.000144
5.2	.000252	.000219	.000194	.000176	.000156	.000138	.000120
5.4	.000214	.000189	.000164	.000150	.000133	.000117	.000101
5.6	.000183	.000159	.000140	.000129	.000115	.000100	.000085
5.8	.000157	.000137	.000121	.000111	.000099	.000086	.000073
6.0	.000136	.000118	.000104	.000097	.000086	.000074	.000062
6.2	.000118	.000103	.000091	.000085	.000076	.000065	.000054
6.4	.000103	.000090	.000079	.000075	.000067	.000057	.000047
6.6	.000091	.000079	.000070	.000066	.000059	.000050	.000040
6.8	.000080	.000070	.000062	.000059	.000053	.000044	.000035
7.0	.000071	.000062	.000055	.000053	.000047	.000039	.000031
7.2	.000063	.000055	.000049	.000048	.000043	.000036	.000028
7.4	.000056	.000049	.000043	.000043	.000038	.000032	.000025
7.6	.000050	.000044	.000039	.000040	.000036	.000029	.000022
7.8	.000045	.000039	.000035	.000036	.000032	.000026	.000020
8.0	.000041	.000035	.000031	.000034	.000030	.000024	.000018
8.2	.000037	.000031	.000027	.000032	.000028	.000023	.000017

Tabel B.1 Hoekverdelingsfunctie  $f(x)$  als functie van B volgens de benaderde N.S.W.-theorie (MAR 67). De cursief gedrukte waarden zijn door interpolatie of extrapolatie bepaald.

Tenslotte wordt  $\theta$  gegeven door  $\theta = x \cdot X_c \sqrt{B}$  (in het programma PROSIM heeft de variabele  $x$  de naam factor gekregen, omdat een variabele  $x$  al gebruikt was als  $x$ -coördinaat van de positie van het proton).

Hoek  $\phi$  wordt gegeven door  $\phi = 2\pi \cdot u_2$ , waarbij  $u_2$  uniform verdeeld is op  $[0,1]$ .

#### B.3.4 Het programma INTEGR

Hieronder is een listing afgedrukt van het programma INTEGR, waarmee, uitgaande van de getabelleerde kansverdelingsfunctie  $f(x)$ ,  $x=0(0.2)8.2$  (zie tabel B.1), voor één waarde van  $B$  de functie  $F_x^{-1}(u)$ ,  $u=0(0.002)1$  berekend wordt. Deze functiewaarden worden weggeschreven in een file. De files die zo voor  $B=6,7,8,9,10,11$  en  $12$  worden gemaakt worden aaneengevoegd tot een nieuwe file, BDATA.DAT, waarvan gebruik wordt gemaakt in het programma PROSIM. De file BDATA.DAT bevat slechts de waarden van  $x=0$  tot  $x=0.998$  voor  $F_x^{-1}(u)$  omdat  $F_x^{-1}(1)$  altijd gelijk is aan  $8$ .

```
PROGRAM integraal
C
REAL*4 f(821),ff(821),inttot,xx,xx1,xx2,yy1,yy2,intpol,
* a,b,c,x0,x1,x2,dx,s,jj,fo,fm,on,bo,mi,hoogte,
* nauwk,ffinv(1001)
INTEGER*2 i,j,k,device,stap,ix0,ix1,ix2,intho,inthm,nulp,
* stapgr
LOGICAL*1 wacht,flnm1(14),flnm2(14)
intpol(xx,xx1,xx2,yy1,yy2)=yy1+(xx-xx1)*(yy2-yy1)/(xx2-xx1)
TYPE*, ' input file ?'
TYPE*
CALL GTLIN(flnm1)
TYPE*, ' output file ?'
TYPE*
CALL GTLIN(flnm2)
TYPE*, ' device ? (7=term. 6=printer) '
ACCEPT*, device
nauwk=1.5E-7
TYPE*, ' stapgrootte (2) '
ACCEPT*, stapgr
OPEN (UNIT=2,NAME=flnm1,TYPE='OLD',ERR=300)
DO 10 i=1,821,20
READ (2,*,ERR=301) f(i)! 41 waarden voor f(i) inlezen, x=(i-1)/100
10 CONTINUE
CLOSE (UNIT=2,ERR=302)

C
C -----
C kwadratisch interpoleren om andere f(x)-en te bepalen
C -----
C
stap=20
s=.2
DO 11 ix0=1,781,stap! hoofdlus voor interpolatie
ix1=ix0+stap
ix2=ix1+stap
x0=FLOAT((ix0-1)/100.)
x1=x0+s
x2=x1+s
a=(f(ix2)+f(ix0)-2*f(ix1))/2/s/s
b=(f(ix1)-f(ix0))/s-a*(2*x0+s)
c=f(ix0)-a*x0*x0-b*x0
C
DO 12 j=ix0+1,ix1-1,1
jj=FLOAT((j-1)/100.)
f(j)=a*jj*jj+b*jj+c
12 CONTINUE
C
IF ((ix0+2*stap).LT.821) GOTO 11
C
DO 13 j=ix1+1,ix2-1,1
jj=FLOAT((j-1)/100.)
f(j)=a*(jj*jj)+b*jj+c
13 CONTINUE
C
11 CONTINUE! hooflus voor interpolatie
```

C  
C  
C  
C  
C  
C  
  
C  
D  
D  
1717  
17  
C  
C  
C  
C  
C  
C  
C  
  
19  
C  
C  
C  
C  
C

```
-----  
f(k) bevat f(x) voor x=0(.01)8.2, ongenormeerd, (k=100*x+1)  
Nu verdelingsfunctie F(x) berekenen van 0 tot 8  
-----
```

```
ff(1)=0.  
DO 16 i=2,821,1! integreren  
ff(i)=ff(i-1)+(2.*i-3.)*(f(i)+f(i-1))  
CONTINUE  
inttot=ff(801)  
  
DO 17 i=1,821,1  
WRITE (device,1717) (i-1)/100.,f(i),ff(i)/inttot,f(i)+ff(i)/inttot  
FORMAT(' i,F ',F5.3,3F9.6)  
CONTINUE
```

```
-----  
ff(k) bevat nu F(x) voor x=0(.01)8 (k=100*x+1), ongenormeerd  
-----
```

```
DO 18 k=1,821,1  
ff(k)=ff(k)/inttot! normeren  
CONTINUE
```

```
-----  
inverse van F(x) berekenen  
-----
```

```
ffinv(1)=0.  
DO 19 k=2,1001,1  
hoogte=FLOAT((k-1)/1000.)  
on=0.  
bo=8.  
mi=(on+bo)/2.! mbv Bisectie bepalen voor welke x F(x)=hoogte  
nulp=0  
intho=INT(100.*on)  
inthm=INT(100.*mi)  
fo=intpol(on,intho/100.,(intho+1)/100.,ff(intho+1),ff(intho+2))  
fm=intpol(mi,inthm/100.,(inthm+1)/100.,ff(inthm+1),ff(inthm+2))  
IF((fo-hoogte)*(fm-hoogte).LT.0.) nulp=1  
IF(nulp.EQ.1) bo=mi  
IF(nulp.NE.1) on=mi  
IF((bo-on).GT.nauwk) GOTO 20  
ffinv(k)=(bo+on)/2.  
CONTINUE
```

```
-----  
ffinv(k) bevat de inverse van F(x), k=x*1000+1  
-----
```

```
OPEN (UNIT=3,NAME=f1nm2,TYPE='NEW',ERR=300)  
WRITE (3,107,ERR=304) (ffinv(k),k=1,1001,stapgr)  
FORMAT(X,10F7.5)  
CLOSE (UNIT=3,DISPOSE='SAVE',ERR=302)
```

107

```
C
C
C
D
105 WRITE(device,105)
    FORMAT('      x          f(x)          n(x)          F(x)
*      F-1(x)      hoogte d(F-1)')
D
WRITE(device,104) ((k-1)/100.,f(k),f(k)/inttot,ff(k),ffinv(k),
D
*      (k-1)/1000.,ffinv(k)-ffinv(k-1), k=1,821,stapgr)
104 FORMAT(X,F7.2,3F14.9,F16.9,F8.3,F10.5)
D
WRITE(device,106) (ffinv(k),(k-1)/1000.,ffinv(k)-ffinv(k-1),
D
*      k=830,1001,stapgr)
106 FORMAT(50X,F16.9,F8.3,F10.5)
C
-----
GOTO 9999
300 TYPE*, 'FOUT IN OPEN'
301 TYPE*, 'FOUT IN READ'
302 TYPE*, 'FOUT IN CLOSE'
304 TYPE*, 'FOUT IN WRITE'
9999 END
```



#### B.4 De stopping power van verrijkte gassen

De verstrooiing van een proton aan een (gas-) folie is o.a. afhankelijk van de energie van het proton. De energie van het proton neemt af als het het target doorloopt. De energie-afremming wordt bepaald door de stopping power van het targetmateriaal. Voor alle elementen zijn de berekende waarden van de stopping power als functie van de protonen-energie getabelleerd (JAN 82). Deze waarden zijn echter berekend voor elementen in hun natuurlijke isotopische samenstelling, en dus niet voor verrijkte materialen.

De stopping power van verrijkte materialen kan verkregen worden uit de getabelleerde waarde van de stopping power voor het element in zijn natuurlijke samenstelling, door te corrigeren voor de atoommassa  $A$  van het verrijkte materiaal. De stopping power is bij benadering omgekeerd evenredig met  $A$  (JAN 82). Omdat de gemiddelde atoommassa voor natuurlijk xenon 131,3 is wordt de stopping power voor  $^{124}\text{Xe}$  gegeven door:

$$dE/pdx(^{124}\text{Xe}) = (131,3/124) \cdot dE/pdx(\text{Xe}_{\text{nat}}).$$

Hierbij is aangenomen dat de verrijkingsgraad 100% is.

De aldus gecorrigeerde waarden van de stopping power als functie van de energie van het proton zijn getabelleerd in tabel B.2.

Energie (MeV)	$-dE/\rho dx$ (MeV/kgm <sup>-2</sup> )
1	8,971
2	6,175
3	4,872
4	4,100
5	3,566
6	3,194
7	2,875
8	2,624
9	2,422
10	2,246
11	2,108
12	1,983
13	1,875
14	1,781
15	1,695
16	1,620
17	1,552
18	1,490
19	1,434
10	1,381
21	1,334
22	1,290
23	1,248
24	1,210
25	1,175
26	1,143
27	1,111

Tabel B.2 Stopping power van <sup>124</sup>Xe voor protonen

### B.5 Invloed van de parameters $N_{folie}$ en $N_{prot}$

De invloed van de parameter  $N_{folie}$  is weergegeven in fig. B.3. Bij eenzelfde target is de FWHM van de uittredende bundel bepaald bij verschillende waarden van  $N_{folie}$ . Geconcludeerd kan worden dat bij  $N_{folie} < ca. 100$  de berekende verstrooiing vrijwel onafhankelijk is van de parameter  $N_{folie}$ . Bij waarden van  $N_{folie}$  groter dan 100 à 200 treedt waarschijnlijk een te grote cumulatie van afrondfouten op, waardoor de verstrooiing niet meer correct wordt berekend.

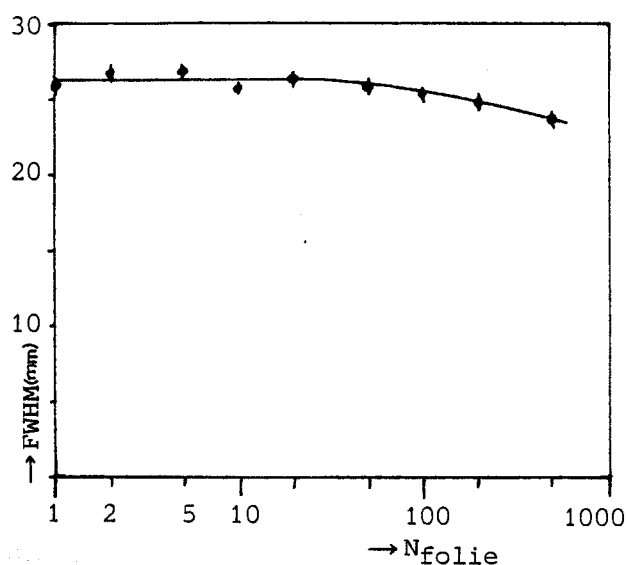


Fig. B.3 FWHM van het bundelprofiel na het doorlopen van een Xe-target met een druk van 3 bar en een lengte van 375 mm als functie van  $N_{folie}$ .

In fig. B.4 zijn een aantal profielen van de bundel aan het einde van een Xe-target met een lengte van 50 cm en een druk van 2 bar weergegeven. De parameter die gevariëerd werd is  $N_{prot}$ . Er werden resp. 1000, 2000, 4000, 8000, 16000 en 32000 protonen gelanceerd. Aan deze figuren is te zien dat een waarde van 16000 of 32000 voor  $N_{prot}$  acceptabele resultaten levert. In het algemeen is het zo dat bij een grotere verstrooiing meer protonen gelanceerd moeten worden om dezelfde nauwkeurigheid van het profiel te bereiken.

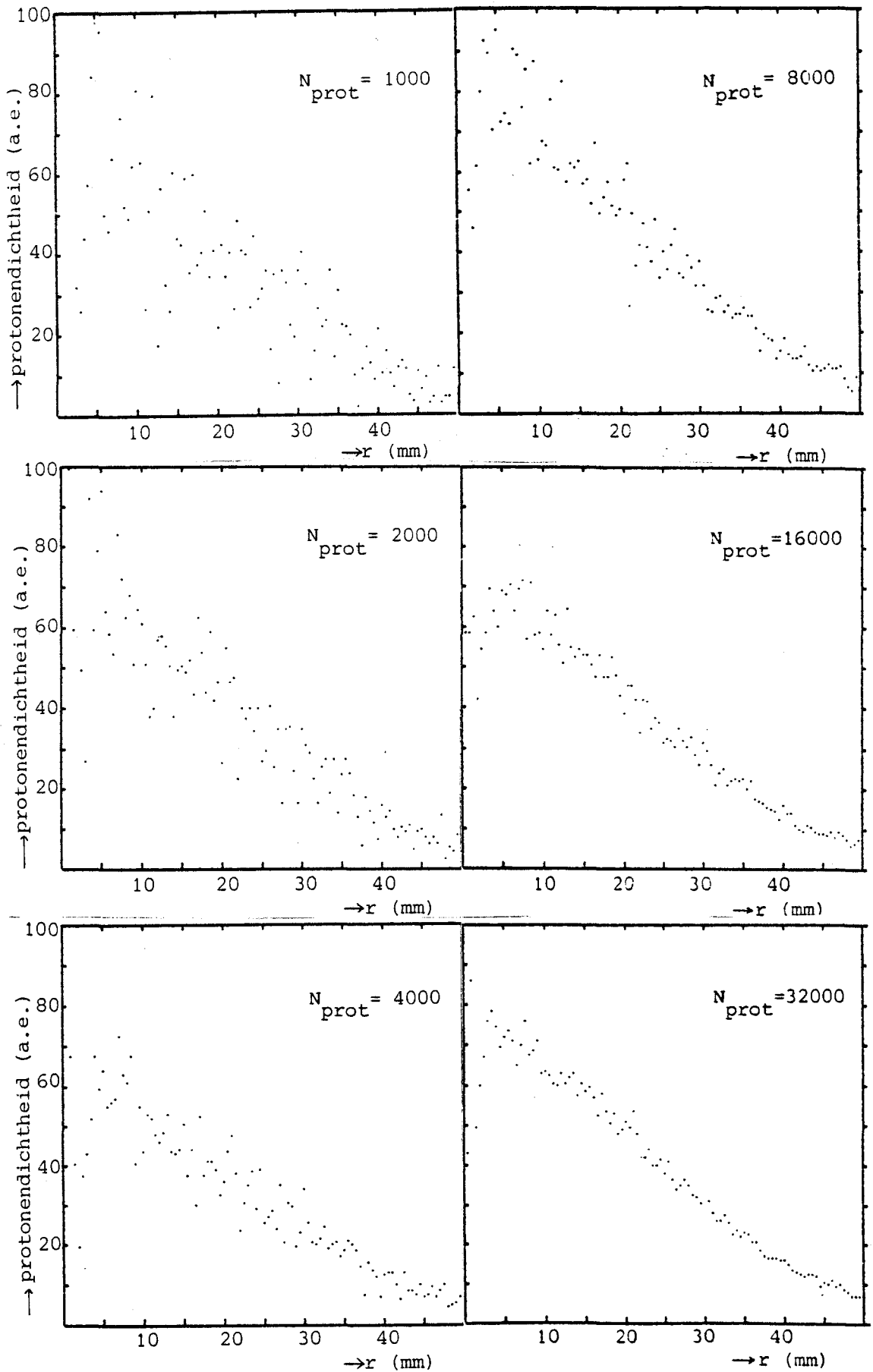


Fig. B.4 Bundelprofiel na het doorlopen van een Xe-target van 2 bar na 500 mm als functie van  $N_{\text{prot}}$ .

### B.6 Invloed van de divergentie van de bundel en de dichtheidsreductie

In fig. B.5 is de FWHM-waarde van de bundel aan het einde van een Xe-target met een lengte van 500 mm en een druk van 2 bar uitgezet als functie van de divergentie van de bundel (of om precies te zijn als functie van de afstand van het 'convergentiepunt' van de bundel tot het target-folie).

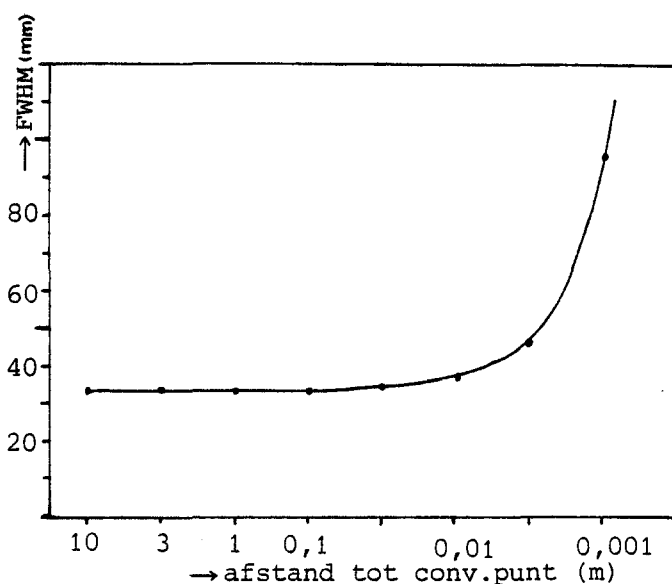


Fig. B.5 FWHM van het bundelprofiel na het doorlopen van een Xe-target met een druk van 2 bar en een lengte van 500 mm. De parameter die gevarieerd werd is de afstand van het 'convergentiepunt' van de bundel tot het targetfolie.

Uit deze figuur kan afgeleid worden dat de invloed van de divergentie van de bundel op de verstrooiing klein is als het 'convergentiepunt' meer dan 10 cm voor het target ligt.

De invloed van de dichtheidsreductie is onderzocht door een dichtheidsprofiel als weergegeven in fig. B.6 in het programma PROSIM in te voeren. De berekende verstrooiing is vrijwel gelijk aan de verstrooiing in een target met gelijke afmetingen, maar met een dichtheid die in het gehele target

gelijk is aan de minimale dichtheid (de dichtheid op de as van de targethouder) van het target mét dichtheidsreductie.

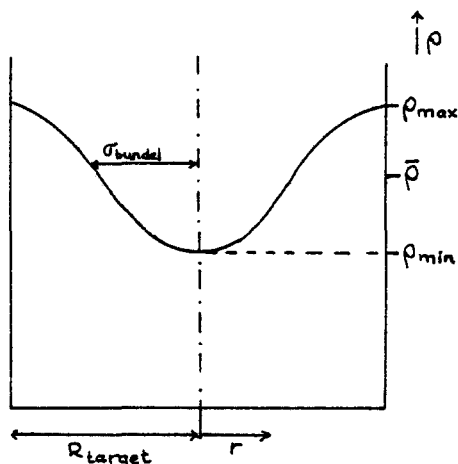


Fig. B.8 Dichtheidsprofiel waarmee de invloed van de dichtheidsreductie is bepaald. Voor de verhouding  $\rho_{\max}/\rho_{\min}$  is een waarde van 1,8 gekozen, voor de verhouding  $\sigma_{\text{bundel}}/R_{\text{target}}$  een waarde 0,5.

B.7 Vergelijking van de met het programma PROSIM berekende verstrooiing van protonen aan een dun nikkel-folie met de door Marion en Zimmerman theoretisch afgeleide verstrooiing.

Marion en Zimmerman (MAR 67) hebben voor protonen van 10, 50 en 100 MeV, die verstrooid worden door een dun Ni-folie (dikte 10 g/m<sup>2</sup>) de hoekverdelingsfuncties berekend, uitgaande van de NSW-theorie (zie hoofdstuk 4). Een hoekverdelingsfunctie geeft de protonendichtheid (uitgedrukt in aantal protonen per steradiaal) als functie van de verstrooiingshoek. De verstrooiing is ook met het programma PROSIM berekend. De verkregen hoekverdelingsfuncties blijken zeer goed overeen te stemmen met de hoekverdelingsfuncties die door Marion en Zimmerman berekend zijn (zie fig. B.9).

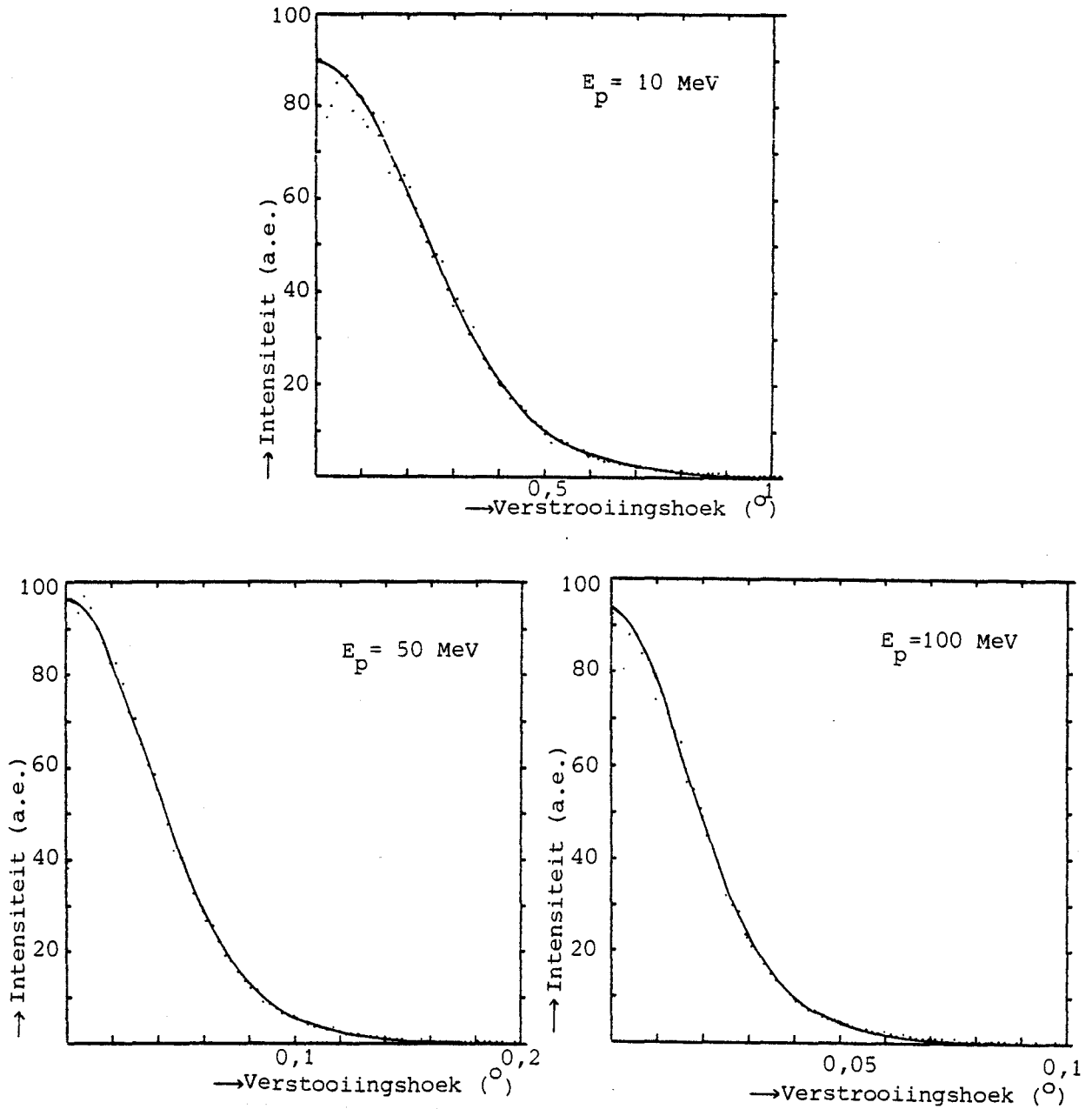


Fig. B.9 Hoekverdelingsfuncties van protonen van 10, 50 en 100 MeV, na verstrooiing in een Ni-folie met een dikte van  $10 \text{ kg/m}^2$ . De met het programma PROSIM berekende hoekverdeling wordt weergegeven door de puntjes, de door Marion en Zimmerman berekende hoekverdeling door de getrokken curven.

## APPENDIX C

*Uitgaande van de berekende protonenverstrooiing en de exitatiefuncties voor de reacties  $^{124}\text{Xe}(p,2n)^{123}\text{Cs}$  en  $^{124}\text{Xe}(p,pn)^{123}\text{Xe}$  kan de opbrengst van  $^{123}\text{I}$  in targets met verschillende afmetingen berekend worden. Hiertoe is een programma 'PROGRA' geschreven. In deze appendix wordt een toelichting gegeven op dit programma.*

### C.1 Inleiding

Met het programma PROGRA kunnen de resultaten van het programma PROSIM gevisualiseerd kunnen worden, en kan de 'optimale' vorm van een targethouder bij een bepaalde hoeveelheid gas met een bepaalde target-druk berekend worden.

De listing van het programma PROGRA en een overzicht van de in dit programma gebruikte variabelen is afgedrukt in § C.2. In § C.3 zijn, in tabelvorm de gebruikte exitatiefuncties afgedrukt.

### C.2 Listing van het programma PROGRA en overzicht van de gebruikte variabelen

Op de volgende pagina's is een listing van het programma PROGRA en een overzicht van de in dit programma gebruikte variabelen afgedrukt. De listing is op beperkte wijze van commentaar voorzien.

In het programma worden subroutines uit de 'programma's' IO en GRAF gebruikt (zie appendix A en B).

Alle FORTRAN-statements en STANDAARD-FORTRAN subroutines zijn met hoofdletters geschreven, alle zelf-gedefiniëerde variabelen en subroutines met kleine letters.



PROGRAM progra

C

```
COMMON /F/Afol,Agas,Booran,dikfol,diverg,drho,Ebegin,Eeind,Ef,
* flnaam,fostap,hoek,j,lengas,Nfolie,Npass,Nprot,opbr,
* Rafkap,Rdiafr,rest,rho0,rhofol,rin,ruit,sigger,sigmaE,
* sigmax,Zfol,Zgas,Zproj
REAL*4 abund,Afol,Agas,AINTst,Barbui,Barvol(12),bo,Booran,breed,
* cP1,cP2,cP3,cross1(28),cross2(28),crossf(10),
* crossg(10),crosst(10),cuml00,cumopb(10,100),dEdxfo,
* dikfol,diverg,drho,Ef(10),Ebegin,Eeind,fb,fk,fm,fo,
* gmperc(12),h,hoogte,intpol,l,lengas,max1,max2,max3,mi,
* mopbr,mspec,nauwk,on,opp,oppkl,oppgr,perc(12,10),pi,
* Rafkap(10),Rdiafr,rhofol,rho0,rhoSTP,rs,sf1,sf2,sf3,
* sfgas,sigmaE,sigmax,sigger,siguit,somkw,stap,
* straal(12,10),sx,sy,xoff,yoff,volbui,volume(12),
* Zfol,Zgas,Zproj
INTEGER*2 aantnp,buisbr,device,f,flniew,folnr,fostap,hoek(100),il,
* intEf,INThb,INThm,INTHo,INTstr,j,k,keuze,m,Nfolie,
* Nprot,Npass(10),nr,nulp,offset,opbr(10,100),
* opperv,rest,ruit(100),vorig0,rin(100)
LOGICAL*1 crosfl,datum(10),flnaam(14),floud,vraag,wacht
DATA pi,rs/3.1415926536,5E-4/
intpol(x,x1,x2,y1,y2)=y1+(x-x1)*(y2-y1)/(x2-x1)
CALL DATE(datum)
TYPE 108,(datum(k),k=1,10)
```

C

C

C

C

-----  
default parameters  
-----

```
nauwk=.001
breed=20.
device=7
-----
```

C

C

700

```
TYPE*, ' filenaam ? '
TYPE*
CALL GTLIN(flnaam)
crosfl=flnaam(1)
OPEN (UNIT=2,NAME=flnaam,TYPE='OLD')
READ (2,30) k,Nfolie,Nprot,Booran,rest,fostap,Zproj
30 FORMAT (3I6,F8.0,2I6,F5.1)
CALL leesin
CLOSE (UNIT=2,DISPOSE='SAVE')
c=drho*(.0005*.0005)/(.05*.05)
DO 31 k=1,10
DO 32 m=1,100
opbr(k,m)=opbr(k,m)*(1+c*m*m)
32 CONTINUE
31 CONTINUE
floud=crosfl
flniew=1
```

C

C

C

-----  
GOTO 40

```
600 flniew=0
TYPE*, ' Nprot,Nfolie ',Nprot,Nfolie
TYPE*, ' oude breedte: ',breed
TYPE*, ' nieuwe breedte ?'
ACCEPT*,breed
TYPE*, ' oude device :',device
TYPE*, ' device ? (7=scherm, 6=printer)'
ACCEPT*,device
TYPE 108,(flnaam(k),k=1,14)
TYPE*, ' Kr of Xe ?'
ACCEPT 1,crosfl
IF (crosfl.EQ.'') crosfl=floud
C
-----
40 IF(crosfl.EQ.'K') OPEN(UNIT=2,NAME='DK:CROSS.KR1',TYPE='OLD')
IF(crosfl.EQ.'X') OPEN(UNIT=2,NAME='DK:CROSS.XE1',TYPE='OLD')
DO 41 k=1,28
READ (2,*) cross1(k)
41 CONTINUE
CLOSE (UNIT=2,DISPOSE='SAVE')
C
IF(crosfl.EQ.'K') OPEN(UNIT=2,NAME='DK:CROSS.KR2',TYPE='OLD')
IF(crosfl.EQ.'X') OPEN(UNIT=2,NAME='DK:CROSS.XE2',TYPE='OLD')
DO 42 k=1,28
READ (2,*) cross2(k)
42 CONTINUE
CLOSE (UNIT=2,DISPOSE='SAVE')
IF (crosfl.EQ.'X') GOTO 43
sfgas=.4! Kr
abund=.72
rhoSTP=3.67
mopbr=1.48E9! max op 40 mCi/uAh
mspec=9.25E9! max op 250 mCi/Barl.uAh
GOTO 44
43 sfgas=5.7142857! Xe
abund=.998
rhoSTP=5.56
mopbr=1.48E8! max op 4 mCi/uAh
mspec=1.85E9! max op 50 mCi/Barl.uAh
44 opperv=INT((breed/50.)**2.*pi*50.*50.)
IF (flniew.EQ.1) buisbr=0! nieuwe buis
C
-----
C berekenen van schaalfactoren sf, verm.fact. cP en maxima max
C
-----
C sf1=5.0E-19! max op 4.8E20 protonen/m2
type*, 'rho0,lengas,abund,agas',rho0,lengas,abund,agas
sf2=4.8E-10*sfgas*3.733*.25*1./83.8/(rho0*lengas*abund/Agas)
C ! max op 5E11 Bq/m2.uAh.folie
sf3=9.6E-6*sfgas*3.733*.25*1./83.8/(rho0*lengas*abund/Agas)
C ! max op 2.5E7 Bq/ uAh.folie
C
cP1=2.25E16/(Npass(1)*pi*rs*rs)
cP2=2.25E16*1.E-28*abund*6.023E26*(lengas/10.)*rho0
* /(Npass(1)*Agas*pi*rs*rs)
```

```
CP3=2.25E16*1.E-28*abund*6.023E26*(lengas/10.)*rho0
*  /(Npass(1)*Agas)
C
max1=4.8E20*1E-6!      prot/mm2
max2=5E11*1E-6*(rho0*lengas*abund/Agas)/(sfgas*3.733*.25*1./
*      83.8)!          Bq/mm2.uAh.folie
max3=2.5E7*(rho0*lengas*abund/Agas)/(sfgas*3.733*.25*1./83.8)
!          Bq/      uAh.folie
-----
C
C
C
DO 50 k=1,10
intEf=INT(Ef(k))
crossf(k)=intpol(Ef(k),intEf+0.,intEf+1.,cross1(intEf),
*      cross1(intEf+1))
crossg(k)=intpol(Ef(k),intEf+0.,intEf+1.,cross2(intEf),
*      cross2(intEf+1))
crosst(k)=(crossf(k)*cross1(28)+crossg(k)*cross2(28))
50 CONTINUE
C
C
C
-----
C
C
C
berekenen van cumopb(i)
-----
C
DO 60 k=1,10
cumopb(k,1)=opbr(k,1)*(crosst(k))! cumulatieve opbrengst
60 CONTINUE
DO 61 k=1,10
DO 62 m=2,100
cumopb(k,m)=cumopb(k,m-1)+opbr(k,m)*crosst(k)
62 CONTINUE
61 CONTINUE
C
C
C
-----
C
C
C
berekenen van sigma (van ruit)
-----
C
siguit=0.
totuit=0.
opp=pi*rs*rs
DO 70 k=1,100
fk=FLOAT(k)
h=ruit(k)/((k*k-(k-1)*(k-1))*opp)
siguit=siguit+h*((fk-.5)*rs)**2.
totuit=totuit+h
70 CONTINUE
siguit=SQRT(siguit/(totuit-1.))
C
C
C
-----
C
C
C
menu
-----
80 CALL REGex1
TYPE 81
TYPE 82,breed,device,crosfl
TYPE 81
82 FORMAT(' breedte,device,gas:',F8.3,I2,A2)
```

```
81  FORMAT(///)
    TYPE*, ' maak keuze: 1. dN/dA (r)'! 100
    TYPE*, '                    2. dN/dA (A)'
    TYPE*, '                    3. d(opbrengst)/dA (r)'! 200
    TYPE*, '                    4. d(opbrengst)/dA (A)'
    TYPE*, '                    5. cumulatieve opbrengst (r)'! 300
    TYPE*, '                    6. cumulatieve opbrengst (A)'
    TYPE*, '                    7. eindprofiel (r)'! 400
    TYPE*, '                    8. eindprofiel (A)'
    TYPE*, '                    9. gegevens printen'! 500
    TYPE*, '                   10. buizen tekenen (eerst keuze 9)'! 800
    TYPE*, '                   11. breedte of device veranderen'! 600
    TYPE*, '                   12. nieuwe file inladen'! 700
    TYPE*, '                   13. opbrengstgrafieken'! 900
    TYPE*, '                   14. stoppen'! 1000
    TYPE 81
    ACCEPT*,keuze
    GO TO (100,100,200,200,300,300,400,400,500,800,600,700,900,1000)
    *      keuze
    GOTO 80
```

C  
C  
C  
C  
C

-----  
dN/dA  
-----

```
100  offset=1
D    GOTO 111
    CALL frame
111  CALL text(25.,468.)
    IF (keuze.EQ.1) TYPE 103,INT(breed)
    IF (keuze.EQ.2) TYPE 104,opperv
103  FORMAT(' breed:',I4,' mm"')
104  FORMAT(' breed:',I5,' mm2"')
    CALL text(25.,488.)
    TYPE 106,max1*1E-14
106  FORMAT(' hoogte:',F5.2,' E14"')
    CALL text(185.,488.)
    TYPE 107
107  FORMAT(' prot/mm2.uAh"')
    CALL text(700.,488.)
    TYPE 108,(flnaam(k),k=1,14)
    TYPE 109
    CALL text(700.,468.)
    TYPE 108,(datum(k),k=1,10)
    TYPE 109
108  FORMAT(' X',14A1)
109  FORMAT(X,'"')
C
110  DO 101 folnr=1,10
C
    DO 102 k=1,INT(100.*breed/50.)
    fk=FLOAT(k)*1.6
    h=FLOAT(opbr(folnr,k))/(k*k-(k-1)*(k-1))
D    TYPE*,h*sf1*cPl
```

```
D      GOTO 102
      IF (keuze.EQ.1) CALL plot (fk*50./breed+160.*(folnr-offset)+1.,
*          sfl*cP1*h+288.-48.*offset)!      als functie van r
      IF (keuze.EQ.2) CALL plot ((fk*50./breed)**2./100.+
*          160.*(folnr-offset)+1.,sfl*cP1*h+288.-48.*offset)! A
102    CONTINUE
      CALL text(25.+160.*(folnr-offset),496.-48.*offset)
      TYPE 105,(FLOAT(Npass(folnr))/FLOAT(Npass(1)))*2.25
105    FORMAT(' int:',F5.3,'E16 p"')
      IF (folnr.GE.5) offset=6

C
101    CONTINUE
      ACCEPT 1,wacht
1      FORMAT(A1)
      GOTO 80

C
C      -----
C      d(opbr)/dA
C      -----
C
200    offset=1
D      GOTO 211
      CALL frame
211    CALL text(30.,468.)
      IF (keuze.EQ.3) TYPE 103,INT(breed)
      IF (keuze.EQ.4) TYPE 104,opperv
      CALL text(30.,488.)
      TYPE 206,max2*1E-6! kBq/mm2.uAh.fol
206    FORMAT(' hoogte:'F6.3,'")
      CALL text(190.,488.)
      TYPE 207
207    FORMAT(' MBq/mm2.uAh.f"')
      CALL text(700.,488.)
      TYPE 108,(flnaam(k),k=1,14)
      TYPE 109
      CALL text(700.,468.)
      TYPE 108,(datum(k),k=1,10)
      TYPE 109

C
210    DO 201 folnr=1,10
C
      DO 202 k=1,INT(100.*breed/50.)
      fk=FLOAT(k)*1.6
      h=FLOAT(opbr(folnr,k))*crosst(folnr)/((k*k-(k-1)*(k-1))
D      TYPE*,h*sf2*cP2
D      GOTO 202
      IF (keuze.EQ.3) CALL plot (fk*50./breed+160.*(folnr-offset)+1.,
*          sf2*cP2*h+288.-48.*offset)!      als functie van r
      IF (keuze.EQ.4) CALL plot ((fk*50./breed)**2./100.+
*          160.*(folnr-offset)+1.,sf2*cP2*h+288.-48.*offset)! A
202    CONTINUE
      CALL text(30.+160.*(folnr-offset),496.-48.*offset)
      TYPE 205,(cP3*cumopb(folnr,100))*1E-6)
205    FORMAT(' int:',F5.1,'MBq/f"')
      IF (folnr.GE.5) offset=6
```

```
C
201  CONTINUE
      ACCEPT 1,wacht
      GOTO 80

C
C-----
C
C      cumulatieve opbrengst
C-----
C
300  offset=1
D    GOTO 311
      CALL frame
311  CALL text(30.,310.)
      IF (keuze.EQ.5) TYPE 103,INT(breed)
      IF (keuze.EQ.6) TYPE 104,opperv
      CALL text(30.,330.)
      TYPE 306,max3*1E-6
306  FORMAT(' hoogte: 'F6.1, ''')
      CALL text(190.,330.)
      TYPE 307
307  FORMAT(' MBq/folie.uAh''')
      CALL text(700.,488.)
      TYPE 108,(flnaam(k),k=1,14)
      TYPE 109
      CALL text(700.,468.)
      TYPE 108,(datum(k),k=1,10)
      TYPE 109

C
310  DO 301 folnr=1,10
C
      DO 302 k=1,INT(100.*breed/50.)
      fk=FLOAT(k)*1.6
D    TYPE*,cumopb(folnr,k)*sf3*cP3
D    GOTO 302
      IF (keuze.EQ.5) CALL plot (fk*50./breed+160.*(folnr-offset)+1.,
* sf3*cP3*cumopb(folnr,k)+288.-48.*offset)! als functie van r
      IF (keuze.EQ.6) CALL plot ((fk*50./breed)**2./100.+1.+160.*
* (folnr-offset),sf3*cP3*cumopb(folnr,k)+288.-48.*offset)! A
302  CONTINUE
      CALL text(30.+160.*(folnr-offset),486.-48.*offset)
      TYPE 305,(cP3*cumopb(folnr,100)*1E-6)
D    TYPE 308,(cP3*cumopb(folnr,100)*1E-6)
305  FORMAT(' int:',F5.1,'MBq/f''')
308  FORMAT(' int:',E15.5,'MBq/fol''')
      IF (folnr.GE.5) offset=6

C
301  CONTINUE
      ACCEPT 1,wacht
      GOTO 80

C
```

```
C -----
C      eindprofiel
C -----
C
400 versch=8.
      horsch=5.
D      GOTO 411
C
      CALL rand
411 CALL text(550.,458.)
      IF (keuze.EQ.7) TYPE 403,INT(breed)
      IF (keuze.EQ.8) TYPE 404,opperv
403 FORMAT(' breed:',I4,' mm"')
404 FORMAT(' breed:',I5,' mm2"')
      CALL text(550.,478.)
      TYPE 406,maxl*1E-14/versch
406 FORMAT(' hoogte:',F5.2,'E14 prot/mm2.uAh"')
C
410 DO 402 k=1,INT(100.*breed/50.)
      fk=FLOAT(k)*1.6
      h=FLOAT(ruit(k))/(k*k-(k-1)*(k-1))
D      TYPE*,h*sfl*cPl*versch
D      GOTO 402
      IF (keuze.EQ.7) CALL plot (fk*50./breed*horsch+1.,
*          sfl*cPl*h*versch)!      als functie van r
      IF (keuze.EQ.8) CALL plot ((fk*50./breed)**2./100.*horsch+1.,
*          sfl*cPl*h*versch)!      als functie van A
402 CONTINUE
      CALL text(550.,438.*offset)
      TYPE 405,(Npass(10)/Npass(1))*2.25
405 FORMAT(' int:',F5.3,'E16 pr"')
C
      ACCEPT 1,wacht
      GOTO 80
C
C -----
C      gegevens printen
C -----
500 WRITE (device,511) (flnaam(k),k=1,14),(datum(k),k=1,10)
      WRITE (device,501) Nprot,Nfolie,rest+1,fostap,Booran,
      WRITE (device,502) Zfol,Afol,rhofol,dikfol*1000,Rdiafr*1000
      WRITE (device,503) Zgas,Agas,rho0,lengas*1000,drho
      WRITE (device,504) sigmax,sigger,siguit*2355,
*          diverg*1000,Ebegin,Eeind,sigmaE,Zproj
      WRITE (device,505) (k,k=1,10)
      WRITE (device,506) (Rafkap(k)*1000,k=1,10)
      WRITE (device,507) (Ef(k),k=1,10)
      WRITE (device,508) (crosst(k)*1E6,k=1,10)
      WRITE (device,509) (Npass(k),k=1,10)
      WRITE (device,510) (cumopb(k,100),k=1,10)
501 FORMAT (' Nprot,Nfolie,fol1,folstap,Boorange',4I9,F9.1)
502 FORMAT (' Zfol,Afol,rhofol,dikfol,Rdiafragma',5F9.3)
503 FORMAT (' Zgas,Agas,rho0,lengas,rhored.(%)',5F9.3)
```

```
504 FORMAT (' sigx,sigger,BHHuit,divg          ',3F9.3,/,  
*      ' Eb,Ee,sigmaE,z                      ',4F9.3)  
505 FORMAT (' folnr ',10I7)  
506 FORMAT (' Rafkap ',10F7.3)  
507 FORMAT (' Ef      ',10F7.3)  
508 FORMAT (' crosst* ',10F7.3)  
509 FORMAT (' Npass  ',10I7)  
510 FORMAT (' cumopb ',10F7.4)  
511 FORMAT (' 0',14A1,60X,10A1)
```

C  
C

```
-----  
cuml00=0.  
DO 528 nr=1,10  
cuml00=cuml00+cumopb(nr,100)*cP3! (Bq/uAh)  
528 CONTINUE! maximale opbrengst van buis met stralen Rafkap(i)
```

C  
C

```
-----  
IF (buisbr.EQ.1) GOTO 534  
DO 523 f=1,12  
fr=f*2.-4.  
IF(f.EQ.1) fr=.5  
IF(f.EQ.2) fr=1.  
hoogte=fr/100.*240./sf2  
vorig0=0
```

C

```
DO 522 nr=1,10
```

C

```
-----  
straal(f,nr)=.0  
IF (vorig0.EQ.1) GOTO 522  
aantnp=0  
somkw=0.  
DO 521 il=1,99! snijpunt zoeken tussen l en l+1 in folie f  
l=FLOAT(il)
```

C

```
on=1  
bo=on+1.  
oppkl=1*1-(1-1.)*(1-1.)  
oppgr=(1+1.)*(1+1.)-1*1  
fo=opbr(nr,il)/oppkl*crosst(nr)*cP2  
fb=opbr(nr,il+1)/oppgr*crosst(nr)*cP2  
IF ((fo-hoogte)*(fb-hoogte).GT. 0.) GOTO 521! geen nulpunt
```

C

C

```
-----  
mbv. bisectie r-waarde zoeken waar d(opbr)/dA=hoogte  
-----
```

C

C

C

520

```
mi=(on+bo)/2.  
fo=intpol(on,1,l+1.,opbr(nr,il)/oppkl,  
*      opbr(nr,il+1)/oppgr)*crosst(nr)*cP2  
fm=intpol(mi,1,l+1.,opbr(nr,il)/oppkl,  
*      opbr(nr,il+1)/oppgr)*crosst(nr)*cP2  
nulp=0  
IF ((fo-hoogte)*(fm-hoogte) .LT. 0.) nulp=1  
IF (nulp.EQ.1) bo=mi
```



```
IF (nulp.NE.1) on=mi
IF ((bo-on).GT.nauwk) GOTO 520
aantnp=aantnp+1
somkw=somkw+mi*mi
521 CONTINUE! volgende l
IF (aantnp.NE.0) straal(f,nr)=SQRT(somkw/aantnp)*rs
IF (straal(f,nr) .LT. 0.001) vorig0=1
C
522 CONTINUE! volgend folienr
C
C
DO 525 nr=1,10! nr = folienummer
perc(f,nr)=0.
INTstr= INT(straal(f,nr)/rs)
AINTst=AINT(straal(f,nr)/rs)
IF ((straal(f,nr) .GE. 0.001) .AND. (cumopb(nr,100).NE.0))
*   perc(f,nr)=intpol(straal(f,nr)/rs,AINTst,AINTst+1.,
*   cumopb(nr,INTstr),cumopb(nr,INTstr+1))/cumopb(nr,100)
525 CONTINUE! fractie van max. opbrengst per folie binnen straal(f,nr)
C
volume(f)=0.
Barvol(f)=0.
volbui=0.
Barbui=0.
DO 526 nr=1,10
volume(f)=volume(f)+pi*straal(f,nr)**2*lengas/10.
Barvol(f)=Barvol(f)+pi*straal(f,nr)**2*lengas/10.*
*   (1.+5*drho*straal(f,nr)**2/.0025)*rho0/rhoSTP
volbui=volbui+pi*Rafkap(nr)**2*lengas/10.
Barbui=Barbui+pi*Rafkap(nr)**2*lengas/10.*
*   (1.+5*drho*Rafkap(nr)**2/.0025)*rho0/rhoSTP
526 CONTINUE! totale volume en 'Barvolume' van buis f
C
gmperc(f)=0.
DO 527 nr=1,10
INTstr= INT(straal(f,nr)/rs)
AINTst=AINT(straal(f,nr)/rs)
IF (straal(f,nr) .GE. 0.001)
* gmperc(f)=gmperc(f)+intpol(straal(f,nr)/rs,AINTst,
*   AINTst+1.,cumopb(nr,INTstr),cumopb(nr,INTstr+1))
C TYPE*, 'gmperc(tussen)', gmperc(f)
527 CONTINUE! totale opbrengst van buis f
gmperc(f)=gmperc(f)*cP3/cum100
C
523 CONTINUE! volgende f
buisbr=1
C
C
534 WRITE (device,529) cuml00/3.7E7,volbui*1E6,barbui*1E6,
*   cuml00/3.7E10/barbui
```

```
DO 535 f=1,12
fr=f*2.-4.
IF(f.EQ.1) fr=.5
IF(f.EQ.2) fr=1.
WRITE (device,530) fr,gmperc(f)*cuml00/3.7E7,volume(f)*1E6,
* Barvol(f)*1E6,gmperc(f)*cuml00/3.7E10/Barvol(f),
* gmperc(f)*100.
IF (device.EQ.7) WRITE (device,531) (nr, nr=1,10)
WRITE (device,532) (straal(f,nr)*1000, nr=1,10)
WRITE (device,533) (perc(f,nr)*100, nr=1,10)
535 CONTINUE
C
529 FORMAT(/,'OTOTAAL : opbr=',F6.3,'mCi vol=',F6.1,'ml=',F7.1,
* 'Barml spec.opbr=',F6.2,'mCi/Barl (/uAh op Tcal)')
530 FORMAT('Of=',F4.1,'% opbr=',F6.3,'mCi vol=',F6.2,'ml=',F7.2,
* 'Barml spec.opbr=',F6.2,'mCi/Barl opbr/max=',F5.2,'%')
531 FORMAT(' nr ',10I7)
532 FORMAT(' straal',10F7.2)
533 FORMAT(' perc ',10F7.2)
IF (device.EQ.6) CLOSE (UNIT=6)
C
C -----
C plaatjes van buizen tekenen
C -----
vraag='N'
TYPE*, 'Wil je een plaatje van de buizen'
ACCEPT 1,vraag
IF (vraag .NE. 'J') GOTO 80
800 CONTINUE! doet niets
D GOTO 540
CALL omrand
CALL line(1.,120.,800.,120.)
CALL line(1.,240.,800.,240.)
CALL line(1.,360.,800.,360.)
CALL line(267.,0.,267.,479.)
CALL line(533.,0.,533.,479.)
C
540 sx=266./0.5
sy=60./0.025
C
DO 541 f=1,12
xoff=MOD(f-1,3)*266.+1.
yoff=420-((f-1)/3)*120.
CALL line(xoff,yoff+Rdiafr*sy,xoff+lengas/20.*sx,
* yoff+straal(f,1)*sy)
CALL line(xoff,yoff-Rdiafr*sy,xoff+lengas/20.*sx,
* yoff-straal(f,1)*sy)
DO 542 nr=1,9
IF (straal(f,nr).LT. 0.001) GOTO 543
CALL line(xoff+(nr-.5)*lengas/10.*sx,yoff+straal(f,nr)*sy,
* xoff+(nr+.5)*lengas/10.*sx,yoff+straal(f,nr+1)*sy)
CALL line(xoff+(nr-.5)*lengas/10.*sx,yoff-straal(f,nr)*sy,
* xoff+(nr+.5)*lengas/10.*sx,yoff-straal(f,nr+1)*sy)
542 CONTINUE
```

```
C
CALL line(xoff+(9.5)*lengas/10.*sx,yoff+straal(f,10)*sy,
*       xoff+lengas*sx,yoff+intpol(10.,8.5,9.5,
*       straal(f,9),straal(f,10))*sy)
CALL line(xoff+(9.5)*lengas/10.*sx,yoff-straal(f,10)*sy,
*       xoff+lengas*sx,yoff-intpol(10.,8.5,9.5,
*       straal(f,9),straal(f,10))*sy)
543 CALL text(xoff+2.,yoff+78.)
TYPE 544,gmperc(f)*100.
544 FORMAT(' rl.o:',F5.1,'%''')
CALL text(xoff+2.,yoff-20.)
TYPE 545,gmperc(f)*cuml00/3.7E10/volume(f)*rhoSTP/rho0
545 FORMAT(' sp.o:',F5.1,'mCi/Barl''')
541 CONTINUE
CALL text(625.,60.)
TYPE 546,cuml00/3.7E7
546 FORMAT(' maxopb:',F5.2,'mCi/uAh''')
CALL text(700.,130.)
TYPE 108,(flnaam(k),k=1,14)
TYPE 109
CALL text(700.,110.)
TYPE 108,(datum(k),k=1,10)
TYPE 109
ACCEPT1,wacht
GOTO 80
```

```
C
C
C
C
C
```

---

opbrengst-grafieken

---

```
900 CONTINUE! doet niets
D   GOTO 902
CALL omrand
CALL line(1.,420.,800.,420.)
CALL line(401.,0.,401.,479.)
stap=800./32.
DO 901 k=1,32
CALL line(k*stap+1.,0.,k*stap+1.,5.)
CALL line(k*stap+1.,414.,k*stap+1.,419.)
901 CONTINUE
DO 902 k=42,420,42
fk=FLOAT(k)
CALL line(1.,fk,6.,fk)
CALL line(795.,fk,800.,fk)
902 CONTINUE
CALL text(2.,500.)
TYPE*, 'OPBRENGST''
CALL text(403.,500.)
TYPE*, 'SPEC.OPBR''
CALL text(2.,480.)
TYPE 903,mopbr/3.7E7
CALL text(403.,480.)
TYPE 904,mspec/3.7E7
```

```
CALL text(2.,460.)
TYPE 905,INT(1600.*breed/20.)
CALL text(403.,460.)
TYPE 905,INT(1600.*breed/20.)
CALL text(700.,498.)
TYPE 108,(flnaam(k),k=1,14)
TYPE 109,
CALL text(700.,478.)
TYPE 108,(datum(k),k=1,10)
TYPE 109,
903 FORMAT(' hoogte :',F6.1,' mC1/uAh''')
904 FORMAT(' hoogte :',F6.1,' mC1/Barl.uAh''')
905 FORMAT(' breedte:',I6,' Barml''')
C
sx=1E6*(20./breed)*400./1600.
sy=420.
DO 906 f=1,11
CALL line(Barvol(f)*sx,gmperc(f)*cuml00/mopbr*sy,
*         Barvol(f+1)*sx,gmperc(f+1)*cuml00/mopbr*sy)
CALL line(Barvol(f)*sx+401.,gmperc(f)*cuml00/(Barvol(f)*1E3)
*         /mspec*sy,Barvol(f+1)*sx+401.,gmperc(f+1)*cuml00/
*         (Barvol(f+1)*1E3)/mspec*sy)
906 CONTINUE
ACCEPT1,wacht
GOTO 80
C
C -----
C stoppen
C -----
1000 END
```

Overzicht van de in het programma PROGRA gebruikte variabelen

aantnp	aantal punten in de opbrengstdichtheid-curve met y-coördinaat=fr.maximale opbrengstdichtheid
abund	abundantie van isotoop in het target
AINTst	$AINT\{ \text{straal}(f,n)/rs \}$
Barbui	hoeveelheid gas in een buis met stralen Rafkap(i) ( $\text{barm}^3$ )
Barvol(f)	hoeveelheid gas in buis f ( $\text{barm}^3$ )
bo	bovengrens bij bisectie
breed	schaalfactor
buisbr	vlag die gezet wordt als de 12 targetbuizen berekend zijn
c	hulpvariabele voor dichtheidsreductie
crossf1	aanduiding targetmateriaal: X=xenon, K=krypton)
crossf(f)	waarde van de werkzame doorsnede van de 1e reactie in opbrengstfolie f ( $\text{barn}/\text{kgm}^{-3}$ )
crossg(f)	waarde van de werkzame doorsnede van de 2e reactie in opbrengstfolie f ( $\text{barn}/\text{kgm}^{-3}$ )
crossf(f)	$\text{crossf}(f).\text{cross1}(28)+\text{crossg}(f).\text{cross2}(28)$ ( $\text{barn}/\text{kgm}^{-3}$ )
cross1(i)	i=1,27 :waarde van de werkzame doorsnede voor de 1e reactie bij i MeV ( $\text{barn}/\text{kgm}^{-2}$ )
cross1(28)	factor waarmee de productiesnelheid P ( $\text{s}^{-1}/\mu\text{Ah}$ ) vermenigvuldigd moet worden om de opbrengst op een bepaald tijdstip (in $\text{Bq}/\mu\text{Ah}$ ) te verkrijgen
cross2(i)	als cross1(i), voor 2e reactie
cross2(28)	als cross1(28), voor 2e reactie
cumN(f,n)	cumulatief aantal protonen in folie f binnen segment n
cumopb(f,n)	(cumulatieve opbrengst in folie f binnen segment n)/cP3
cum100	totale opbrengst in alle folies ( $\text{Bq}/\mu\text{Ah}$ )
cP1	$2,25 \cdot 10^{16} / \{ N_{\text{pass}}(1) \cdot n(rs)^2 \}$
cP2	$2,25 \cdot 10^{16} \cdot 10^{-28} N_A \cdot (\text{lengas}/10) \cdot \rho_{\text{gas}} / \{ N_{\text{pass}}(1) \cdot \pi rs^2 \cdot A_{\text{gas}} \}$

cP3  $2,25 \cdot 10^{16} \cdot 10^{-28} N_A \cdot (\text{lengas}/10) \cdot \text{rhogas}/\{\text{Npass}(1) \cdot \text{Agas}\}$

datum array dat de datum bevat

device devicenr (6=printer, 7=display)

fb opbrengsdichtheid bij r=on

fm opbrengsdichtheid bij r=mi

fo opbrengsdichtheid bij r=bo

flniew vlag die gezet wordt als er een nieuwe file voor de werkzame doorsnede wordt ingelezen

floud vorige waarde van crosfi

fk hulpvariabele

fr fractie va maximale opbrengstdichtheid

gmperc(f) percentage van de totale opbrengst van buis f t.o.v. de buis met straal(i)=Rafkap(i), i=1,10

hoogte schaalfactor

horsch schaalfactor

il algemene index

INTEf INT(E(f)) (MeV)

INTstr INT(straal(f,n)/rs)

INThb INT(hb)

INThm INT(hm)

INTho INT(ho)

k hulpvariabele

keuze hulpvariabele bij menu

l FLOAT(il)

m algemene index

max1(2/3) met sf1(2/3) samenhangende schaalfactor

mi (on+bo)/2

mopbr schaalfactor

mspec schaalfactor

nauwk nauwkeurigheid bij bisectie

nr index voor folie-nummer

nulp vlag die bij bisectie aangeeft of de opbrengstdichtheidcurve de lijn y=hoogte snijdt.

offset offset bij plaatjes

on ondergrens bij bisectie

opp  $\pi r_s^2$  (m<sup>2</sup>)  
opperv schaalfactor (m<sup>2</sup>)  
oppgr  $b_0^2 - (b_0 - 1)^2$   
oppki  $a_0^2 - (a_0 - 1)^2$   
perc(f,n) percentage van totale opbrengst bij buis f in  
folie n binnen r=straal(f,n) (%)  
rhoSTP dichtheid van target bij T=273K en p=1bar (kg/m<sup>3</sup>)  
sfgas schaalfactor  
sf1 schaalfactor voor protonendichtheid dN/dA  
sf2 schaalfactor voor opbrengstdichtheid d(opbr)/dA  
sf3 schaalfactor voor cumulatieve opbrengst opbr(A)  
siguit standaarddeviatie van het bundelprofiel aan het  
eind van het target (m)  
somkw hulpvariabele bij bisectie  
straal(f,n) straal van folie n bij buis f (met fractie fr)  
(m)  
sx,sy schaalfactoren  
totuit hulpvariabele  
versch schaalfactor  
volbui volume van buis met straal(i)=Rafkap(i), i=1,10  
(m<sup>3</sup>)  
volume(f) volume van buis f (m<sup>3</sup>)  
vorig0 vlag die gezet wordt als de berekende straal in  
een folie kleiner dan 1 mm is  
vraag hulpvariabele  
wacht hulpvariabele  
xoff,yoff horizontale resp. verticale offset bij plaatjes

De overige variabelen die gebruikt worden in het programma  
PROGRA zijn beschreven bij het programma PROSIM in appendix  
B.

C.3 Exitatiefuncties.

In tabel C.1 zijn de exitatiefuncties die gebruikt zijn bij de berekening van de opbrengst getabelleerd.

Energie (MeV)	$\sigma_1$ (barn)	$\sigma_2$ (barn)
1	0,0000	0,0000
2	0,0000	0,0000
3	0,0000	0,0000
4	0,0000	0,0000
5	0,0000	0,0000
6	0,0000	0,0000
7	0,0000	0,0000
8	0,0000	0,0000
9	0,0000	0,0000
10	0,0000	0,0000
11	0,0000	0,0000
12	0,0000	0,0000
13	0,0000	0,0000
14	0,0000	0,0000
15	0,0005	0,0015
16	0,0084	0,0259
17	0,0264	0,0770
18	0,0543	0,1323
19	0,0882	0,1732
20	0,1383	0,2321
21	0,1961	0,2627
22	0,2598	0,2748
23	0,3146	0,2907
24	0,3561	0,2912
25	0,3767	0,2748
26	0,3810	0,2657
27	0,3810	0,2456

Tabel C.1 Crossectie in barn ( $=10^{-28}m^2$ ) voor de reacties  $^{124}Xe(p,2n)^{123}Cs$  ( $\sigma_1$ ) en  $^{124}Xe(p,pn)^{123}Xe$  ( $\sigma_2$ ).



## APPENDIX D

*In deze appendix worden enkele fysische eigenschappen van (natuurlijk) xenon gegeven. Van een aantal van deze gegevens is gebruik gemaakt bij de berekening van het warmtetransport in een xenon gastarget. De gegevens zijn ontleend aan COO 61, HAN 83, JAN 77, TEC 84 en VAR 75.*

atoomnummer	54	
atoommassa	131,3	u
dichtheid (gasvormig, T=290K)	5,89	kg/m <sup>3</sup>
dichtheid (vloeibaar, T=164K)	3520	kg/m <sup>3</sup>
dichtheid (vast, T=133K)	2700	kg/m <sup>3</sup>
smeltpunt	161	K
kookpunt	165	K
smeltwarmte bij smeltpunt	13,8	kJ/kg
verdampingswarmte bij kookpunt	96,2	kJ/kg
Cp (T=288K, p=1atm)	167	J/kg.K
Cp/Cv (T=288K, p=1atm)	1,66	
dampspanning bij smeltpunt	0,82	mPa
kritieke temperatuur	289,7	K
kritieke druk	5,84	MPa
warmtegeleidingscoëfficiënt (T=77K)	0,4	W/K (schatting)

De warmtegeleidingscoëfficiënt  $\lambda$ , de dynamische viscositeit  $\mu$  en de verzadigingsdampdruk  $p_d$  van xenon zijn temperatuurafhankelijk. In fig. D.1 en D.2 zijn de waarden voor  $\lambda$ ,  $\mu$  en  $p_d$  uitgezet als functie van de absolute temperatuur.

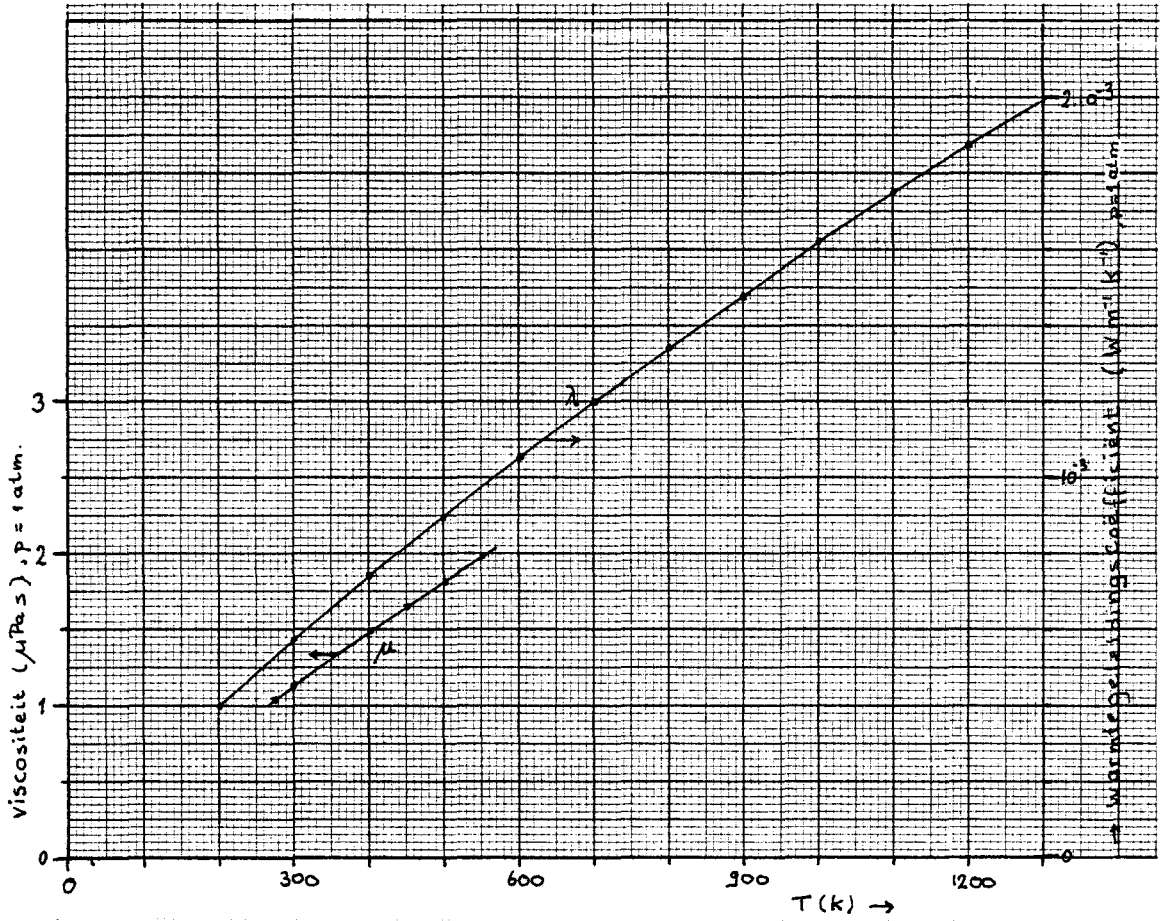


Fig. D.1 Warmtegeleidingscoëfficiënt ( $\lambda$ ) en dynamische viscositeit ( $\mu$ ) van natuurlijk Xe als functie van de absolute temperatuur

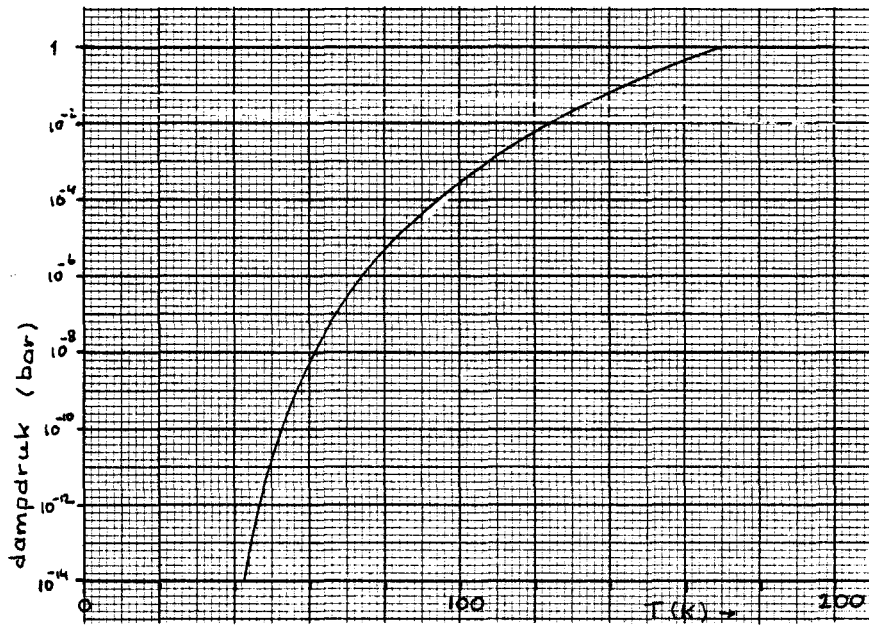


Fig. D.2 Verzadigingsdampdruk van xenon als functie van de absolute temperatuur