AERE-R 9170

AERE - R 9170 C

D

United Kingdom Atomic Energy Authority

HARWELL

SIMON
A Fortran Monte-Carlo computer program to simulate a time-of-flight neutron diffractometer using a PDP-11 computer

J. H. Clarke
Materials Physics Division
AERE Harwell, Oxfordshire
October 1978

PRICE £2.00 net from H.M. Stationery Office

C8

AERE R-9170

SIMON

A FORTRAN MONTE-CARLO COMPUTER PROGRAM TO SIMULATE A TIME-OF-FLIGHT NEUTRON DIFFRACTOMETER USING A PDP-11 COMPUTER

J. H. Clarke*

ABSTRACT

A computer code has been written in FORTRAN to simulate a time-of-flight diffractometer operating on a pulsed neutron source driven by an electron or proton accelerator. The code can be modified to simulate a time-of-flight diffractometer on a reactor with an incident neutron beam pulsed by a chopper.

The program simulates the fast neutron emission from the accelerator target, the time spread of neutrons emitted from the moderator, the geometry of the sample and diffractometer and the efficiency of the neutron detector.

The final neutron intensity is corrected for the effect of solid angle. A figure-of-merit may thus be evaluated when varying the geometry of the experiment to optimize momentum-transfer resolution and intensity.

The Q-resolution of the Harwell Total-Scattering and Back-Scattering spectrometers has been simulated and compared with experimental results. The effect of varying certain instrument parameters in relation to the design of time-of-flight spectrometers is discussed.

*Physics Laboratories, University of Kent at Canterbury

Materials Physics Division,
AERE HARWELL

October 1978

HL78/3874 (C8)

ADAB

CONTENTS

ISBN 0-70-580640-5

CONTENTS (Contd.)

TABLES

ILLUSTRATIONS

1. Introduction

A time-of-flight neutron diffractometer may be used to study the molecular structure of amorphous materials (glasses, liquids and gases) or may be used as a powder diffractometer for the study of crystallographic structures. Several such diffractometers have been constructed and described in the literature[1,2]. Figure 1 shows a schematic diagram of the total-scattering spectrometer on the AERE Harwell electron linac for the study of glasses and liquids.

Pulses of electrons are incident on the uranium target producing fast neutrons. These are slowed down to energies in the region of 1 eV suitable for diffraction experiments by the polyethylene moderator. The energy distribution of the neutrons is now that of a Maxwellian with a higher energy epithermal component[3]. These neutrons are scattered by the sample into the detector and recorded as an event occurring at a certain time-of-flight since the firing of the electron pulse. Thus knowledge of the flight path travelled, the angle through which it has been scattered and its time-of-flight provide the energy of the neutron and its momentum transfer. The intensity distribution of neutrons as a function of this momentum transfer scattered from a sample can be used to determine the structure of the sample.

The purpose of this program is to provide a simple method of designing new diffractometers and experiments. It is based on a simpler geometrical resolution simulation algorithm first proposed by Dr. J. M. Carpenter of Argonne National Laboratory. The dimensions, orientations and positions of the moderator, sample and detector can all be varied so affecting the momentum-transfer resolution and the detected neutron intensity.

The program enables the instrument designer to achieve the optimum balance between resolution and intensity by varying the parameters of the moderator, detectors and diffractometer. Similarly the experimental neutron scientist may obtain a similar balance by adjusting his sample size and orientation and by varying the height of the detectors.

2. Calculations used in the program

We shall follow a neutron through its path or history from its creation in the uranium target to its absorption in the detector. The symbols which occur in brackets are the names of the variables as used in the computer code. At each stage the assumptions made and the relevant formulae used will be stated. Because this is a Monte-Carlo simulation the computer program will cycle over these operations many times (NHIST). At each interaction of the neutron with the moderator, sample or detector a random number is generated to decide on the value of some physical quantity. The final time-of-flight of the neutron is accumulated into a scoring array (ISCORE) which is a distribution exactly as would be recorded in the real neutron experiment. In the following sections the symbol R̲ is used to represent a random number between 0 and 1.0.

(a) **The emission time from the fast neutron target (TSRC)**

Figure 2 shows the shape and duration of the electron or proton pulse from the accelerator incident upon the uranium target. The pulse is assumed to be rectangular and the centre is taken as the origin of the time-of-flight scale. The width (TESRCE μsecs) is an input parameter in the program, so that the intrinsic time 'jitter' of the fast neutron source which is calculated in subroutine CALC of the program is

$$TSRC = (\underline{R} - \tfrac{1}{2})\,TESRCE \qquad (1)$$

(b) **The emission time from the moderator (TMOD)** ✓

The neutron density in an infinite moderator at energies large compared with the chemical binding energy and temperature is

$$\rho(E,t) = \frac{1}{2E}\,\frac{y^{2/\gamma}\,e^{-y}}{\Gamma(\frac{2}{\gamma})} \qquad (2)$$

where

$$y = \frac{\xi \Sigma_s v t}{\gamma} \qquad (3)$$

and Σ_s is the macroscopic scattering cross section, ξ is the mean logarithmic energy change per collision, v is the velocity, t is the time and γ is a function of the mass number of the moderator (see §6a in ref. [4]).

The χ^2 distribution for a variable x with n degrees of freedom is

$$\chi^2(n,x) = \frac{x^{\frac{n}{2}-1}\,e^{-\frac{x}{2}}}{2^{\frac{n}{2}}\,\Gamma(\frac{n}{2})} \qquad (4)$$

Making the substitutions,

$$x = 2y \qquad (5)$$

$$n = 2 + 4/\gamma \qquad (6)$$

and

$$\Gamma(n+1) = n\Gamma(n) \qquad (7)$$

✓ This section describes the simulation of the moderator emission time in the epithermal region only. A parameterization of a Maxwellian pulse width could be inserted into the computer program subroutine CALC at the point where TMOD is calculated.

equation (2) becomes:

$$\rho(E,t) = \chi^2(n,x)\left(\frac{n-2}{2}\right) \quad (8)$$

where

$$t = \frac{\gamma x}{2\xi\Sigma_s v} \quad (9)$$

In the computer code the moderator is defined by two parameters, the mean free path (PATH) and the number of degrees of freedom (FREEDM). These are obtained from table 7 of reference [4] for an infinite medium.

ie $$\text{PATH} = \frac{\gamma}{\xi\Sigma_s} \quad \text{and} \quad \text{FREEDM} = 2 + \frac{4}{\gamma} \quad (10)$$

so that since the mean and variance of the χ^2 -distribution are n and 2n,

$$\bar{t} = \frac{\text{FREEDM*PATH}}{2 \times \text{VELOCITY}} \quad (11)$$

and

$$\sigma(t) = \frac{\text{PATH}}{2 \times \text{VELOCITY}} \sqrt{2 \times \text{FREEDM}} \quad (12)$$

and

$$t_{MOD} = \frac{\text{PATH}}{2 \times \text{VELOCITY}} * \text{CHISQ}(\underline{R}, \text{FREEDM}) \quad (13)$$

is the moderator emission time where CHISQ(R,FREEDM) is the inverse χ^2 value obtained from a random number $0 \leqslant \underline{R} \leqslant 1$.

At this stage we define a right-handed cartesian coordinate system for the diffractometer as shown in figure 3.

(i) The origin is taken to be the centre of the moderator.

(ii) The x-axis is the direction of the unscattered or incident neutron beam.

(iii) The y-axis lies in the scattering plane (normally horizontal).

(iv) The z-axis is perpendicular to the scattering plane (normally vertical).

(v) The moderator, sample and detector orientation angles (THETAM, THETAS and THETAD) are positive in the anticlockwise direction ($\underline{x} \rightarrow \underline{y}$) as is the scattering angle (THETAØ).

(vi) The moderator and sample angles are zero when parallel to the $\underline{y}$-axis, the scattering angle is zero when parallel to the $\underline{x}$-axis and the detector angle is zero when it is perpendicular to the scattered beam.

(c) **The emission point from the moderator (XNM, YNM, ZMP)**

The moderator is assumed to be a sheet of zero thickness whose width (YM m) height (ZM m) and orientation (THETAM degrees) are input parameters of the program.

In subroutine CALC the program produces the random variables YMP and ZMP and thus the coordinates of the emission point on the surface of the moderator.

$$\text{YMP} = (\underline{\text{R}} - \tfrac{1}{2})\,\text{YM} \quad (14a)$$

$$\text{XNM} = -\text{YMP}\sin(\text{THETAM}) \quad (14b)$$

$$\text{YNM} = \text{YMP}\cos(\text{THETAM}) \quad (14c)$$

$$\text{ZMP} = (\underline{\text{R}} - \tfrac{1}{2})\,\text{ZM} \quad (14d)$$

The effect of the thickness of the moderator is in the $\underline{x}$-direction on the neutron flight path is included in the calculation of TMOD §2b. However the change in scattering angle that would be caused by a finite thickness moderator is neglected.

(d) **The scattering point in the sample (XNS, YNS, ZSP)**

The sample may be a flat plate whose width (YS m), height (ZS m), thickness (TS m), orientation (THETAS degrees) and distance from the moderator (RMSØ m) are input parameters to the program.

Alternatively the sample may be a cylinder whose diameter (YS m) and height (ZS m) are input parameters. RMSØ is referred to as the initial or incident flight path.

Two mutually exclusive logical variables (ROD and PLATE) control the code when calculating sample coordinates.

For a plate sample the random variables YSP, TSP and ZSP are produced and thus the scattering point coordinates.

$$\text{YSP} = (\underline{\text{R}} - \tfrac{1}{2})\,\text{YS} \quad (15a)$$

$$\text{TSP} = (\underline{\text{R}} - \tfrac{1}{2})\,\text{TS} \quad (15b)$$

so that

$$\text{XNS} = \text{RMSØ} - \text{YSP}\sin(\text{THETAS}) + \text{TSP}\cos(\text{THETAS}) \quad (15c)$$

$$\text{YNS} = \text{YSP}\cos(\text{THETAS}) + \text{TSP}\sin(\text{THETAS}) \quad (15d)$$

$$\text{ZSP} = (\underline{\text{R}} - \tfrac{1}{2})\,\text{ZS} \quad (15e)$$

For a cylindrical sample the random variables YSP, THP and ZSP are produced and thus the scattering point coordinates are,

$$THP = 2\pi \underline{R} \qquad (15f)$$

$$XNS = RMSØ - YSP \sin(THP) \qquad (15g)$$

$$YNS = YSP \cos(THP) \qquad (15h)$$

$$ZSP = (\underline{R} - ½) ZS \qquad (15i)$$

(e) **The stopping point in the detector (XND, YND, ZDP)**

The detector is a rectangular plate whose width (YD m), height (ZD m), thickness (TD m), orientation (THETAD degrees), scattering angle (THETAØ degrees) and distance from the sample (RSDØ m) are input parameters to the program.

The orientation (THETAD) is also referred to as the Q-focussing angle and the distance (RSDØ) is known as the scattered, secondary or final flight path. An approximation is made by considering the rectangular plate detector as representing a single cylindrical detector or a bank of cylindrical detectors. The thickness of the detector should be chosen to represent the equivalent slab thickness of a cylinder

$$TD = \pi/2 \text{ x radius of cylindrical detector} \qquad (16)$$

If the absorption coefficient of the detector at 1Å is μ (EFFD) then the efficiency of the detector at a wavelength λ (WAVE) is

$$\epsilon(\lambda) = 1 - \exp(-\mu\lambda TD) \qquad (17)$$

The assumption is made that the probability of the neutron reaching a point $\bar{t}$ is $e^{-\mu\lambda\bar{t}}$. By integrating this probability and normalizing by the area of the function a unit probability function is obtained.

$$P(t) = \frac{1-e^{-\mu\lambda t}}{1-e^{-\mu\lambda t_o}} \qquad O \leqslant P(t) \leqslant 1 \qquad (18)$$

where t_o is the thickness of the detector.

In subroutine CALC the program produces the random variables YDP, ZDP and TDP and thus the coordinates of the absorption point within the detector. In using equation (18) it is necessary to know the effective thickness of the detector for this neutron path ie TD/cos(THETAD). The approximation is made in choosing this limit that the neutron is scattered from the centre of the sample to the centre of the detector. This is a similar approximation to that made in § 2c concerning the moderator thickness.

In subroutine CALC the program produces the random variables YDP, ZDP and TDP and thus the coordinates of the absorption point in the detector,

$$YDP = (\underline{R} - ½) YD \tag{19a}$$

$$TDP = \frac{\log_e}{(EFFD \times WAVE)} \left[1 - \underline{R} \left(1 - \exp\left[- \frac{EFFD \times WAVE \times TD}{\cos(THETAD)} \right] \right) \right] \tag{19b}$$

The coordinates in the detector thus become

$$XND = RMSØ + RSDØ \cos(THETAØ) - YDP \sin(THDPØ) + TDP \cos(THDPØ) \tag{19c}$$

$$YND = RSDØ + YDP \cos(THDPØ) - TDP \sin(THDPØ) \tag{19d}$$

$$ZDP = (\underline{R} - ½) ZD \tag{19e}$$

where

$$THDPØ = THETAØ + THETAD \tag{19f}$$

For scattering angles below 90° the aperture of the detector may be chosen as a section of a Debye-Scherrer cone to maintain the angular resolution and the solid angle of the detector. The computer code is controlled by a logical variable DSCONE when calculating the detector variables. Figure 4 shows the detector aperture at a scattering angle (THETAØ degrees) or (THETRØ radians), the radius of the aperture is given by

$$RCONE = RSDØ \times THETRØ \tag{19g}$$

The approximation is made in the code, that when Debye-Scherrer cones are specified the focussing angle (THETAD) is zero. This is not unreasonable as Q-focussed banks are used at backward scattering angles where the Debye-Scherrer cones are unnecessary. Different random variables are used in this case, referring to figure 4 we have

$$THETAC = 2 \sin^{-1} [ZD/(2 \times RCONE)] \tag{19h}$$

and thus

$$XND = RMSØ + RSDØ \cos(THETAØ) - YDP \sin(THETAØ) + TDP \cos(THETAØ) \tag{19c}$$

$$YND = RSDØ + YDP \cos(THETAØ) - TDP \sin(THETAØ) \tag{19d}$$

$$ZDP = (\underline{R} - ½) THETAC \times ZD \tag{19i}$$

(f) The time-of-flight of the neutron

The Q-value (QØ) is specified as part of the input data to the program

$$Q = \frac{4\pi \sin\theta/2}{\lambda} \quad \text{where } \theta = THETAØ \tag{20}$$

or

$$Q = \frac{4\pi \sin\theta/2 \times 252.7 \times \text{Total-flight-path}}{\text{Time-of-flight}} \tag{21}$$

and from this equation (20) the incident wavelength λ (WAVE) is derived.

The scattering from the sample is a delta function in momentum transfer $\delta(\hbar Q)$ so that the Q-value resolution as simulated by the code is due solely to the source and moderator time dependence and to the geometry of the diffractometer. From the input parameters there are derived three variables,

(i) QØ the Q-value;

(ii) WAVE the wavelength of the neutron;

(iii) TØ = WAVE × 252.7 × (RMSØ + RSDØ)

the time-of-travel of the central neutron ie a path through the centres of the moderator, sample and detector.

The three sets of coordinates produced in § 2c-§ 2e give the total flight path and the angle of scatter by application of the cosine rule;

$$\text{RMS} = \sqrt{(\text{XNS-XNM})^2 + (\text{YNS-YNM})^2 + (\text{ZSP-ZMP})^2} \tag{22a}$$

is the moderator to sample distance,

$$\text{RSD} = \sqrt{(\text{XND-XNS})^2 + (\text{YND-YNS})^2 + (\text{ZDP-ZSP})^2} \tag{22b}$$

$$\text{RMD} = \sqrt{(\text{XND-XNM})^2 + (\text{YND-YNM})^2 + (\text{ZDP-ZMP})^2} \tag{22c}$$

are the sample to detector and moderator to detector distances.

$$\text{RMSD} = \text{RMS} + \text{RSD} \tag{22d}$$

is the total flight path.

$$\cos\theta_t = \frac{\text{RMS}^2 + \text{RSD}^2 - \text{RMD}^2}{2\text{RMS} \times \text{RSD}} \tag{22e}$$

is the cosine of the scattering angle, so that

$$\sin\frac{\theta_t}{2} = \text{SINTHT} = \sqrt{\frac{1 + \cos\theta_t}{2}} \tag{22f}$$

As the scattering is a delta function in Q the time-of-travel from the moderator surface to the detector is given by,

$$t = 4\pi \sin\frac{\theta_t}{2} \; \frac{252.7 \times \mathrm{RMSD}}{Q\emptyset} \tag{23}$$

and thus the velocity is

$$\frac{Q\emptyset}{4\pi \sin\frac{\theta_t}{2} \times 252.7} \tag{24}$$

This velocity can be used to obtain a value for TMOD from the expressions discussed in § 2b. Thus the final time-of-flight of the neutron is

$$t_f = \left(4\pi \sin\frac{\theta_t}{2} \; \frac{252.7 \times \mathrm{RMSD}}{Q\emptyset}\right) + \mathrm{TSRC} + \mathrm{TMOD} \tag{25}$$

The approximation used in deriving TMOD in § 2b is that the velocity is calculated using the mean emission time. This is because it is not possible to calculate the true moderator emission time without knowledge of the velocity which itself depends upon the moderator emission time.

This cyclic argument can be used as a basis for iterating the calculation and obtaining a more accurate value of TMOD.

(g) **The scoring of the event**

The computer program at this stage adds the neutron into the scoring array (ISCORE). The array has 100 elements (NTRANG) and the central neutron defined in § 2f(iii) is positioned in the centre of this array, so that if the time channel width is DT μsecs, the channel number of the neutron in equation (25) is

$$\mathrm{NT} = (t_f/\mathrm{DT}) - (\mathrm{T}\emptyset/\mathrm{DT}) + (\mathrm{NTRANG}/2) \tag{26}$$

At the beginning of the program the subroutine PRELIM sets two parameters NTMIN to (TØ/DT) + 2 and NTMAX to 1. The scoring array ISCORE is initially zero and each time NT is greater than NTMAX or less than NTMIN these two parameters are given new values. If however NT becomes negative or greater than NTRANG the code enters a subroutine called HALT which doubles the width of the time-of-flight channel and restarts the program.

When the requested number of neutron histories have occurred (NHIST) the program enters the subroutine OUTDAT. This routine calculates the zeroth, first and second moments of the array ISCORE which is the time-of-flight Q-value resolution function. The zeroth moment or area is

$$S\varnothing = \sum_{NTMIN}^{NTMAX} ISCORE(NT) \qquad (27)$$

$$S1 = \sum_{NTMIN}^{NTMAX} NT \times ISCORE(NT) \qquad (28)$$

is the first moment, so that the mean time-of-flight after correcting for the change in the origin of ISCORE in equation (26) is

$$TMEAN = \left(\frac{S1}{S\varnothing} - \frac{NTRANG}{2} \right) DT + T\varnothing \qquad (29)$$

the second moment is

$$S2 = \sum_{NTMIN}^{NTMAX} NT^2 \times ISCORE(NT) \qquad (30)$$

so that the standard deviation σ (SIGMAT) is

$$SIGMAT = DT \times \sqrt{\frac{S2}{S\varnothing} - \left(\frac{S1}{S\varnothing} \right)^2} \qquad (31)$$

(h) **The calculation of the intensity correction terms**

The calculations used in this program is of the form known as a 'forced' Monte Carlo code in that every random event is made to appear in the final time-of-flight distribution. This procedure was chosen in order to make the most efficient use of a small computer.

While varying experimental parameters such as flight paths, detector and moderator sizes the solid angles of the spectrometer and consequently the simulated intensity should reflect these variations and enable a figure-of-merit to be evaluated for different experimental arrangements. Thus the improvement in resolution which can be obtained by for example increasing the flight paths can be balanced against the subsequent reduction in intensity.

The subroutine RSCALE corrects for the effects of solid angle by applying a scale factor at the end of the Monte Carlo calculation subroutine CALC. This correction is applied once to the final time-of-flight distribution. The correction factor is

$$SCALER = \frac{A_m}{L_o^2} \times A_s \times N_s \times \frac{A_d}{L_s^2} \times \frac{1}{\Delta t} \qquad (32)$$

where A_m, A_S and A_d are the areas of the moderator, sample and detector, L_0 and L_S are the initial and secondary flight paths, N_S is the number of scattering centres per unit area of sample and Δt is the time channel width.

Thus, the final intensity simulated by the program will be proportional to the sizes of the moderator, sample and detector and to the thickness of the sample. The intensity will be corrected for the time channel width but not for the effect of the incident spectrum shape. Thus all comparisons between intensity and resolution should either be made at the same incident neutron energy or should include a spectrum shape parameter.

The individual terms are as follows,

$$\text{AREAMS} = \text{ZM*YM*COS(THETAM)} \qquad (33)$$

is the area of the moderator viewed by the sample for a plate sample. The area of sample facing the moderator is

$$\text{AREASM} = \text{ZS*YS*COS(THETAS)} \qquad (34)$$

and for a cylindrical sample it is

$$\text{AREASM} = \text{ZS*YS} \qquad (35)$$

The number of scattering centres per unit area as for a plate sample is proportional to TS the thickness. For a cylindrical sample it is proportional to the volume of the sample divided by the area.

$$N_S = \frac{\pi\text{*YS}^2\text{*ZS}}{\text{4*(YS*ZS)}} = \frac{\pi\text{*YS}}{4} \qquad (36)$$

The area of the detector viewing the sample is

$$\text{AREADS} = \text{ZD*YD*COS(THETAD)}$$

for a rectangular detector. For a detector defined by a Debye-Scherrer cone aperture as discussed in § 2e

$$\text{AREADS} = \left[\text{(RCONE + 0.5*YS)}^2 - \text{(RCONE - 0.5*YS)}^2 \right] \frac{\text{*THETA*}\ \pi}{2\pi} \qquad (37)$$

$$\text{AREADS} = \text{RCONE*YS*THETAC} \qquad (38)$$

where RCONE and THETAC are defined in § 2e.

Thus the correction term in equation (32) is given by,

$$SCALER = \frac{AREASM*AREAMS*AREADS*N_S}{RMSØ^2*RSDØ^2*DT} \qquad (39)$$

and is the scaling factor proportional to the geometrical sizes and shape of the spectrometer but is independent of the incident neutron spectrum and flux.

3. The path of the computer program

Figure 5 shows a schematic flow-chart of the program. The code is overlaid in the computer to use a small amount of core. The path of the program is as follows:

(a) The main program initialises all the parameters to the case of the Total Scattering Spectrometer via the BLOCKDATA routine.

(b) The parameters are displayed one-by-one and can be changed through the keyboard and the routine READAT and its subsidiaries.

(c) The routine LPDATA then lists all the instrumental parameters on the lineprinter.

(d) The routine PRELIM sets values for all the working variables of the program.

(e) If the display option is in use then a plan is drawn of the spectrometer geometry by the routine DRWDAT and its subsidiaries.

(f) The main calculation routine CALC performs the Monte Carlo simulation via the routines MODTR, SAMPLT or SAMROD, CHISQQ and DETYDS or DETNDS. The ray diagrams are displayed on the screen by the routine GEOMPS.

(g) When the display buffer is full the screen is cleared and an elevation view of the moderator, sample and detector surfaces is displayed. The interaction points are then displayed until the buffer is full.

During the calculation process the number of channels filled is monitored by the program. If this exceeds a specified number (NTRANG) as described in § 2g the routine HALT is entered and the intrinsic channel width is doubled. The program then restarts at subroutine PRELIM.

(h) At the end of the Monte Carlo sequence the routine RSCALE is entered and the intensity correction factors are applied as described in § 2h.

(i) The routine OUTDAT computes the mean and standard deviation of the simulated time-of-flight distribution.

(j) PLTDAT displays the distribution on the screen with the values of the variables computed in OUTDAT. The routine ASK prompts for four different commands; SUM, RESTORE, RESTART or STOP. SUM will prompt for the number of channels to be summed, RESTORE returns the data to its intrinsic time channel spacing, RESTART returns the program to READAT for another simulation and STOP ends the program. RESTART and STOP ask whether to PLOT or NOPLOT, this calls the routine VERDAT which will write the contents of the screen to a disk file for subsequent plotting on the VERSATEC graph plotter.

4. How to use the program

(a) Under RT-11 Operating system

The command is simply

.RUN SIMON.

(b) Under RSX-11 M Operating system

The user logs on under his own identifier and password and then types the command

>RUN DK1: [100,101] SIMON

The program can then run under a batch mode with all the input data in a data file or in an interrogative mode with the data being typed in at the user's console. Under batch mode the data are arranged in a file in the format shown in Figure 6. In the interrogative mode the terminal types the default value of the parameter and requests a T or F reply to indicate whether a new value is required. Figure 7 shows a typical input stream. After the Monte-Carlo calculations are completed the data are displayed on the screen and the user can type one of several commands.

SUM	This allows the user to sum channels for display
RESTORE	This restores the data to single channel spacing
RESTART	The program moves to the next case
STOP	The program completes and exits
PLOT	The screen image is written to a file for subsequent plotting
NOPLOT	No plotting output.

The graphs are plotted subsequently by a task called RASM. The line printer output is spooled to a file called LP.SPL, this can be printed by the command

>PIP LP.SPL/SP

Similarly additional graph copies may be produced by the command

>RUN RASM

and whatever images are stored in the graphics files

VECTR1.BIN and PARM.BIN will be plotted.

5. Examples of use

(a) The Harwell Total-Scattering Spectrometer

The Q-resolution for this spectrometer has been simulated for all the detector angles. The Q-ranges chosen were designed to cover the wavelength range where the moderator pulse width is expected to obey the formalism of equation (4).

The experimental parameters of the instrument as used in the simulation are listed in Table 1. The detectors below the scattering angle of 90° were fitted with Debye-Scherrer cone apertures as discussed in § 2e. The sample chosen for the simulation was a 5 mm diameter powder.

Figure 8 shows the diffraction peaks from the imaginary powder sample at a range of Q-values for the 150° detector. Figure 9 shows the variation of the fractional Q-resolution with Q-value for each detector together with the FWHM measurements of reference[5].

The agreement between the measured Bragg peaks and the simulated data is poor in the wavelength region used to record these nickel diffraction peaks. The principal source of error is the moderator pulse width as these peaks occur in the region of the Maxwellian component of the spectrum rather than in the epithermal region where the pulse width is better described. The horizontal arrows on the left of Figure 9 represent some other measurements of Sinclair (private communication).

TABLE 1

Parameters used to simulate the Total-Scattering Spectrometer

Group	Parameter	Value
Source	Pulse width	2 μs
Moderator	Angle Width Height Mean-free-path	- 10° 0.1m 0.1m 0.0053m ⸸⸸
Incident flight path		4.324m
Sample	Angle Width (diameter) Height	cylinder 5.0 mm 0.025m
Scattered flight path		0.46m
Detector	Thickness Efficiency constant Angle Width Height Focussing Angle Angle Width Height Angle Width Height Angle Width Height Angle Width Height Angle Width Height	0.025m* ≡ 0.0196m 17.65 m^{-1} 150° 0.175m 0.30m 54.3° 90° 0.01m 0.30m 58° ⸸ 0.008m 0.25m 35° ⸸ 0.005m 0.14m 20° ⸸ 0.005m 0.085m 10° ⸸ 0.005m 0.085m

* Cylinder diameter

⸸ Debye-Scherrer cone

⸸⸸ [6]

(b) **The Harwell Back-Scattering Spectrometer**

This spectrometer has been described in detail elsewhere[2]. The Q-resolution was simulated for the parameters tabulated in table 2. The fractional Q-resolution is shown in figure 10 together with the measured and calculated resolution from the work of[2]. The agreement is poor at lower Q-values due to the formalism used to describe the intrinsic moderator pulse width in § 2b. At high Q-values where the epithermal neutron pulse width is adequately described by equation (4) the agreement is better. Figure 11 shows the diffraction peaks from the imaginary sample.

TABLE 2

Parameters used to simulate the Back-Scattering Spectrometer

Group	Parameter	Value
Source	Pulse width	2 μs
Moderator	Angle Width Height Mean-free-path	10° 0.1m 0.1m 0.0053m //
Incident flight path		12.04m
Sample	Angle Width Height Thickness	0° 0.05m 0.05m 0.0025m
Scattered flight path		2.0m
Detector	Thickness Efficiency constant Angle Width Height Focussing angle Angle Width Height Focussing angle	0.025* ≡ 0.0196m 17.65 m^{-1} 170° 0.3m 0.3m 17.1° -170° 0.3m 0.3m 17.1°

// [6]

6. Simulation of the Q-resolution function whilst varying specific instrumental parameters

As a test of the computer code and also to verify some of the formulae which are commonly used in calculating the Q-resolution of time-of-flight spectrometers, specific instrumental configurations were simulated. To separate the contributions from each part of the spectrometer, one or all of the solids (ie moderator, sample or detector) were given small dimensions to reduce them effectively to points or in some cases slits or lines.

In a similar manner the time-broadening produced by the source and moderator was disabled by setting the appropriate constants to zero.

(a) **The neutron source pulse**

The moderator, sample and detector were given minute dimensions (0.001m × 0.001m) and the mean free path in the moderator was set to zero. Figure 12 shows the time-resolution function generated for electron pulse widths of 0.5, 1.0 and 5.0 μs at a Q-value of 30 $Å^{-1}$. The functions are rectangular and have the intrinsic width of the pulse. This is as would be expected because all the other instrumental broadening effects have been eliminated.

(b) **The moderator pulse width**

With a point moderator, sample and detector as in § 6a and with a zero electron pulse width, the resolution function was simulated for a range of moderator mean free paths of 0.003, 0.006 and 0.009m for a solid with 6-degrees of freedom in the thermalization calculations[4]. The functions are shown in figure 13 and have the shape of the expression in § 2b(2). The mean emission time of the moderator in the epithermal region is given by,

$$\bar{t} = \frac{\text{FREEDM*PATH}}{\text{2*VELOCITY}} \tag{11}$$

and the half-width is

$$\sigma(t) = \frac{\text{PATH*}\sqrt{\text{2*FREEDM}}}{\text{2*VELOCITY}} \tag{12}$$

Table 3 lists the parameters used in the simulation and the calculated and observed mean emission values and half widths

TABLE 3

Verification of the Moderator Pulse Width Function

Degrees of Freedom	Velocity [m/μs]	Mean Free path [m]	$\bar{t}_{calc}$ [μs]	$\bar{t}_{obs}$ [μs]	σ_{calc} [μs]	σ_{obs} [μs]
6	0.00945	0.003	0.953	0.8	0.55	0.6
6	0.00945	0.006	1.905	1.7	1.10	1.1
6	0.00945	0.009	2.858	2.7	1.65	1.7

Sections (a) and (b) have simply verified that the computer code is correctly simulating the formulae used to describe the source and moderator time dependence. The next sections will examine the behaviour of the resolution function as the geometry of the instrument is changed.

(c) **The Angular Dependence of the Detector Resolution**

If

$$t = \frac{4\pi}{Q} \sin\frac{\theta}{2} \text{ [Flight path} \times 252.7] \qquad (21)$$

and as we are discussing Q-value resolution the form of Q is a δ-function so that the fractional time resolution is

$$\frac{\Delta t}{t} = \cot\frac{\theta}{2}\frac{\Delta\theta}{2} \qquad (40)$$

For this simulation the moderator and sample were regarded as points, the detector was a horizontal slit 0.005 m wide by 0.0001 m high and both the source and moderator time resolution contributions were set to zero. Q-resolution functions were simulated as a function of scattering angle and were compared with the calculated values. These functions were rectangular and their widths are plotted in Figure 14 together with those calculated from equation (40). Because the functions were rectangular the angular width was taken to be [0.005/0.46].

Table 4 lists the observed and calculated resolution widths as a function of angle, the agreement is good showing that the $\cot\frac{\theta}{2}$ dependence is valid in this simple case.

TABLE 4

Simulation of the Angular Dependence of the Resolution Function

Flight paths 5.6 and 0.46 m

$\Delta\theta = 0.005/0.46 = 0.0109$

θ degrees	$\cot\theta/2$	t [μs]	Δt obs[μs]	$\Delta t/t$ obs	$\Delta t/t$ calc
2.5	45.83	83	20.0	0.241	0.249
5.0	22.90	167	20.0	0.120	0.124
10.0	11.43	335	20.0	0.0597	0.062
20.0	5.67	668	20.0	0.0299	0.031
30.0	3.73	996	19.5	0.0196	0.020
60.0	1.73	1924	18.0	0.00936	0.0094
90.0	1.0	2721	15.0	0.0055	0.0054
150.0	0.268	3717	5.4	0.00145	0.00146
170.0	0.087	3834	1.9	0.00049	0.000475

(d) **The Q-Focussing Properties of the Moderator, Sample and Detector**

With an extended detector at scattering angles above 90° (where the change in the absorption corrections etc. with angle is small), it is possible to Q-focus the scattered neutrons[7,8].

Each time-of-flight channel from several detectors lying on a focussed locus will have the same Q-value. This is achieved by balancing the change in flight path by the change in scattering angle. This technique has the advantage of providing a large solid angle of detector for one time-of-flight scaler and has been an important feature in the successful use of the Harwell Total-Scattering and Back-Scattering spectrometers[1,2].

The focussing technique has been extended to cover moderators and samples so that if the initial and final flight paths are L_1 and L_2, and the detector angle is θ_o the three focussing angles are:

$$\tan \theta_d = \frac{(L_1 + L_2)}{2L_2} \cot \frac{\theta_o}{2} \tag{41}$$

$$\tan \theta_m = \frac{(L_1 + L_2)}{2L_1} \cot \frac{\theta_o}{2} \tag{42}$$

and for a flat plate sample

$$\tan \theta_s = \frac{\cos \theta_d \tan \theta_m + \sin (\theta_o + \theta_d)}{2 \sin \frac{\theta_o}{2} \sin \left(\frac{\theta_o}{2} + \theta_d \right)} \tag{43}$$

In order to test these focussing criteria satisfactorily by simulation in the program, the moderator, sample and detector were replaced by points or slits where appropriate to decouple the resolution contributions from each component of the spectrometer. Figure 15 shows the effect of varying the moderator focussing angle, Figure 16 shows the detector angle and Figure 17 shows the sample angle. For the sample angle two cases were considered (a) with a point moderator and detector, and (b) with line and slit components set at the correct focussing angles. The minima in all the focussing tests agreed with the calculated values from reference [8].

(e) **The Intensity Correction Terms**

The effect of varying certain parameters of the diffractometer has been considered. The TSS Mk I, as outlined in Table 1 of Section 5, has been simulated for the 150° detector. The Q-resolution and intensity have been studied as functions of instrumental dimensions as an example of the type of design study possible using computer codes of this form.

Figure 18 shows the fractional Q-value resolution for the 150° detector bank at a Q-value of 30 Å^{-1}. The incident and scattered flight paths have been varied and the fixed area detector bank has been rotated to maintain Q-focussing. The dashed lines indicate the peak area and the solid lines the fractional Q-resolution. An experimentalist requiring a specific Q-resolution could thus evaluate whether a small sacrifice in resolution might yield a larger increase in intensity or vice-versa. The criterion for an instruments' figure-of-merit really depends on the type of experiment, but as for example, in Figure 18 for the incident flight path, no real gain in resolution is achieved above 10 m but the intensity decreases appreciably, this is assuming a uniform incident spectrum intensity which is clearly a gross simplification.

Figure 19 shows the variation of fractional resolution with moderator dimensions for the 10° and 150° detectors. The 150° data show relatively little deterioration in resolution up to quite large (and impractical) moderator dimensions. The intensity (which assumes a uniform surface brightness) increases rapidly.

For the 10° data the moderator width plays an important role in determining the fractional Q-resolution with a reduced gain in intensity above 0.20 m in YM. The moderator height is (as would be expected) far less important.

These dependences upon width and height have been investigated for other parameters. Variables such as sample and detector height have been found to affect resolution only slightly, whereas sample widths and diameters have a greater effect. There are only general conclusions and for a full investigation of an instrument the incident spectrum shape should be considered when changing flight paths and considering the same Q-values. Also the incident pulse shape in the Maxwellian region must be parameterised wherever possible to avoid the obvious approximation of an infinite medium epithermal moderator.

Acknowledgements

The author wishes to thank Dr. J. M. Carpenter of the Argonne National Laboratory for the idea of simulating a time-of-flight spectrometer by a Monte-Carlo computer code, and for a copy of his geometric resolution code TOFDIF.

The author also thanks Mr. D. A. G. Johnson for assistance in implementing the PDP-11 programs and Dr. R. N. Sinclair and Dr. C. G. Windsor for suggestions concerning the intensity calculations.

Financial support is acknowledged from AERE, Harwell in the form of an EMR fellowship.

References

1. SINCLAIR, R. N., JOHNSON, D. A. G., DORE, J. C., CLARKE, J. H. and WRIGHT, A. C., Nuc. Instr. Meth. *117*, 445 (1974).

2. WINDSOR, C. G., BUNCE, L. J., BORCHERDS, P. H., COLE, I., FITZMAURICE, M., JOHNSON, D. A. G. and SINCLAIR, R. N., Nuc. Instr. Meth. *140,* 241 (1977).

3. MILDNER, D. F. R., BOLAND, B., SINCLAIR, R. N., WINDSOR, C. G., BUNCE, L. J., CLARKE, J. H. Nuc. Instr. Meth. *152*, 437 (1978)

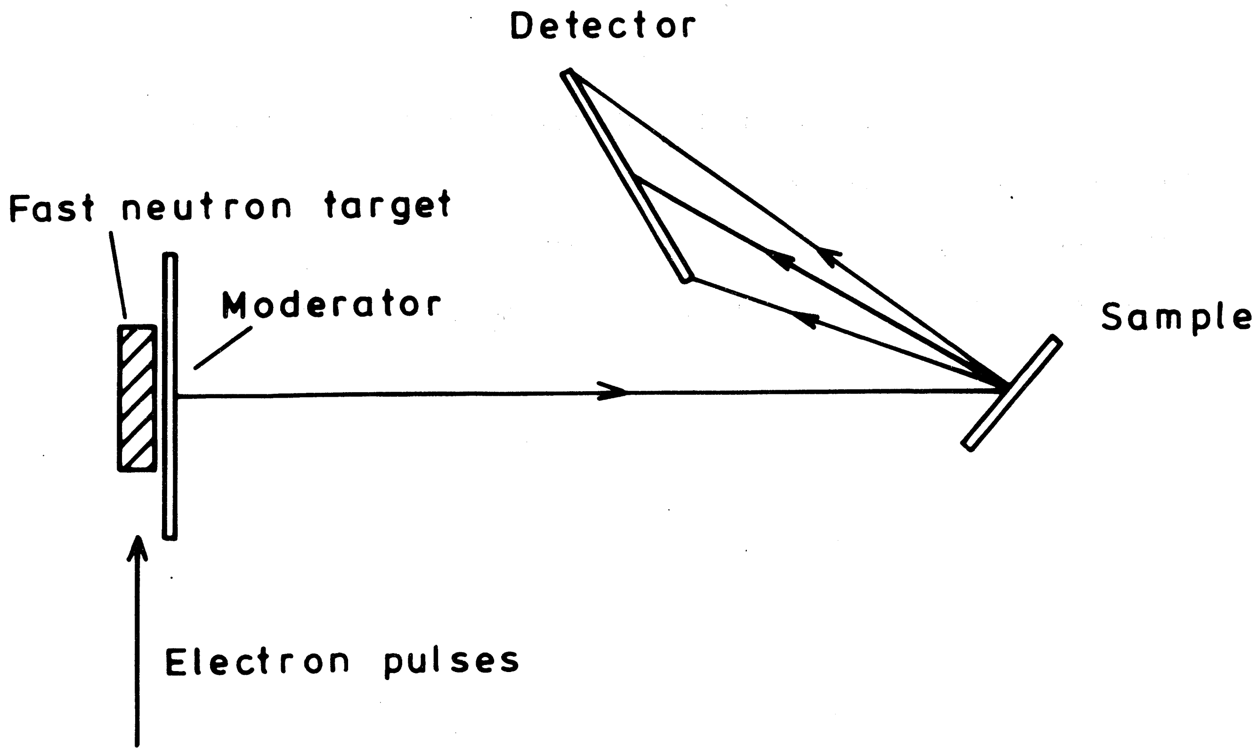
4. CARPENTER, J. M., Nuc. Instr. Meth. *145*, 91 (1977)

5. JOHNSON, P. A. V., WRIGHT, A. C. and SINCLAIR, R. N., AERE Report R-8925 (1978).

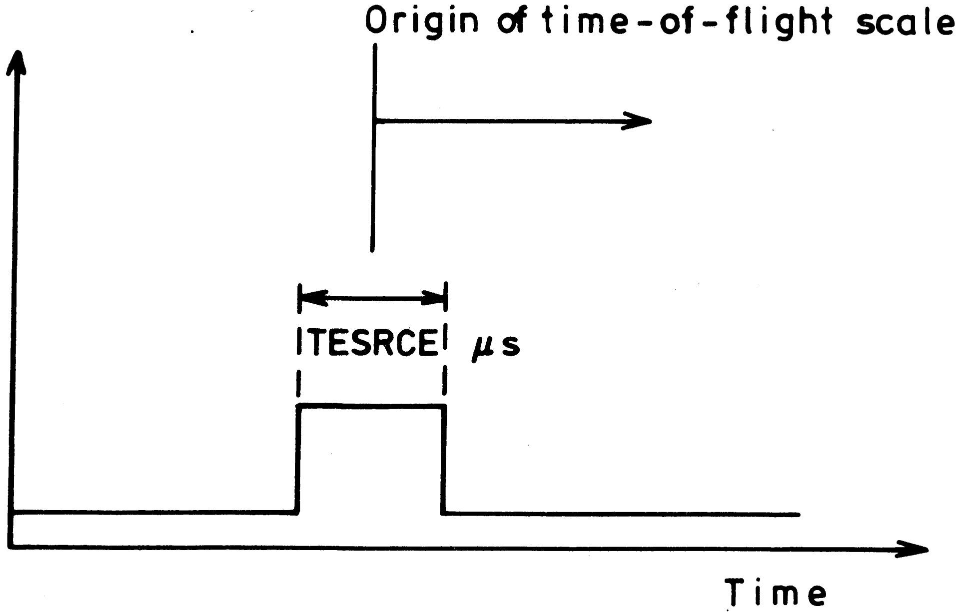
6. MOXON, M. C., Private communication.

7. CARPENTER, J. M., Nuc. Instr. and Meth. *47*, 179 (1967).

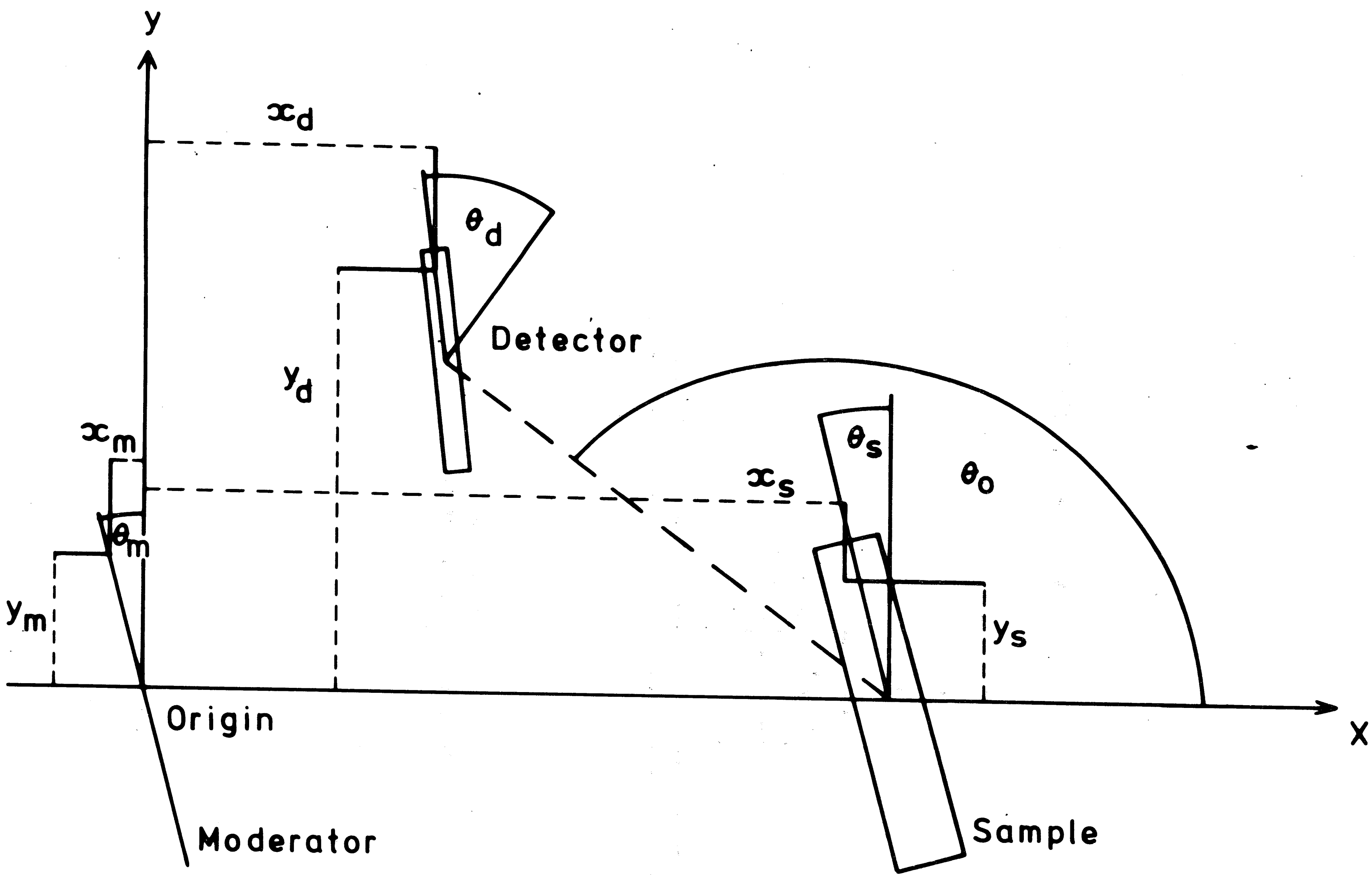
8. GRAHAM, K. F. and CARPENTER, J. M., Nuc. Instr. and Meth. *85* 163, (1970).



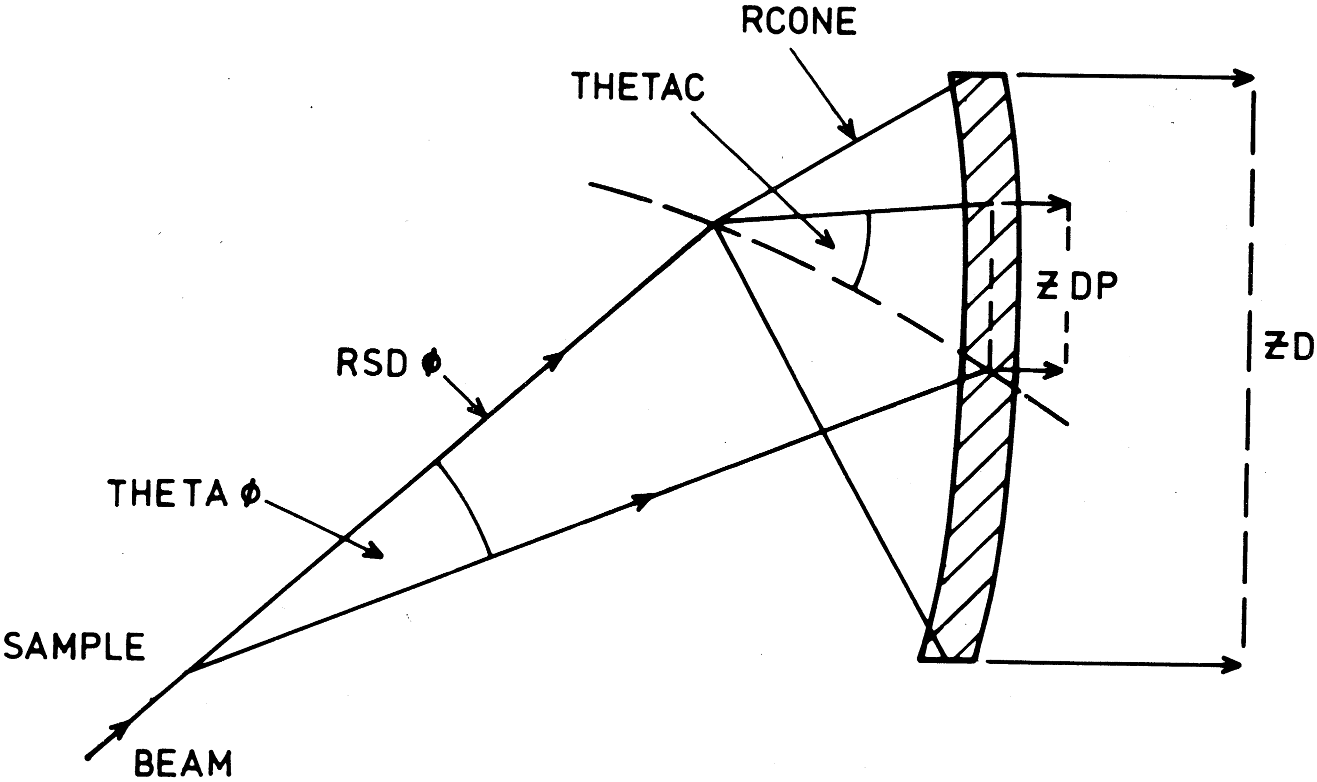
AERE - R.9170 Fig. 1
The Harwell Total-Scattering Spectrometer



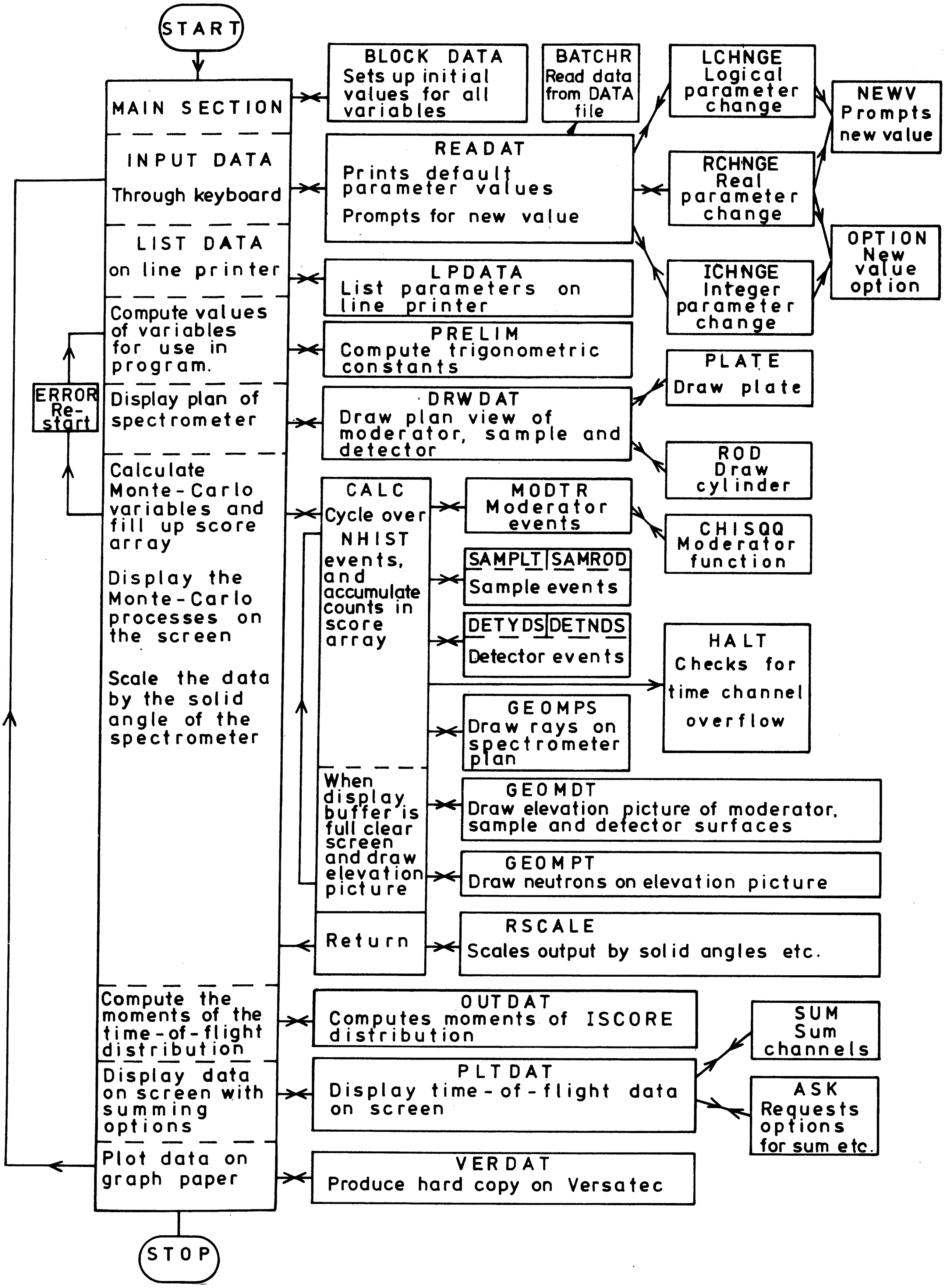
AERE - R.9170 Fig. 2
The shape and duration of the electron pulse



AERE - R.9170 Fig. 3
The coordinate system chosen for the diffractometer



AERE - R.9170 Fig. 4
The detector aperture using the Debye-Scherrer cone option



AERE - R.9170 Fig. 5
The schematic flow chart of the program

```
TEST DATA FOR R-9170 TSS 150 Q=5          !TITLE
TTTTFF               !6 LOGICAL VARIABLES IN FORMAT 6L1
                     !LINEPRINTER T OR F
                     !GT44 SCREEN T OR F
                     !DISPLAY ON SCREEN T OR F
                     !ROD SAMPLE T OR F
                     !PLATE SAMPLE T OR F
                     !DEBYE-SCHERRER CONE T OR F

     15000           !NUMBER OF EVENTS I10 FORMAT
-10.                 !MODERATOR ANGLE
.1                   !MODERATOR WIDTH
.1                   !MODERATOR HEIGHT
0.                   !SAMPLE ANGLE
.005                 !SAMPLE WIDTH OR DIAMETER
.025                 !SAMPLE HEIGHT
.0025                !SAMPLE THICKNESS OR RADIUS
54.3                 !DETECTOR FOCUSSING ANGLE
.175                 !DETECTOR WIDTH
.3                   !DETECTOR HEIGHT
.0196                !DETECTOR THICKNESS
17.65                !DETECTOR EFFICIENCY CONSTANT
150.                 !DETECTOR ANGLE
4.324                !INITIAL FLIGHT PATH
0.46                 !SECOND FLIGHT PATH
5.                   !Q-VALUE
2.                   !ELECTRON PULSE WIDTH
.0053                !MODERATOR MEAN FREE PATH
0.5                  !TIME CHANNEL WIDTH
                     !BLANK LINE
RESTART              !RESTART OR STOP COMMAND
PLOT                 !PLOT OR NOPLOT COMMAND
```

AERE - R.9170 Fig. 6
Data file for input to SIMON running under batch mode

```
RUN DK1:[100,101]SIMON

DO YOU WANT TO RUN UNDER BATCH ? T OR F? F

TYPE IN TITLE UP TO 72 CHARACTERS
TEST DATA TO DEMONSTRATE USE OF INTERROGATIVE MODE
```

Fig. 7a

```
LINEPRINTER OPTION IS        F
CR FOR NO CHANGE,   T  TO CHANGE PARAMETER T
NEW VALUE=T

GT44 SCREEN IS       F
CR FOR NO CHANGE,   T  TO CHANGE PARAMETER T
NEW VALUE=T

DISPLAY OPTION IS       F
CR FOR NO CHANGE,   T  TO CHANGE PARAMETER T
NEW VALUE=T

MODERATOR ANGLE=           0.00000
CR FOR NO CHANGE,   T  TO CHANGE PARAMETER T
NEW VALUE=-10.

MODERATOR WIDTH=           0.10000 METRES
CR FOR NO CHANGE,   T  TO CHANGE PARAMETER

MODERATOR HEIGHT=          0.10000 METRES
CR FOR NO CHANGE,   T  TO CHANGE PARAMETER

 IS THE SAMPLE A  FLAT PLATE T OR F? F

SAMPLE DIAMETER            0.02500 METRES
CR FOR NO CHANGE,   T  TO CHANGE PARAMETER

SAMPLE HEIGHT=             0.02500 METRES
CR FOR NO CHANGE,   T  TO CHANGE PARAMETER

DETECTOR ANGLE=           60.50000
CR FOR NO CHANGE,   T  TO CHANGE PARAMETER

DETECTOR WIDTH=            0.17500 METRES
CR FOR NO CHANGE,   T  TO CHANGE PARAMETER

DETECTOR HEIGHT=           0.30000 METRES
CR FOR NO CHANGE,   T  TO CHANGE PARAMETER

IS THE DETECTOR DEFINING SLIT SHAPED TO FIT
 THE DEBYE SCHERRE CONE?? T OR F ?F

DETECTOR THICKNESS=        0.01963 METRES
CR FOR NO CHANGE,   T  TO CHANGE PARAMETER

DETECTOR EFFICIENCY AT 1.0 ANGSTROM =   17.65000
CR FOR NO CHANGE,   T  TO CHANGE PARAMETER
```

Fig. 7b

```
SCATTERING ANGLE=          150.00000
CR FOR NO CHANGE,  T  TO CHANGE PARAMETER____

MODERATOR TO SAMPLE=        5.60000 METRES
CR FOR NO CHANGE,  T  TO CHANGE PARAMETER____

SAMPLE TO DETECTOR=         0.46000 METRES
CR FOR NO CHANGE,  T  TO CHANGE PARAMETER____

Q-VALUE=                   10.00000 ANGSTROM**-1
CR FOR NO CHANGE,  T  TO CHANGE PARAMETER T
NEW VALUE=15.0

RECTANGULAR SOURCE PULSE WIDTH=     2.00000 USECS
CR FOR NO CHANGE,  T  TO CHANGE PARAMETER____

MODERATOR MEAN FREE PATH=           0.00530 METRES
CR FOR NO CHANGE,  T  TO CHANGE PARAMETER____

NO. OF DEGREES OF FREEDOM=          6.

TIMECHANNEL WIDTH=          1.00000 USECS
CR FOR NO CHANGE,  T  TO CHANGE PARAMETER____

NO. OF PATHS=                   500
CR FOR NO CHANGE,  T  TO CHANGE PARAMETER T
NEW VALUE=1000
```

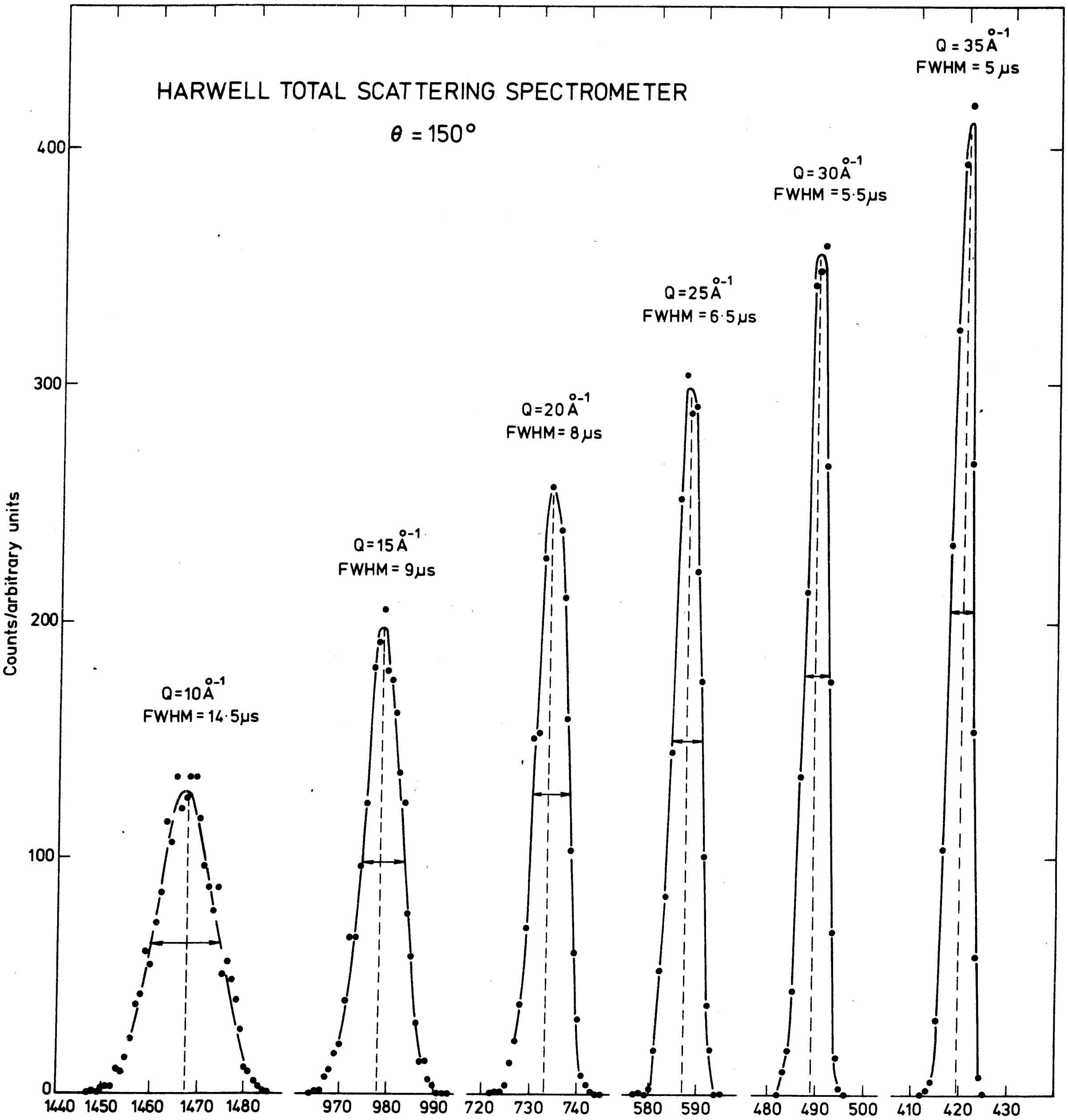
AERE - R.9170 Fig. 7

Sample input stream for input to SIMON running under the interrogative console mode

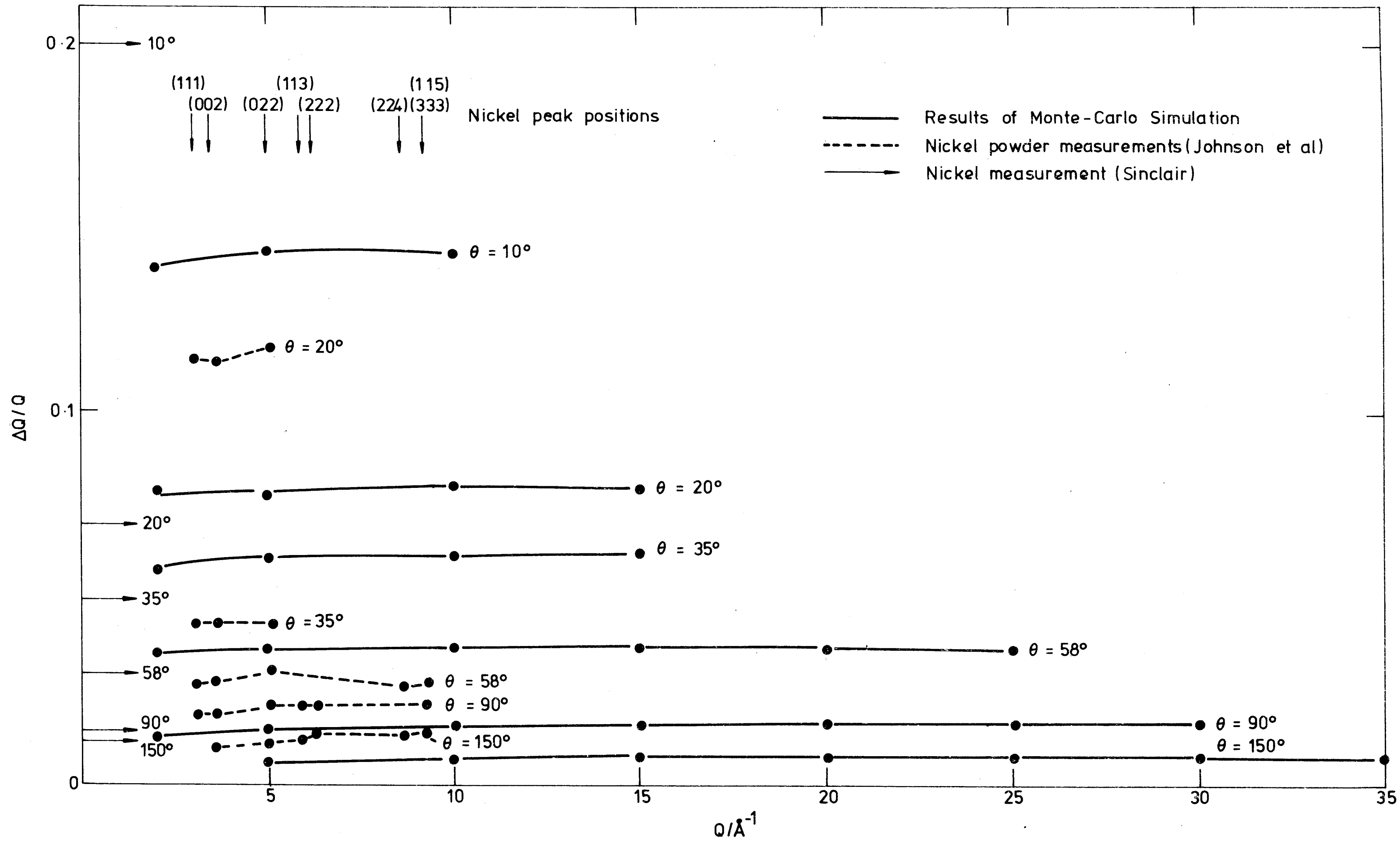
Fig. 7c

TIME USECS	SCORE
1222.00	0.00
1223.00	470.83
1224.00	1412.50
1225.00	1412.50
1226.00	1412.50
1227.00	1883.33
1228.00	6120.84
1229.00	2825.00
1230.00	12241.67
1231.00	11770.84
1232.00	11300.01
1233.00	14595.84
1234.00	28720.85
1235.00	25425.02
1236.00	31075.02
1237.00	31545.85
1238.00	31545.85
1239.00	37195.86
1240.00	30604.19
1241.00	32958.36
1242.00	36254.19
1243.00	31545.85
1244.00	22600.02
1245.00	22600.02
1246.00	15537.51
1247.00	12712.51
1248.00	6120.84
1249.00	3295.84
1250.00	941.67
1251.00	2354.17
1252.00	941.67
1253.00	470.83
1254.00	470.83
1255.00	0.00
1256.00	0.00
1257.00	470.83
1258.00	0.00

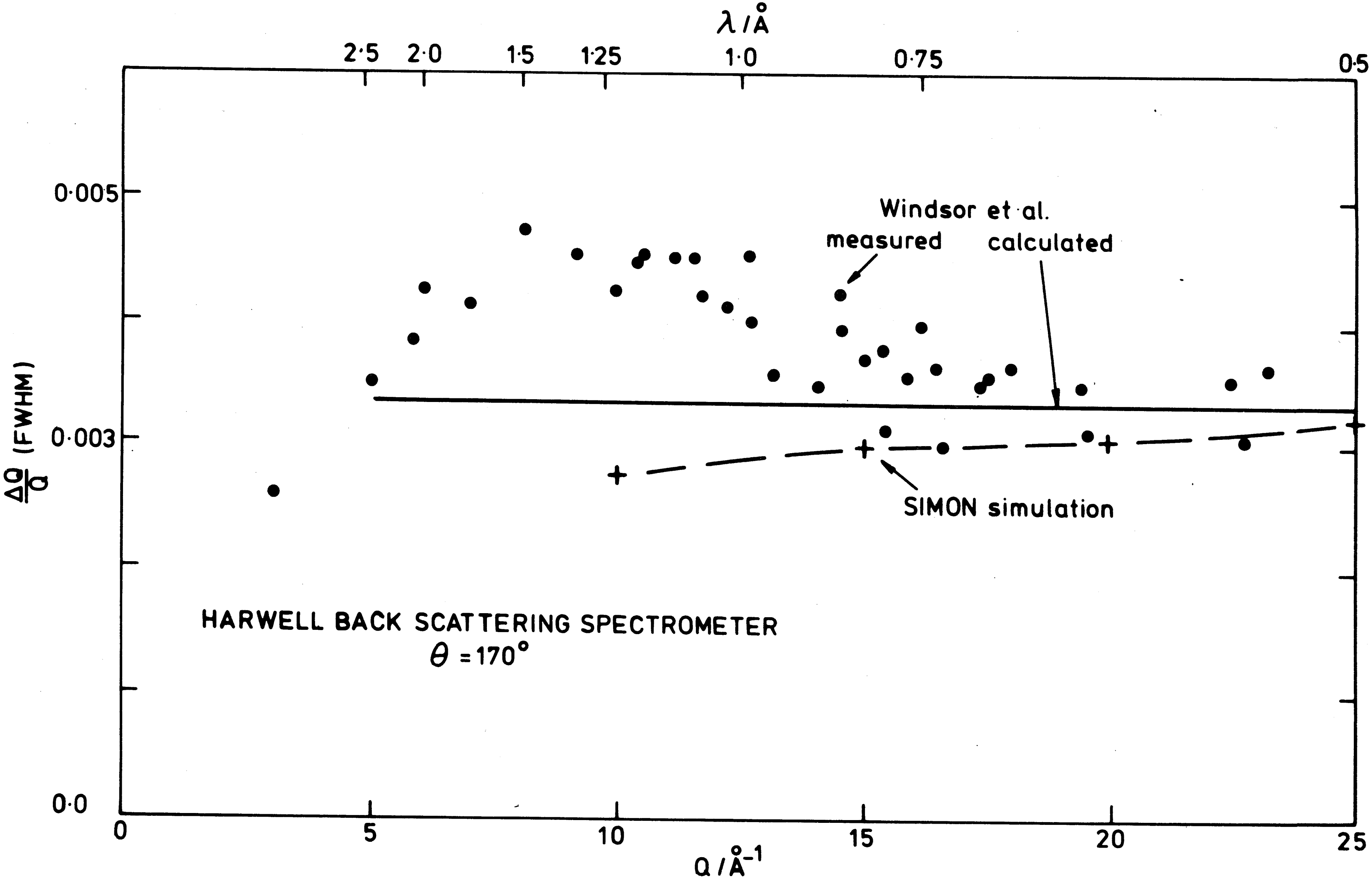
```
SO          =  470833.656 TO    =  1239.2 USECS TMEAN     =  1238.9 USECS
SIGMAT      =   5.125 USECS
WAVELENGTH=     0.809 ANGSTROM
Q-VALUE     =  15.000 ANGSTROM**-1 D-SPACING = 0.41888 ANGSTROM1
SUM OR RESTORE; THEN RESTART OR STOP SUM
NO. OF CHANNELS FOR SUMMING?2
SUM OR RESTORE; THEN RESTART OR STOP RESTORE
SUM OR RESTORE; THEN RESTART OR STOP STOP
FOR THIS PICTURE ON VERSATEC TYPE PLOT OR NOPLOT PLOT
MAPPED - VECTOR
```



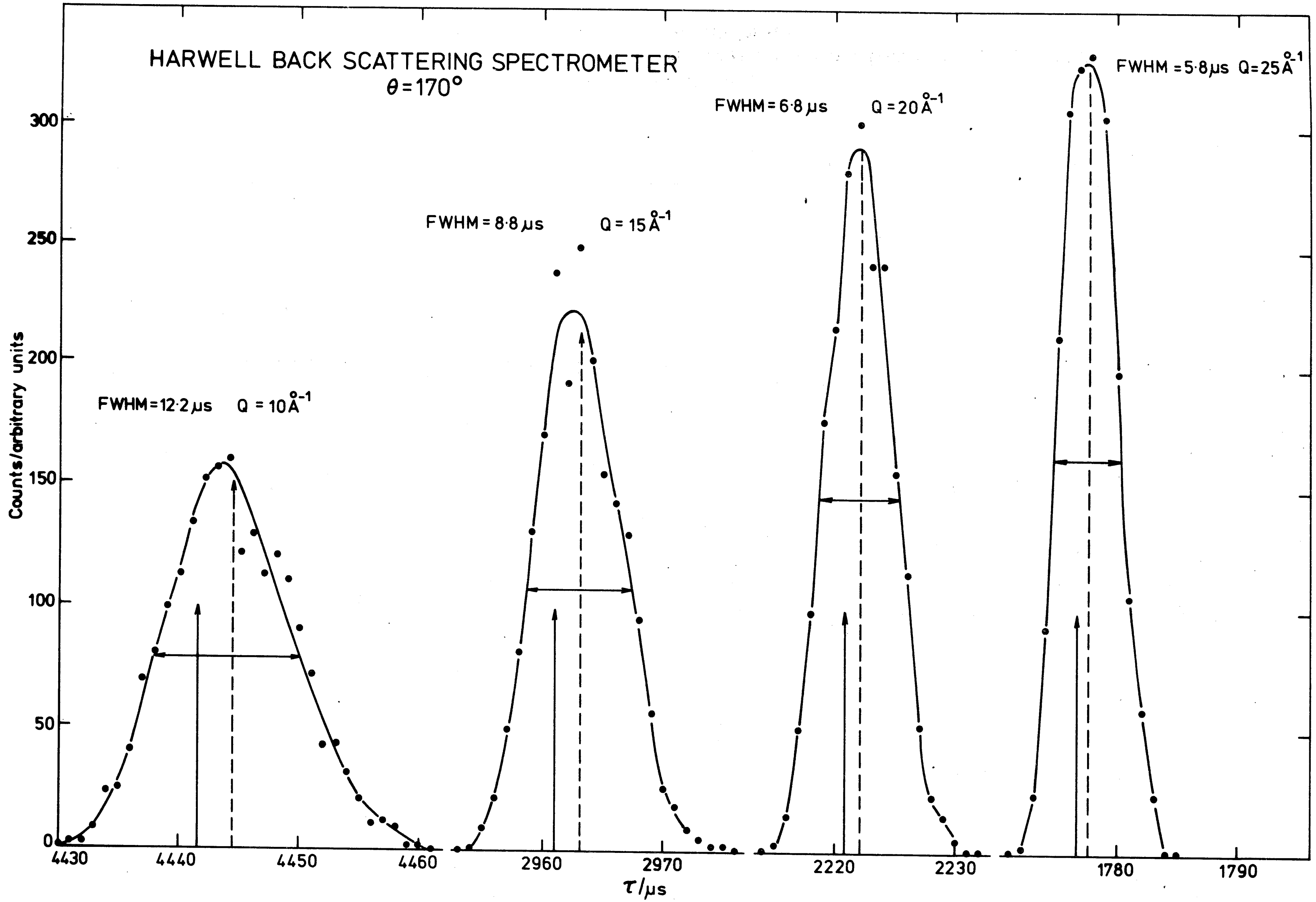
AERE - R.9170 Fig. 8
Simulated diffraction peaks for the TSS spectrometer at 150° scattering angle



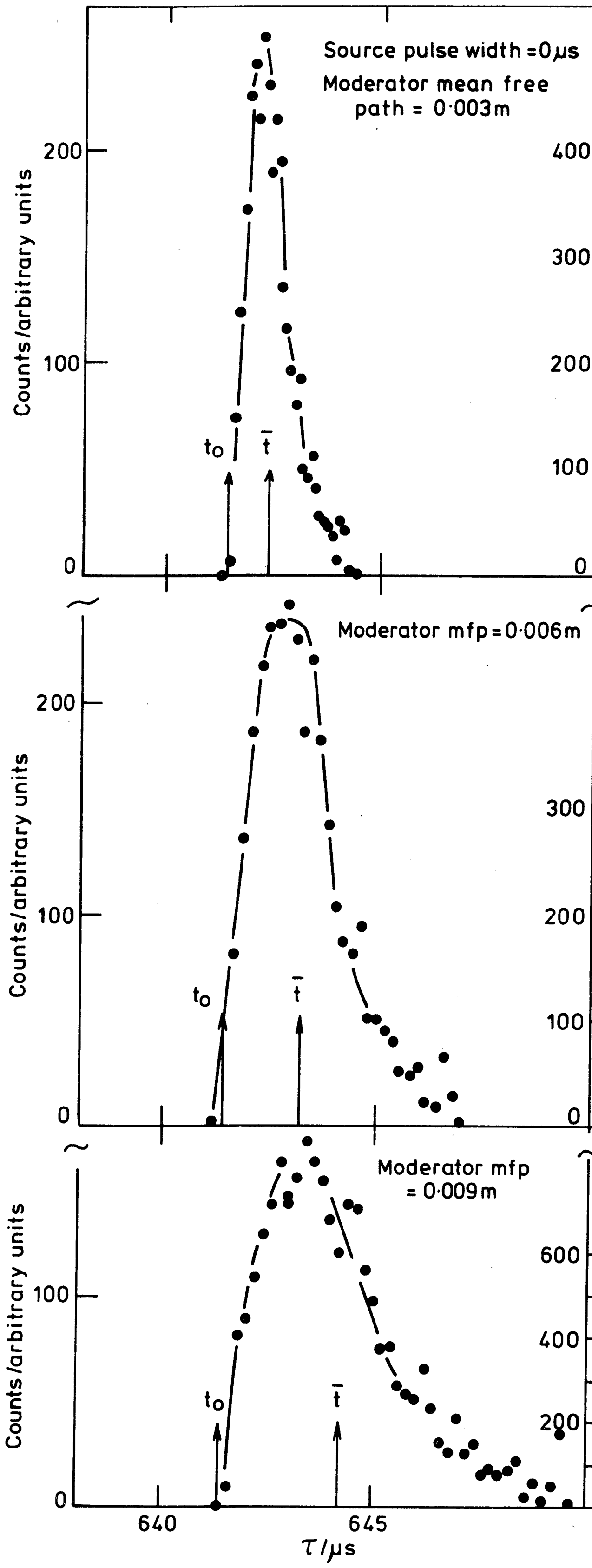
AERE - R.9170 Fig. 9
Variation of the fractional FWHM Q-resolution with Q-value for the TSS spectrometer together with some experimental results



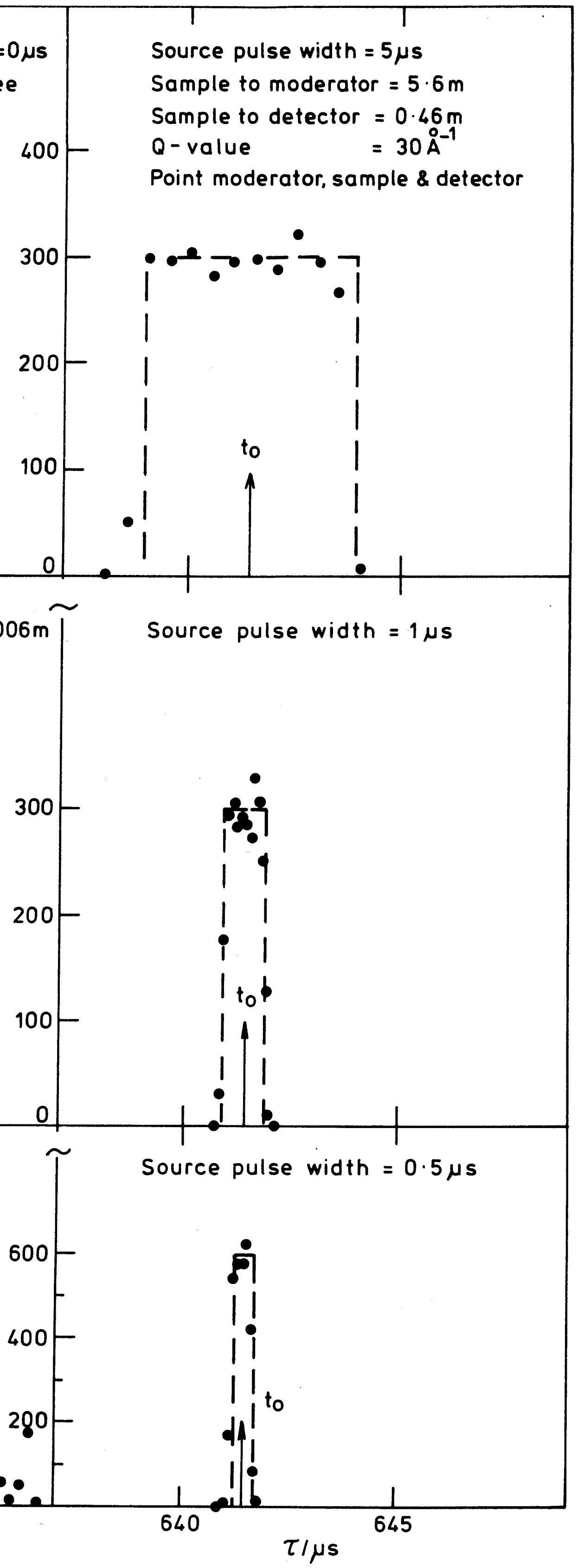
AERE - R.9170 Fig. 10
Fractional Q-value resolution for the BSS spectrometer as a function of Q-value compared with the experimental results of [2]



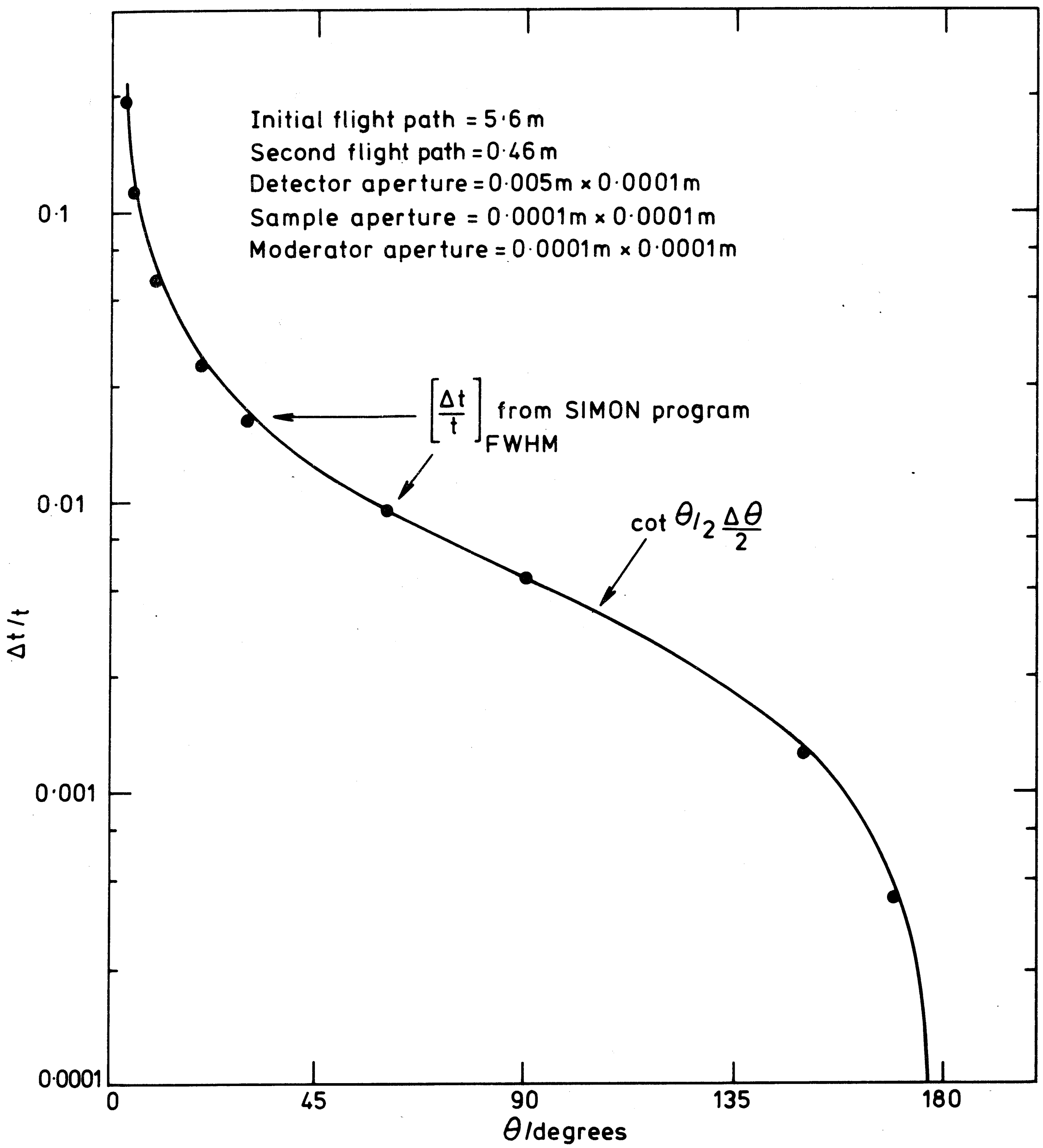
AERE - R.9170 Fig. 11
Simulated diffraction peaks for the BSS spectrometer at a scattering angle of 170°



AERE - R.9170 Fig. 13
Contribution to the resolution from the moderator pulse width

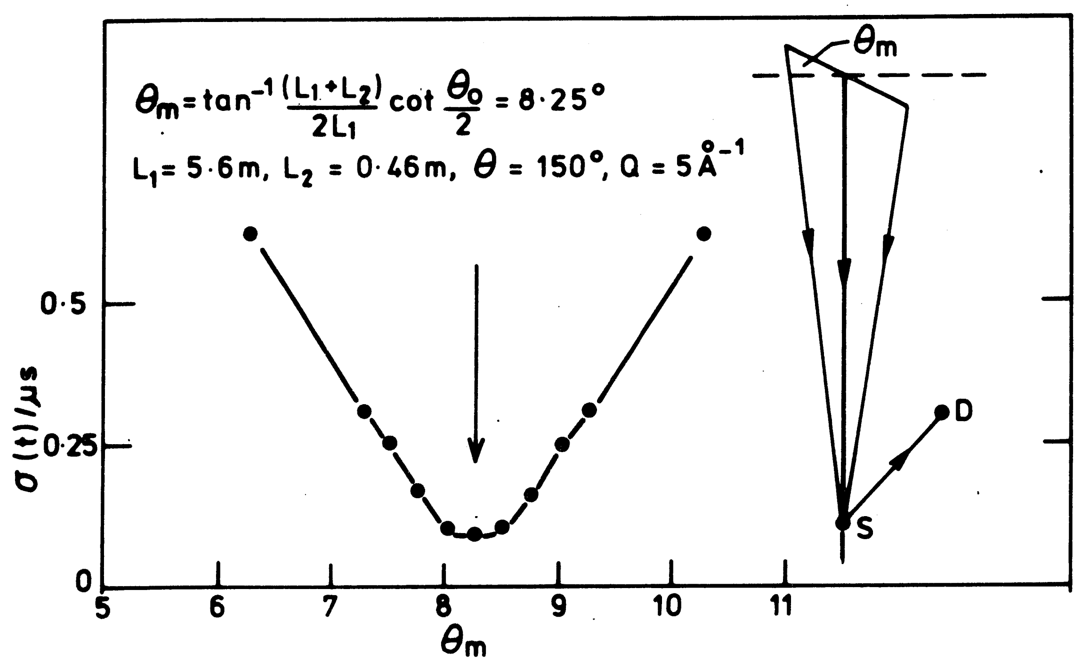


AERE - R.9170 Fig. 12
Contribution to the resolution from the electron pulse width

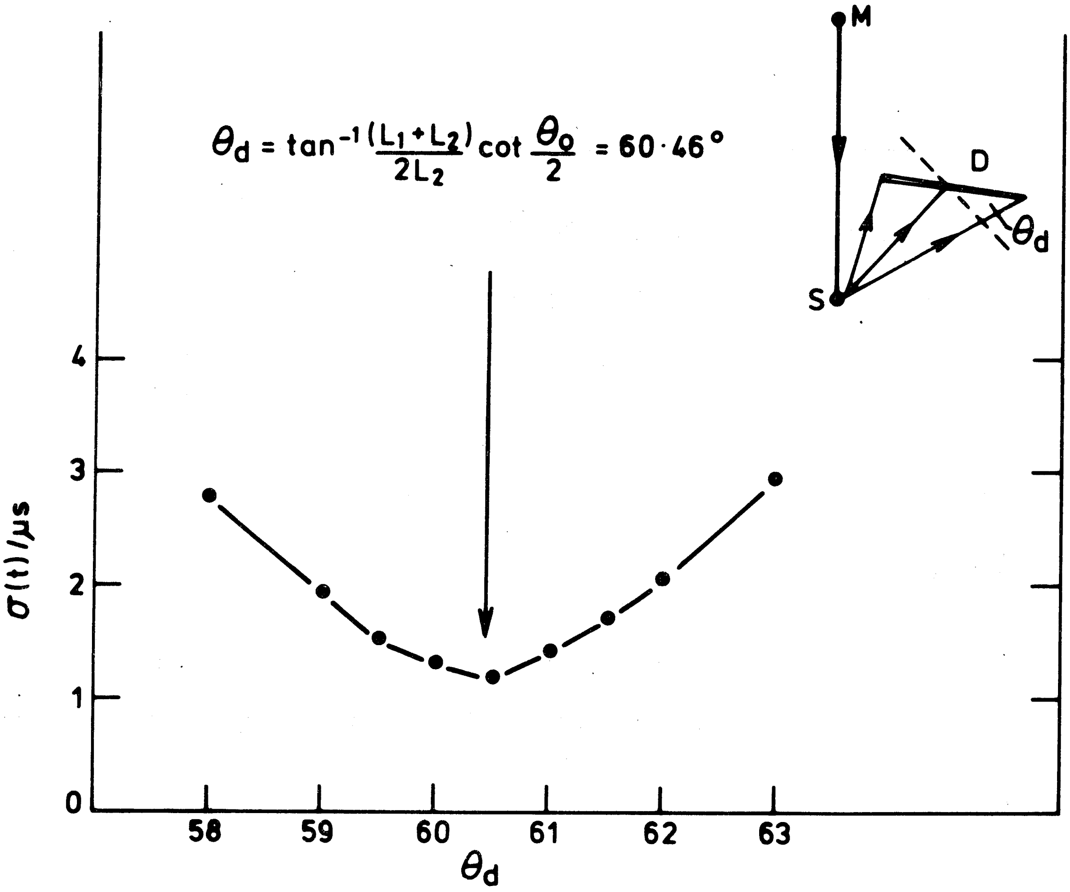


AERE - R.9170 Fig. 14

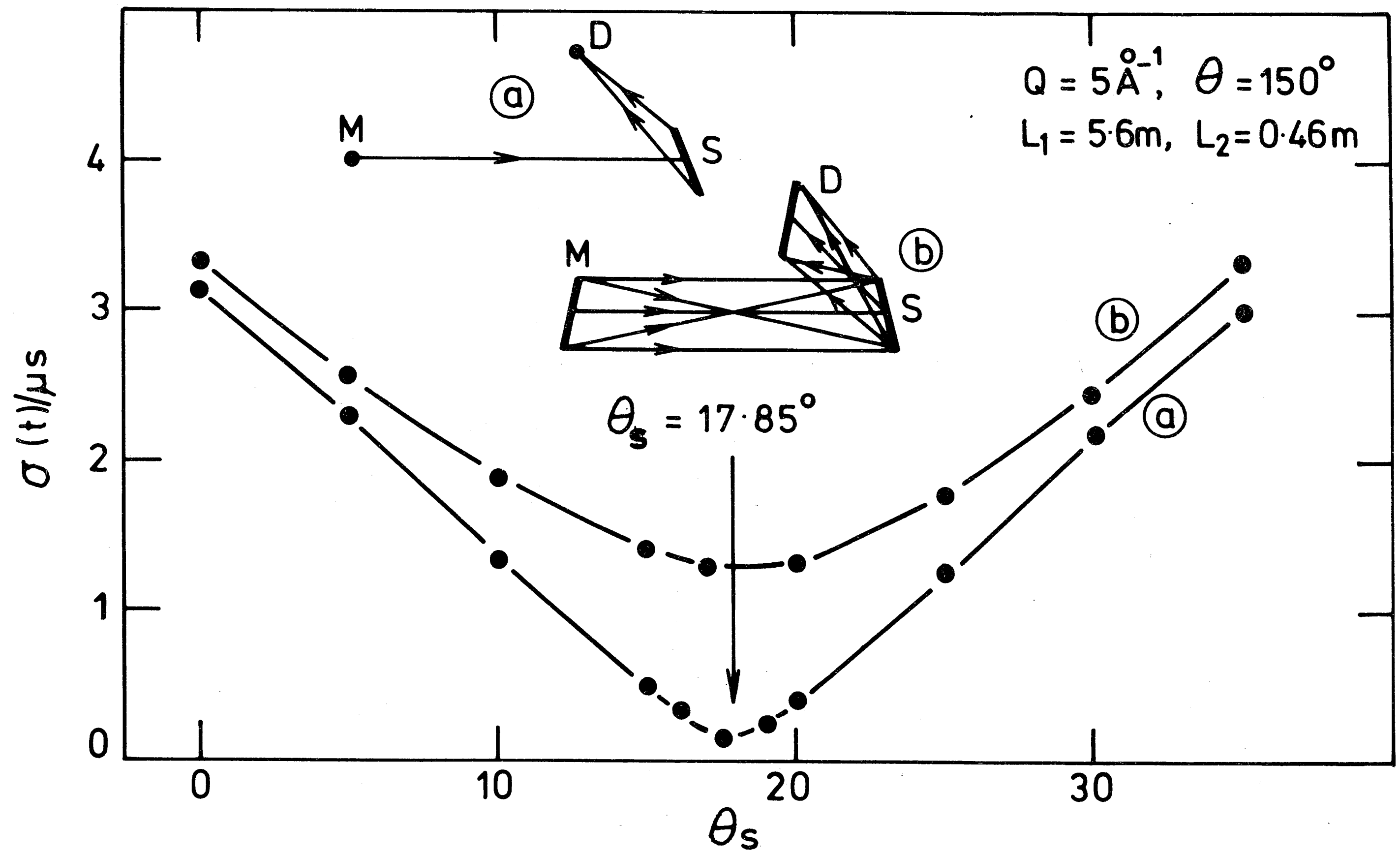
Variation of Q-value resolution with diffraction angle θ, for a slit sample showing both simulated and calculated points



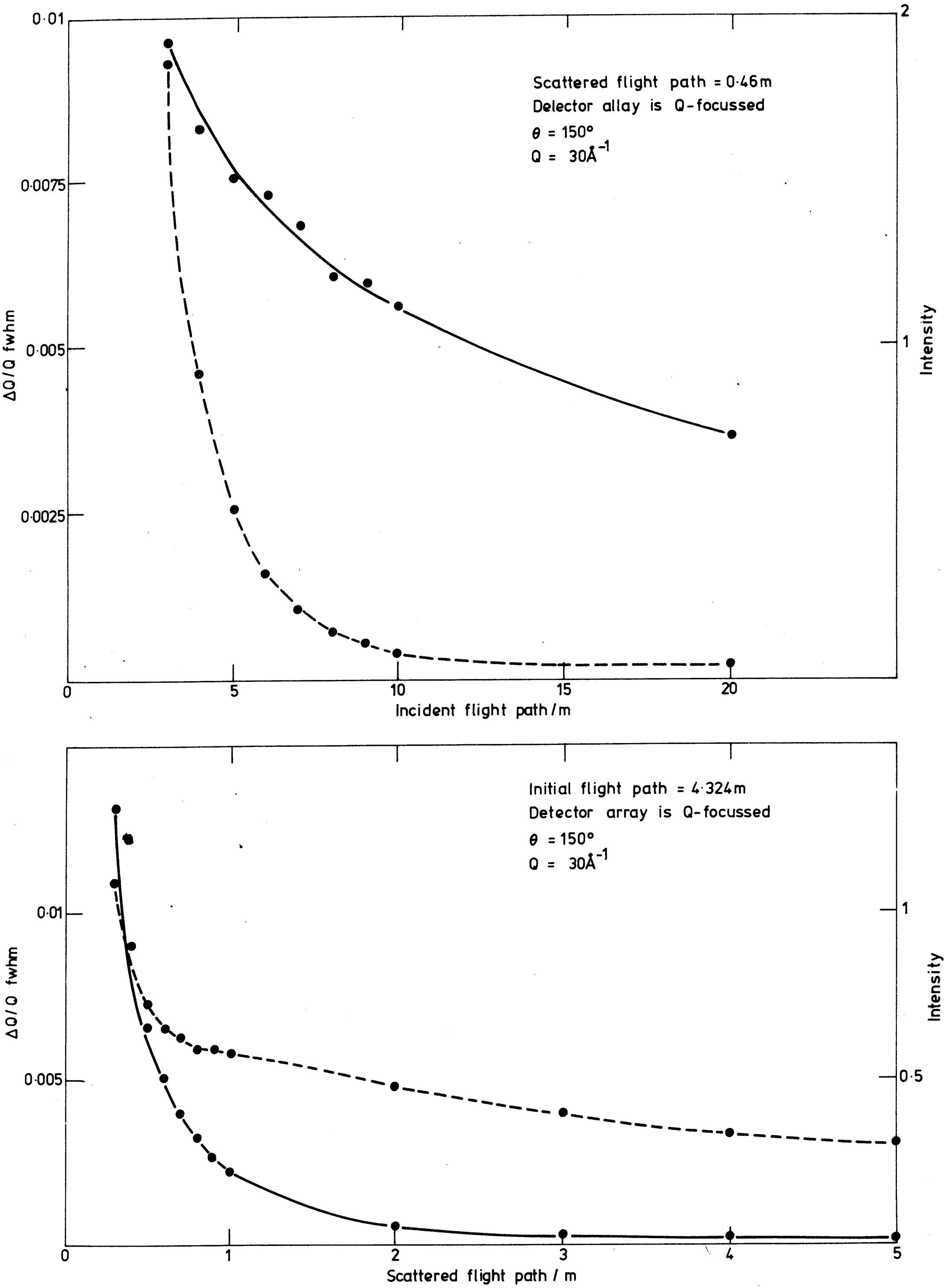
AERE - R.9170 Fig. 15
Variation of Q-value resolution with moderator focussing angle



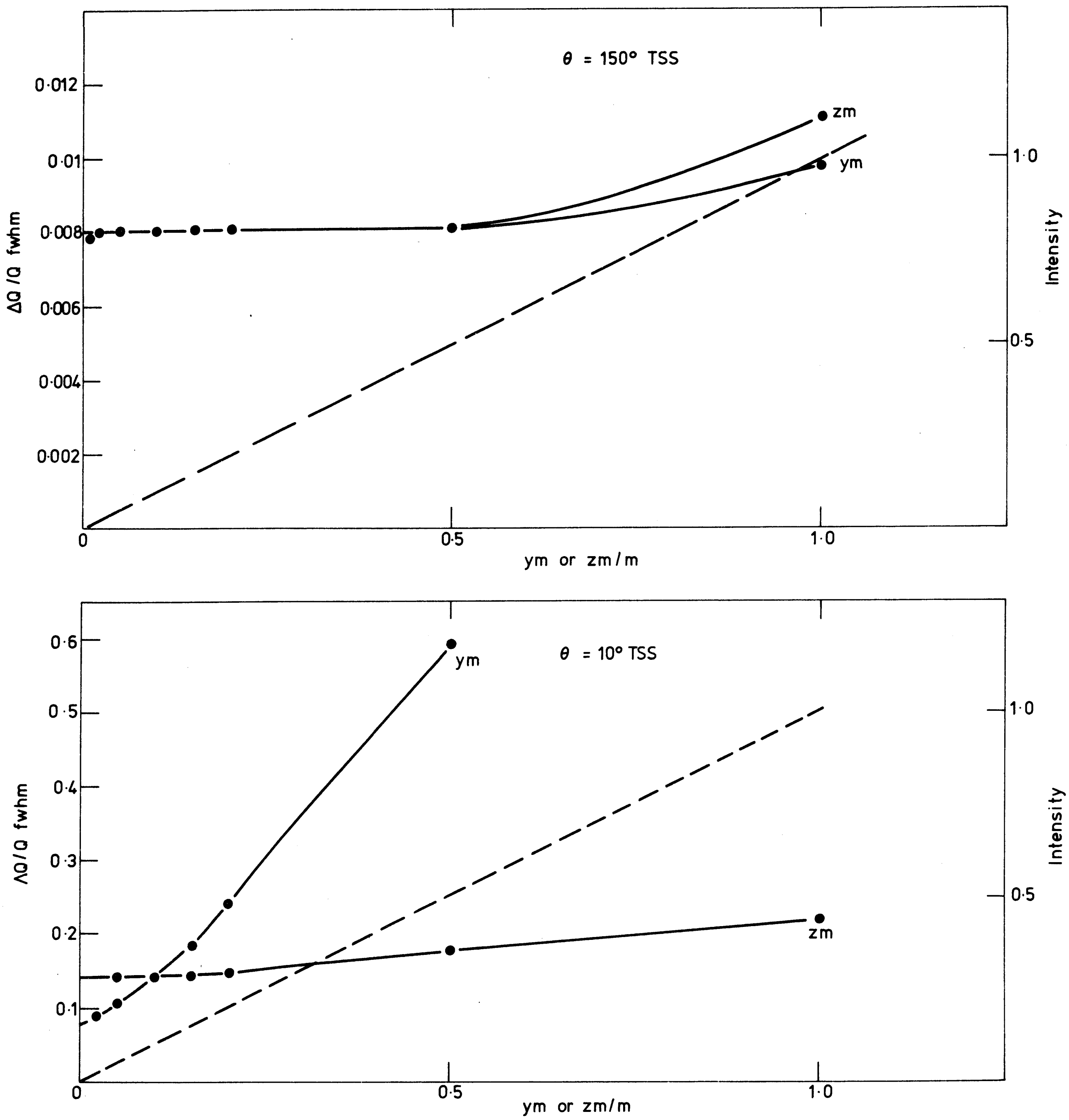
AERE - R.9170 Fig. 16
Variation of Q-value resolution with detector focussing angle



AERE - R.9170 Fig. 17
Variation of Q-value resolution with sample focussing angle



AERE - R.9170 Fig. 18
Variation of fractional Q-resolution and intensity with initial and secondary flight paths for the TSS as outlined in Table I



AERE - R.9170 Fig. 19
Variation of fractional Q-resolution and intensity with moderator dimensions for the TSS as outlined in Table I

```
PIP *.CRF;*/DE
PIP *.BIN;*,*.LST;*,*.MAP;*,SIMON.TSK;*/DE
PIP *.*/PU
PIP DK0:*.*/PU
>
```

Indirect file CLEAN.CMD for clearing disk prior to

Task-building SIMON.TSK

```
SIMON/-CP,SIMON/-SP/SH/CR=TEST/MP
UNITS=8
ASG=GR0:1
COMMON=DFILE:RW
MAXBUF=512
//
>
```

Indirect file TEST.CMD for Task-building SIMON.TSK with

overlaid subroutines.

```
        .ROOT   SIMON-BLKDAT-F8-F9-*(F1,F2,F3,F4,F5,F6,F7)
F1:     .FCTR   READAT-ICHNGE-RCHNGE-LCHNGE-OPTION-NEWV-G4-BATCHR
F2:     .FCTR   LPDATA
F3:     .FCTR   PRELIM
F4:     .FCTR   DRWDAT-ROD-PLATE-G4
F5:     .FCTR   CALC-HR-G1-G2-G4
F6:     .FCTR   OUTDAT
F7:     .FCTR   PLTDAT-SAVDAT-LINES-DATPLT-ASK-SUM-SUMDAT-AXESS-DATCAP-G4
F8:     .FCTR   VERDAT-G3
F9:     .FCTR   ENDP
G1:     .FCTR   MODTR-SAMPLT-SAMROD-DETYDS-DETNDS
G2:     .FCTR   CHISQQ-HALT-GEOMPS-GEOMDT-GEOMPT-RSCALE
G3:     .FCTR   [1,1]MAPPED-[1,1]PEPLIB/LB
G4:     .FCTR   [1,1]GLIB/LB
        .END
>
```

Overlay Descriptor Language file TEST.ODL for specifying the overlay

format used in Task-building SIMON.TSK

```
PIP *.CRF;*,*.LST;*,*.OBJ;*/DE
PIP DK0:*.*/PU,DK1:*.*/PU
FOR DK1:SIMON,SIMON/NOSP=DK0:SIMON/NOSN
FOR DK1:ICHNGE,ICHNGE/NOSP=DK0:ICHNGE/NOSN
FOR DK1:OPTION,OPTION/NOSP=DK0:OPTION/NOSN
FOR DK1:HALT,HALT/NOSP=DK0:HALT/NOSN
FOR DK1:PLATE,PLATE/NOSP=DK0:PLATE/NOSN
FOR DK1:RCHNGE,RCHNGE/NOSP=DK0:RCHNGE/NOSN
FOR DK1:HR,HR/NOSP=DK0:HR/NOSN
FOR DK1:LCHNGE,LCHNGE/NOSP=DK0:LCHNGE/NOSN
FOR DK1:SAMPLT,SAMPLT/NOSP=DK0:SAMPLT/NOSN
FOR DK1:MODTR,MODTR/NOSP=DK0:MODTR/NOSN
FOR DK1:CHISQQ,CHISQQ/NOSP=DK0:CHISQQ/NOSN
FOR DK1:SAMROD,SAMROD/NOSP=DK0:SAMROD/NOSN
FOR DK1:GEOMPS,GEOMPS/NOSP=DK0:GEOMPS/NOSN
FOR DK1:NEWV,NEWV/NOSP=DK0:NEWV/NOSN
FOR DK1:DATPLT,DATPLT/NOSP=DK0:DATPLT/NOSN
FOR DK1:AXESS,AXESS/NOSP=DK0:AXESS/NOSN
FOR DK1:DETNDS,DETNDS/NOSP=DK0:DETNDS/NOSN
FOR DK1:BLKDAT,BLKDAT/NOSP=DK0:BLKDAT/NOSN
FOR DK1:LPDATA,LPDATA/NOSP=DK0:LPDATA/NOSN
FOR DK1:RSCALE,RSCALE/NOSP=DK0:RSCALE/NOSN
FOR DK1:DETYDS,DETYDS/NOSP=DK0:DETYDS/NOSN
FOR DK1:LINES,LINES/NOSP=DK0:LINES/NOSN
FOR DK1:SAVDAT,SAVDAT/NOSP=DK0:SAVDAT/NOSN
FOR DK1:SUM,SUM/NOSP=DK0:SUM/NOSN
FOR DK1:GEOMPT,GEOMPT/NOSP=DK0:GEOMPT/NOSN
FOR DK1:ROD,ROD/NOSP=DK0:ROD/NOSN
FOR DK1:DATCAP,DATCAP/NOSP=DK0:DATCAP/NOSN
FOR DK1:ASK,ASK/NOSP=DK0:ASK/NOSN
FOR DK1:SUMDAT,SUMDAT/NOSP=DK0:SUMDAT/NOSN
FOR DK1:READAT,READAT/NOSP=DK0:READAT/NOSN
FOR DK1:PRELIM,PRELIM/NOSP=DK0:PRELIM/NOSN
FOR DK1:CALC,CALC/NOSP=DK0:CALC/NOSN
FOR DK1:DRWDAT,DRWDAT/NOSP=DK0:DRWDAT/NOSN
FOR DK1:OUTDAT,OUTDAT/NOSP=DK0:OUTDAT/NOSN
FOR DK1:PLTDAT,PLTDAT/NOSP=DK0:PLTDAT/NOSN
FOR DK1:VERDAT,VERDAT/NOSP=DK0:VERDAT/NOSN
FOR DK1:ENDP,ENDP/NOSP=DK0:ENDP/NOSN
FOR DK1:GEOMDT,GEOMDT/NOSP=DK0:GEOMDT/NOSN
FOR DK1:BATCHR,BATCHR/NOSP=DK0:BATCHR/NOSN
@CLEAN
TKB  @TEST
>
```

Indirect file for compiling and Task-building SIMON.TSK

```
 C==========     SIMON    ==========================================
C        SIMULATION OF A TIME OF FLIGHT DIFFRACTOMETER               =
C        MODERATOR IS AN INFINITELY THIN SHEET YM X ZM               =
C        SAMPLE IS SHEET YS X ZS THICKNESS TS                        =
C                       OR A CYLINDER DIAMETER YS HEIGHT ZS          =
C        DETECTOR IS A SHEET YD X ZD THICKNESS TD                    =
C=====================================================================
C
        COMMON  /LOGICS/LPRINT,L(4),IER,RESTRT,PLOTX,DISPLY,BATCH
        REAL    FNAME(4)
        LOGICAL LPRINT,IER,RESTRT,PLOTX,L,DISPLY,PLOTY,BATCH
        DATA PLOTY/F/,BATCH/F/,LUN/8/,FNAME/'    ','    ','    ','    '/
C
                        TYPE 6          !PROMPT FOR BATCH OPTION
                        ACCEPT 7,BATCH  !ACCEPT BATCH OPTION
        IF(BATCH)       TYPE 8          !PROMPT FOR FILENAME
        IF(BATCH)       ACCEPT 9,FNAME  !BATCH FILENAME
C====   ASSIGN ACCEPT DEVICE TO FILE FNAME
        IF(BATCH)       CALL ASSIGN(LUN,FNAME,16)
C====   ASSIGN LINEPRINTER TO SPOOL FILE
                        CALL ASSIGN(6,'DK1:LP.SPL',10)
                        WRITE(6,5)      !PRINT HEADING
2                       CALL READAT     !INPUT DATA
        IF(LPRINT)      CALL LPDATA     !LIST DATA ON LINEPRINTER
1                       CALL PRELIM     !COMPUTE CONSTANTS
        IF(DISPLY)      CALL DRWDAT     !DRAW PLAN OF GEOMETRY
                        CALL CALC       !DO MONTE-CARLO LOOP
        IF(IER)         GOTO1           !RESTART AFTER ERROR
                        CALL OUTDAT     !OUTPUT RESULTS
                        CALL PLTDAT     !DISPLAY DATA
        IF(PLOTX)       CALL VERDAT     !SET UP VERSATEC PLOTTING
        IF(PLOTX )      PLOTY=.TRUE.    !SET END OF PLOT VARIABLE
        IF(RESTRT)      GOTO2           !GO TO RESTART
                        CALL ENDP(PLOTY)!TIDY UP PLOTS
                        CALL CLOSE(LUN)
                        STOP
C
C=====================================================================
6       FORMAT('$ DO YOU WANT TO RUN UNDER BATCH ? T OR F? ')
7       FORMAT(L1)
8       FORMAT('$ TYPE IN FILENAME EG. DK1:TSS.DAT ')
9       FORMAT(4A4)
5       FORMAT(1X,'SIMULATION BY MONTE-CARLO (SIMON)',/,
        1       1X,'A CODE TO SIMULATE A TOF DIFFRACTOMETER',/,
        2       1X,'ON A PULSED SOURCE,JHCLARKE JULY 1978',/)
C
C=====================================================================
                        END
```

```
      SUBROUTINE READAT        !READS IN ALL CONTROL DATA AND PARAMETERS
C==================================================================
      COMMON  /LOGICS/LPRINT,GT44,SROD,SPLATE,DSCONE,L(3),DISPLY,BATCH
     2        /ANGLES/THETAM,THETAS,THETAD,THETA0,
     3                THETRM,THETRS,THETRD,THETR0
     4        /SIZES /YM,ZM, XC,YC,ZC, YS,ZS,TS,RS, YD,ZD,TD,HTD,TC,RC
     5        /FPATHS/Q0,RMS0,RSD0,WAVE,T0,EFFD,F1,F2
     6        /TCONS /TESRCE,PATH,DT,FREEDM
     7        /INTGRS/TITLE(18),NTRANG,NHIST,I1,I2,I(6)
      LOGICAL LPRINT,GT44,SROD,SPLATE,DSCONE,L,DISPLY,BATCH
C
      IF(BATCH)       CALL BATCHR       !READ BATCH PARAMETERS
      IF(BATCH)       GOTO1000          !RETURN AND CONTINUE WITH SIMON
                      TYPE 98           !PROMPT FOR TITLE
                      ACCEPT99,TITLE    !READ IN TITLE
C
                      TYPE 1,LPRINT     !SWITCH ON LINEPRINTER
                      CALL LCHNGE(LPRINT)
C
                      TYPE 2,GT44       !CHECK SCREEN SIZE GT40 OR GT44
                      CALL LCHNGE(GT44)
                      TYPE 201,DISPLY !SWITCH ON DISPLAY
                      CALL LCHNGE(DISPLY)
C
                      TYPE 3,THETAM     !MODERATOR ANGLE
                      CALL RCHNGE(THETAM)
                      TYPE 4,YM         !MODERATOR WIDTH
                      CALL RCHNGE(YM)
                      TYPE 5,ZM         !MODERATOR HEIGHT
                      CALL RCHNGE(ZM)
C
D                     TYPE 6,XC         !COLLIMATOR POSITION
D                     CALL RCHNGE(XC)
D                     TYPE 7,YC         !COLLIMATOR WIDTH
D                     CALL RCHNGE(YC)
D                     TYPE 8,ZC         !COLLIMATOR HEIGHT
D                     CALL RCHNGE(ZC)
C
100                   SPLATE=.FALSE.
                      SROD  =.FALSE.
                      TYPE 9            !CHECK SAMPLE IS ROD OR PLATE
                      ACCEPT10,SPLATE
      IF(.NOT.SPLATE) SROD=.TRUE.
      IF(     SPLATE) SROD=.FALSE.
C
      IF(SPLATE)      TYPE11,THETAS     !SAMPLE ANGLE IF A PLATE SAMPLE
      IF(SPLATE)      CALL RCHNGE(THETAS)
      IF(.NOT.SPLATE.AND..NOT.SROD)GOTO100
      IF(     SPLATE.AND.     SROD)GOTO100
C
      IF(SPLATE)      TYPE 12,YS        !SAMPLE WIDTH
      IF(SROD  )      TYPE 13,YS        !SAMPLE DIAMETER
                      CALL RCHNGE(YS)
      IF(SROD  )      RS=.5*YS
                      TYPE 14,ZS        !SAMPLE HEIGHT
                      CALL RCHNGE(ZS)
      IF(SPLATE)      TYPE 15,TS        !SAMPLE THICKNESS
      IF(SPLATE)      CALL RCHNGE(TS)
C
```

```
                        TYPE 16,THETAD  !DETECTOR ANGLE
                        CALL RCHNGE(THETAD)
                        TYPE 17,YD      !DETECTOR WIDTH
                        CALL RCHNGE(YD)
                        TYPE 18,ZD      !DETECTOR HEIGHT
                        CALL RCHNGE(ZD)
                        TYPE 19         !DEBYE-SCHERRER CONE
                        ACCEPT10,DSCONE
      IF(DSCONE)        TYPE 190
      IF(DSCONE)        THETAD=0.       !THETAD IS SET ZERO WITH D-S TRUE
                        TYPE 20,TD      !DETECTOR THICKNESS
                        CALL RCHNGE(TD)
                        TYPE 200,EFFD   !DETECTOR EFFICIENCY
                        CALL RCHNGE(EFFD)
C
                        TYPE 21,THETA0  !SCATTERING ANGLE
                        CALL RCHNGE(THETA0)
C
                        TYPE 22,RMS0    !MODERATOR TO SAMPLE
                        CALL RCHNGE(RMS0)
                        TYPE 23,RSD0    !SAMPLE TO DETECTOR DISTANCE
                        CALL RCHNGE(RSD0)
C
                        TYPE 24,Q0      !Q-VALUE
                        CALL RCHNGE(Q0)
C
                        TYPE 25,TESRCE  !RECTANGULAR SOURCE PULSE WIDTH
                        CALL RCHNGE(TESRCE)
C
                        TYPE 26,PATH    !MODERATOR MEAN FREE PATH
                        CALL RCHNGE(PATH)
                        TYPE 261,FREEDM !NO. OF DEGREES OF FREEDOM
C
                        DT=1.           !RESET DT TO SMALLEST VALUE
                        TYPE 27,DT      !TIME CHANNEL WIDTH
                        CALL RCHNGE(DT)
C
                        TYPE 28,NHIST   !NUMBER OF MONTE CARLO EVENTS
                        CALL ICHNGE(NHIST)
                        TYPE 29
C
                        AC=.0174532925  !CONVERT ANGLES FROM DEGREES
                        THETRM=THETAM*AC!TO RADIANS
                        THETRS=THETAS*AC
                        THETRD=THETAD*AC
                        THETR0=THETA0*AC*.5
C
      IF(SROD )         TS=YS
                        RETURN
```

```
1000                    TYPE 3,THETAM    !MODERATOR ANGLE
                        TYPE 4,YM        !MODERATOR WIDTH
                        TYPE 5,ZM        !MODERATOR HEIGHT
        IF(SPLATE)      TYPE11,THETAS    !SAMPLE ANGLE IF A PLATE SAMPLE
        IF(SPLATE)      TYPE 12,YS       !SAMPLE WIDTH
        IF(SROD  )      TYPE 13,YS       !SAMPLE DIAMETER
                        TYPE 14,ZS       !SAMPLE HEIGHT
        IF(SPLATE)      TYPE 15,TS       !SAMPLE THICKNESS
                        TYPE 16,THETAD   !DETECTOR ANGLE
                        TYPE 17,YD       !DETECTOR WIDTH
                        TYPE 18,ZD       !DETECTOR HEIGHT
        IF(DSCONE)      TYPE 190
                        TYPE 20,TD       !DETECTOR THICKNESS
                        TYPE 200,EFFD    !DETECTOR EFFICIENCY
                        TYPE 21,THETAO   !SCATTERING ANGLE
                        TYPE 22,RMSO     !MODERATOR TO SAMPLE
                        TYPE 23,RSDO     !SAMPLE TO DETECTOR DISTANCE
                        TYPE 24,QO       !Q-VALUE
                        TYPE 25,TESRCE   !RECTANGULAR SOURCE PULSE WIDTH
                        TYPE 26,PATH     !MODERATOR MEAN FREE PATH
                        TYPE 261,FREEDM  !NO. OF DEGREES OF FREEDOM
                        TYPE 27,DT       !TIME CHANNEL WIDTH
                        TYPE 28,NHIST    !NUMBER OF MONTE CARLO EVENTS
C
                        AC=.0174532925  !CONVERT ANGLES FROM DEGREES
                        THETRM=THETAM*AC                  !TO RADIANS
                        THETRS=THETAS*AC
                        THETRD=THETAD*AC
                        THETRO=THETAO*AC*.5
                        RETURN
```

```
C
C====   INPUT SECTION FORMATS ===================================
1       FORMAT(1H1,/,' LINEPRINTER OPTION IS ',L6)
2       FORMAT(/,' GT44 SCREEN IS ',L6)
201     FORMAT(/,' DISPLAY OPTION IS ',L6)
3       FORMAT(/,' MODERATOR ANGLE=',T25,F10.5)
4       FORMAT(/,' MODERATOR WIDTH=',T25,F10.5,' METRES')
5       FORMAT(/,' MODERATOR HEIGHT=',T25,F10.5,' METRES')
6       FORMAT(/,' COLLIMATOR POSITION=',T25,F10.5,' METRES')
7       FORMAT(/,' COLLIMATOR WIDTH=',T25,F10.5,' METRES')
8       FORMAT(/,' COLLIMATOR HEIGHT=',T25,F10.5,' METRES')
9       FORMAT(/,'$ IS THE SAMPLE A  FLAT PLATE T OR F? ')
10      FORMAT(L1)
11      FORMAT(/,' SAMPLE ANGLE=',T25,F10.5)
12      FORMAT(/,' SAMPLE WIDTH=',T25,F10.5,' METRES')
13      FORMAT(/,' SAMPLE DIAMETER',T25,F10.5,' METRES')
14      FORMAT(/,' SAMPLE HEIGHT=',T25,F10.5,' METRES')
15      FORMAT(/,' SAMPLE THICKNESS=',T25,F10.5,' METRES')
16      FORMAT(/,' DETECTOR ANGLE=',T25,F10.5)
17      FORMAT(/,' DETECTOR WIDTH=',T25,F10.5,' METRES')
18      FORMAT(/,' DETECTOR HEIGHT=',T25,F10.5,' METRES')
19      FORMAT(/,' IS THE DETECTOR DEFINING SLIT SHAPED TO FIT ',/,
        1        '$ THE DEBYE SCHERRE CONE?? T OR F ?')
190     FORMAT(/,' DEBYE-SCHERRE CONES ARE USED SO FOCUSSING ANGLE',/,
        1       ' THETAD IS ZERO')
20      FORMAT(/,' DETECTOR THICKNESS=',T25,F10.5,' METRES')
200     FORMAT(/,' DETECTOR EFFICIENCY AT 1.0 ANGSTROM =',T40,F10.5)
21      FORMAT(/,' SCATTERING ANGLE=',T25,F10.5)
22      FORMAT(/,' MODERATOR TO SAMPLE=',T25,F10.5,' METRES')
23      FORMAT(/,' SAMPLE TO DETECTOR=',T25,F10.5,' METRES')
24      FORMAT(/,' Q-VALUE=',T25,F10.5,' ANGSTROM**-1')
25      FORMAT(/,' RECTANGULAR SOURCE PULSE WIDTH=',T35,F10.5,' USECS')
26      FORMAT(/,' MODERATOR MEAN FREE PATH=',T35,F10.5,' METRES')
261     FORMAT(/,' NO. OF DEGREES OF FREEDOM=',T35,F5.0)
27      FORMAT(/,' TIMECHANNEL WIDTH=',T25,F10.5,' USECS')
28      FORMAT(/,' NO. OF PATHS=',T25,I10)
29      FORMAT( /////////////////)
99      FORMAT(18A4)
97      FORMAT(' DO YOU WANT TO RUN IN BATCH MODE (T OR F)?')
98      FORMAT(/,' TYPE IN TITLE UP TO 72 CHARACTERS ')
C===============================================================
        END
```

```
      SUBROUTINE BATCHR       !READS IN ALL CONTROL DATA AND PARAMETERS
C===================================================================
      COMMON  /LOGICS/LPRINT,GT44,SROD,SPLATE,DSCONE,L(3),DISPLY,BATCH
      2       /ANGLES/THETAM,THETAS,THETAD,THETAO,
      3               THETRM,THETRS,THETRD,THETRO
      4       /SIZES /YM,ZM, XC,YC,ZC, YS,ZS,TS,RS, YD,ZD,TD,HTD,TC,RC
      5       /FPATHS/QO,RMSO,RSDO,WAVE,TO,EFFD,F1,F2
      6       /TCONS /TESRCE,PATH,DT,FREEDM
      7       /INTGRS/TITLE(18),NTRANG,NHIST,I1,I2,I(6)
C===================================================================
      REAL    R(19)
      LOGICAL LPRINT,GT44,SROD,SPLATE,DSCONE,L,DISPLY,BATCH,LDD(6)
C====  READ ALL PARAMETERS IN A BATCH FORM ==============================
                        SPLATE=.FALSE.
                        SROD  =.FALSE.
                        READ(8,1)TITLE,LDD,IDD,R
      IF(LDD( 1))       LPRINT  =.TRUE.
      IF(LDD( 2))       GT44    =.TRUE.
      IF(LDD( 3))       DISPLY  =.TRUE.
      IF(LDD( 4))       SROD    =.TRUE.
      IF(LDD( 5))       SPLATE  =.TRUE.
      IF(LDD( 6))       DSCONE  =.TRUE.
      IF(.NOT.LDD(1)) LPRINT  =.FALSE.
      IF(.NOT.LDD(2)) GT44    =.FALSE.
      IF(.NOT.LDD(3)) DISPLY  =.FALSE.
      IF(.NOT.LDD(4)) SROD    =.FALSE.
      IF(.NOT.LDD(5)) SPLATE  =.FALSE.
      IF(.NOT.LDD(6)) DSCONE  =.FALSE.
      IF(R( 1).NE.0.) THETAM  =R( 1)
      IF(R( 2).NE.0.) YM      =R( 2)
      IF(R( 3).NE.0.) ZM      =R( 3)
      IF(R( 4).NE.0.) THETAS  =R( 4)
      IF(R( 5).NE.0.) YS      =R( 5)
      IF(R( 6).NE.0.) ZS      =R( 6)
      IF(R( 7).NE.0.) TS      =R( 7)
      IF(R( 8).NE.0.) THETAD  =R( 8)
      IF(R( 9).NE.0.) YD      =R( 9)
      IF(R(10).NE.0.) ZD      =R(10)
      IF(R(11).NE.0.) TD      =R(11)
      IF(R(12).NE.0.) EFFD    =R(12)
      IF(R(13).NE.0.) THETAO  =R(13)
      IF(R(14).NE.0.) RMSO    =R(14)
      IF(R(15).NE.0.) RSDO    =R(15)
      IF(R(16).NE.0.) QO      =R(16)
      IF(R(17).NE.0.) TESRCE  =R(17)
      IF(R(18).NE.0.) PATH    =R(18)
      IF(R(19).NE.0.) DT      =R(19)
      IF(IDD  .NE.0 ) NHIST   =IDD
      IF(.NOT.LDD(5)  )       SROD=.TRUE.
      IF(LDD(6)       )       THETAD=0.
      IF(LDD(4)      )        RS=0.5*YS          !SET RADIUS OF ROD SAMPLE
      IF(LDD(4)  )      TS=YS
1     FORMAT(18A4,/,6L1,/,I10,/,19(F12.5,/))
                        RETURN
                        END
```

```
      SUBROUTINE LCHNGE(L)      !ROUTINE TO READ CHANGED LOGICAL
      LOGICAL L,LD              !PARAMETERS FOR READAT
      CALL OPTION(LD)
      IF(.NOT.LD)RETURN
      CALL NEWV
      ACCEPT1,L
1     FORMAT(L4)
      RETURN
      END
      SUBROUTINE RCHNGE(R)      !REAL PARAMETERS
      LOGICAL LD
      CALL OPTION(LD)
      IF(.NOT.LD)RETURN
      CALL NEWV
      ACCEPT3,R
3     FORMAT(F12.5)
      RETURN
      END
      SUBROUTINE ICHNGE(I)      !INTEGER PARAMETERS
      LOGICAL LD
      CALL OPTION(LD)
      IF(.NOT.LD)RETURN
      CALL NEWV
      ACCEPT3,I
3     FORMAT(I10)
      RETURN
      END
      SUBROUTINE OPTION(LD)     !REQUEST ROUTINE
      LOGICAL LD
      TYPE 4
4     FORMAT('$CR FOR NO CHANGE,  T  TO CHANGE PARAMETER ')
      ACCEPT1,LD
1     FORMAT(L4)
      RETURN
      END
      SUBROUTINE NEWV           !NEW VALUE REQUEST
      TYPE2
2     FORMAT('$NEW VALUE=')
      RETURN
      END
```

```
      SUBROUTINE LPDATA       !PRINT ALL DATA ON LINEPRINTER
C=================================================================
      COMMON  /LOGICS/LPRINT,GT44,ROD,PLATE,DSCONE,L(5)
     2        /ANGLES/THETAM,THETAS,THETAD,THETAO,
     3                THETRM,THETRS,THETRD,THETRO
     4        /SIZES /YM,ZM, XC,YC,ZC, YS,ZS,TS,RS, YD,ZD,TD,HTD,TC,RC
     5        /FPATHS/QO,RMSO,RSDO,WAVE,TO,EFFD,F1,F2
     6        /TCONS /TESRCE,PATH,DT,FREEDM
     7        /INTGRS/TITLE(18),NTRANG,NHIST,I1,I2,I(6)
C=================================================================
      LOGICAL LPRINT,GT44,ROD,PLATE,DSCONE,L
      WRITE(6,1)TITLE,THETAM,THETAS,THETAD,THETAO
     1      ,YM,ZM, XC,YC,ZC, YS,TS,ZS, YD,ZD,TD,EFFD
     2      ,RMSO,RSDO,QO,DT,NTRANG,NHIST,TESRCE,PATH,FREEDM
      IF(ROD   )WRITE(6,2)
      IF(PLATE )WRITE(6,3)
      IF(DSCONE)WRITE(6,4)
C
C====  LINEPRINTER FORMATS=======================================
1     FORMAT(1X,18A4,/,
     1      ' MODERATOR ANGLE',T35,'=',F10.5,' DEGREES',/,
     1       ' SAMPLE ANGLE',T35,'=',F10.5,' DEGREES',/,
     2       ' DETECTOR ANGLE',T35,'=',F10.5,' DEGREES',/,
     3       ' 1/2 SCATTERING ANGLE',T35,'=',F10.5,' DEGREES',/,
     4       ' MODERATOR WIDTH',T35,'=',F10.5,' METRES',/,
     5       ' MODERATOR HEIGHT',T35,'=',F10.5,' METRES',/,
     6       ' COLLIMATOR POSITION=',T35,'=',F10.5,' METRES',/,
     7       ' COLLIMATOR WIDTH=',T35,'=',F10.5,' METRES',/,
     8       ' COLLIMATOR HEIGHT=',T35,'=',F10.5,' METRES',/,
     6       ' SAMPLE WIDTH',T35,'=',F10.5,' METRES',/,
     6       ' SAMPLE THICKNESS',T35,'=',F10.5,' METRES',/,
     7       ' SAMPLE HEIGHT',T35,'=',F10.5,' METRES',/,
     8       ' DETECTOR WIDTH',T35,'=',F10.5,' METRES',/,
     9       ' DETECTOR HEIGHT',T35,'=',F10.5,' METRES',/,
     1       ' DETECTOR THICKNESS',T35,'=',F10.5,' METRES',/,
     2       ' DETECTOR EFFICIENCY AT 1 ANGSTROM',T35,'=',F10.5,/,
     1       ' MODERATOR TO SAMPLE',T35,'=',F10.5,' METRES',/,
     2       ' SAMPLE TO DETECTOR',T35,'=',F10.5,' METRES',/,
     3       ' Q-VALUE',T35,'=',F10.5,' ANGSTROMS**-1',/,
     4       ' TIME CHANNEL WIDTH',T35,'=',F10.5,' MICROSECS',/,
     5       ' TIME SLOT WIDTH',T35,'=',I4,' CHANNELS',/,
     6       ' NUMBER OF PATHS',T35,'=',I8,/,
     7       ' PULSE WIDTH OF SOURCE',T35,'=',F3.1,' USECS',/,
     8       ' MEAN FREE PATH OF MODERATOR',T35,'=',F6.4,' METRES',/,
     9       ' NO. OF DEGREES OF FREEDOM',T35,'=',F6.3,//)
2     FORMAT(' SAMPLE IS A CYLINDER')
3     FORMAT(' SAMPLE IS A PLATE')
4     FORMAT(' DETECTORS LIE ON DEBYE-SCHERRER CONE')
      RETURN
      END
```

```
      SUBROUTINE PRELIM        !PRELIMINARY CONSTANTS ARE CALCULATED
      IMPLICIT LOGICAL (L-L)
C===================================================================
      COMMON  /ANGLES/THETAM,THETAS,THETAD,THETAO,
      2               THETRM,THETRS,THETRD,THETRO
      3       /LOGICS/LPRINT,L(9)
      4       /FPATHS/QO,RMSO,RSDO,WAVE,TO,EFFD,F1,F2
      5       /TCONS /TESRCE,PATH,DT,FREEDM
      4       /PRECON/XSO,XDO,CTHDPO,STHDPO,CTHM,STHM,CTHS,STHS,
      5               XDOC2T,XDOS2T,CTHD
      6       /INTGRS/TITLE(18),NTRANG,NPOS,I1,I2,NTMAX,NTMIN,NTO,I(3)
C===================================================================
      LOGICAL LPRINT
      DATA    PI/3.141592654/,FPI/12.56637061/
C
      TTHETR  =2.*THETRO                  !THETA SCATTERING
      XSO     =RMSO                       !SAMPLE X-COORDINATE
      XDO     =RSDO                       !DETECTOR TO SAMPLE DISTANCE
      THDPO   =THETRD+TTHETR              !THETA-SCATTERING+THETA-DETECTOR
      CTHDPO  =COS(THDPO)                 !COSINE OF THETAD+THETAO
      STHDPO  =SIN(THDPO)                 !SINE OF     "       "
      CTHM    =COS(THETRM)                !COSINE OF THETA MODERATOR
      STHM    =SIN(THETRM)                !SINE OF     "       "
      CTHS    =COS(THETRS)                !COSINE OF THETA  SAMPLE
      STHS    =SIN(THETRS)                !SINE OF THETA SAMPLE
      CTHD    =COS(THETRD)                !COSINE OF DETECTOR ANGLE
C
      XDOC2T  =XDO*COS(TTHETR)+XSO        !X-COORDINATE OF DETECTOR
      XDOS2T  =XDO*SIN(TTHETR)            !Y-COORDINATE OF DETECTOR
C
      WAVE    =ABS(FPI*SIN(THETRO)/QO)!WAVELENGTH OF NEUTRON
      NTMAX   =1                          !INITIAL VALUE OF TOP CHANNEL
      TO      =252.7*WAVE*(RMSO+RSDO) !TOF OF CENTRAL PATH
      IF(TO/DT.LT.32760.)     GOTO2       !INTEGER OVERFLOW UNLIKELY
      DT2     =2.*DT                      !SO DOUBLE CHANNEL WIDTH
                      TYPE1,DT,DT2        !AND SAY SO TO LINEPRINTER
      IF(LPRINT)      PRINT1,DT,DT2       !AND TELETYPE
      DT      =DT2                        !RESET DT TO NEW VALUE
2     NTO     =IFIX(TO/DT)                !CHANNEL NUMBER OF CENTRAL PATH
      NTMIN   =NTO+2                      !INITIAL VALUE OF BOTTOM CHANNEL
                      RETURN
C
1     FORMAT(' INTEGER OVERFLOW LIKELY , TIME CHANNEL TOO NARROW',/,
      1   ' OLD=',F5.2,' NEW=',F5.2)
                      END
```

```
      SUBROUTINE DRWDAT          !DRAWS PLAN OF SYSTEM
C
C=====================================================================
      COMMON  /ANGLES/THETAM,THETAS,THETAD,THETAO,
     1                THETRM,THETRS,THETRD,THETRO
     2        /SIZES /YM,ZM, XC,YC,ZC, YS,ZS,TS,RS, YD,ZD,TD,HTD,TC,RC
     3        /FPATHS/Q0,RMSO,RSDO,WAVE,TO,EFFD,F1,F2
     5        /LOGICS/LPRINT,GT44,LROD,LPLATE,L(6)
C============================================================================
      LOGICAL LPRINT,GT44,LROD,LPLATE,L
C
      DMSO   =RMSO*.05                   !SCALE DOWN RMSO BY 20
      TTHETR=2.*THETRO                   !SCATTERING ANGLE
      TTHEPD=TTHETR+THETRD               !THETA-SCATTER+THETA-DETECTOR
      C2TH   =COS(TTHETR)                !COSINE THETA-SCATTER
      S2TH   =SIN(TTHETR)                !SINE     "      "
      CTHPD  =COS(TTHEPD)                !COSINE THETA-DETECTOR
      STHPD  =SIN(TTHEPD)                !SINE THETA-DETECTOR
C
      XMO=.5*YM                          !MODERATOR CENTRE
      D   =YM*.5*SIN(THETRM)
      XM1=XMO+D                          !MODERATOR LOW X
      XM2=XMO-D                          !MODERATOR UPPER X
C
      YMO=0.                             !MODERATOR CENTRE
      D   =YM*.5*COS(THETRM)
      YM1=YMO-D
      YM2=YMO+D
C
      XSO=XMO+DMSO                       !SAMPLE CENTRE
      D   =.5*YS*SIN(THETRS)
      E   =.5*TS*COS(THETRS)
      XS1=XSO+D+E
      XS2=XSO-D+E
      XS3=XSO-D-E
      XS4=XSO+D-E
      XSMAX=AMAX1(XS1,XS2,XS3,XS4)       !MAXIMUM X
C
      YSO=YMO
      D   =.5*YS*COS(THETRS)
      E   =.5*TS*SIN(THETRS)
      YS1=YSO-D+E
      YS2=YSO+D+E
      YS3=YSO+D-E
      YS4=YSO-D-E
      YSMAX=AMAX1(YS1,YS2,YS3,YS4)       !MAXIMUM Y
      YSMIN=AMIN1(YS1,YS2,YS3,YS4)
```

```
C
      XD0=XS0+RSD0*C2TH                  !DETECTOR CENTRE
      D  =YD*.5*STHPD
      E  =TD*.5*CTHPD
      F  =YD*.5*CTHPD
      G  =TD*.5*STHPD
      XD1=XD0-D-E
      XD2=XD0-D+E
      XD3=XD0+D+E
      XD4=XD0+D-E
      XDMAX=AMAX1(XD1,XD2,XD3,XD4)       !MAXIMUM X
      YD0=RSD0*S2TH+YS0
      YD1=YD0+F-G
      YD2=YD0+F+G
      YD3=YD0-F+G
      YD4=YD0-F-G
      YDMAX=AMAX1(YD1,YD2,YD3,YD4)       !MAXIMUM Y
      YDMIN=AMIN1(YD1,YD2,YD3,YD4)
C
C
      AYMAX=1.01*AMAX1(YDMAX,YSMAX,YM1,YM2,YS0+YS*.5,0.)
      AXMAX=1.01*AMAX1(XDMAX,XSMAX)
      AYMIN=     AMIN1(YDMIN,YM1,YM2,YSMIN,YS0-YS*.5,0.)
      AXMIN=     AMIN1(XD1,XD2,XD4,XD3,XM1,XM2,0.)
      IF(.NOT.GT44)AYMAX=(AYMAX-AYMIN)*1024./768.
      AXYMAX=AMAX1(AYMAX-AYMIN,AXMAX-AXMIN)
C
      CALL INIT(2048)
      CALL SCAL(AXMIN,AYMIN,AXYMAX+AXMIN,AXYMAX+AYMIN)
C
      CALL APNT(XM1,YM1,-1,-6,-1,1)     !POSITION BOTTOM MODERATOR
      CALL VECT(XM2-XM1,YM2-YM1)        !JOIN TO TOP MODERATOR
C
      IF(LPLATE)
      1 CALL PLATE(XS1,YS1,XS2,YS2,XS3,YS3,XS4,YS4,XM0,YM0,XS0,YS0)    !PLATE
C
      IF(LROD)CALL ROD(RS,XM0,YM0,XS0,YS0)     !CYLINDRICAL SAMPLE
C
      CALL PLATE(XD1,YD1,XD2,YD2,XD3,YD3,XD4,YD4,XS0,YS0,XD0,YD0)      !DETECT
      RETURN
      END
```

```
C       ROUTINE TO DRAW A PLATE SAMPLE OR DETECTOR
        SUBROUTINE PLATE(A,B,C,D,E,F,G,H,P,Q,R,S)
        CALL APNT(P,Q,-1,-7,-1,2)          !POSITION CENTRE MODERATOR
        CALL VECT(R-P,S-Q)                 !JOIN CENTRE SAMPLE
        CALL APNT(A,B,-1,-6,-1,1)          !POSITION BOTTOM SAMPLE
        CALL VECT(C-A,D-B)                 !CORNER
        CALL VECT(E-C,F-D)                 !JOIN UP CORNERS
        CALL VECT(G-E,H-F)
        CALL VECT(A-G,B-H)
        RETURN
        END
        SUBROUTINE ROD(R,A,C,B,D)          !DRAW A CYLINDRICAL SAMPLE
C       RADIUS R,MODERATOR A,C SAMPLE B,D
        DATA M/60/                         !NO OF ANGLE STEPS
        CALL APNT(A,C,-1,-7,-1,2)          !POSITION CENTRE MODERATOR
        CALL VECT(B-A,D-C)                 !JOIN CENTRE SAMPLE
        CALL APNT(B+R,D,-1,-6,-1,1)        !POSITION X START
        DT=6.28318/FLOAT(M)                !ANGLE INCREMENT
        X1=R
        Y1=0.
        DO 1 N=1,M                !STEP ROUND 2 PI
        DP=FLOAT(N)*DT            !ANGLE INCREMENT
        X=R*COS(DP)               !NEW X
        Y=R*SIN(DP)               !NEW Y
        CALL VECT(X-X1,Y-Y1)      !JOIN OLD TO NEW
        X1=X                      !RESET OLD X TO NEW
        Y1=Y                      !RESET OLD Y TO NEW
1       CONTINUE
        RETURN
        END
```

```
      SUBROUTINE CALC           !MAIN MONTE CARLO SECTION
C
      IMPLICIT LOGICAL (L-L)
C=====================================================================
      COMMON  /LOGICS/LPRINT,GT44,ROD,PLATE,DSCONE,LR,L(2),DISPLY,BATCH
      1       /ANGLES/A(7),THETRO
      3       /TCONS /TESRCE,PATH,DT,FREEDM
      3       /SIZES /YM,ZM, XC,YC,ZC, YS,ZS,TS,RS, YD,ZD,TD,HTD,TC,RC
      5       /FPATHS/Q0,RMSO,RSDO,WAVE,TO,EFFD,F1,F2
      4       /PRECON/XSO,XDO,CTHDPO,STHDPO,CTHM,STHM,CTHS,STHS,
      5              XDOC2T,XDOS2T,CTHD
      6       /SCORE /RSCORE(100)
      7       /ELEVTN/YMP,ZMP,YSP,ZSP,YDP,ZDP,TDP,YMO,YSO,YD10,YD20,NYM
      8       /PLAN  /XNM,YNM,XNS,YNS,XND,YND,RMS1,XMO1
      9       /INTGRS/TITLE(18),NTRANG,NHIST,I1,I2,NMAX,NMIN,NTO,N1,N2,NUM
C=====================================================================
      LOGICAL LPRINT,GT44,ROD,PLATE,DSCONE,DISPLY,BATCH
C
      LR   =.FALSE.                               !SET ERROR TO FALSE
      LC   =.FALSE.
      LD   =.FALSE.
      CON  =12.56637061*252.7/Q0
      NT1  =NTO-NTRANG/2
      RMS1 =RMSO                          !MOD-SAMPLE DISTANCE FOR GEOMPS
      XMO1 =.5*YM                         !MOD X-CENTRE FOR GEOMPS
C                                         !MEAN MODERATOR TIME CONSTANT
      TMDMN=FREEDM*PATH*.5*TO/(RMSO+RSDO)
      TMDMN=PATH*.5*(TO+TMDMN)
      HTD  =TD*.5                         !PENETRATION DEPTH OF DETECTOR
      F1   =EFFD*WAVE                     !MU*LAMBDA
      F2   =1.-EXP(-F1*TD/CTHD)           !1-EXP(-MU*LAMBDA*T/COS(THETAD))
      RC   =XDOS2T                        !RADIUS OF DEBYE SCHERRER CONE
C
C====  CHECK THAT CONE IS LESS THAN A SEMICIRCLE =====================
      IF(DSCONE.AND.ZD.GE.2.*RC)          ZD=1.9*RC
      TC   =2.*ATAN(ZD*.5/RC)             !ANGLE OF CONE HEIGHT
C
      DO 600 NT=1,100                     !INITIALIZE COUNT ARRAY
600   RSCORE(NT)=0.
C
      DO 500 NYM=1,NHIST                  !SCAN ALL PATHS
      CALL MODTR                          !MODERATOR COORDINATES
      IF(PLATE)        CALL SAMPLT        !PLATE SAMPLE
      IF(ROD  )        CALL SAMROD        !ROD SAMPLE
      IF(      DSCONE) CALL DETYDS        !DETECTOR WITH D-S CONE
      IF(.NOT.DSCONE) CALL DETNDS         !DETECTOR WITHOUT D-S CONE
C
      RMSS  =DI(XNS,XNM,YNS,YNM,ZSP,ZMP)      !(MODERATOR-SAMPLE)**2
      RSDS  =DI(XND,XNS,YND,YNS,ZDP,ZSP)      !(SAMPLE-DETECTOR)**2
      RMDS  =DI(XND,XNM,YND,YNM,ZDP,ZMP)      !(MODERATOR-DETECTOR)**2
      RMS   =SQRT(RMSS)
      RSD   =SQRT(RSDS)
      RMSD  =RMS+RSD                          !FLIGHT PATH
C
      COSTHT=.5*(RMSS+RSDS-RMDS)/(RMS*RSD)    !COS 180-2*THETR
      SINTHT=SQRT((1.+COSTHT)*.5)
      TINT  =CON*SINTHT*RMSD                  !INTRINSIC T-O-F
      TSRC  =HR(I1,I2)*TESRCE                 !SOURCE EMISSION TIME
      TMOD  =CHISQQ(I1,I2)*TMDMN/RMSD         !MODERATOR EMISSION TIME
```

```
C
      NT    =IFIX((TINT+TSRC+TMOD)/DT)-NT1                  !TIME CHANNEL
      IF(NT.GT.NMAX)                    NMAX=NT              !IN SHIFTED SYSTEM
      IF(NT.LT.NMIN)                    NMIN=NT              !UPPER & LOWER BOUNDS
      IF(NT.GT.NTRANG.OR.NT.LT.1)       CALL HALT(DT,LR)!ERROR T CHANNEL TOO SM
      IF(LR)                            RETURN               !ERROR ESCAPE
      RSCORE(NT)=RSCORE(NT)+1.                               !UPDATE SCORE ARRAY
C
C====  DISPLAY PROCESSES ON SCREEN ==========================================
      IF(.NOT.DISPLY)                   GOTO500              !ESCAPE DISPLAY
      IF(RMS1.NE.-999.999)              CALL GEOMPS          !DRAW PLAN
      IF(RMS1.EQ.-999.999.AND..NOT.LD)LC=.TRUE.              !END OF PLAN
      IF(.NOT.LC)                       GOTO499
                                        CALL INIT            !CLEAR SCREEN
                                        CALL GEOMDT
                                        LD=.TRUE.            !SWITCH ON ELEVATION
499                                     LC=.FALSE.           !SWITCH OFF PLAN
      IF(LD.AND.YMO.NE.-999.999)        CALL GEOMPT          !DRAW ELEVATION
500                                     CONTINUE
                                        CALL RSCALE          !SCALE RSCORE BY
                                        RETURN               !SOLID ANGLES ETC.
                                        END
C
      FUNCTION DI(A,B,C,D,E,F)
      DI=(A-B)**2+(C-D)**2+(E-F)**2
      RETURN
      END
```

```
      SUBROUTINE MODTR           !MODERATOR COORDINATES
      COMMON  /SIZES /YM,ZM, R(13)
      1       /PRECON/R2(4),CTHM,STHM,R3(5)
      2       /ELEVTN/YMP,ZMP,R4(9),NYM
      3       /PLAN  /XNM,YNM,R5(6)
      4       /INTGRS/TITLE(18),N1,N2,I1,I2,N3(6)
      YMP= YM*HR(I1,I2)          !Y POSITION
      XNM=-YMP*STHM
      YNM= YMP*CTHM
      ZMP= ZM *HR(I1,I2)
      RETURN
      END
      SUBROUTINE SAMPLT                  !PLATE SAMPLE COORDINATES
      COMMON  /SIZES /R1(5), YS,ZS,TS, R2(7)
      1       /PRECON/XSO,R3(5),CTHS,STHS,R4(3)
      2       /ELEVTN/YMP,ZMP,YSP,ZSP,R5(7),N
      3       /PLAN  /XNM,YNM,XNS,YNS,R6(4)
      4       /INTGRS/TITLE(18),N1,N2,I1,I2,N3(6)
      YSP=YS*HR(I1,I2)                   !Y-COORDINATE
      TSP=TS*HR(I1,I2)                   !THICKNESS
      XNS=XSO-YSP*STHS+TSP*CTHS
      YNS=    YSP*CTHS+TSP*STHS
      ZSP=ZS*HR(I1,I2)
      RETURN
      END
      SUBROUTINE SAMROD                  !ROD SAMPLE COORDINATES
      COMMON  /SIZES /R1(5), YS,ZS,TS,RS, R2(6)
      1       /PRECON/XSO,R3(5),CTHS,STHS,R4(3)
      2       /ELEVTN/YMP,ZMP,YSP,ZSP,R5(7),NYM
      3       /PLAN  /XNM,YNM,XNS,YNS,R6(4)
      4       /INTGRS/TITLE(18),N1,N2,I1,I2,N3(6)
      THP=RAN(I1,I2)*6.28318             !RANDOM ANGLE
      YSP=RAN(I1,I2)*RS                  !RANDOM RADIUS
      XNS=XSO-YSP*SIN(THP)
      YNS=YSP*COS(THP)
      ZSP=ZS*HR(I1,I2)
      RETURN
      END
      SUBROUTINE HALT(D,I)      !ERROR RETRUN WHEN T CHANNEL TO NARROW
      LOGICAL I
      E=2.*D            !DOUBLE DELTAT
      TYPE 2,D,E        !PRINT MESSAGE
      D=E               !REPLACE DELTAT
      I=.TRUE.          !SET LOGICAL TO
      RETURN            !ENABLE TRANSFER TO PRELIM FROM MAIN
2     FORMAT(' TIME CHANNEL TOO NARROW, OLD =',F5.2,' NEW=',F5.2)
      END
>
```

```
      SUBROUTINE DETYDS                       !DETECTOR COORDINATES WITH
C                                             !DEBYE-SCHERRER CONE
      COMMON  /INTGRS/TITLE(18),N1,N2,I,J,N3(6)
      1       /SIZES /R1(9),YD,ZD,TD,HTD,TC,RC
      2       /PRECON/XS0,XD0,CTHDP0,STHDP0,R3(4),XDOC2T,XDOS2T,CTHD
      3       /FPATHS/R4(6),F1,F2
      4       /ELEVTN/R5(4),YDP,ZDP,TDP,R6(4),N
      5       /PLAN  /R7(4),XND,YND,R8(2)
      6       /ANGLES/R9(7),THETR0
      TDP=CTHD*ALOG(1./(1.-RAN(I,J)*F2))/F1-HTD!TRACK IN DETECTOR
      ADP=TC*HR(I,J)
      YDP=YD*HR(I,J)-RC*(1.-COS(ADP))
      XND=XDOC2T-YDP*STHDP0+TDP*CTHDP0
      YND=XDOS2T+YDP*CTHDP0-TDP*STHDP0
      ZDP=RC*ADP
      RETURN
      END
      SUBROUTINE DETNDS                       !DETECTOR COORDINATES WITHOUT
C                                             !DEBYE-SCHERRER CONE
      COMMON  /INTGRS/TITLE(18),N1,N2,I,J,N3(6)
      1       /SIZES /R1(9),YD,ZD,TD,HTD,R2(2)
      2       /PRECON/XS0,XD0,CTHDP0,STHDP0,R3(4),XDOC2T,XDOS2T,CTHD
      3       /FPATHS/R4(6),F1,F2
      4       /ELEVTN/R5(3),ZSP,YDP,ZDP,TDP,R6(4),N
      5       /PLAN  /R7(2),XNS,YNS,XND,YND,R8(2)
      6       /ANGLES/R9(6),THETRD,THETR0
      YDP   =YD*HR(I,J)                              !1ST DETECTOR Y-COORDINATE
      ZDP   =ZD*HR(I,J)                              !1ST DETECTOR Z-COORDINATE
      TDP   =CTHD*ALOG(1./(1.-RAN(I,J)*F2))/F1-HTD!TRACK IN DETECTOR
      XND   =XDOC2T-YDP*STHDP0+TDP*CTHDP0      !3RD X-COORDINATE
      YND   =XDOS2T+YDP*CTHDP0-TDP*STHDP0      !3RD Y-COORDINATE
      RETURN
      END
      FUNCTION CHISQQ(I1,I2)
C     !CHISQUARED ROUTINE WITH 6 DEGREES OF FREEDOM
      DIMENSION P(13),C(13)
      DATA P/.01,.02,.05,.1,.2,.3,.5,.7,.8,.9,.95,.98,.99/,
      1    C/16.812,15.033,12.592,10.645,8.558,7.231,5.348,
      2       3.828,3.070,2.204,1.635,1.134,.872/
                         R=RAN(I1,I2)     !RANDOM NO.
      IF(R.LT..01)       R=.01            !UPPER AND LOWER BOUNDS
      IF(R.GT..99)       R=.99            !OF TABULATED VALUES OF XSQ
      DO 10 K=1,13
      IF(R-P(K))         30,20,10
10    CONTINUE                            ! < SO INCREMENT
20    CHISQQ=C(K)                         ! EXACTLY =
      RETURN
30    J=K-1                               ! < K & > K-1 SO INTERPOLATE
      CHISQQ=C(J)+(R-P(J))*(C(K)-C(J))/(P(K)-P(J))
      RETURN
      END
>
```

```
      SUBROUTINE GEOMPS          !DRAWS VECTORS ON PLAN VIEW
      COMMON   /PLAN  /A,B,C,D,E,F,G,H
                      CALL DPTR(I)
      IF(I.GT.990)    GOTO3
                      CALL APNT(A+H,B,-1,-1,-1,1)
                      CALL VECT(C-.95*G-A,D-B)
                      CALL VECT(E-C,F-D)
                      RETURN
3                     G=-999.999
                      RETURN
                      END
C     DRAWS ELEVATION VIEW OF MODERATOR,SAMPLE AND DETECTOR FACES
      SUBROUTINE GEOMDT
C
      COMMON   /LOGICS/PRINT,GT44,L(2),DSCONE,L1(4)
      1        /SIZES /YM,ZM, XC,YC,ZC, YS,ZS,TS,RS, YD,ZD,TD,HTD,TC,RC
      2        /ELEVTN/D(7),YMO,YSO,YD10,YD20,NYM
      LOGICAL L,GT44,PRINT,DSCONE,L1
      YMO =YM*.7                 !CENTRES OF PICTURES MODERATOR
      YSO =YMO+YM+YS*.7          !SAMPLE
      YD10=YSO+YS+YD             !DETECTOR FACE
      YD20=YD10+YD+TD            !DETECTOR EDGE
C     SCALES ETC.
                      YMX =AMAX1(ZM,ZS,ZD)*.55          !Y MAXIMUM
                      YMN =-YMX                         !Y MINIMUM
      IF(.NOT.GT44)   YMX=1024.*(YMX-YMN)/768.+YMN
                      XMX =(YD20+TD)*1.2
                      A   =AMAX1(YMX-YMN,XMX)
C
                      CALL SCAL(O.,YMN,A,A)
                      CALL BOX(YMO,YM,ZM)        !MODERATOR FACE
                      CALL BOX(YSO,YS,ZS)        !SAMPLE FACE
      IF(.NOT.DSCONE) CALL BOX(YD10,YD,ZD)       !DETECTOR FACE
                      CALL BOX(YD20,TD,ZD)       !DETECTOR EDGE
                      RETURN
                      END
C
      SUBROUTINE BOX(Y,T,W)     !DISPLAY BOX CENTRE Y,WIDTH T,HEIGHT W
      DATA Z/O./
      CALL APNT(Y-T*.5,-W*.5,-1,-5,-1,1)
      CALL VECT(T,Z)
      CALL VECT(Z,W)
      CALL VECT(-T,Z)
      CALL VECT(Z,-W)
      RETURN
      END
      SUBROUTINE GEOMPT         !DISPLAY POINTS ON ELEVATION SURFACES
      COMMON   /ELEVTN/A,B,C,D,E,F,G,P,Q,R,S,N
                      CALL DPTR(I)
      IF(I.GT.990)    GOTO3
                      CALL APNT(P+A,B,-1,5,-1,1)
                      CALL APNT(Q+C,D,-1,5,-1,1)
                      CALL APNT(R-E,F,-1,5,-1,1)        !DETECTOR FROM BEHIND
                      CALL APNT(S+G,F,-1,5,-1,1)
                      RETURN
3                     P=-999.999
                      RETURN
                      END
```

```
C       SCALES RSCORE ACCORDING TO THE APPROPIATE SOLID ANGLES
C       THIS IS TO ENABLE A PROPER FIGURE OF MERIT TO BE EVALUATED
        SUBROUTINE RSCALE
        IMPLICIT LOGICAL(L-L)
        COMMON  /ANGLES/THETAM,THETAS,THETAD,THETAO,
        1               THETRM,THETRS,THETRD,THETRO
        2       /SIZES /YM,ZM, XC,YC,ZC, YS,ZS,TS,RS, YD,ZD,TD,HTD,TC,RC
        3       /FPATHS/Q0,RMSO,RSDO,WAVE,TO,EFFD,F1,F2
        4       /SCORE/RSCORE(100)
        5       /INTGRS/TITLE(18),I(4),N2,N1,I2(4)
        6       /LOGICS/L1,L2,LROD,LPLATE,DSCONE,L(5)
        7       /TCONS /TESRCE,PATH,DT,FREEDM
C
        LOGICAL DSCONE
C
C       AREA OF MODERATOR VIEWED BY SAMPLE
                        AREAMS=ZM*YM*COS(THETRM)
C
C       AREA OF SAMPLE VIEWING MODERATOR
        IF(LPLATE)      AREASM=ZS*YS*COS(THETRS)
        IF(LROD  )      AREASM=ZS*YS
C
C       THICKNESS OF SAMPLE
        IF(LPLATE)      RNS=TS
        IF(LROD  )      RNS=3.14159*YS*.25
C
C       AREA OF DETECTOR VIEWING SAMPLE
        IF(.NOT.DSCONE) AREADS=ZD*YD*COS(THETRD)
        IF(     DSCONE) AREADS=RC*YS*TC
C
C       SOLID ANGLE OF SAMPLE SEEN BY MODERATOR
                        SMS=AREASM*AREAMS/RMSO**2
C
C       SOLID ANGLE OF DETECTOR SEEN BY SAMPLE
                        SSD=AREADS/RSDO**2
                        SCALER=RNS*1.E12*SMS*SSD/DT
C
C       APPLY SCALING TO DATA ARRAY
                        DO 11 K=N1,N2
11                      RSCORE(K)=RSCORE(K)*SCALER
C
                        RETURN
                        END
```

```
      SUBROUTINE OUTDAT        !COMPUTES MOMENTS OF RSCORE
C
C===============================================================
      COMMON  /SCORE /RSCORE(100)  /TCONS /TESRCE,PATH,DT,FREEDM
     4        /LOGICS/LPRINT,L(9)   /FPATHS/Q,F1(2),WAVE,TO,F2(3)
     7        /SCALES/XMIN,YMIN,XMAX,YMAX,TMEAN,SIGMAT
     8        /INTGRS/TITLE(18),NTRANG,NHIST,I1,I2,NTMAX,NTMIN,NTO,N1,N2,NUM
C===========================================================
      LOGICAL LPRINT,L
C
      TYPE 2
      IF(LPRINT)WRITE(6,2)
2     FORMAT('1',2X,'TIME USECS',2X,'SCORE')
C
      N1=NTMIN-1
      N2=NTMAX+1
      NT1=NTRANG/2-NTO                  !OFFSET STEP IN TIME FOR DISPLAY
C
      YMAX=0.                           !SET CONSTANTS TO ZERO BEFORE
      SO=0.                             !COMPUTING MOMENTS ETC.
      S1=0.
      S2=0.
      DO 1 NT=N1,N2
      T=FLOAT(NT-NT1)*DT                          !TIME IN USEC
      IF(RSCORE(NT).GT.YMAX)YMAX=RSCORE(NT)       !FIND YMAX
      SO=SO+RSCORE(NT)                            !ZEROTH MOMENT (AREA)
      S1=S1+RSCORE(NT)*FLOAT(NT)                  !FIRST MOMENT
      S2=S2+RSCORE(NT)*FLOAT(NT*NT)               !SECOND MOMENT
1     TYPE 4,T,RSCORE(NT)                         !TELETYPE OUTPUT
      IF(.NOT.LPRINT)GOTO7
      DO 6 NT=N1,N2
      T     =FLOAT(NT-NT1)*DT
6     WRITE(6,4)T,RSCORE(NT)                      !LINEPRINTER OUTPUT
7     SIGMAT=SQRT(S2/SO-(S1/SO)**2)*DT            !STANDARD DEVIATION
      TMEAN =(S1/SO-FLOAT(NT1))*DT                !TMEAN IN REAL TIME
      D     =6.28318/Q                            !D-SPACING
            TYPE 5,   SO,TO,TMEAN,SIGMAT,WAVE,Q,D
      IF(LPRINT)WRITE(6,5)SO,TO,TMEAN,SIGMAT,WAVE,Q,D
C
      XMIN  =FLOAT(N1)*DT                         !XMIN,XMAX &YMAX FOR PLOTTING
      XMAX  =FLOAT(N2)*DT
      NUM   =N2-N1+1
      YMAX  =1.1*YMAX
      RETURN
C
4     FORMAT(1X,F8.2,F12.2)
5     FORMAT(///,' SO         =',F12.3,' TO          =',F8.1,' USECS',
     1           ' TMEAN      =',F8.1,' USECS',/,
     1           ' SIGMAT     =',F8.3,' USECS',/,
     1           ' WAVELENGTH=',F8.3,' ANGSTROM',/,
     1           ' Q-VALUE    =',F8.3,' ANGSTROM**-1',
     1           ' D-SPACING =',F8.5,' ANGSTROM',1H1)
      END
```

```
 SUBROUTINE PLTDAT        !DISPLAYS DATA ON SCREEN NO LIGHT PEN
=====================================================================
 COMMON  /SCORE /RSCORE(100)
 3       /SCALES/XMIN,YMIN,XMAX,YMAX,TMEAN,S1
 4       /INTGRS/TITLE(18),NTRANG,NHIST,I1,I2,NTMAX,NTMIN,NTO,N1,N2,NUM
 3       /LOGICS/PRINT,GT44,L(4),RESTRT,PLOT,DISPLY,BATCH
 5       /FPATHS/Q,RMSO,RSDO,WAVE,TO,EFFD,F1,F2
 6       /TCONS /TESRCE,PATH,DT,FREEDM
 REAL*8  P,YP,NP,YES,NO,RSAVE*4(100)
 LOGICAL L,RESTRT,PLOT,GT44,PRINT,SUMM,REST,LSTOP,DISPLY,BATCH
 DATA YP /'PLOT    '/,NP/'NOPLOT  '/,
 1    YES/'Y       '/,NO/'N       '/

 CALL SAVDAT(RSCORE,RSAVE)                  !SAVE ONE'S

 RESTRT=.FALSE.                             !RESTART VARIABLE
 LSTOP =.FALSE.                             !STOP
 PLOT  =.FALSE.                             !PLOT VARIABLE
 NT1   =NTO-NTRANG/2                        !FIRST CHANNEL OF WINDOW
 CALL INIT(2048)                            !RESTART POINT AFTER SUMMING
 CALL SUBP(1)                               !SET UP SUBPICTURE
 CALL AXESS                                 !AXES
 CALL DATCAP(TO,TMEAN,Q,S1)                 !DRAW VARIABLES
 CALL SCAL  (XMIN,YMIN,XMAX,YMAX)           !SCALE SYSTEM
 CALL LINES (TO,TMEAN,NT1,DT,YMAX/1.1)      !DRAW LINES
 CALL DATPLT(N1,N2,RSCORE,DT)               !DATA PLOT
 CALL ESUB                                  !CLOSE SUBPICTURE




ASK FOR OPTIONS ON RESTART & SUMMING ETC.========================
                CALL ASK(LSTOP,RESTRT,SUMM,REST,BATCH)
IF(SUMM)        CALL SUM(N1,N2,RSCORE,RSAVE)     !GOTOSUM OPTION
IF(REST)        CALL SAVDAT(RSAVE,RSCORE)        !RESTORE UNSUMMED DATA
IF(SUMM.OR.REST)        CALL OFF(1)              !CLEAR SUBPICTURES
IF(SUMM.OR.REST)        GOTO1                    !REPLOT NEW DATA
IF((RESTRT.OR.LSTOP).AND..NOT.BATCH)
1                       TYPE3                    !ASK ABOUT PLOTTING
IF(BATCH)               READ(8,4)P               !ASK PLOTTING OPTION FOR BATCH
IF(.NOT.BATCH)          ACCEPT 4,P               !ASK PLOTTING OPTION FROM TI:
IF(P.NE.YP.AND.P.NE.NP.AND.P.NE.YES.AND.P.NE.NO)GOTO5
IF(P.EQ.YP.OR.P.EQ.YES) PLOT=.TRUE.
IF(P.EQ.NP.OR.P.EQ.NO ) PLOT=.FALSE.
                        RETURN                   !GOTO TOFDIF AND PLOT

                        TYPE6,P                  !INVALID COMMAND
                        GOTO7                    !ASK AGAIN
FORMAT('$FOR THIS PICTURE ON VERSATEC TYPE PLOT OR NOPLOT ')
FORMAT(A8)
FORMAT('COMMAND MUST BE PLOT OR NOPLOT, NOT:',A8)
                        END
```

```
      SUBROUTINE ASK(L1,L2,L3,L4,L5)  !ASK FOR OPTION VIA KEYBOARD
      IMPLICIT REAL*8(P-R),LOGICAL(L-L)
      DIMENSION R(4)
      DATA R/'STOP    ','RESTART ','SUM     ','RESTORE '/
      L1=.FALSE.
      L2=.FALSE.
      L3=.FALSE.
      L4=.FALSE.
C
      IF(.NOT.L5)                     TYPE 1
5     IF(     L5)                     READ(8,2)P       !BATCH READ COMMAND
      IF(.NOT.L5)                     ACCEPT 2,P       !TI: READ COMMAND
      IF(P.NE.R(1).AND.P.NE.R(2).AND.
     1  P.NE.R(3).AND.P.NE.R(4))      GOTO3            !CHECK WORD
      IF(P.EQ.R(1))                   L2=.FALSE.       !STOP
      IF(P.EQ.R(1))                   L1=.TRUE.
      IF(P.EQ.R(2))                   L2=.TRUE.        !RESTART
      IF(P.EQ.R(3))                   L3=.TRUE.        !SUMMING
      IF(P.EQ.R(4))                   L4=.TRUE.        !UN-SUMMING
                                      RETURN
3                                     TYPE 4,P         !PRINT INVALID COMMAND
                                      GOTO5            !TRY AGAIN
C
1     FORMAT('$SUM OR RESTORE; THEN RESTART OR STOP ')
2     FORMAT(A8)
4     FORMAT(' INVALID COMMAND ',A8,/,
     1       ' VALID COMMANDS ARE STOP,RESTART,SUM OR RESTORE')
                                      END


      SUBROUTINE SUM(I,J,R,S) !SUMMING SUBROUTINE
      DIMENSION R(100),S(100)
      TYPE 1                  !QUESTION ON HOW MANY CHANNELS FOR SUMMING
      ACCEPT 2,M
      CALL SUMDAT(R,S,M)
      RETURN
1     FORMAT('$NO. OF CHANNELS FOR SUMMING?')
2     FORMAT(I2)
      END


      SUBROUTINE ENDP(PLOTY)
      LOGICAL PLOTY
                      DATA RNAME/6RRASM  /
                      CALL PLOT(0.,0.,999)    !FINISH PLOTTING
      IF(PLOTY)       CALL REQUES(RNAME)      !PLOT GRAPHS
      RETURN
      END
```

```
      SUBROUTINE SAVDAT(R,S)  !SAVES REAL DATA
      DIMENSION R(100),S(100)
      DO 1 K=1,100
1     S(K)=R(K)
      RETURN
      END
C
      SUBROUTINE DATCAP(TO,TMN,Q,S1)  !DATA CAPTIONS
      DATA X/56./
      CALL NOSC
      CALL APNT(X,600.,-1,-4,-1,1)
      CALL TEXT('SIGMA=')
      CALL NMBR(10,S1,8,'(F8.1)')
      CALL APNT(X,500.,-1,-4,-1,1)
      CALL TEXT('TMEAN=')
      CALL NMBR(11,TMN,8,'(F8.1)')
      CALL APNT(X,400.,-1,-4,-1,1)
      CALL TEXT('TO   =')
      CALL NMBR(12,TO,8,'(F8.1)')
      CALL APNT(X,300.,-1,-4,-1,1)
      CALL TEXT('Q     =')
      CALL NMBR(13,Q,8,'(F8.1)')
      RETURN
      END
C
      SUBROUTINE LINES(TO,TM,N,DT,Y)
      DATA Z/0./
      TOR=TO-FLOAT(N)*DT
      TMR=TM-FLOAT(N)*DT
      CALL APNT(TMR,Z,-1,-3,-1,2)      !DRAW VERTICAL LINE
      CALL VECT(Z,Y)                   !FOR TMEAN
      CALL `APNT(TOR,Z,-1,-3,-1,3)     !DRAW VERTICAL LINE
      CALL VECT(Z,Y)                   !FOR TO
      RETURN
      END
C
      SUBROUTINE DATPLT(I,J,R,D)       !DISPLAY DATA POINTS
      DIMENSION R(100)
      DO 1 K=I,J                       !PLOT POINTS
1     CALL APNT(FLOAT(K)*D,R(K),-1,5,-1,1)
      RETURN
      END
C
      SUBROUTINE AXESS                             !DRAWS AXES
      DATA Z/0./,Y/1024./
      CALL NOSC                                    !NOSCALING
      CALL APNT(Z,Z,-1,-3,-1,1)                    !ORIGIN
      CALL VECT(Y,Z)                               !X-AXIS
      CALL APNT(Z,Z,-1,-3,-1,1)                    !ORIGIN
      CALL VECT(Z,Y)                               !Y-AXIS
      CALL APNT(412.,32.,-1,-3,-1,1)
      CALL TEXT('TIME-OF-FLIGHT MICROSECS')        !X CAPTION
      RETURN
      END
>
```

```
      SUBROUTINE VERDAT                  !RSX11-M VERSATEC PLOTTING ROUTINE
C======================================================================
      COMMON  /SCORE /RSCORE(100)
      1       /ANGLES/THETAM,THETAS,THETAD,THETA0,TTT(4)
      2        /INTGRS/TITLE(18),NTRANG,NPOS,I1,I2,NTMAX,NTMIN,NT0,N1,N2,NUM
      3        /FPATHS/Q0,RMS0,RSD0,WAVE,T0,EFFD,F1,F2
      4        /TCONS /TESRCE,PATH,DT,FREEDM
      5        /SCALES/XMIN,XMAX,YMIN,YMAX,TMEAN,SIGMAT
C======================================================================
      REAL X(102),Y(102),TX(5),TY(3)                   !PLOTTING ARRAYS
      DATA Z/0./,H/.1/,LM/-30584/,T/5./,R/999./        !STARTING VALUES
      DATA TX/'TIME','-OF-','FLIG','HT U','SECS'/,
      1    TY/'INTE','NSIT','Y   '/
      YMAX  =0.                                        !START VALUE
      TSTART=FLOAT(NT0-NTRANG/2)*DT                    !OFFSET IN TIME
      DO 10 N=N1,N2
      N3    =N+1-N1
      Y(N3) =RSCORE(N)
      IF(Y(N3).GT.YMAX)YMAX=Y(N3)
10    X(N3) =FLOAT(N)*DT+TSTART          !TIME CHANNEL
      CALL PLOTS(0,0,0)                  !INITIALIZE PLOTTER
      CALL SCALE(Y,10.,N3,1)             !SCALE Y-ARRAY
      CALL SCALE(X, 7.,N3,1)             !SCALE X-ARRAY
      M=N3+1
      N=N3+2
      CALL SYMBOL(T*.5,10.,H,TITLE,Z,72)
      CALL AXIS   (.5,.5,TX,-20, 7.,  Z,X(M),X(N))!X-AXIS
      CALL AXIS   (.5,.5,TY, 12,10.,90.,Y(M),Y(N))!Y-AXIS
      CALL GRID   (.5,.5,IFIX(7.*X(N)/DT),DT/X(N),20,.5,LM)
      CALL PLOT   (.5,.5,-3)                    !REORIGIN
      CALL LINE   (X,Y,N3,1,-1,14)              !PLOT *'S
      CALL PLOT   (Z,Z,-3)                      !REORIGIN
      CALL SYMBOL(T,9.,H,'Q=    ',Z,6)          !ANNOTATE Q0
      CALL NUMBER(R,R,H,Q0,Z,2)
      CALL SYMBOL(T,8.5,H,'T0=   ',Z,6)         !ANNOTATE T0
      CALL NUMBER(R,R,H,T0,Z,4)
      CALL SYMBOL(T,8.,H,'TMEAN=',Z,6)          !ANNOTATE TMEAN
      CALL NUMBER(R,R,H,TMEAN,Z,4)
      CALL SYMBOL(T,7.5,H,'ANGLE=',Z,6)         !ANNOTATE THETA0
      CALL NUMBER(R,R,H,THETA0,Z,4)
      CALL OFFSET(X(M),X(N),Y(M),1.)
      CALL PLOT   (T0,Z,13)                     !DRAW T0 LINE
      CALL PLOT   (T0,10.,12)
      CALL PLOT   (TMEAN,Z,13)
      CALL PLOT   (TMEAN,10.,12)
      CALL PLOT   (Z,Z,-999)                    !END OF FRAME
      TYPE 20
20    FORMAT(' GRAPH PLOTTED')
      RETURN
      END
```

```
        BLOCK DATA
C======================================================================
C=====  COMMON BLOCKS FOR SIMON PASSING DATA THROUGH THE OVERLAY STRUCTURES
C
        COMMON  /ANGLES/THETAM,THETAS,THETAD,THETAO,
        1                THETRM,THETRS,THETRD,THETRO
        2       /SIZES /YM,ZM, XC,YC,ZC, YS,ZS,TS,RS, YD,ZD,TD,HTD,TC,RC
        3       /FPATHS/Q0,RMSO,RSDO,WAVE,TO,EFFD,F1,F2
        4       /TCONS /TESRCE,PATH,DT,FREEDM
        5       /LOGICS/LPRINT,GT44,SROD,SPLATE,DSCONE,IER,RESTRT,PLOT,DIS,BATCH
        6       /INTGRS/TITLE(18),NTRANG,NHIST,I1,I2,NTMAX,NTMIN,NTO,N1,N2,NUM
        7       /SCALES/XMIN,YMIN,XMAX,YMAX,TMEAN,S1
        4       /PRECON/PRECOR(11)       /SCORE /RSCORE(100)
        5       /ELEVTN/ELEVR(11),NYM    /PLAN  /PLANR(8)
C
C======================================================================
C=====  COMMON BLOCKS FOR DISPLAY FILES GRAPHICS LIBRARY AND VERSATEC LIBRARY
C
        COMMON  /DFILE/IDISP(2048)      /GRDAT/IDUMM(142)
        1       /PPEP1/IDUMM1(56)       /IOCOM/IDUMM2(268)      /MULTI/IDUMM3
C
        LOGICAL LPRINT,GT44,SROD,SPLATE,DSCONE,IER,RESTRT,PLOT,DIS,BATCH
C
        DATA    THETAM,THETAS/2*0./,THETAD/60.5/,THETAO/150./,
        1       YM,ZM,YC,ZC/4*.1/,YS,ZS,TS/3*.025/,ZD/.3/,Q0/10./,
        2       XC/5.6/,YD/.175/,EFFD/17.65/,TD/.019634954/,
        2       RMSQ/5.6/,RSDO/.46/,DT/2./,NTRANG/100/,NHIST/500/,
        3       XMIN,YMIN,XMAX,YMAX,TMEAN,S1/6*0./,
        6       DIS,BATCH,GT44,IER,RESTRT,PLOT,LPRINT,DSCONE,SROD,SPLATE
        7       /10*.FALSE./,
        7       PATH/.0053/,TESRCE/2./,N1,N2,NUM,I1,I2/5*0/,
        8       RSCORE/100*0./,FREEDM/6./,
        9       TITLE/18*'    '/
        END
```