[Supplement](#HadjiSupplFirstCite).. Source of the programme used to analyse the here considered experimental data.

!! The following programme source is in Fortran 90. !!

! Absolute Atom concentration etc. determinations.

! The PROGRAMME uses the "Dichotomy" Method.

! - The effective mass must be provided to get the absolute atom concentration etc.

! - This programme source copy belongs to Noureddine HADJI and is "given" free of charge

! to whoever want(s) to use it as it is or to modify it before use.

 module Some\_Const\_Params0

! Module containing definitions of parameters needed to compute the Atom Concentration

 real(8) , parameter :: Ar0=5.292d-11 ! Bohr radius in m

 real(8) , parameter :: ECHARGe=1.602d-19 ! electron charge in C

 real(8) , parameter :: MASS0=9.110d-31 ! electron rest mass in kg

 real(8) , parameter :: HPlckConst=6.626d-34 ! Planck's constant in Js

 real(8) , parameter :: HBAR=1.055d-34 ! Planck's constant over two pi in Js

 real(8) , parameter :: SpeedC=2.998d8 ! speed of light in ms-1

 real(8) , parameter :: EPSILO=8.854D-12 ! permitivity of space in Fm-1

 double precision, parameter :: Pi = 3.14159265358979323845

 end module Some\_Const\_Params0

 module For\_Concentrations\_etc0

! Module containing definitions of variables needed to compute the Atom Concentration

 real(8) X,Var1,Y ,Q,STAR,XLAST,Var2, Var3

 integer IL

 end module For\_Concentrations\_etc0

!

 module Incident\_electron\_Dta0

 real(8) V0,D,WaveLengt,K0,ElecSpeed,Gama

 end module Incident\_electron\_Dta0

!

 module Get\_interval\_A1\_3\_0

 real(8) , parameter :: P0InitVal = 1.D-15

 integer SubType

 real(8) A1,A2,A3,Y1,Y2,Y3,Alow,Aup,AplausMin

 real(8) VA1,VA3, DeltaA,Var1LT,Ysmallest,Var1Ysmallest,P0

 real(8) Yf, Yg

 character\*52 OutCome

 end module Get\_interval\_A1\_3\_0

!

 module Calculated\_Factors0

 real(8) PlPeaP, LambdaP, ElectConc, FEnerg,EFoverEp

 real(8) LOGKF, LOGElectConc,Lambda,ElectConcToThird

 real(8) Thetac,ThetaE, ElecEfMas,KF,Alfa,ThetaF,Nk1

 real(8) ThetaFto3over2, KC

 character(1) RPAorExCor

 logical errKc, errThetaC

 end module Calculated\_Factors0

!

 subroutine If\_SubType13

 use Get\_interval\_A1\_3\_0

 implicit none

 if (SubType.eq.13) then

 call Elect\_Conc\_Dispersion\_etc\_Eff\_M\_Known\_FULL\_Kc13

 end if

 return

 end subroutine If\_SubType13

!

 subroutine Dichotoming !! Dichotomy Method for solving nonlinear equations.

 use Get\_interval\_A1\_3\_0

 use For\_Concentrations\_etc0

 implicit none

 integer i

 i=0

 if (A1.eq.A3) then

 A2=A1

 goto 1200

 end if

! A2=(A1+A3)/2.

 A2=(1.+A3/A1)/2.\*A1

 Var1=A1

 call If\_SubType13

 Y1=Y

 Var1=A2

 call If\_SubType13

 Y2=Y

 Var1=A3

 call If\_SubType13

 Y3=Y

 do

 if (Y1\*Y2.lt.0.0) then

 A3=A2

 Y3=Y2

 else

 A1=A2

 Y1=Y2

 end if

 A2=(1.+A3/A1)/2.\*A1

 if (abs(Y).le.P0) then

 write(\*,\*)' Condition abs(Y)<=P0 satisfied (in 1) after ',i,' small loops'

 ! A2 Returned value

 A1=A2

 A3=A1

 goto 1200

 end if

 Var1=A2

 call If\_SubType13

 Y2=Y

 i=i+1

 end do

1200 continue

 return

 end subroutine Dichotoming

!

 subroutine Get\_Incident\_electron\_Dta\_Final0

 use Some\_Const\_Params0

 use Incident\_electron\_Dta0

 implicit none

760 write (\*,\*) 'Enter accelerating voltage, in kilovolts'

 write (\*,\*) 'E.g. 100.'

 write (\*,\*) ''

 read(\*,\*,err=760) V0

 V0=V0\*1000.0\_8

! Get k0=2pi/lambda

 D=SQRT(V0+0.97845e-6\*V0\*V0)/1.22639\*1.e9

! Get wave length

 WaveLengt=1.0\_8/D

 K0=2.\_8\*Pi\*D

! Get incident electron velocity.

! First work out D.

 D=D\*(HPlckConst/MASS0)

! Incident electron velocity is: ElecSpeed

 ElecSpeed=D/SQRT(1.\_8+(D/SpeedC)\*(D/SpeedC))

! Get Gamma

 Gama=1./SQRT(1.\_8-(ElecSpeed/SpeedC)\*(ElecSpeed/SpeedC))

 print \*, '!------------------------------------------------------&

 ----------------------!'

 print \*, ' '

 print \*, ' Have obtained the following "incident-electron" data:'

 print \*, ' '

 print \*, '!------------------------------------------------------&

 ----------------------!'

 write(\*,2510)' Incident electron wave length: ',WaveLengt\*1.e10,'&

 angs'

 print \*, '!------------------------------------------------------&

 ----------------------!'

 write(\*,2520)' Incident electron wave vector: ',K0,' in m-1'

 print \*, '!------------------------------------------------------&

 ----------------------!'

 write(\*,2520)' Incident electron velocity: ',ElecSpeed,' m/s'

 print \*, '!------------------------------------------------------&

 ----------------------!'

 write(\*,2510)' Gamma : ',Gama

 print \*, '!------------------------------------------------------&

 ----------------------!'

!

2510 format(A,F14.5,A)

2520 format(A,ES15.5,A)

 return

 end subroutine Get\_Incident\_electron\_Dta\_Final0

!

 SUBROUTINE Elect\_Conc\_Dispersion\_etc\_Eff\_M\_Known\_FULL\_Kc13

 use Some\_Const\_Params0

 use Incident\_electron\_Dta0

 use For\_Concentrations\_etc0

 use Get\_interval\_A1\_3\_0

 use Calculated\_Factors0

! NOTE: This routine is not based on the equations given in the paper

! by N. Hadji: Two Analytical Techniques for the Determination

! of the Absolute Atom Concentration Using Electron Energy

! Loss Spectroscopy (EELS). Int. J. Mod. Phys. B, 16, 4775 (2002).

! The equations given in the above text are used. That is in the text

! More Accurate Formulas for Determination of Absolute Atom Concentration Using Electron

! Energy-Loss Spectroscopy.

 implicit none

! Q=PlSctCSec(1) is PSCA per atom of the first element entered.

! Var1=ElectConc

! Work out the plasmon peak energy

 PlPeaP=(HBAR\*(Var1)\*\*(1./2.)/ElecEfMas\*\*(1./2.)/EPSILO\*\*(1./2.))\*ECHARGe

! Work out the Fermi vector

 KF=(3.\*Pi\*Pi\*Var1)\*\*(1./3.)

! Work out the Fermi energy

 FEnerg=(HBAR\*KF)/MASS0\*(HBAR\*KF)/2.

! Work out the ratio of the Fermi energy to the first plasmon peak position energy

 EFoverEp=FEnerg/PlPeaP

 if (RPAorExCor .eq. 'k' .or. RPAorExCor .eq. 'K') then

 !! Value for alfa was provided

 else

 !! Either RPA or ExCor considered.

 Alfa=3./5.\*(EFoverEp)

 if ( RPAorExCor .eq. 'E' .or. RPAorExCor .eq. 'e') then

 !! ExCor considered.

 Alfa=Alfa\*(1.-(1./EFoverEp)\*(1./EFoverEp)/16.)

 else

 !! RPA considered.

 end if

 end if

 if (Alfa .eq. 0.5) then

 KC = KF/2.\*(PlPeaP/FEnerg)

 else

 if (1.-(2\*Alfa-1.)\*PlPeaP/FEnerg .le. 0) then

 errKc = .true.

 write(\*,\*) ' There is error !!:'

 write(\*,\*) '(1.-(2\*Alfa-1.)\*PlPeaP/FEnerg) is negative'

 return

 end if

 KC = KF/(2.\*Alfa-1.)\*(1.-DSQRT(1.-(2\*Alfa-1.)\*PlPeaP/FEnerg))

 end if

 ThetaE=PlPeaP/(Gama\*MASS0\*ElecSpeed\*ElecSpeed)

 if ((KC/K0 - ThetaE)\*(KC/K0 + ThetaE) .le. 0) then

 errThetaC = .true.

 write(\*,\*) ' There is error !! I cannot calculate Thetac:'

 write(\*,\*) '(KC/K0 - ThetaE)\*(KC/K0 + ThetaE) is negative'

 return

 end if

 Thetac = DSQRT((KC/K0 - ThetaE)\*(KC/K0 + ThetaE))

!! The following next instruction is the equation to solve.

!! Var1 is the unknown to be determined.

!! The solution is obtained when Y=0.

 Y=1.-Gama\*STAR\*ThetaE/(Q\*Var1)/Ar0\*DLOG(KC/K0/ThetaE)

 Var1LT = (Var1LT-Var1)/Var1

 IF (Y.eq.0.0.OR.ABS(Var1LT).le.P0) IL=1

 Var1LT=Var1

 if (Ysmallest.gt.abs(Y)) then

 Ysmallest=abs(Y)

 Var1Ysmallest=Var1

 endif

 return

 END SUBROUTINE Elect\_Conc\_Dispersion\_etc\_Eff\_M\_Known\_FULL\_Kc13

!

 subroutine get\_interval\_A1\_A3\_Lt

! This routine gets the interval [A1, A3] enclosing the solution Var1

! for a non-linear equation defined in another subroutine that this

! subroutine calls. This is for PHYSICAL QUANTITIES ONLY, and therefore

! for strictly positive quantities. To get it working for other quantities,

! e.g. including negative values for Var1, one needs to change

! the local variables Alow and Aup accordingly.

 use Some\_Const\_Params0

 use For\_Concentrations\_etc0

 use Get\_interval\_A1\_3\_0

 use Incident\_electron\_Dta0

 use Calculated\_Factors0

 implicit none

 integer L, k

 real(8) Mas0

!---------------------Get interval containing solution ------------------

 Var1LT=1.e-38

 errKc = .false.

 errThetaC = .false.

 write(\*,\*) ' Started Working, please wait for the results !!!!'

 write(\*,\*) ' '

940 Mas0=MASS0

 Alow=0.0

 k=0

970 Aup=AplausMin\*100.

 if (SubType.eq.6) Aup=AplausMin\*500.

 Var1=1.e38

 Ysmallest=1.e38

980 DeltaA=A2/100.

985 continue

 if (SubType.eq.5) DeltaA=A2/3.

990 A1=A2

 A3=A2

 L=0

 Var1LT=3.\*Var1

 Var1=A2

995 continue

 call If\_SubType13

 if (errKc) return

 if (errThetaC) return

 Y2=Y

 if (abs(Y).le.P0.or.DeltaA/Var1.le.P0InitVal) then

 A2=Var1 ! Returned value

 A1=A2

 A3=A1

 return

 end if

 Y1=Y2

 Y3=Y2

1020 if (Y1\*Y3>0.0) then

 A1=A1-DeltaA

 if (A1<=Alow) then

 if (A3>=Aup) then

 L=1

 goto 1120

 end if

 A1=A1+DeltaA-DeltaA/10.

 else if (A3>=Aup) then

 if (A1<=Alow) then

 L=1

 goto 1120

 end if

 A3=A3-DeltaA+DeltaA/10.

 else

 end if

 Var1=A1

 if (Var1<=0.0) then

 goto 1060

 end if

 call If\_SubType13

 if (errKc) return

 if (errThetaC) return

 Y1=Y

 if (abs(Y).le.P0.or.DeltaA/Var1.le.P0InitVal) then

 A2=Var1 ! Returned value

 A1=A2

 A3=A1

 return

 end if

1060 A3=A3+DeltaA

 Var1=A3

 call If\_SubType13

 if (errKc) return

 if (errThetaC) return

 Y3=Y

 if (abs(Y).le.P0.or.DeltaA/Var1.le.P0InitVal) then

 A2=Var1 ! Returned value

 A1=A2

 A3=A1

 return

 end if

 goto 1020

 else

 if (Y1\*Y2>0.0) then

 A1=A3-DeltaA

 Var1=A1

 call If\_SubType13

 if (errKc) return

 if (errThetaC) return

 Y1=Y

 else if (Y3\*Y2>0.0) then

 A3=A1+DeltaA

 Var1=A3

 call If\_SubType13

 if (errKc) return

 if (errThetaC) return

 Y3=Y

 else if (Y1.eq.0.0) then

 A3=A1

 A2=A1

 VA1=A1

 VA3=A3

 return

 else if (Y3.eq.0.0) then

 A1=A3

 A2=A3

 VA1=A1

 VA3=A3

 return

 end if

 VA1=A1

 VA3=A3

 end if

1120 if (L.eq.1) then

 write(\*,\*) ' '

 write(\*,\*) '!------------------------------------------------------&

 ----------------------!'

 write(\*,\*) ' Enter a number you expect is much larger than the'

 write(\*,\*) ' '

 write(\*,\*) ' solution you are looking for. Use MKSA units.'

 write(\*,\*) '!------------------------------------------------------&

 ----------------------!'

 write(\*,\*) ' '

1140 write(\*,\*) ' If you do not know what to enter, try', Aup\*5.

 write(\*,\*) ' '

 write(\*,\*) ' To return to the Menu enter 0.0'

1150 write(\*, '(A)', ADVANCE = "NO")' => '

 read (\*,\*,err=1140)Aup

 if (Aup.eq.0.0) return

 if (Aup.lt.AplausMin) then

 write(\*,\*) '!------------------------------------------------------&

 ----------------------!'

 write(\*,\*) ' PLEASE try a larger number.'

 goto 1140

 end if

 L=0

 A2=AplausMin\*2.

 goto 980

 else if (L.eq.0) then

 end if

 return

 end subroutine get\_interval\_A1\_A3\_Lt

!

 program Absol\_Atom\_Concent\_Determ\_Need\_Eff\_Mass\_FULL

! NOTE: This programme is built on the basis of the equations given in the text

! " More Accurate Formulas for Determination of Absolute Atom Concentration Using Electron

! Energy-Loss Spectroscopy."

! These equations are for obtaining results more precise

! than those obtained using the equations given in the article "Two Analytical

! Techniques for the Determination of the Absolute Atom Concentration

! Using Electron Energy Loss Spectroscopy (EELS)". See International Journal

! of Modern Physics B, Vol. 16, No 31, pp 4775-4781 (2002)' N. Hadji.

! The electron concentration is the parameter varied to get the function Y zeroed.

 use Some\_Const\_Params0

 use Incident\_electron\_Dta0

 use For\_Concentrations\_etc0

 use Get\_interval\_A1\_3\_0

 use Calculated\_Factors0

 implicit none

 character\*2, dimension(:), allocatable :: ElemNames

 character\*1 choix,Choice, ChoiceP, RPAorExCorP

 character(255) file

 INTEGER(KIND=4)Ierr

 integer n

 integer i, ILLa,ILLb

 integer AllocateStatus,DeAllocateStatus

 real(8) X0

 real(8) Alfa1

 real(8), dimension(:), allocatable :: Nk

 real(8), dimension(:), allocatable :: Pelect,PlSctCSec

 print \*, ' This programme uses equations more precise than those'

 print \*, ' considered in the article "Two Analytical Techniques for'

 print \*, ' the Determination of the Absolute Atom Concentration'

 print \*, ' Using Electron Energy Loss Spectroscopy (EELS)".'

 print \*, ' [See International Journal of Modern Physics B,'

 print \*, ' Vol. 16, No 31, pp 4775-4781 (2002)'

 print \*, ' by N. Hadji.]'

 print \*, ' '

 print \*, ' The ELECTRON EFFECTIVE MASS is required.'

 print \*, ' '

 write(\*,\*) '/////////////////////////////////////////////////////////'

 print \*, '/ This programme is entirely free of charge. /'

 print \*, "/ So please don't pay for it. /"

 write(\*,\*) '/////////////////////////////////////////////////////////'

! Write equation to solve in subroutine Elect\_Conc\_Dispersion\_etc\_Eff\_M\_Known\_FULL\_Kc13

 print \*, ' '

100 do

 RPAorExCor = 't'

 write(\*,\*) '-------------------------- Menu -------------------- &

 ------------!'

 write(\*,\*) ' '

 write(\*,\*) ' Q to quit'

 write(\*,\*) ' A for Technique no 1'

 write(\*,\*) ' B for Technique no 2'

 if (ILLa.eq.1) then

 write(\*,\*) ' D to get thickness (Technique no 1)'

 write(\*,\*) ' P to print results as Resutlts\_AAC\_etc.txt file.'

 else if (ILLb.eq.1) then

 write(\*,\*) ' E to get thickness (Technique no 2)'

 write(\*,\*) ' P to print results as Resutlts\_AAC\_etc.txt file.'

 end if

 read (\*,\*) Choice

 P0=P0InitVal

 if (Choice=='Q'.or.Choice=='q') then

 stop

 else if (Choice=='A'.or.Choice=='a') then

 IF (ILLb.eq.1.or.ILLa.eq.1) THEN

 deallocate (ElemNames, STAT = DeAllocateStatus)

 deallocate ( Nk, STAT = DeAllocateStatus )

 deallocate ( Pelect, STAT = DeAllocateStatus )

 deallocate ( PlSctCSec, STAT = DeAllocateStatus )

 END IF

 ILLa=0.

 IL=0

 ChoiceP = 'A'

 Alfa = 0.0

 Alfa1 = 0.0

 goto 700

 else if (Choice=='B'.or.Choice=='b') then

 IF (ILLb.eq.1.or.ILLa.eq.1) THEN

 deallocate (ElemNames, STAT = DeAllocateStatus)

 deallocate ( Nk, STAT = DeAllocateStatus )

 deallocate ( Pelect, STAT = DeAllocateStatus )

 deallocate ( PlSctCSec, STAT = DeAllocateStatus )

 END IF

580 write(\*,\*) 'Enter first plasmon peak position in electron volts.'

 read(\*,\*,err=580) PlPeaP

 PlPeaP=PlPeaP\*ECHARGe

 ILLb=0.

 ChoiceP = 'B'

 Alfa = 0.0

 Alfa1 = 0.0

 goto 700

 else if (Choice=='P'.or.Choice=='p') then

 goto 40000

 else if (Choice=='D'.or.Choice=='d'.or. &

 Choice=='E'.or.Choice=='e') then

 if (ILLa.eq.0.and.ILLb.eq.0) goto 100

610 print \*, 'Enter thickness to plasmon mean free path ratio'

 print \*, ' enter the number of atoms per unit are &

 of element ',ElemNames(1)

 read (\*,\*,err=610) X0

 if (X0.lt.5.0.and.X0.gt.0.001) then

 print \*, ' '

 print \*, ' Thickness = ',X0\*LambdaP,' m'

 else if (X0>1.e20) then

 print \*, ' '

 print \*, ' Thickness = ',X0/Nk(1),' m'

 else

 print \*, ' '

 print \*,' Error !!!, thickness = ',X0\*LambdaP,' m'

 end if

 print \*, ' '

 pause 'Press return to go back to menu'

 else

 end if

 end do

700 write(\*,\*) ' Enter R for RPA, E for ExCor or K for '

 write(\*,\*) ' known volume plasmon dispersion coefficient.'

 write(\*,\*) ' (NOTE: to use approximate cutoff vector'

 write(\*,\*) ' enter k and type in 0.5 for Alfa).'

 read(\*,\*, err=700) RPAorExCor

 RPAorExCorP = RPAorExCor

 if (RPAorExCor .ne. 'R' .and. RPAorExCor .ne. 'r' &

 & .and. RPAorExCor .ne. 'E' .and. RPAorExCor .ne. 'e' &

 & .and. RPAorExCor .ne. 'k' .and. RPAorExCor .ne. 'K') goto 700

 if (RPAorExCor .eq. 'k' .or. RPAorExCor .eq. 'K') then

710 write(\*,\*) 'Enter value for plasmon dispersion coefficient'

 read(\*,\*, err=710) Alfa

 end if

730 print \*, ' Enter electron effective mass in MKSA units '

 read (\*,\*,err=730) ElecEfMas

 if (ElecEfMas .ge. 1.e-25) then

740 write(\*,\*) ' The entered electron effective mass is ',ElecEfMas

 write(\*,\*) " Isn't it too large ? If it is not then enter N."

 write(\*,\*) 'Otherwise, enter any character to continue or m to Menu.'

 read(\*,\*, err= 740) choix

 if (choix .ne. 'n' .and. choix .ne.'N') goto 730

 if (choix .eq. 'm' .or. choix .eq.'M') goto 100

 end if

760 write (\*,\*) 'Enter the number of chemical elements in specimen'

 read(\*,\*, err=760) n

 if (Choice=='B'.or.Choice=='b') then

 if (n.lt.2) then

 print \*, 'Technique No 2 cannot be used, the &

 & numbers of elements in specimen less than 2'

 goto 100

 end if

 else

 end if

 allocate (ElemNames(n))

 IF (AllocateStatus /= 0) STOP "\*\*\* ElemNames(:) Not enough &

 & memory \*\*\*"

 allocate ( Nk(n), STAT = AllocateStatus )

 IF (AllocateStatus /= 0) STOP "\*\*\* Nk(:) Not enough memory \*\*\*"

 allocate ( Pelect(n), STAT = AllocateStatus )

 IF (AllocateStatus /= 0) STOP "\*\*\* Pelect(:) Not enough &

 & memory \*\*\*"

 allocate ( PlSctCSec(n), STAT = AllocateStatus )

 IF (AllocateStatus /= 0) STOP "\*\*\* PlSctCSec(:) Not enough &

 & memory \*\*\*"

 print \*, ' '

 print \*, ' '

 if (Choice.eq.'A'.or.Choice.eq.'a') print \*, ' &

 & We are using Technique No 1'

 if (Choice.eq.'B'.or.Choice.eq.'b') print \*, ' &

 & We are using Technique No 2'

 print \*, ' '

 print \*, ' '

 write (\*,\*) ' Data PPROVISION (in MKSA units).'

 print \*, ' '

 print \*, ' '

 write (\*,\*) ' Enter data for the DETECTABLE chemical'

 print \*, ' elements in the following order:'

 print \*, ' '

 write (\*,\*) "element's name; valency; plasmon scattering cross&

 & section per element."

 print \*, ' '

 if (Choice.eq.'A'.or.Choice.eq.'a') then

 write (\*,\*) 'E.g.: name: Si, valency: 4., Plasm Scat Cros Sect&

 & per Si at = 1.77e-22 to get:'

 write(\*,\*) 'Si,4.,1.77e-22'

 else if (Choice.eq.'B'.or.Choice.eq.'b') then

 write (\*,\*) 'E.g.: name: Si, valency: 4., Plasm Scat Cros Sect&

 & per Si at = 1.93e-22 to get:'

 write(\*,\*) 'Si,4.,1.93e-22'

 end if

 print \*, ' '

 do i=1,n

 write (\*,\*) 'Element No ',i

 print \*, ' '

 if (i.eq.n) then

 if (Choice.eq.'B'.or.Choice.eq.'b') then

800 print \*, ' NOW: enter just the name and the valency for &

 the element'

 print \*, ' which is NOT EASY TO DETECT.'

 print \*, ' For example: H,1.0'

 read(\*,\*,err=800) ElemNames(i),Pelect(i)

 goto 900

 end if

 end if

840 read(\*,\*,err=840) ElemNames(i),Pelect(i),PlSctCSec(i)

900 continue

 end do

 call Get\_Incident\_electron\_Dta\_Final0

 print \*, '-----------------------------------------------'

 print \*, ' Entered data: '

 print \*, '-----------------------------------------------'

 print \*, ' '

 print \*, 'Electron effective mass ',ElecEfMas,' kg'

 print \*, ' '

 print \*, 'Element ; Valency ; plasmon scatt. &

 cr-section'

 do i=1,n

 print\*, ' '

 if (i.eq.n) then

 if (Choice.eq.'B'.or.Choice.eq.'b') then

 print \*, ElemNames(i),' ',Pelect(i)

 print \*, ' '

 print \*, 'Plasmon energy ', PlPeaP/ECHARGe,' eV'

 print \*, ' '

 goto 920

 end if

 end if

 print \*, ElemNames(i),' ',Pelect(i),PlSctCSec(i),' m2 &

 & per ',ElemNames(i),' atom'

920 continue

 end do

 print \*, ' '

940 print\*, ' Is this OK ? Enter yes or no.'

 read(\*,\*) choix

945 if (choix.ne.'y'.or.choix.ne.'Y') then

 if (choix.eq.'N'.or.choix.eq.'n') then

 deallocate (ElemNames, STAT = DeAllocateStatus)

 deallocate ( Nk, STAT = DeAllocateStatus )

 deallocate ( Pelect, STAT = DeAllocateStatus )

 deallocate ( PlSctCSec, STAT = DeAllocateStatus )

 if (Aup.eq.0.0) goto 100

 goto 700

 end if

 else

 print \*, ' '

 print \*, ' Please enter yes or no.'

 goto 940

 end if

 If (Choice=='A'.or.Choice=='a') Then

 !! USING Tehnique 1.

! Get STAR, STAR is equal to the electron concentration

! divided by the atom concentration of the first element entered.

 STAR=0.0

 do i=1,n

 STAR=STAR+Pelect(i)\*(PlSctCSec(1)/PlSctCSec(i))

 end do

 Q=PlSctCSec(1)

!---------------------Get interval containing solution -----------------------!

 A2=2.e31 !!

 i=0

 SubType=13 !! For calling Elect\_Conc\_Dispers\_etc\_Eff\_M\_Known\_FULL\_Kc13

 AplausMin=5.e29 !! AplausMin\*10. is the initial value starting

 !! with when looking for the appropriate upper

 !! limit of the interval containing the

 !! actual electron concentration.

 call get\_interval\_A1\_A3\_Lt

 if (errKc) then

 write(\*,\*) 'Interruption, entered data cannot be analyzed'

 write(\*,\*) 'Cutoff vector cannot be calculated'

 write(\*,\*) ' OK ? Enter any character to continue'

 read(\*,\*) RPAorExCor

 goto 100

 else if (errThetaC) then

 write(\*,\*) 'Interruption, entered data cannot be analyzed'

 write(\*,\*) 'Cutoff angle cannot be calculated'

 write(\*,\*) ' OK ? Enter any character to continue'

 read(\*,\*) RPAorExCor

 goto 100

 end if

 if (Aup.eq.0.0) then

 choix='n'

 goto 945

 end if

 call Dichotoming

1200 continue

! The electron concentration has been obtained

 ElectConc=A2

! Work out the first plasmon peak position.

 PlPeaP=(HBAR\*(ElectConc)\*\*(1./2.)/ElecEfMas\*\*(1./2.)/EPSILO\*\*(1./2.))\*ECHARGe

 IF (RPAorExCor == 'k' .or. RPAorExCor == 'K') THEN

!! Alfa = Entered Alfa

 ELSE

 Alfa1 = 3./5.\*FEnerg/PlPeaP

! Work out Alfa.

 Alfa=Alfa1

 if (RPAorExCor == 'E' .or. RPAorExCor == 'e') then

 !! ExCor considered

 Alfa=Alfa\*(1.-(1./EFoverEp)\*(1./EFoverEp)/16.)

 else

 !! RPA considered

 end if

 END IF

! Work out the various absolute atom concentrations

 Nk(1)=ElectConc/STAR

 do i=2,n

 Nk(i)=Nk(1)\*PlSctCSec(1)/PlSctCSec(i)

 end do

! Get plasmon mean free path

 LambdaP=1.0/Nk(1)/PlSctCSec(1)

 if (Alfa .eq. 0.5) then

 KC = KF/2.\*(PlPeaP/FEnerg)

 else

 KC = KF/(2.\*Alfa-1.)\*(1.-DSQRT(1.-(2\*Alfa-1.)\*PlPeaP/FEnerg))

 end if

 ILLa=1 ! Now we can get the specimen thickness via Technique 1

 ILLb=0

 Else If (Choice=='B'.or.Choice=='b') Then

 !! USING Technique 2.

! Effective mass known, first plasmon peak position known and PlSctCSec(n) not known

! Work out the electron concentration

 ElectConc=(ElecEfMas/HBAR)\*(PlPeaP/ECHARGe)\*\*2.\*(EPSILO/HBAR)

! Work out the Fermi vector

 KF=(3.\*Pi\*Pi\*ElectConc)\*\*(1./3.)

! Get the Fermi energy

 FEnerg=(HBAR\*KF)/MASS0\*(HBAR\*KF)/2.

! Get EFoverEp

 EFoverEp=FEnerg/PlPeaP

! Get Alfa1

 IF (RPAorExCor == 'k' .or. RPAorExCor == 'K') THEN

 ! Value entered for alfa

 ELSE

 Alfa1 = 3./5.\*EFoverEp

! Get Alfa

 if (RPAorExCor == 'E' .or. RPAorExCor == 'e') then ! ExCor considered

 Alfa = Alfa1\*(1.-(1./(Alfa1/3.\*5.)/4.)\*\*2.)

 else ! RPA considered

 Alfa = Alfa1

 end if

 END IF

 if (Alfa .eq. 0.5) then

 KC = KF/2.\*(PlPeaP/FEnerg)

 else

 if (1.-(2\*Alfa-1.)\*PlPeaP/FEnerg .le. 0) then

 errKc = .true.

 write(\*,\*) ' There is error !!:'

 write(\*,\*) '(1.-(2\*Alfa-1.)\*PlPeaP/FEnerg) is negative'

 write(\*,\*) ' OK ? Enter any character to continue.'

 read(\*,\*) RPAorExCor

 goto 100

 end if

 KC = KF/(2.\*Alfa-1.)\*(1.-DSQRT(1.-(2\*Alfa-1.)\*PlPeaP/FEnerg))

 end if

! And third get Thetac:

 if ((KC/K0 - ThetaE)\*(KC/K0 + ThetaE) .le. 0) then

 errThetaC = .true.

 write(\*,\*) ' There is error !! I cannot calculate Thetac:'

 write(\*,\*) '(KC/K0 - ThetaE)\*(KC/K0 + ThetaE) is negative'

 write(\*,\*) ' OK ? Enter any character to continue.'

 read(\*,\*) RPAorExCor

 goto 100

 end if

 Thetac = DSQRT((KC/K0 - ThetaE)\*(KC/K0 + ThetaE))

! Get Theta E

 ThetaE=PlPeaP/(Gama\*MASS0\*ElecSpeed\*ElecSpeed)

! Get plasmon men free path

 LambdaP=Ar0/(Gama\*ThetaE)/DLOG(1.+(Thetac/ThetaE)\*(Thetac/ThetaE))\*2.

! Work out the various absolute atom concentrations relative to the detectable species.

 do i=1,n-1

 Nk(i)=1./LambdaP/PlSctCSec(i)

 end do

! Work out the contribution to the electron concentration due to the detectable

! elements: use X0 for that.

 X0=0.0

 do i=1,n-1

 X0=X0+Pelect(i)\*Nk(i)

 end do

! Work out the absolute atom concentration for the non-easily detectable species by

! subtracting the contributions of the detectable elements from the total electron

! concentration and dividing by the valency of the non-easily detectable element..

 Nk(n)=(ElectConc-X0)/Pelect(n)

! Work out the plasmon scattering cross section per nth atom species (i.e. that of the

! non-easily detectable element).

 PlSctCSec(n)=PlSctCSec(1)\*Nk(1)/Nk(n)

 ILLb=1 ! Now we can get the specimen thickness via Technique 2.

 ILLa=0

 Else

 End If

 print \*,' '

 print \*,' '

 if (ILLa.eq.1) print \*,' / Technique No 1 &

 was used /'

 if (ILLb.eq.1) print \*,' / Technique No 2 &

 was used /'

 print \*,' '; print \*,' '

 !--------------------------- Results are given hereafter. -----------------

 print \*, '!------------------------------------------------------&

 ----------------------!'

 if (RPAorExCor == 'r' .or. RPAorExCor == 'R') then

 write(\*,\*) '////////// Random Phase Approximation considered.&

 //////////////////////.'

 print \*, "!-------------------------------------------------------&

 ----------------------!"

 else if (RPAorExCor == 'e' .or. RPAorExCor == 'E') then

 write(\*,\*) '////////// Exchange plus Correlation considered.&

 //////////////////////.'

 print \*, ' '

 print \*, '!-------------------------------------------------------&

 ---------------------!'

 else if (RPAorExCor == 'k' .or. RPAorExCor == 'K') then

 write(\*,\*) '////////// Known dispersion coefficient considered.&

 ///////////////////.'

 print \*, '!--------------------------------------------------&

 ----------------------!'

 end if

 print \*, ' '

 print \*, ' Calculated physical quantities.'

 print \*, ' '

 print \*, '!-------------------------------------------------------&

 ---------------------!'

 print \*, ' '

 write(\*,2530)' Plasmon mean free path is: ',LambdaP\*1.e10,' &

 angs-1'

 print \*, '!------------------------------------------------------&

 ----------------------!'

 print \*, ' '

 write(\*,2520) ' Electron concentration is: ',ElectConc,' electron/m3'

 print \*, '!------------------------------------------------------&

 ----------------------!'

 print \*, ' '

 write(\*,2520) ' The Fermi vector is: ',KF,' 1/m'

 print \*, '!------------------------------------------------------&

 ----------------------!'

 print \*, ' '

 write(\*,2510) ' The Fermi energy is: ',FEnerg/1.602e-19,' eV'

 print \*, '!------------------------------------------------------&

 ----------------------!'

 print \*, ' '

 write(\*,2510) ' Theta E is: ',ThetaE\*1000.,' mrad'

 print \*, '!------------------------------------------------------&

 ----------------------!'

 print \*, ' '

 write(\*,2520) ' Plasmon cutoff vector (kc) ',KC,' 1/m'

 print \*, '!------------------------------------------------------&

 ----------------------!'

 print \*, ' '

 write(\*,2510) ' Plasmon cutoff angle (Theta c) ',Thetac\*1000.,' mrad'

 print \*, '!------------------------------------------------------&

 ----------------------!'

 if (Choice.eq.'A'.or.Choice.eq.'a') then

 print \*, '!------------------------------------------------------&

 ----------------------!'

 print \*, ' '

 write(\*,2510) ' First plasmon peak position is: ',PlPeaP/1.602D-19,' eV'

 print \*, '!------------------------------------------------------&

 ----------------------!'

 print \*, ' '

 else if (Choice.eq.'B'.or.Choice.eq.'b') then

 print \*, '!------------------------------------------------------&

 ----------------------!'

 print \*, ' '

 write(\*,2540) ' Plasmon Scatt. Cros Sect for ',ElemNames(n),'is: '&

 ,PlSctCSec(n),' m2 per ',ElemNames(n),'atom'

 print \*, '!------------------------------------------------------&

 ----------------------!'

 print \*, ' '

 else if (Choice.eq.'C'.or.Choice.eq.'c') then

 print \*, '!------------------------------------------------------&

 ----------------------!'

 print \*, ' '

 end if

 print \*, ' Element Absolute atom concentration &

 (in at/m3).'

 print \*, '!------------------------------------------------------&

 ----------------------!'

 do i=1,n

 print \*,' '

 write(\*,2550) ' ', ElemNames(i),' ',Nk(i)

 if (Nk(i).lt.0.0) print \*,' This is negative, maybe this should be zero?'

 end do

 if (n>=2) then

 print \*, '!------------------------------------------------------&

 ----------------------!'

 print \*, ' '

 print \*, ' The relative content ratios are:'

 print \*, ''

 do i=2,n

 write(\*,2560)' ',ElemNames(i),'/',&

 ElemNames(1),': ',Nk(i)/Nk(1)

 end do

 end if

 print \*, ' '

 print \*, ' '

 print \*, '!-------------------------------------------------------&

 ----------------------!'

 print \*, ' '

 if (RPAorExCor == 'r' .or. RPAorExCor == 'R') then

 write(\*,2570) ' Alfa (RPA) = (3/5)EF/Ep = ',Alfa1

 print \*, '!-------------------------------------------------------&

 ----------------------!'

 else if (RPAorExCor == 'e' .or. RPAorExCor == 'E') then

 print \*, ' '

 write(\*,2570) ' Alfa1 = (3/5)EF/Ep = ',Alfa1

 print \*, '!-------------------------------------------------------&

 ----------------------!'

 print \*, ' '

 write(\*,2570) ' Alfa = Alfa1(1-Ep/4EF)\*\*2) = ',Alfa

 print \*, '!-------------------------------------------------------&

 ----------------------!'

 else if (RPAorExCor == 'k' .or. RPAorExCor == 'K') then

 print \*, ' '

 write(\*,2570) ' Entered Alfa = ',Alfa

 print \*, '!-------------------------------------------------------&

 ----------------------!'

 end if

2570 format(A,f15.5)

 print \*, ' '; print \*, ' '

 if (Choice.eq.'C'.or.Choice.eq.'c') ILLa=1

1400 print \*, '!------------------------------------------------------&

 ----------------------!'

 print \*, ' !-------------------------------------&

 --------------!'

 print \*, ' // The entered data is the following&

 : //'

 if (Choice.eq.'B'.or.Choice.eq.'b'.or.&

 Choice.eq.'C'.or.Choice.eq.'c') then

 print \*, '!------------------------------------------------------&

 ----------------------!'

 print \*, ' '

 write(\*,2100)'First plasmon peak energy: ',PlPeaP/1.602E-19,' eV'

 end if

 print \*, ' !-------------------------------------&

 --------------!'

 print \*, '!------------------------------------------------------&

 ----------------------!'

 print \*, ' '

 write(\*,2520)' Electron effective mass is: ',ElecEfMas,' kg'

 print \*, ' '

 print \*, '!------------------------------------------------------&

 ----------------------!'

 print \*, ' '

 write(\*,2200) 'Accelarating voltage: ',V0/1000.,' kV'

 print \*, '!------------------------------------------------------&

 ----------------------!'

 print \*, ' '

 print \*, 'Element; Valency; plasmon scatt. &

 cross-section'

 do i=1,n

 print \*, '!------------------------------------------------------&

 ----------------------!'

 print\*, ' '

 if (i.eq.n) then

 if (Choice.eq.'B'.or.Choice.eq.'b') then

 write(\*,2400) ElemNames(i),' ',Pelect(i)

 print \*, ' '

 print \*, '!------------------------------------------------------&

 ----------------------!'

 print \*, 'Plasmon enargy ',PlPeaP/ECHARGe,' eV'

 print \*, ' '

 goto 1240

 end if

 end if

 write(\*,2300) ElemNames(i),' ',Pelect(i),' ',&

 PlSctCSec(i),' m2 per ',ElemNames(i),' atom'

1240 continue

 end do

 print\*, ' ' ; print\*, ' '

 print\*, ' ' ; print\*, ' '

 do

 print \*, ' Enter Q to quit or'

 print \*,' any other key to return to the menu'

 write(\*, '(A)', ADVANCE = "NO")' => '

 read (\*,\*,err=100) choix

 if (choix=='Q'.or.choix=='q') stop

 Choice='~'

 goto 100

 end do

2100 format(1x, A,F8.3,A)

2200 format(1x, A,ES12.3,A)

2300 format(3x,A2,A,F10.1,A,ES12.3,A,A2,A)

2400 format(3x,A2,A,F10.1)

2510 format(A,F14.5,A)

2520 format(A,ES15.5,A)

2530 format(A,F10.2,A)

2540 format(A,A2,A,ES14.5,A,A2,A)

2550 format(A,A2,A,ES14.5)

2560 format(A,A2,A,A2,A,F10.5)

 print \*, ' '

40000 continue

 if (ChoiceP.eq.'B'.or.ChoiceP.eq.'b') then

 file = 'Resutlts\_AAC\_etc\_tec\_2.txt'

 else if (ChoiceP.eq.'A'.or.ChoiceP.eq.'a') then

 file = 'Resutlts\_AAC\_etc\_tec\_1.txt'

 end if

 OPEN (7,STATUS='new',File=file,IOSTAT = Ierr)

 IF(Ierr .ne. 0)THEN

40010 if (ChoiceP.eq.'B'.or.ChoiceP.eq.'b') then

 write(\*,\*) ' File: Resutlts\_AAC\_etc\_2.txt exists, filing as Resutlts\_AAC\_etc\_04.txt.'

 file = 'Resutlts\_AAC\_etc\_tec\_4.txt'

 else if (ChoiceP.eq.'A'.or.ChoiceP.eq.'a') then

 write(\*,\*) ' File: Resutlts\_AAC\_etc\_1.txt exists, filing as Resutlts\_AAC\_etc\_03.txt.'

 file = 'Resutlts\_AAC\_etc\_tec\_3.txt'

 end if

 write(\*,\*) ' OK ? Yes/No.'

 read(\*,\*) RPAorExCor

 if (RPAorExCor == 'n' .or. RPAorExCor == 'N') then

 write(\*,\*) ' Enter a file name.'

 read(\*,\*) file

 file = trim(file)//'.txt'

 else if (RPAorExCor == 'y' .or. RPAorExCor == 'Y') then

 else

 goto 40010

 end if

 OPEN(7, STATUS = 'unknown', FILE = file, IOSTAT = Ierr)

 REWIND(7)

 ENDIF

 write(7,\*)' '

 write(7,\*) ' !-------------------------------------------&

 &--------------!'

 write(7,\*) ' // The entered data is the following&

 & : //'

 write(7,\*) ' !-------------------------------------------&

 &--------------!'

 write(7,\*) ' '

 write(7,\*) '!------------------------------------------------------&

 &----------------------!'

 write(7,\*) ' '

 if (RPAorExCorP == 'r' .or. RPAorExCorP == 'R') then

 write(7,2570) ' Random phase approximation used.'

 write(7,\*)' '

 write(7,\*) ' !-------------------------------------------&

 &--------------!'

 else if (RPAorExCorP == 'e' .or. RPAorExCorP == 'E') then

 write(7,\*)' '

 write(7,2570) ' Exchange and electron correlation used.'

 write(7,\*) ' !-------------------------------------------&

 &--------------!'

 else if (RPAorExCorP == 'k' .or. RPAorExCorP == 'K') then

 write(7,\*)' '

 write(7,2570) ' Constant value for Alfa entered.'

 write(7,\*) ' !-------------------------------------------&

 &--------------!'

 write(7,\*)' '

 end if

 write(7,2520)' Electron effective mass is: ',ElecEfMas,' kg'

 if (ChoiceP.eq.'B'.or.ChoiceP.eq.'b'.or. &

 & Choice.eq.'C'.or.Choice.eq.'c') then

 write(7,\*) '!------------------------------------------------------&

 &----------------------!'

 write(7,\*) ' '

 write(7,2100)' First plasmon peak energy: ',PlPeaP/1.602E-19,' eV'

 end if

 write(7,\*) '!------------------------------------------------------&

 &----------------------!'

 write(7,\*) ' '

 write(7,2200) 'Accelarating voltage: ',V0/1000.,' kV'

 write(7,\*) '!------------------------------------------------------&

 &----------------------!'

 write(7,\*) ' '

 write(7,\*) 'Element; Valency; plasmon scatt.cross-&

 &section'

 do i=1,n

 write(7,\*) '!------------------------------------------------------&

 &----------------------!'

 write(7,\*) ' '

 if (i.eq.n) then

 if (ChoiceP.eq.'B'.or.ChoiceP.eq.'b') then

 write(7,2400) ElemNames(i),' ',Pelect(i)

 goto 1340

 end if

 end if

 write(7,2300) ElemNames(i),' ',Pelect(i),' ', &

 & PlSctCSec(i),' m2 per ',ElemNames(i),' atom'

1340 continue

 end do

 write(7,\*) ' '

 write(7,\*) ' Calculated physical quantities.'

 write(7,\*) ' '

 write(7,\*) '!-------------------------------------------------------&

 &---------------------!'

 write(7,\*) ' '

 write(7,2530)' Plasmon mean free path is: ',LambdaP\*1.e10,' &

 & angs-1'

 write(7,\*) '!------------------------------------------------------&

 & ----------------------!'

 write(7,\*) ' '

 write(7,2520)' Electron concentration is: ',ElectConc,' electron/m3'

 write(7,\*) '!------------------------------------------------------&

 &----------------------!'

 write(7,\*) ' '

 write(7,2520) ' The Fermi vector is: ',KF,' radians/m'

 write(7,\*) '!------------------------------------------------------&

 &----------------------!'

 write(7,\*) ' '

 write(7,2510) ' The Fermi energy is: ',FEnerg/1.602e-19,' eV'

 write(7,\*) '!------------------------------------------------------&

 &----------------------!'

 write(7,\*) ' '

 write(7,2510) ' Theta E is: ',ThetaE\*1000.,' mrad'

 write(7,\*) '!------------------------------------------------------&

 &----------------------!'

 write(7,\*) ' '

 write(7,2510) ' Plasmon cutoff angle (Theta c) ',Thetac\*1000.,' mrad'

 write(7,\*) ' '

 write(7,\*) '!------------------------------------------------------&

 &----------------------!'

 write(7,\*) ' '

 write(7,2520) ' Critical wavevector (kc) ',KC,' radians/m'

 if (ChoiceP.eq.'A'.or.ChoiceP.eq.'a') then

 write(7,\*) '!------------------------------------------------------&

 &----------------------!'

 write(7,\*) ' '

 write(7,2510) ' First plasmon peak position : ',PlPeaP/1.602E-19,' eV'

 write(7,\*) '!------------------------------------------------------&

 &----------------------!'

 write(7,\*) ' '

 else if (ChoiceP.eq.'B'.or.ChoiceP.eq.'b') then

 write(7,\*) '!------------------------------------------------------&

 &----------------------!'

 write(7,\*) ' '

 write(7,2540) ' Plasmon Scatt. Cros Sect for ',ElemNames(n),'is:'&

 &,PlSctCSec(n),' m2 per ',ElemNames(n),'atom'

 write(7,\*) '!------------------------------------------------------&

 &----------------------!'

 write(7,\*) ' '

 else if (ChoiceP.eq.'C'.or.ChoiceP.eq.'c') then

 write(7,\*) '!------------------------------------------------------&

 &----------------------!'

 write(7,\*) ' '

 end if

 write(7,\*) ' Element Absolute atom concentration &

 & (in at/m3).'

 write(7,\*) '!------------------------------------------------------&

 &----------------------!'

 do i=1,n

 write(7,\*)' '

 write(7,2550) ' ', ElemNames(i),' ',Nk(i)

 if (Nk(i).lt.0.0) write(7,\*)' This is negative, maybe this should be zero?'

 end do

 if (n>=2) then

 write(7,\*) '!------------------------------------------------------&

 &----------------------!'

 write(7,\*) ' '

 write(7,\*) ' The relative content ratios are:'

 write(7,\*) ''

 do i=2,n

 write(7,2560)' ',ElemNames(i),'/', &

 &ElemNames(1),': ',Nk(i)/Nk(1)

 end do

 end if

 write(7,\*) ' '

 if (RPAorExCorP == 'r' .or. RPAorExCorP == 'R') then

 write(7,2570) ' Alfa (RPA) = (3/5)EF/Ep = ',Alfa1

 write(7,\*) ' !-------------------------------------------&

 &--------------!'

 else if (RPAorExCorP == 'e' .or. RPAorExCorP == 'E') then

 write(7,\*)' '

 write(7,2570) ' Alfa1 = (3/5)EF/Ep = ',Alfa1

 write(7,\*) ' !-------------------------------------------&

 &--------------!'

 write(7,\*)' '

 write(7,2570) ' Alfa = Alfa1(1-Ep/4EF)\*\*2) = ',Alfa

 write(7,\*) ' !-------------------------------------------&

 &--------------!'

 else if (RPAorExCorP == 'k' .or. RPAorExCorP == 'K') then

 write(7,\*)' '

 write(7,2570) ' Entered Alfa value = ',Alfa

 write(7,\*) ' !-------------------------------------------&

 &--------------!'

 end if

 close(7)

 goto 100

 end program Absol\_Atom\_Concent\_Determ\_Need\_Eff\_Mass\_FULL