



Description of the MOPS-III programme

Ølgaard, Povl Lebeck

Publication date:
1972

Document Version
Publisher's PDF, also known as Version of record

[Link back to DTU Orbit](#)

Citation (APA):
Ølgaard, P. L. (1972). *Description of the MOPS-III programme*. Risø-M No. 1450

General rights

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

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

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

Risø - M - 1450

Title and author(s) Description of the MOPS-III Programme by P.L. Ølgaard	Date July 1972 Department or group Reactor Physics Group's own registration number(s) P-667
pages + 0 tables + 1 illustrations	
Abstract The MOPS-III programme is intended for calculations of subsurface neutron moisture gauges. The source-detector geometry of the probe is a line detector for thermal and epithermal neutrons and a fast neutron point source situated anywhere on the detector axis. MOPS-III allows calculation of calibration curves for such gauges with the probe situated in a medium of known composition. It allows also experimental input count rates to be compared with corresponding calculated count rates. The probe may be situated in an infinite medium or in a finite spherical medium with the source at the centre of the sphere. In this report the theoretical basis of the programme is discussed. Also instructions on the use of the programme, a print-out of the complete ALGOL programme and a test example with description of input and output are given.	Copies to Standard distribution
Available on request from the Library of the Danish Atomic Energy Commission (Atomenergikommisjonenens Bibliotek), Risø, Roskilde, Denmark. Telephone: (03) 35 51 01, ext. 334, telex: 5072.	Abstract to

List of Contents

	Page
1. Introduction	4
2. Theoretical Basis	5
2.1. Calculation of detector count rate	5
2.2. Average detector flux'es	7
2.2.1. Thermal neutron flux, infinite medium	7
2.2.2. Epithermal neutron flux, infinite medium .	11
2.2.3. Thermal neutron flux, finite sphere	12
2.2.4. Epithermal neutron flux, finite sphere ...	14
2.2.5. Linear extrapolation length	15
3. Limitations of the Programme	16
4. Use of the Programme	17
4.1. Input data	17
4.2. Print-out of programme	17
4.3. Output and computing time	19
5. References	22
Appendix I. Integrals of Relevance to the Calculations of $\bar{\Phi}_D$	23
Appendix II. Input Data Sheet for MOPS-III	25
Appendix III. Print-out of the MOPS-III Programme	29
Appendix IV. Output from the MOPS-III Programme	41
Appendix V. Output from the MOPS-III Programme	42

1. Introduction

The MOPS-III programme is a modified version of the MOPS-II-C programme described in ref. 1. The MOPS-II-C programme was designed to calculate the calibration curve of a neutron moisture gauge, situated in an infinite medium. The MOPS-III programme can perform any calculation which can be executed by the use of MOPS-II-C, but in addition a number of extra features are built into MOPS-III, which allows the programme to perform other types of calculations.

Both MOPS-II-C and MOPS-III are based on a fast neutron point source and a line detector with the source situated on the detector axis. In the case of MOPS-II-C the source must be situated outside the detector, while in the case of MOPS-III the source may be situated at any point on the detector axis, inside as well as outside the detector.

Both programmes are based on the three-group-diffusion theory. As regards detector characteristics, MOPS-II-C assumes that the detector may be grey to thermal and white to epithermal (group 2) neutrons or black to thermal and either grey or black to epithermal neutrons. MOPS-III allows the same combinations, but in addition the detector may be grey to both thermal and epithermal neutrons.

In MOPS-II-C the normalization or correction factor C is determined by normalization of the calculated rate to a measured count rate, obtained in a reference medium with known dry soil density and moisture content. The count rate, dry bulk density and moisture content of the reference medium are input data. The same procedure may be used in MOPS-III, but for this programme C may also be read directly into the computer in which case the input C-value is used in the calculations.

MOPS-II-C can only be used in connection with infinite media. However MOPS-III can also treat finite, spherical media, surrounded by vacuum, provided the source is at the centre of the sphere and the detector is completely inside the medium.

In the case of MOPS-II-C the macroscopic absorption cross section of the dry medium is calculated by use of the composition of the medium. This procedure can also be used in the case of MOPS-III, but in addition the macroscopic 2200 m/sec absorption cross section per gramme of the dry medium ($\Sigma_{a,dry}^{2200}/\rho_{dry}$) may be an input value and used in the calculation of the macroscopic absorption

cross section of the dry medium. This option is of interest if the composition of the medium is not well known, in which case $\Sigma_{a,dry}^{2200} / \rho_{dry}$ may be used as a fitting parameter.

MOPS-II-C can be used to calculate calibration curves for neutron moisture gauges. The same is the case for MOPS-III, but this programme can also be used to perform direct comparisons between measured and calculated count rates. When such calculations are performed, sets of experimentally determined values of count rate and corresponding dry soil density, moisture content and medium radius are read into the computer. For each of these sets and by use of the input dry soil density, moisture content and medium radius the programme calculates the theoretical count rate and compares it with the experimental input count rate. This option is obviously of interest when the theoretical model contained in MOPS-III is to be compared with measurements.

Finally in the case of calibration curve calculations, MOPS-III contains some options with respect to the amount of output given by the programme, options which are not included in MOPS-II-C.

2. Theoretical Basis

As stated in the introduction MOPS-III is a modified version of MOPS-II-C. Hence MOPS-III uses to a large extent the same theoretical basis as MOPS-II-C. This basis is described in refs. 2 and 3. We shall therefore limit the following considerations to the new features which have been included in MOPS-III.

2.1. Calculation of detector count rate

As discussed in ref. 3, the total count rate (CR_{total}) is equal to the sum of the thermal neutron count rate (CR_{th}) and the epithermal neutrons count rate (CR_{epi}), i.e.

$$CR_{total} = CR_{th} + CR_{epi}.$$

For detectors with different neutron absorption characteristics, different expressions for the count rate CR should be used.

In agreement with refs. 2 and 3 we have the following expressions:

White detector (the detector is transparent to the neutrons of the energy group considered and therefore does not absorb any of them):

$$CR = 0.$$

Grey detector (the detector will absorb some of the neutrons of the energy group considered which enter the detectors):

$$CR = \bar{\Phi}_D \cdot V_{eff} \cdot \bar{\Sigma}_{a,D} \cdot 60 \quad (\text{cpm})$$

where $\bar{\Phi}_D$ is the neutron flux of the detector in the energy group considered; $\bar{\Phi}_D$ is averaged over the detector volume. V_{eff} is the effective detector volume, and $\bar{\Sigma}_{a,D}$ is the macroscopic neutron absorption cross section of the detector for the process which gives rise to the detector pulses. $\bar{\Sigma}_{a,D}$ is the energy averaged absorption cross section of the energy group considered.

For the thermal neutron group, $\bar{\Sigma}_{a,D}^{th}$ is equal to

$$\bar{\Sigma}_{a,D}^{th} = \Sigma_{a,D}^{2200} \cdot \frac{\sqrt{\pi}}{2} \cdot \sqrt{\frac{293}{T_n}} = \Sigma_{a,D}^{2200} \cdot 0.8862 \sqrt{293/T_n}, \quad (\text{cf. ref. 2})$$

where $\Sigma_{a,D}^{2200}$ is the 2200 m/sec macroscopic absorption cross section of the detector for the process which gives detector pulses at T_n is the neutron temperature in $^{\circ}\text{K}$.

For the epithermal neutron group, $\bar{\Sigma}_{a,D}^{epi}$ is assumed equal to

$$\bar{\Sigma}_{a,D}^{epi} = \Sigma_{a,D}^{2200} \frac{2\sqrt{293/(5 T_n)}}{\ln(E_{2,u}/(5k T_n))}, \quad (\text{see ref. 3})$$

where $E_{2,u}$, the upper energy limit of the group 2 neutrons, is 2 MeV.

Black detector (the detector absorbs all neutrons of the energy group considered which enter the detector):

$$CR = S_D \cdot \frac{\bar{\Phi}_D}{4} \cdot 60, \quad (\text{cpm})$$

where S_D is the surface area of the detector. $\bar{\Phi}_D$ is the unperturbed neutron flux, averaged over the surface of the detector.

It should be noted that this expression is an approximation only. In MOPS-III the following combinations of detector sensitivity towards thermal and epithermal neutrons have been incorporated:

c	Thermal neutrons	Epithermal neutrons
1	grey	white
2	black	grey
3	black	black
4	grey	grey

For these combinations we have the following expressions for the count rate:

$$\begin{aligned}
 c = 1 \quad CR &= \bar{\phi}_{th,D} \cdot V_{eff} \cdot \bar{\Sigma}_{a,D}^{th} \cdot 60 \\
 c = 2 \quad CR &= \left[S_D \cdot \frac{\bar{\phi}_{th,D}}{4} + \bar{\phi}_{epi,D} \cdot V_{eff} \cdot \bar{\Sigma}_{a,D}^{epi} \right] \cdot 60 \\
 c = 3 \quad CR &= \left[S_D \cdot \frac{\bar{\phi}_{th,D}}{4} + S_D \cdot \frac{\bar{\phi}_{epi,D}}{4} \right] \cdot 60 \\
 c = 4 \quad CR &= \left[\bar{\phi}_{th,D} \cdot V_{eff} \cdot \bar{\Sigma}_{a,D}^{th} + \bar{\phi}_{epi,D} \cdot V_{eff} \cdot \bar{\Sigma}_{a,D}^{epi} \right] \cdot 60
 \end{aligned}$$

2.2. Average detector flux'es

The only terms contained in the expressions of the count rate given above which are not readily determined is $\bar{\phi}_{th,D}$ and $\bar{\phi}_{epi,D}$. We shall therefore discuss the derivation of the formulas for the neutron flux'es used in MOPS-III. As mentioned in the introduction of this report both infinite media and finite, spherical media with the neutron source at the centre of the sphere can be handled by the programme, and for both cases flux formulas are derived below.

2.2.1. Thermal neutron flux, infinite medium

If we consider a fast neutron point source situated in an infinite medium, the thermal neutron flux, ϕ_{th}^{∞} , as a function of the distance from the source, r , is given by the following expression according to ref. 2:

$$\begin{aligned}
 \phi_{th}^{\infty}(r) &= \frac{q}{4\pi D_3 r} \frac{L_3^2}{L_1^2 - L_2^2} \left[\frac{L_1^2}{L_1^2 - L_3^2} (e^{-r/L_1} - e^{-r/L_3}) - \frac{L_2^2}{L_2^2 - L_3^2} (e^{-r/L_2} - e^{-r/L_3}) \right] \\
 &= \frac{q L_3^2}{4\pi D_3 r} \left[K_1 e^{-r/L_1} + K_2 e^{-r/L_2} + K_3 e^{-r/L_3} \right],
 \end{aligned}$$

where

$$K_1 = \frac{L_1^2}{(L_1^2 - L_2^2)(L_1^2 - L_3^2)}; \quad K_2 = \frac{L_2^2}{(L_2^2 - L_1^2)(L_2^2 - L_3^2)}; \quad K_3 = \frac{L_3^2}{(L_3^2 - L_1^2)(L_3^2 - L_2^2)};$$

$$\text{and} \quad K_3 = -(K_1 + K_2).$$

q is the source strength, and D_3 is the thermal neutron diffusion coefficient. L_1 and L_2 are the slowing-down length of the upper and lower fast neutron groups, and L_3 the diffusion length of the thermal group, all evaluated for the medium considered.

The expression for $\phi_{th}^{\infty}(r)$ is valid for the unperturbed flux around the source, not taking into account the perturbation introduced by the detector. We are however interested in the true detector flux. In order to obtain this flux we introduce a correction factor C , which among other things corrects for the flux depression through the grey detector or for the approximative character of the expression used for the count rate of the black detector. C should in general be somewhat less than 1.

For the source-detector geometry of fig. 1A, an expression for the average thermal neutron flux of the detector $\bar{\phi}_{th,D}^{\infty}$ is obtained by integration (cf. ref. 2 and Appendix I):

$$\begin{aligned}
 \bar{\phi}_{th,D}^{\infty}(a) &= \frac{C}{a} \int_0^a \phi_{th}^{\infty}(r) dr = C \frac{q}{4\pi D_3 a} \frac{L_3^2}{L_1^2 - L_2^2} \times \\
 &\quad \left[\frac{L_1^2}{L_1^2 - L_3^2} (E_1(a/L_3) - E_1(a/L_1)) + \ln(L_1/L_3) \right] -
 \end{aligned}$$

$$\begin{aligned}
 & - \frac{L_2^2}{L_2^2 - L_3^2} (E_1(a/L_3) - E_1(a/L_2) + \ln(L_2/L_3)) \Big] \\
 & = C \frac{q L_3^2}{4\pi D_3 a} \left[K_1 (\ln(L_1) - E_1(a/L_1)) + K_2 (\ln(L_2) - E_1(a/L_2)) \right. \\
 & \left. + K_3 (\ln(L_3) - E_1(a/L_3)) \right].
 \end{aligned}$$

In the last expression for $\bar{\phi}_{th,D}^\infty(a)$, the term $\ln(L)$ must be interpreted as the natural logarithm of L divided by one unit length, so that the argument of the logarithmic function is dimensionless. $E_1(x)$ is the exponential integral

$$E_1(x) = \int_x^\infty \frac{e^{-y}}{y} dy.$$

If we consider the source-detector geometry of fig. 1B, the expression for $\bar{\phi}_{th,D}^\infty$, obtained by integration, is (cf. ref. 3):

$$\begin{aligned}
 \bar{\phi}_{th,D}^\infty(b,c) &= \frac{C}{c-b} \int_b^c \bar{\phi}_{th}^\infty(r) dr = C \frac{q L_3^2}{4\pi D_3 (c-b)} \times \\
 & \left[K_1 (E_1(b/L_1) - E_1(c/L_1)) + K_2 (E_1(b/L_2) - E_1(c/L_2)) \right. \\
 & \left. + K_3 (E_1(b/L_3) - E_1(c/L_3)) \right].
 \end{aligned}$$

By use of the two functions $\bar{\phi}_{th,D}^\infty(a)$ and $\bar{\phi}_{th,D}^\infty(b,c)$ the average thermal flux of the detector $\bar{\phi}_{th,D}^\infty$ for the four geometries considered in fig. 2 can easily be calculated

$$g = 1 \quad \bar{\phi}_{th,D}^\infty = \bar{\phi}_{th,D}^\infty(d-l, d+l)$$

$$g = 2 \quad \bar{\phi}_{th,D}^\infty = \bar{\phi}_{th,D}^\infty(2l)$$

Fig. 1. Source-detector geometry

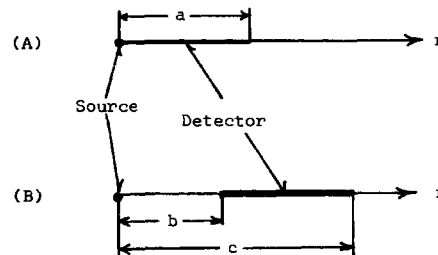
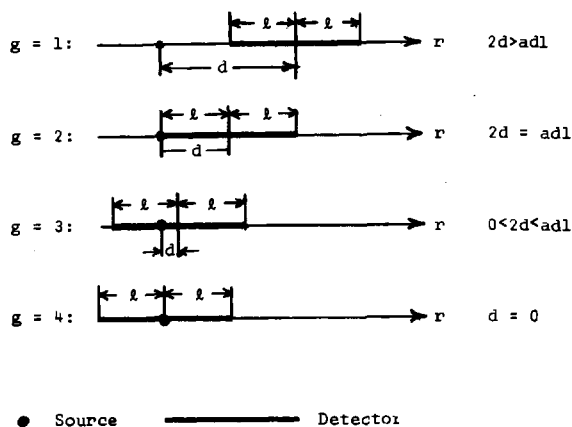


Fig. 2. Source-detector geometries considered by MOPS-III (adl = 2 * l)



$$g = 3 \quad \bar{\phi}_{th,D}^{\infty} = \frac{1}{2L} \left[\bar{\phi}_{th,D}^{\infty}(\ell-d) \cdot (\ell-d) + \bar{\phi}_{th,D}^{\infty}(\ell+d) \cdot (\ell+d) \right]$$

$$g = 4 \quad \bar{\phi}_{th,D}^{\infty} = \bar{\phi}_{th,D}^{\infty}(\ell) .$$

2.2.2. Epithermal neutron flux, infinite medium

We consider again a fast neutron point source, situated in an infinite medium. According to ref. 3, the epithermal flux (lower fast neutron group, $\phi_{epi}^{\infty}(r)$, is given by:

$$\phi_{epi}^{\infty}(r) = \frac{q}{4\pi D_2 r} \frac{L_2^2}{L_1^2 - L_2^2} (e^{-r/L_1} - e^{-r/L_2}) .$$

D_2 is the diffusion coefficient of the epithermal neutron flux.

Here again we have to introduce a correction factor K which corrects for the epithermal neutron flux depression in a grey detector or for the approximative nature of the count rate expression used for a black detector.

For the source-detector geometry of fig. 1A, the expression used for the average epithermal neutron flux of the detector $\bar{\phi}_{epi,D}^{\infty}$ is derived by integration:

$$\bar{\phi}_{epi,D}^{\infty}(a) = \frac{K}{a} \int_0^a \phi_{epi}^{\infty}(r) dr = K \frac{q}{4\pi D_2 a} \cdot \frac{L_2^2}{L_1^2 - L_2^2} \times$$

$$\left[E_1(a/L_2) - E_1(a/L_1) + 2n(L_1/L_2) \right]$$

$$= K \frac{q L_2^2}{4\pi D_2 a} \left[\frac{1}{L_1^2 - L_2^2} (2n(L_1) - E_1(a/L_1)) + \frac{1}{L_2^2 - L_1^2} (2n(L_2) - E_1(a/L_2)) \right]$$

For the source-detector geometry of fig. 1B we obtain

$$\bar{\phi}_{epi,D}^{\infty}(b,c) = \frac{K}{c-b} \int_b^c \phi_{epi}^{\infty}(r) dr = \frac{K \cdot q}{4\pi D_2 (c-b)} \frac{L_2^2}{L_1^2 - L_2^2} \times$$

$$\left[E_1(b/L_1) - E_1(c/L_1) - E_1(b/L_2) + E_1(c/L_2) \right]$$

in agreement with ref. 4.

By use of the expressions for $\bar{\phi}_{epi,D}^{\infty}(a)$ and $\bar{\phi}_{epi,D}^{\infty}(b,c)$ we can immediately determine the average epithermal flux of the detector geometrics given in fig. 2 (cf. section 2.2.1.):

$$g = 1 \quad \bar{\phi}_{epi,D}^{\infty} = \bar{\phi}_{epi,D}^{\infty}(d-\ell, d+\ell)$$

$$g = 2 \quad \bar{\phi}_{epi,D}^{\infty} = \bar{\phi}_{epi,D}^{\infty}(2\ell)$$

$$g = 3 \quad \bar{\phi}_{epi,D}^{\infty} = \frac{1}{2L} \left[\bar{\phi}_{epi,D}^{\infty}(\ell-d) \cdot (\ell-d) + \bar{\phi}_{epi,D}^{\infty}(\ell+d) \cdot (\ell+d) \right]$$

$$g = 4 \quad \bar{\phi}_{epi,D}^{\infty} = \bar{\phi}_{epi,D}^{\infty}(\ell) .$$

2.2.3. Thermal neutron flux, finite sphere

If the fast neutron point source is situated in the centre of an spherical medium with the radius R , the thermal neutron flux, ϕ_{th}^R , in the medium is, according to ref. 4, given by

$$\phi_{th}^R(r) = \frac{q L_3^2}{4\pi D_3 r} \left[K_1 \frac{\sinh((R-r)/L_1)}{\sinh(R/L_1)} + K_2 \frac{\sinh((R-r)/L_2)}{\sinh(R/L_2)} \right.$$

$$\left. + K_3 \frac{\sinh((R-r)/L_3)}{\sinh(R/L_3)} \right]$$

$$= \frac{q L_3^2}{4\pi D_3 r} \left[\frac{K_1}{1 - e^{-2R/L_1}} (e^{-r/L_1} - e^{-2R/L_1} e^{r/L_1}) + \right.$$

$$\begin{aligned}
 & + \frac{K_2}{1-e^{-2R/L_2}} (e^{-r/L_2} - e^{-2R/L_2}) e^{r/L_2} \\
 & + \frac{K_3}{1-e^{-2R/L_3}} (e^{-r/L_3} - e^{-2R/L_3}) e^{r/L_3} \Big] \\
 & = \frac{q L_3^2}{4\pi D_3 r} \left[K_1 e^{-r/L_1} + K_2 e^{-r/L_2} + K_3 e^{-r/L_3} \right. \\
 & \left. - K_1 C_1 (e^{r/L_1} - e^{-r/L_1}) - K_2 C_2 (e^{r/L_2} - e^{-r/L_2}) - K_3 C_3 (e^{r/L_3} - e^{-r/L_3}) \right],
 \end{aligned}$$

where

$$C_1 = \frac{e^{-2R/L_1}}{1-e^{-2R/L_1}}; \quad C_2 = \frac{e^{-2R/L_2}}{1-e^{-2R/L_2}}; \quad C_3 = \frac{e^{-2R/L_3}}{1-e^{-2R/L_3}}.$$

For the source-detector geometry of fig. 1A we obtain the following expression for $\bar{\phi}_{th,D}^R$ by integration (cf. ref. 4 and Appendix I) and by introduction of the correction factor C:

$$\begin{aligned}
 \bar{\phi}_{th,D}^R(a) & = \frac{C}{a} \int_0^a \phi_{th}^R(r) dr = C \frac{q L_3^2}{4\pi D_3 a^2} \times \\
 & \left[K_1 (E_1(a/L_3) - E_1(a/L_1) + \ln(L_1/L_3)) \right. \\
 & + K_2 (E_1(a/L_3) - E_1(a/L_2) + \ln(L_2/L_3)) \\
 & - K_1 C_1 (E_1(a/L_1) - E_1(-a/L_1)) - K_2 C_2 (E_1(a/L_2) - E_1(-a/L_2)) \\
 & \left. - K_3 C_3 (E_1(a/L_3) - E_1(-a/L_3)) \right].
 \end{aligned}$$

For the source-detector geometry of fig. 1B we obtain the following expression for $\bar{\phi}_{th,D}^{\infty}$:

$$\begin{aligned}
 \bar{\phi}_{th,D}^R(b,c) & = \frac{C}{c-b} \int_b^c \phi_{th}^R(r) dr = C \frac{q L_3^2}{4\pi D_3 (c-b)} \times \\
 & \left[K_1 (E_1(b/L_1) - E_1(c/L_1) - e^{-2R/L_1} (E_1(-b/L_1) - E_1(-c/L_1))) / (1 - e^{-2R/L_1}) \right. \\
 & + K_2 (E_1(b/L_2) - E_1(c/L_2) - e^{-2R/L_2} (E_1(-b/L_2) - E_1(-c/L_2))) / (1 - e^{-2R/L_2}) \\
 & \left. + K_3 (E_1(b/L_3) - E_1(c/L_3) - e^{-2R/L_3} (E_1(-b/L_3) - E_1(-c/L_3))) / (1 - e^{-2R/L_3}) \right]
 \end{aligned}$$

By use of $\bar{\phi}_{th,D}^R(a)$ and $\bar{\phi}_{th,D}^R(b,c)$ the average thermal flux of the detector geometries of fig. 2 can be determined by use of the expressions given at the end of section 2.2.1. if only " ∞ " is replaced by "R".

2.2.4. Epithermal neutron flux, finite sphere

The epithermal flux in a finite sphere, originating from a fast neutron point source situated at the centre of the sphere, is, according to ref. 4, given by the expression:

$$\begin{aligned}
 \phi_{epi}^R(r) & = \frac{q}{4\pi D_2 r} \cdot \frac{L_2^2}{L_1^2 - L_2^2} \left[\frac{\sinh((R-r)/L_1)}{\sinh(R/L_1)} - \frac{\sinh((R-r)/L_2)}{\sinh(R/L_2)} \right] \\
 & = \frac{q}{4\pi D_2 r} \cdot \frac{L_2^2}{L_1^2 - L_2^2} \left[(e^{-r/L_1} - e^{-2R/L_1}) e^{r/L_1} / (1 - e^{-2R/L_1}) \right. \\
 & \quad \left. - (e^{-r/L_2} - e^{-2R/L_2}) e^{r/L_2} / (1 - e^{-2R/L_2}) \right] \\
 & = \frac{q}{4\pi D_2 r} \cdot \frac{L_2^2}{L_1^2 - L_2^2} \left[(e^{-r/L_1} - e^{-r/L_2}) - C_1 (e^{r/L_1} - e^{-r/L_1}) \right. \\
 & \quad \left. + C_2 (e^{r/L_2} - e^{-r/L_2}) \right].
 \end{aligned}$$

For the geometry of fig. 1A the following expression of $\bar{\phi}_{epi,D}^R$ is obtained by integration (cf. Appendix I) and introduction of the correction factor K:

$$\bar{\phi}_{epi,D}^R(a) = \frac{K}{a} \int_0^a \phi_{epi}^R(r) dr = K \cdot \frac{q}{4\pi D_2 a} \cdot \frac{L_2^2}{L_1^2 - L_2^2} \times$$

$$\left[E_1(a/L_2) - E_1(a/L_1) + \ln(L_1/L_2) \right.$$

$$\left. - C_1(E_1(a/L_1) - E_1(-a/L_1)) + C_2(E_1(a/L_2) - E_1(-a/L_2)) \right]$$

and for the geometry of fig. 1B the expression of $\bar{\phi}_{epi,D}^R$ is found to be

$$\bar{\phi}_{epi,D}^R(b,c) = \frac{K}{c-b} \int_b^c \phi_{epi}^R(r) dr = K \cdot \frac{q}{4\pi D_2 (c-b)} \cdot \frac{L_2^2}{L_1^2 - L_2^2} \times$$

$$\left[(E_1(b/L_1) - E_1(c/L_1)) e^{-2R/L_1} (E_1(-b/L_1) - E_1(-c/L_1)) / (1 - e^{-2R/L_1}) \right.$$

$$\left. - (E_1(b/L_2) - E_1(c/L_2)) e^{-2R/L_2} (E_1(-b/L_2) - E_1(-c/L_2)) / (1 - e^{-2R/L_2}) \right].$$

By use of $\bar{\phi}_{epi,D}^R(a)$ and $\bar{\phi}_{epi,D}^R(b,c)$ the average epithermal flux of the detector geometries of fig. 2 can be determined by use of the expressions given at the end of section 2.2.2., if only "a" is replaced by "R".

2.2.5. Linear extrapolation length

The radius R used in section 2.2.3. and 2.2.4. is the extrapolated radius of the spherical medium. Since the physical radius of the sphere, R_m , is related to R by

$$R = R_m + \text{lel},$$

where lel is the linear extrapolation length of the medium, lel

has to be calculated before R can be determined from R_m .

The method used in calculating lel is that described in ref. 4. Since the neutron leakage of the three energy groups has to be known before lel can be calculated and since lel has to be known before the neutron leakage can be computed, lel is found by iteration.

3. Limitations of the Programme

A number of limitations, due in part to the inherent limitations of diffusion theory, in part to numerical problems and in part to the computational procedure chosen, must be taken into account when the programme is used.

In the case of three-group calculations on typical soil media, L_1 and L_2 are usually of the same order of magnitude. At low moisture contents L_1 is usually smaller than L_2 while at high contents $L_1 > L_2$. This means that there will be a moisture range in which $L_1 \approx L_2$. Since the formulas used are based on $L_1 \neq L_2$, one may in this moisture range get erroneous results for the count rate on account of rounding-off errors.

If in the case of $g = 1$ $d \approx 0.001$ cm (see fig. 2), g should be put equal to 2 and d equal to 2. The same is true in the case of $g = 3$ and $d \approx 0.001$ cm. If this is not done one may get erroneous results owing to numerical difficulties.

Too short a detector may also give rise to significant errors owing to rounding-off. For this reason the active detector length (adl of fig. 2) should not be less than 1 cm.

The detector must in case of a finite medium be well inside the sphere, i.e. $d + \text{lel} < R_m$, since the flux distributions used to calculate the count rate are only valid inside the sphere and may give negative count rates outside the medium. The physical radius of the medium, R_m , must also be appreciably larger than lel. This condition is a basic requirement of the diffusion theory.

It is important to bear in mind these limitations when evaluating the results obtained with the MOPS-III programme.

4. Use of the Programme

4.1. Input data

An input data sheet is given in appendix II, where the significance of the necessary input data is also explained. However the following, additional remarks may be of interest.

vr, roJr, cpmr and Rmr are moisture content, dry bulk density, count rate and radius of the reference medium. If C = -1 in the input, the programme will calculate C by normalization of the calculated count rate to that of the reference medium. The composition of the reference medium must either be that of the medium considered (input data for the composition are given on page 2 of the input data sheet) or pure water, in which case vr = 100 and roJr = 0.

As earlier mentioned the programme can only be used for finite spherical media. However cylindrical media are often used when calibration measurements are performed. The problem of "converting" a cylindrical medium into a spherical one is briefly discussed in ref. 5, pag. 104.

The programme has been tested with radius values up to 500 cm without any numerical difficulties occurring. Usually it is a good approximation to assume the medium to be infinite for media with a radius of 500 cm or larger.

The programme may be used either for calculation of calibration curves or for comparison between measured and calculated count rates. In both cases all data required in the two first pages of the input data sheet must be given. In addition the input data required on page 3 should be given in the case of a calibration curve calibration, ignoring page 4, while the input data required on page 4 should be given in the case of comparison between experimental and calculated count rates, ignoring page 3.

4.2. Print-out of programme

In appendix III a print-out of the complete MOPS-III programme is given. It has been written exclusively in GIER ALGOL III.

Three standard procedures of the GIER library have been used, "AEK-ADM 1B", "HYP" and "expoint". The ADM 1B procedure is designed to facilitate the programming and proper arrangement of the output.

The procedure starts in the second line of the programme and continues to: NOW COMES THE PROGRAMME; it also includes the last four lines of the programme. The HYP procedure calculates with pre-determined accuracy (eps) a zero (x) of an arbitrary function (F), given as real procedure, in a prescribed interval (between x1 and x2). If the values of the function at x1 and x2 are different from zero and have the same sign, a jump to a prescribed label (error) will occur. The real procedure expoint calculates the exponential integral of first order, $E_1(x)$.

In addition the six real procedures ØAV(L), Ø2AV(L), sinh(x), lel(x), F2AV(L), and FIAV(L) are included in the programme. ØAV(L) determines the average thermal-neutron flux of a line detector with the length L and with the fast neutron source at the end of the detector (fig. 1A), situated in an infinite medium. FIAV(L) determines the same average flux for the same source-detector geometry except that the medium is a finite sphere with the neutron source at its centre. Ø2AV(L) determines the average epithermal flux of a line detector with the length L, with the neutron source at the end of the detector (fig. 1A), the source and the detector being situated in an infinite medium. F2AV(L) determines the same average flux for the same source-detector geometry as Ø2AV(L), the only difference being that the medium is not infinite, but a finite sphere with the source at its centre. sinh(x) determines the hyperbolic sine function and lel(x) the linear extrapolation length.

A calculation of a theoretical calibration curve by use of MOPS-III is performed in the following way:

The input data are read into the computer, the density of water at tm°C is calculated, and the major part of the input data are printed out. Next the macroscopic cross sections of the 7 energy sub-groups (including slowing-down power as well as transport and absorption cross sections) are determined. For water the cross sections per cm³ (Σ) are calculated, while for the dry medium the cross sections per gramme (Σ/ρ) are computed. The reason for this is that this procedure is convenient when the programme is used to perform comparisons between measurements and calculations, in which case the dry medium density is not constant.

The next major step in the calculation is to determine the count rate of the reference medium. This is done by successive

computation for this medium of the necessary cross sections, neutron temperature, diffusion and slowing-down lengths as well as diffusion coefficients of the three energy groups and finally the average neutron flux'es of the detector. From these the value of C is obtained.

Now the calibration curve, i.e. corresponding values of water volume per cent and count rate, can be calculated for a fixed dry medium density. The procedure is to recalculate for each moisture content the cross sections, the neutron temperature, the diffusion properties of the medium, the average detector flux'es and the count rate. The calculation is repeated for the required moisture contents, and the results are printed out.

The programme can, as earlier mentioned, also be used for comparing experimental and calculated count rate values. When the programme is operating in this mode the calculation is performed in the following way:

The first parts of the computation, i.e. up to and including the determination of C, are practically identical to those of a calibration curve calculation.

When C has been fixed, sets of corresponding values of moisture content, dry medium density, count rate and medium radius, obtained experimentally, are read into the computer. For each set a theoretical count rate is calculated by use of the procedure outlined above. The theoretical count rate, the experimental count rate and the relative deviation between these two count rates are printed out together with moisture per cent, dry medium density and medium radius.

For all experimental data sets considered, the numerical values of the differences between the theoretical and the experimental count rates are summed up for all sets and printed out. The same is true of the squares of the differences between the theoretical and the experimental count rates.

4.3. Output and computing time

In appendix IV the result of a calibration curve calculation is shown. The output given corresponds to the input data set of pages 1, 2 and 3 of appendix II. After programme number, problem

number, data and programme name, the heading of the input data sheet follows. Next, practically all the input data are printed out. The designations used in the output are very close to those used in the input data sheet of appendix II.

Following the print-out of the input data, corresponding values of the moisture content (v) and the count rate (CR) are printed out. Since, for the calculation considered, h = 1 (see appendix II), the average thermal flux of the detector (ϕ_{av}) and the corresponding epithermal flux (ϕ_2) are also printed out together with the diffusion or slowing down lengths of the three groups (L, L1, L2) and the diffusion coefficients of the thermal and epithermal groups (D and D2).

In appendix V the result of a comparison between count rates given in the input data and the corresponding, calculated count rates is given. The input data used are those of appendix II pages 1, 2 and 4, except that C has been put equal to 0.4575 (and consequently $vr = roJr = cpmr = Rur = 1$), that SaJ2dr has been put equal to 0.00499 and that a = -1 and m = 1. As, in appendix IV, the headings are followed by the major part of the data in pages 1 and 2 of the input data sheet. Next, sets of the input data for moisture percent (v), dry soil density (roJ), input count rate (CR(exp)) and medium radius (Rm) follow together with calculated count rate (CR(cal)) and the per cent deviation between calculated and input count rate (delta). Delta is given by the expression

$$\Delta = 100 \cdot \frac{CR(cal) - CR(exp)}{CR(exp)}$$

Finally the sum of the numerical differences between corresponding calculated and input count rates and also the sum of the squares of these differences are given.

The count rates, calculated in appendix IV, and the corresponding dry densities and moisture contents are used as input data in appendix V. However while Rm is infinite in appendix IV, it is 500 cm in appendix V. In spite of this difference the deviation between the calculated and the input count rates is in all cases less than 0.05 per cent. This means that a medium with a radius of 500 cm can be assumed to be infinitely large.

Note that while a constant value of the dry soil density and of the medium radius has been used in the input data sets of appendix V, this need not be the case.

The computing time for a calibration curve calculation as that of appendix IV, assuming an infinite medium, is close to 30 sec. on the GIER-computer. Had a finite medium been considered, the computing time would have been increased to about 40 sec.

The computing time for the comparison given in appendix V is about 40 sec. on the GIER-computer. Had an infinite medium been considered, the computing time would have been roughly 30 sec.

5. References

1. P.L. Ølgaard, Description of the MOPS-II-C Programme. Risø-M-481 (1967) 18 pp.
2. P.L. Ølgaard, On the Theory of the Neutronic Method for Measuring Water Content in Soil. Risø Report No. 97 (1965) 44 pp.
3. P.L. Ølgaard and V. Haahr, Comparative Experimental and Theoretical Investigations of the DM Neutron Moisture Probe. Nucl Eng Des 5 (1967) 311-324.
4. P.L. Ølgaard, Description of the SOFIE-III Programme No. P-405. Risø-M-637 (1967) 52 pp.
5. P.L. Ølgaard and V. Haahr, On the Volume Measured by Sub-surface Neutron Moisture Gauges. In: Aspects of Research at Risø. Risø Report No. 256 (1972) 99-111.

Appendix I

Integrals of Relevance to the Calculation of \bar{Q}_D

$$L_1 = \int_{\delta+0}^L \left[\frac{e^{-r/L_a}}{r} - \frac{e^{-r/L_b}}{r} \right] dr = \int_{\delta/L_a}^{L/L_a} \frac{e^{-x}}{x} dx - \int_{\delta/L_b}^{L/L_b} \frac{e^{-x}}{x} dx$$

$$= \int_{\delta/L_a}^{\infty} \frac{e^{-x}}{x} dx + \int_{\infty}^{L/L_a} \frac{e^{-x}}{x} dx - \int_{\delta/L_b}^{\infty} \frac{e^{-x}}{x} dx - \int_{\infty}^{L/L_b} \frac{e^{-x}}{x} dx$$

$$= E_1(L/L_b) - E_1(L/L_a) + \int_{\delta/L_a}^{\delta/L_b} \frac{e^{-x}}{x} dx$$

$$\frac{e^{-x}}{x} = \frac{1}{x} \left[1 - \frac{x}{1!} + \frac{x^2}{2!} - \frac{x^3}{3!} + \dots \right]$$

$$= \frac{1}{x} - 1 + \frac{x}{2} - \frac{x^2}{6} + \dots$$

$$\int_{\delta/L_a}^{\delta/L_b} \frac{e^{-x}}{x} dx = \left(\ln x - x + \frac{x^2}{2} - \frac{x^3}{6} + \dots \right) \Big|_{\delta/L_a}^{\delta/L_b}$$

For $\delta \rightarrow 0$ the higher order terms will vanish. Hence

$$\int_{\delta/L_a}^{\delta/L_b} \frac{e^{-x}}{x} dx \xrightarrow{\delta \rightarrow 0} \ln(\delta/L_b) - \ln(\delta/L_a) = \ln(L_a/L_b)$$

Thus

$$L_1 = E_1(L/L_b) - E_1(L/L_a) + \ln(L_a/L_b)$$

$$I_2 = \int_{\delta+0}^L \left(\frac{e^{r/L}}{r} - \frac{e^{-r/L}}{r} \right) dr = \int_{-\delta/L}^{-L/L} \frac{e^{-x}}{x} dx - \int_{\delta/L}^{L/L} \frac{e^{-x}}{x} dx$$

$$= \int_{-\delta/L}^{\infty} \frac{e^{-x}}{x} dx - \int_{-\infty}^{\infty} \frac{e^{-x}}{x} dx - \int_{\delta/L}^{\infty} \frac{e^{-x}}{x} dx + \int_{L/L}^{\infty} \frac{e^{-x}}{x} dx$$

$$= E_1(L/L) - E_1(-L/L) + \int_{-L/L}^{\delta/L} \frac{e^{-x}}{x} dx$$

$$\int_{-\delta/L}^{\delta/L} \frac{e^{-x}}{x} dx = \ln|x| - x + \frac{x^2}{4} - \frac{x^3}{18} + \dots \Big|_{-\delta/L}^{\delta/L}$$

For $\delta \rightarrow 0$ the higher order terms will vanish. Hence

$$\int_{-\delta/L}^{\delta/L} \frac{e^{-x}}{x} dx \xrightarrow{\delta \rightarrow 0} \ln(\delta/L) - \ln|-\delta/L| = \ln 1 = 0$$

Thus

$$I_2 = E_1(L/L) - E_1(-L/L)$$

$$I_3 = \int_b^c \frac{e^{-r/L}}{r} dr = \int_{b/L}^{c/L} \frac{e^{-x}}{x} dx = \int_{b/L}^{\infty} \frac{e^{-x}}{x} dx + \int_{\infty}^{c/L} \frac{e^{-x}}{x} dx$$

$$= \int_{b/L}^{\infty} \frac{e^{-x}}{x} dx - \int_{c/L}^{\infty} \frac{e^{-x}}{x} dx = E_1(b/L) - E_1(c/L)$$

$$I_4 = \int_b^c \frac{e^{r/L}}{r} dr = \int_{-b/L}^{-c/L} \frac{e^{-x}}{x} dx = \int_{-b/L}^{\infty} \frac{e^{-x}}{x} dx - \int_{-c/L}^{\infty} \frac{e^{-x}}{x} dx$$

$$= E_1(-b/L) - E_1(-c/L)$$

Appendix II

DANISH ATOMIC ENERGY COMMISSION
Research Establishment Risø
Reactor Physics Department

July 1972
P-667

Input Data Sheet for MOPS-III

Pf .1.1..... Problem number

.11..... 8....., .1972....., Day, month, year

[TEST OF MOPS-III] Heading

.20....., tm, physical temperature (°C)

.14₂₀5....., q, neutron source strength (n/sec)

.2....., c, control parameter for detector

If c=1, detector is grey to thermal, white to epithermal neutrons
If c=2, " " black " " , grey " " "
If c=3, " " black " " , black " " "
If c=4, " " grey " " , grey " " "

.6.13592.., Veff, active detector volume (cm³)

In case c=3, the value of Veff is not used, but a value for Veff must be given.

.21.8873.., SaC, macroscopic 2200 m/sec absorption cross section of nuclear process in detector which causes pulses (cm⁻¹).

In case c=3, the value of SaC is not used, but a value for SaC must be given.

.19.536... , SD, surface area of active detector volume (cm²).

In case c=1 or c=4, the value of SD is not used, but a value for SD must be given.

.1.25....., adl, active detector length (cm);

.6.7....., d, distance from centre of source to centre of detector (midpoint of active detector length) (cm)

.1....., g, control parameter for source-detector geometry

If 2d > adl, g=1; if 2d = adl, g=2;
If 0 < 2d < adl, g=3; if d=0, g=4.

The values of adl and d must agree with the value of g.

.0.090....., K, correction factor for depression in detector of epithermal flux. If c=1, the value of K is not used and should be put equal to 0.

-.1....., C, correction factor for depression in detector of thermal flux.

If C > 0, this value will be used in the calculations. If C=-1, the programme will calculate C by normalization of the calculated count rate to that of the reference medium.

.100....., vr, water volume per cent of reference medium.

If C > 0, the value of vr is not used; thus vr may be put equal to 1.

0....., roJr, dry bulk density of reference medium (g/cm³).

If C = 0, the value of roJr is not used; thus roJr may be put equal to 1.

.74105...., cpmr, count rate of detector in reference medium (cpm).

If C > 0, the value of cpmr is not used; thus cpmr may be put equal to 1.

.4....., Rmr, physical radius of reference medium (cm).

If Rmr=-1, the medium is considered infinite.

If C > 0, the value of Rmr is not used; thus Rmr may be put equal to 1.

The following input data must be written line by line:

.0.326...., wt% H in dry medium	0.001....., wt% Li in dry medium
.0.003...., - B - - -	0.506...., - C - - -
.0.052...., - N - - -	53.4818... - O - - -
.0.740...., - Na - - -	1.380...., - Mg - - -
.3.750...., - Al - - -	34.280...., - Si - - -
.0.058...., - P - - -	0.0012...., - S - - -
.0.007...., - Cl - - -	1.730...., - K - - -
.1.550...., - Ca - - -	0.320...., - Ti - - -
.0.034...., - Mn - - -	1.760...., - Fe - - -
.0.0006...., - Co - - -	0.00002...., - Cd - - -

-.1....., SaJ2dr, macroscopic 2200 m/sec absorption cross section per gram of dry medium ($\Sigma_{a, dry}^{2200}$). If SaJ2dr ≤ 0, then $\Sigma_{a, dry}^{2200}$ is calculated from the medium composition. If SaJ2dr > 0, then $\Sigma_{a, dry}^{2200} = roJ \times SaJ2dr$, irrespective of the medium composition.

.5....., a. If a > 0, then a is the step length in volume percent of water, when calculating the calibration curve (see page 3). If a < 0, then the programme calculates count rates for specific values of water volume per cent, and these count rates are compared with experimental count rates read into the programme (see page 4).

.8....., m. If a > 0, then m is the number of steps involved in the calibration curve calculation. If a < 0, then the value of m is not used and may be put equal to 1.

Appendix III

Print-out of the MOFS-III Programme

```

comment MOFS-III, P-667, 12/8 1972;
begin comment: A.E.K. - ADM 1B - April 9th 1964;
integer pagecounter, linecounter, problem no, day, month, year, drum, a;

procedure head; CR(100);

procedure stop;
begin integer a; outtext(←
stop);
a := typechar;
if a > 128 then a := a - 128;
if a = 50 then go to start;
if a = 53 then go to end of program;
end of stop;

procedure CR(a);
value a; integer a;
begin
if linecounter - 6 < a then a := linecounter + 2;
linecounter := linecounter - a;
for a := a - 1 step -1 until 0 do out;

if linecounter < 0 then begin
pagecounter := pagecounter + 1;
linecounter := linecounter + 64;
if pagecounter > 1 then
begin outsp(32); output(←ddd, -pagecounter, outtext(←-)) end;
outtext(←

A.E.K. - Program No. 667 - Problem No);
output(ddd, problem no);
outtext(← - );
output(ddd, day, outtext(←-), month); outtext(←-); output(ddd, year);

comment NOW ONE LINE TO BE PRINTED IN EACH HEADING CAN BE WRITTEN;

outtext(←
MOFS-III

);
end of linecounter < 0
end of CR;
drum := drumplace;

linecounter := 0;
for a := 1 step 1 until 30 do outchar(112); outclear;

start;
drumplace := drum;
pagecounter := 0;
input(problem no);
if problem no < 0 then go to end of program;
input(day, month, year);

```

```

for a := 1 step 1 until 30 do outchar(112);
head;
if year < 1968 v year > 1981 then begin

writetext(←
sar); write(←ddd-ddd, year); writetext(← 1 org.-); write(ddd, problem no);
stop;
end of test;
comment NOW COMES THE PROGRAM;

begin
real tm, roV, SaV22, SaJ22, roJ, vpS1, vpA1, vpFe, vpJ1, vpMn,
vpCa, vpK, vpK, vpKa, vpK, vpO, vpP, vpK, vpC, vpB, vpC1,
vpS, vpL1, vpCo, vpCd, StrJ, v, a, SsaAJ, SsaAV, B,
S1S1, S1S2, S1S3, S1S4, S1S5, S1S6, S1S7, S1S8, S1S9, S1S10, S1S11, S1S12, S1S13,
S1S14, S1S15, S1S16, S1S17, StJ1, StJ2, StJ3, StJ4, StJ5, StJ6, StJ7, StJ8, StJ9, StJ10, StJ11, StJ12,
StJ13, StJ14, StJ15, StJ16, StJ17, b, q, L1, L2, L2a2, tn, Sa, Str, D, L,
Rav, Verr, Sac, l, C, d, K, R2, R2d, roJr, cpmr, vr, A1, B1, CRa, SD, SaJ2dr,
SaJ22dr, SsaAJdr, StrJdr, S1S1dr, S1S2dr, S1S3dr, S1S4dr, S1S5dr,
S1S6dr, S1S7dr, StJ1dr, StJ2dr, StJ3dr, StJ4dr, StJ5dr, StJ6dr, StJ7dr, CRa,
sumr, sumr2, ad1, Ra, Rar, R, le, K1, R2, K3, E1, E2, E3, E1a, E2a, E3a, e1, e2, e3, D1;

integer m, k, n, g, c, h;

real procedure expoint(x);
value x;
real x;

begin
if x > 1 then goto A;
if x > 0 then goto B;
if x > -6 then goto D;
if x > -9 then go to C;

x := abs(6/x);
expoint := exp(6/x) * ((((((((((((((((-87.6579265 * xx + 718.72221261) * xx
-2654.1918821) * xx
+5822.35141945) * xx
-8423.77492395) * xx
+8440.09131190) * xx
-5987.99131136) * xx
+3029.44805047) * xx
-1090.49767874) * xx
+ 276.81688416) * xx
- 48.65976398) * xx
+ 5.7706208) * xx
- 0.41716973) * xx
+ 0.04726299) * xx
+ 0.05510829) * xx
+ 0.16667063) * xx
+ 0.99999999)/6;
goto end;
A: expoint := exp(-x) * (((x+8.5733287401) * xx + 18.0590169730) * xx
+8.634760825) * xx + 0.267773743)/
((((x+9.5733223454) * xx + 25.6329561486) * xx
+21.0596530827) * xx + 3.9584969228) * xx;
goto end;

```

```

B: expoint:=-ln(x) + ((( (1.07857w-3 * x -9.76004w-2)xx
+5.519968w-2)xx
-0.24991055)xx
+0.999999193)xx
-0.57721566;

goto end;

D: x := abs(x/6);
expoint:=-ln(xxx)-(((((((((((5.06913215xx - 12.77625843)xx
+29.88740807)xx
-37.45936457)xx
+37.99866850)xx
-16.20406335)xx
+17.06181298)xx
+ 8.11345875)xx
+13.49321199)xx
+13.43177582)xx
+12.00520932)xx
+ 8.99979305)xx
+ 6.00000323)xx
- 0.577215666;

goto end;
C*x := -x;
expoint := -(8.31262743 * x ^ (-1)
-1.37913164w1 * x ^ (-1/2)
+1.00379747w1 * x ^ (-1/3)
-(+2.88620786
-2.05089143 * x
+6.06413093w-1 * x ^ 2
-9.57253358w-2 * x ^ 3
+8.52559931w-3 * x ^ 4
-4.06747109w-4 * x ^ 5
+8.12355608w-6 * x ^ 6)) * (1+(24-5xx)w-7) * exp(x)/x;

end;
end of real procedure expoint;

real procedure FAV(1);
value 1; real 1;
FAV := q*L2*(L1/2*(expoint(1/L)-expoint(1/L1)+ln(L1/L))
/(L1/2-L1/2)-L2/2*(expoint(1/L)-expoint(1/L2)+ln(L2/L))
/(L2/2-L1/2))/(12.5664wDx1*(L1/2-L2/2));

real procedure F2AV(1);
value 1; real 1;
F2AV := q*L2/2*(expoint(1/12)-expoint(1/L1)+ln(L1/12))/(12.5664w
L2*x1*(L1/2-L2/2));

real procedure sinh(x);
value x; real x;
sinh:=(exp(x)-exp(-x))/0.5;

real procedure lel(x);
value x; real x;
begin
real leak1, leak2, leak3, leakt, x1, x2,x3,sinh1,sinh2,sinh3;
x1 := (Rw*x)/L1; x2 := (Rw*x)/L2; x3 := (Rw*x)/L;
sinh1:=sinh(x1); sinh2:=sinh(x2); sinh3:=sinh(x3);

```

```

leak1 :=x1/sinh1;
leak2 :=L2/2/(L1/2-L2/2)*(x1/sinh1-x2/sinh2);
leak3 :=L2/2*(K1*x1/sinh1+K2*x2/sinh2+K3*x3/sinh3);
leakt:=leak1+leak2+leak3;
lel :=2.13*(D1*leak1+D2*leak2+D3*leak3)/leakt-x;
end of lel;

real procedure F2AV(1);
value 1; real 1;
begin
real C1,C2,E11,E12,Eml1,Eml2;
C1:=exp(-2*xR/L1)/(1-exp(-2*xR/L1));
C2:=exp(-2*xR/L2)/(1-exp(-2*xR/L2));
E11 :=expoint(1/L1);
E12 :=expoint(1/L2);
Eml1 :=expoint(-1/L1);
Eml2 :=expoint(-1/L2);
F2AV :=q*L2/2/(12.56636wD2*(L1/2-L2/2)*x1*
(E12-E11+ln(L1/L2)
-C1*(E11-Eml1)
-C2*(E12-Eml2));
end of F2AV;

real procedure F1AV(1);
value 1; real 1;
begin
real C1,C2,C3,E11,E12,E13,Eml1,Eml2,Eml3;
C1:=exp(-2*xR/L1)/(1-exp(-2*xR/L1));
C2:=exp(-2*xR/L2)/(1-exp(-2*xR/L2));
C3:=exp(-2*xR/L)/(1-exp(-2*xR/L));
E11 :=expoint(1/L1);
E12 :=expoint(1/L2);
E13 :=expoint(1/L);
Eml1 :=expoint(-1/L1);
Eml2 :=expoint(-1/L2);
Eml3 :=expoint(-1/L);
F1AV :=q*L/2/(12.56636wDx1)*
(K1*(E13-E11+ln(L1/L))
+K2*(E13-E12+ln(L2/L))
-K1*C1*(E11-Eml1)
-K2*C2*(E12-Eml2)
-K3*C3*(E13-Eml3));
end of F1AV;

procedure HYP(x,F,x1,x2,eps,error);
value x1,x2;
real x,x1,x2,eps;
label error;

real procedure F;
comment: A.E.K. - August 26th 1965 - this procedure locates a zero for the
function F as described in SA-31/2;
begin
real x3,f1,f2,f3,T,N,d,t;

```

```

f := x2-x1;
f1 := F(x1); f2 := F(x2);
if f1 * f2 > 0 then goto error;
if f1 = 0 then x := x1 else
if f2 = 0 then x := x2 else
goto bis;
goto out;
hyp:
T := 1/f1 - 1/f2;
N := (1/f1-1/f2)/(x1-x2) + (1/f2-1/f1)/(x2-x3);
r := if N ≠ 0 then T/N else 0;
if abs(r3) ≤ abs(r2) ∨ abs(r) > d/2 then
bis:
x := (x1+x2)/2 else
if (x1+x1) * (x1-x2) < 0 then x := x + f;
d := abs(f);
r := F(x);
if f = 0 then goto out;
if sign(r) = sign(r2) then
begin
x3 := x2; f3 := f2;
end
else
begin
x3 := x1; f3 := f1;
x1 := x2; f1 := f2;
end;
x2 := x; f2 := f;
if abs(x1-x2) < eps then goto out;
if abs(x2-x3) > eps then goto hyp;
if F(x-eps*sign(x1-x2)) * f < 0 then goto out;
d := (x1-x2)/2 * sign(r3);
for x := (x1+x2)/2, x+d*sign(F(x)) while abs(d) > eps do
d := d/2;
out:
end of HYP;

```

```

outcopy(⟨[ ]⟩); CR(5);
input(tm,q,c,Veff,SaC,SD,ad1,d,g,K,C,vr,roJr,cprR,Rm,
vpH,vpL,vpB,vpC,vpH,vp0,vpBa,vpBe,vpAl,vpS1,
vpF,vpS,vpCl,vpK,vpCa,vpTi,vpMn,vpFe,vpCo,vpCd,SaW2dr,a,m);
b := 5; l:=ad1/2;
roV := 0.9998u+tm*(6.29575w-5+tm*(-8.20132w-6+4.82612w-8*tm));
outtext(
⟨[ tm q Veff SaC SD d ad1 c g ]⟩;
CR(1); outtext(
⟨[ c n/sec cm3 1/cm cm2 cm cm ]⟩;
CR(1);
output(⟨nd.d⟩,tm); outsp(2);
output(⟨n.d⟩,q); outsp(2);
output(⟨nd.d⟩,Veff); outsp(2);
output(⟨n.d⟩,SaC); outsp(2);
output(⟨nd.d⟩,SD); outsp(2);
output(⟨nd.d⟩,d); outsp(2);

```

```

output(⟨nd.d⟩,ad1); outsp(2);
output(⟨n,c⟩); outsp(2);
output(⟨n,g⟩);
CR(2);

outtext(
⟨[ H Li B C N O Na Mg Al S1 ]⟩;
CR(1); outtext(
⟨[ wto/o wto/o wto/o wto/o wto/o wto/o wto/o wto/o wto/o ]⟩;
CR(1);
output(⟨nd.d⟩,vpH); outsp(2);
output(⟨n.d⟩,vpLi); outsp(2);
output(⟨n.d⟩,vpB); outsp(2);
output(⟨nd.d⟩,vpC); outsp(2);
output(⟨nd.d⟩,vpH); outsp(2);
output(⟨nd.d⟩,vpO); outsp(2);
output(⟨nd.d⟩,vpBa); outsp(2);
output(⟨nd.d⟩,vpBe); outsp(2);
output(⟨nd.d⟩,vpAl); outsp(2);
output(⟨nd.d⟩,vpS1); CR(2);
outtext(
⟨[ P S Cl K Ca Ti Mn Fe Co Ca ]⟩;
CR(1); outtext(
⟨[ wto/o wto/o wto/o wto/o wto/o wto/o wto/o wto/o wto/o ]⟩;
CR(1);
output(⟨nd.d⟩,vpP); outsp(2);
output(⟨nd.d⟩,vpS); outsp(2);
output(⟨n.d⟩,vpCl); outsp(2);
output(⟨nd.d⟩,vpK); outsp(2);
output(⟨nd.d⟩,vpCa); outsp(2);
output(⟨nd.d⟩,vpTi); outsp(2);
output(⟨nd.d⟩,vpMn); outsp(2);
output(⟨nd.d⟩,vpFe); outsp(2);
output(⟨n.d⟩,vpCo); outsp(2);
output(⟨n.d⟩,vpCd); CR(2);
outtext(
⟨[ roJr vr cprR Rm C K SaW2/roJ ]⟩;
if a>0 then
outtext(⟨[ roJ Rm ]⟩);
CR(1);
outtext(
⟨[ g/cm3 o/o cpm cm cm2/g ]⟩;
if a>0 then
outtext(⟨[ g/cm3 cm ]⟩);
CR(1);
output(⟨nd.d⟩,roJr); outsp(2);
output(⟨nd.d⟩,vr); outsp(2);
output(⟨nd.d⟩,cprR); outsp(2);
if Rm<0 then
outtext(⟨[ infinite ]⟩);
if Rm>0 then
output(⟨nd.d⟩,Rm);
outsp(2);
SaW22 := 0.022205*roV; SaWV := 1.3354*roV;
SaW22dr := (0.00331*vpS1+0.005135*vpAl+0.028254*vpBe
+0.072935*vpTi+0.14471*vpBa+0.006612*vpCa
+0.001709*vpMg+0.03189*vpK+0.013229*vpMn
+0.19885*vpFe+0.000075*vpO+0.003889*vpP
+0.08085*vpH+0.000181*vpC+42.027*vpB

```

+0.5742xvpC1+0.009767xvpS+6.162xvpL1
+0.3883xvpC0+13.127xvpCd)/100;

SeoAJdr := (0.0012977xvpS1+0.001158xvpA1+0.002124xvpFe
+0.0010501xvpL1+0.0004589xvpBn+0.001124xvpCa
+0.003667xvpPz+0.0005909xvpK+0.000455xvpPa
+11.85xvpH+0.009882xvpO+0.00313xvpP
+0.03069xvpN+0.02004xvpC+0.0205xvpB
+0.007665xvpC1+0.0006443xvpS+0.01750xvpL1
+0.0012136xvpC0+0.0003336xvpCd)/100;

StrJdr := (0.03559xvpS1+0.0304xvpA1+0.1172xvpFe
+0.04959xvpL1+0.02491xvpBn+0.04433xvpCa
+0.08671xvpPz+0.02271xvpK+0.1017xvpPa
+18.063xvpH+0.15152xvpO+0.09514xvpP
+0.4095xvpN+0.2274xvpC+0.2090xvpB
+0.2667xvpC1+0.02023xvpS+0.10988xvpL1
+0.07073xvpC0+0.03729xvpCd)/100;

N := (vpH+vpO+vpA1+vpS1+vpFe);
vpH := vpH/N; vpO := vpO/N; vpA1 := vpA1/N;
vpS1 := vpS1/N; vpFe := vpFe/N;

S1B1dr := (11.950xvpH+0.017166xvpO+0.002255xvpA1
+0.003366xvpS1+0.00437xvpFe);

S1B2dr := (11.486xvpH+0.016669xvpO+0.002309xvpA1
+0.003091xvpS1+0.002966xvpFe);

S1B3dr := (4.076xvpH+0.014813xvpO+0.005239xvpA1
+0.0017518xvpS1+0.0008629xvpFe);

S1B4dr := (1.7549xvpH+0.009720xvpO+0.003837xvpA1
+0.003200xvpS1+0.0007977xvpFe);

S1B5dr := (1.124xvpH+0.002203xvpO+0.003791xvpA1
+0.003382xvpS1+0.0009605xvpFe);

S1B6dr := (0.8640xvpH+0.008255xvpO+0.003211xvpA1
+0.002545xvpS1+0.0010491xvpFe);

S1B7dr := (0.7593xvpH+0.007347xvpO+0.002951xvpA1
+0.002651xvpS1+0.0011198xvpFe);

S1SV1 := roV * 1.3540; S1SV2 := roV * 1.3001;

S1SV3 := roV * 0.4692; S1SV4 := roV * 0.2050;

S1SV5 := roV * 0.12782; S1SV6 := roV * 0.10401;

S1SV7 := roV * 0.09149;

StJ1dr := (4.254xvpH+0.13702xvpO+0.0305xvpA1
+0.04717xvpS1+0.1225xvpFe);

StJ2dr := (5.890xvpH+0.13318xvpO+0.03113xvpA1
+0.04340xvpS1+0.0828xvpFe);

StJ3dr := (1.3731xvpH+0.11835xvpO+0.07066xvpA1
+0.02459xvpS1+0.02409xvpFe);

StJ4dr := (0.6067xvpH+0.07766xvpO+0.05117xvpA1
+0.04492xvpS1+0.02227xvpFe);

StJ5dr := (0.4017xvpH+0.017606xvpO+0.05112xvpA1
+0.04749xvpS1+0.02682xvpFe);

StJ6dr := (0.3127xvpH+0.06595xvpO+0.04330xvpA1
+0.03571xvpS1+0.02929xvpFe);

StJ7dr := (0.2748xvpH+0.05870xvpO+0.03979xvpA1
+0.03722xvpS1+0.03126xvpFe);

StV1 := roV * 0.59771; StV2 := roV * 0.5468;

StV3 := roV * 0.2588; StV4 := roV * 0.13686;

StV5 := roV * 0.06059; StV6 := roV * 0.09356;

StV7 := roV * 0.08288;

roJ:=roJr; Rm:=Rmr; n:=0; v:=0.01xvr; go to 12;

11:

if a>0 then begin
if h=1 then begin
outtext(
<< v &av CR L L1 L2 D D2 &v);
CR(1); outtext(
<<o/o n/cm2/s cpm cm cm cm cm n/cm2/s);
end;

if h=2 then begin
outtext(
<< v CR le D1 tn);
CR(1); outtext(
<<o/o cpm cm cm C);
end;

if h=0 then begin
outtext(<< v roJ CR); CR(1);
outtext(<<o/o g/cc cpm); end;
end else begin
outtext(<< v roJ CR(cal) CR(exp) delta Rm); CR(1);
outtext(<<o/o g/cm3 cpm cpm o/o cm);
sumer:=0; sumer2:=0; end;
CR(2);

k:=-1;

lab:

k:=k+1;

v := 0.01xvob;

13:

if a<0 then begin
input(v);
if v<0 then go to 1a5;
input(roJ,CR,Rm); v:=0.01xvr;
end;

12:

if SeAJ2dr>0 then SeAJ2 := roJnSeAJ2dr else SeAJ2 := roJnSeAJ22dr;
SeoAJ := SeoAJdrroJ; StrJ := StrJdrroJ;
S1B1 := S1B1drroJ; S1B2 := S1B2drroJ;
S1B3 := S1B3drroJ; S1B4 := S1B4drroJ;
S1B5 := S1B5drroJ; S1B6 := S1B6drroJ;
S1B7 := S1B7drroJ; StJ1 := S1B1drroJ;
StJ2 := StJ2drroJ; StJ3 := StJ3drroJ;
StJ4 := StJ4drroJ; StJ5 := StJ5drroJ;
StJ6 := StJ6drroJ; StJ7 := StJ7drroJ;

tn := tm + 0.92x(tm+275)x(SeAJ2+vrSeAV22)
xsgt(295/(275+tm))/(SeoAJ+vrSeoAV);

Se := 0.8862x(SeAJ2+vrSeAV22)xsgt(295/(275+tm));
Str := Se+StrJ+vr2.156x(0.047+0.933xsgt(295/(tm+275)))xroV;
D := 1/(3xStr); L := sgt(D/Se);
L1 := sgt(0.40547/(3x(StJ5+vrStV5)x(S1B5+vrS1B5)))
+0.28768/(3x(StJ6+vrStV6)x(S1B6+vrS1B6)))
+0.11778/(3x(StJ7+vrStV7)x(S1B7+vrS1B7)));

```

L2m2 := 11.14829/(5*(Stw2+vwStw2)*(S1Sj2+vwS1SV2))
      + 2.3026/(5*(Stw3+vwStw3)*(S1Sj3+vwS1SV3))
      + 0.69315/(5*(Stw4+vwStw4)*(S1Sj4+vwS1SV4));

L2 := sqrt(ln(1.44/(tw8.6167w-5*(tn+273)))/(5*(Stw1+vwStw1)
      *(S1Sj1+vwS1SV1))+L2m2);

D1 := (c.40547/(Stw5+vwStw5)+0.28768/(Stw6+vwStw6)
      +0.11778)/(Stw7+vwStw7))/2.432799;

D2 := (ln(1.44/(tw8.6167w-5*(tn+273)))/(5*(Stw1+vwStw1)
      + 11.14829/(5*(Stw2+vwStw2)) + 2.3026/(5*(Stw3+vwStw3))
      + 0.69315/(5*(Stw4+vwStw4)))
      /ln(1.44/(tw8.6167w-5*(tn+273))) + 14.14404);

K1 := L1/2/((L1/2-L2/2)*(L1/2-L2/2));
K2 := L2/2/((L2/2-L1/2)*(L2/2-L1/2));
K3 := -(K1+K2);

if Rw<0 then begin
  if g=1 then begin
    E1 := expoint((d-1)/L1)-expoint((d+1)/L1);
    E2 := expoint((d-1)/L2)-expoint((d+1)/L2);
    E3 := expoint((d-1)/L)-expoint((d+1)/L);

    Rav := qnL/2/(25.13272xDa1)*(K1xE1+K2xE2+K3xE3);
    Rd := qnL2/2/(25.13272xD2x1*(L1/2-L2/2))*(E1-E2);
  end;

  if g=2 then begin
    Rav := FIAV(2x1);
    Rd := F2AV(2x1);
  end;

  if g=3 then begin
    Rav := (FIAV(1-d)*(1-d)+FIAV(1+d)*(1+d))/(2x1);
    Rd := (F2AV(1-d)*(1-d)+F2AV(1+d)*(1+d))/(2x1);
  end;

  if g=4 then begin
    Rav := FIAV(1);
    Rd := F2AV(1);
  end;
end;

if Rw=0 then begin
  HYP(1e,1e1,0,100,w-5*1e,1b6);
  R:= Rw + 1e;

  if g=1 then begin
    E1 := expoint((d-1)/L1)-expoint((d+1)/L1);
    E2 := expoint((d-1)/L2)-expoint((d+1)/L2);
    E3 := expoint((d-1)/L)-expoint((d+1)/L);
  end;

```

```

E1m:= expoint(-(d-1)/L1)-expoint(-(d+1)/L1);
E2m:= expoint(-(d-1)/L2)-expoint(-(d+1)/L2);
E3m:= expoint(-(d-1)/L)-expoint(-(d+1)/L);

e1:=exp(-2xR/L1); e2:=exp(-2xR/L2); e3:=exp(-2xR/L);

Rav:=qnL/2/(25.13272xDa1)*
      (K1*(E1-e1xE1m)/(1-e1)
      +K2*(E2-e2xE2m)/(1-e2)
      +K3*(E3-e3xE3m)/(1-e3));

Rd:=qnL2/2/(25.13272xD2x1*(L1/2-L2/2))*
      ((E1-e1xE1m)/(1-e1)
      -(E2-e2xE2m)/(1-e2));

end;

if g=2 then begin
  Rav:=FIAV(2x1);
  Rd:=F2AV(2x1);
end;

if g=3 then begin
  Rav:=(FIAV(1-d)*(1-d)+FIAV(1+d)*(1+d))/(2x1);
  Rd:=(F2AV(1-d)*(1-d)+F2AV(1+d)*(1+d))/(2x1);
end;

if g=4 then begin
  Rav:=FIAV(1);
  Rd:=F2AV(1);
end;

end;

if n=0 then begin
  if c=1 then begin
    A1 := 60xVerfT6aCnRavx0.8862xsqrt(295/(tn+273));
    B1 := 0;
  end;
  if c=2 then begin
    A1 := 60xSD/4xRav;
    B1 := 60x2xVerfT6aCnKxRd2xsqrt(295/(tw(tn+273)))
          /ln(2x6/(tw8.6167w-5*(tn+273)));
  end;
  if c=3 then begin
    A1 := 60xSD/4xRav;
    B1 := 60xSD/4xKxRd;
  end;
  if c=4 then begin
    A1 := 60x2xVerfT6aCnRavx0.8862xsqrt(295/(tn+273));
    B1 := 60x2xVerfT6aCnKxRd2xsqrt(295/(tw(tn+273)))
          /ln(2x6/(tw8.6167w-5*(tn+273)));
  end;
end;

```

```

if C<0 then
  C:=(cpm2-B1)/A1;
n:=1;
if a>0 then
  input(r0J,h,Rm);
  output(dn.dddd,C); outsp(2);
  output(dn.dddd,K); outsp(2);
  if SaJ2dr < 0 then
    output(dn.dddd-dJ,SaJ22dr) else
    output(dn.dddd-dJ,SaJ2dr); outsp(2);
  if a>0 then begin
    output(dn.dddd,roJ); outsp(2);
    if Rm>0 then
      output(dn.dddd.dddd,Rm);
    if Rm<0 then
      outtext(⟨<infinite⟩);
    end;
  CR(5);
  go to 13;
end;

if c = 1 then
  Cfa := C*sqrt(SaCk/Avx60x0.8862*sqrt(295/(tn*273)));
if c = 2 then
  Cfa := 60x(CaSD/Av/Av
    +2*sqrt(SaCk/Ka/2d*sqrt(295/(tn*(tn*273))))/
    ln(2u6/(tn8.6167u-5x(tn*273))));
if c = 3 then
  Cfa := 60x(CaSD/Av/Av+KaSD/Av/2d);
if c = 4 then
  Cfa := C*sqrt(SaCk/Avx60x0.8862*sqrt(295/(tn*273)))
    +60x2*sqrt(SaCk/Ka/2d*sqrt(295/(tn*(tn*273))))/
    ln(2u6/(tn8.6167u-5x(tn*273)));

if a>0 then begin
  output(dn.dddd,100xv);
  if h=0 then begin
    outsp(2);
    output(dn.dd,roJ);
    outsp(2);
    output(dn.dddd,CRa);
    CR(1); end;
  if h=1 then begin
    outsp(1);
    output(dn.dd-dJ,Av); outsp(2);
    output(dn.dddd-dJ,CRa); outsp(1);
    output(dn.dddd-dJ,L); outsp(2);
    output(dn.dddd-dJ,L1); outsp(2);
    output(dn.dddd-dJ,L2); outsp(2);
    output(dn.dddd-dJ,D); outsp(2);
    output(dn.dddd-dJ,D2); outsp(1);
    output(dn.dd-dJ,2d); CR(1); end;
  if h=2 then begin
    outsp(2);
    output(dn.dddd-dJ,CRa); outsp(2);
    if Rm>0 then begin
      outsp(1);
      output(dn.dddd-dJ,1e); outsp(2);
    end else begin
      outtext(⟨not cal.⟩); outsp(2); end;
  end;

```

```

output(dn.dddd-dJ,D1); outsp(2);
output(dn.dddd-dJ,tn); CR(1); end;
if k=m then go to 1b6;
go to 1a5;
end;

if a<0 then begin
  output(dn.dddd,100xv); outsp(2);
  output(dn.dddd,roJ); outsp(2);
  output(dn.dddd,CRa); outsp(2);
  output(dn.dddd,CRe); outsp(2);
  output(dn.dddd-dJ,(CRa-CRe)/CRx100); outsp(2);
  if Rm>0 then
    output(dn.dddd-dJ,Rm);
  if Rm<0 then
    outtext(⟨<infinite⟩);
  CR(1);
  sumer:=abs(CRa-CRe)+sumer;
  sumer2:=(CRa-CRe)/2+sumer2;
  go to 13;

1a5:
CR(5);
outtext(⟨<Summation of all |CR(cal)-CR(exp)| : ⟩);
output(dn.dddd-dJ,sumer); CR(2);
outtext(⟨<Summation of all (CR(cal)-CR(exp))/2 : ⟩);
output(dn.dddd-dJ,sumer2);
end;
1b6:
end of calculation;
go to start;

end of program:
outsum; for a := 1 step 1 until 60 do outchar(112);
end of program;

```

Appendix IV

Output from the MOPS-III Programme
(Calibration curve calculation)

A.E.K. - Program No. 667 - Problem No 1 - 11. 8.1972
MOPS-III

TEST OF MOPS-III

tm	q	Veff	ScC	SD	d	adl	c	g						
C	n/sec	cm ³	1/cm	cm ²	cm	cm	cm							
20.0	1.40 ₁₅	6.14	2.19 ₁₀	19.64	6.700	1.250	2	1						
H	L1	B	C	N	O	Na	Mg	Al	Si					
wto/o	wto/o	wto/o	wto/o	wto/o	wto/o	wto/o	wto/o	wto/o	wto/o					
0.326	1.0 ₁₅ ⁻³	3.0 ₁₅ ⁻³	0.506	0.055	53.50	0.740	1.38	3.75	34.28					
P	S	Cl	K	Ca	Ti	Mn	Fe	Co	Cd					
wto/o	wto/o	wto/o	wto/o	wto/o	wto/o	wto/o	wto/o	wto/o	wto/o					
0.058	0.001	7.0 ₁₅ ⁻³	1.730	1.55	0.320	0.034	1.76	6.0 ₁₅ ⁻⁴	2.0 ₁₅ ⁻⁵					
roJr	vr	cpmr	Rur	C	K	ScJ22/roJ	roJ	Ru						
g/cm ³	o/o	cpm	cm	cm ² /g	cm ² /g	g/cm ³	cm	cm						
0.000	100.0	7.410 ₁₅ ⁴	infinite	0.4575	0.0900	4.990 ₁₅ ⁻³	1.460	infinite						
v	Av	CR	L	L1	L2	D	D2	δ						
o/o	n/cm ² /s	cpm	cm	cm	cm	cm	cm	n/cm ² /s						
0.0	2.45 ₁₀	1	6.975 ₁₀	3	1.5037 ₁₀	1	2.1143 ₁₀	1	2.6193 ₁₀	1	1.3903	2.2356	9.80 ₁₀	1
5.0	4.84 ₁₀	1	1.139 ₁₀	4	1.1550 ₁₀	1	1.6360 ₁₀	1	1.8621 ₁₀	1	9.6459 ₁₀ ⁻¹	1.9359	1.27 ₁₀	2
10.0	7.60 ₁₀	1	1.619 ₁₀	4	9.4522	1	1.3647 ₁₀	1	1.4807 ₁₀	1	7.3653 ₁₀ ⁻¹	1.7095	1.55 ₁₀	2
15.0	1.06 ₁₀	2	2.128 ₁₀	4	8.0274	1	1.1828 ₁₀	1	1.2413 ₁₀	1	5.9507 ₁₀ ⁻¹	1.5319	1.81 ₁₀	2
20.0	1.38 ₁₀	2	2.654 ₁₀	4	6.9887	1	1.0497 ₁₀	1	1.0740 ₁₀	1	4.9919 ₁₀ ⁻¹	1.3886	2.06 ₁₀	2
25.0	1.71 ₁₀	2	3.21 ₁₀	4	6.1945	1	9.4677	1	9.4929	1	4.2988 ₁₀ ⁻¹	1.2705	2.29 ₁₀	2
30.0	2.05 ₁₀	2	3.731 ₁₀	4	5.5661	1	8.6426	1	8.5216	1	3.7745 ₁₀ ⁻¹	1.1712	2.51 ₁₀	2
35.0	2.39 ₁₀	2	4.267 ₁₀	4	5.0557	1	7.9626	1	7.7405	1	3.3641 ₁₀ ⁻¹	1.0867	2.71 ₁₀	2
40.0	2.73 ₁₀	2	4.794 ₁₀	4	4.6324	1	7.3903	1	7.0970	1	3.0	1.0137	2.89 ₁₀	2

Appendix V

Output from the MOPS-III Programme
(Comparison between calculated and input count rates)

A.E.K. - Program No. 667 - Problem No 2 - 11. 8.1972
MOPS-III

TEST OF MOPS-III

tm	q	Veff	ScC	SD	d	adl	c	g	
C	n/sec	cm ³	1/cm	cm ²	cm	cm	cm		
20.0	1.40 ₁₅	6.14	2.19 ₁₀	19.64	6.700	1.250	2	1	
H	L1	B	C	N	O	Na	Mg	Al	Si
wto/o	wto/o	wto/o	wto/o	wto/o	wto/o	wto/o	wto/o	wto/o	wto/o
0.326	1.0 ₁₅ ⁻³	3.0 ₁₅ ⁻³	0.506	0.055	53.50	0.740	1.38	3.75	34.28
P	S	Cl	K	Ca	Ti	Mn	Fe	Co	Cd
wto/o	wto/o	wto/o	wto/o	wto/o	wto/o	wto/o	wto/o	wto/o	wto/o
0.058	0.001	7.0 ₁₅ ⁻³	1.730	1.55	0.320	0.034	1.76	6.0 ₁₅ ⁻⁴	2.0 ₁₅ ⁻⁵
roJr	vr	cpmr	Rur	C	K	ScJ22/roJ	roJ	Ru	
g/cm ³	o/o	cpm	cm	cm ² /g	cm ² /g	g/cm ³	cm	cm	
1.000	1.0	1.000	1.000	0.4575	0.0900	4.990 ₁₅ ⁻³	1.460	infinite	
v	roJ	CR(cal)	CR(exp)	delta	Ru				
vto/o	g/cm ³	cpm	cpm	o/o	cm				
0.0	1.460	6.975 ₁₀	6.975 ₁₀	0.0	500.0				
5.0	1.460	1.139 ₁₀	1.139 ₁₀	0.0	500.0				
10.0	1.460	1.619 ₁₀	1.619 ₁₀	0.0	500.0				
15.0	1.460	2.128 ₁₀	2.128 ₁₀	0.0	500.0				
20.0	1.460	2.654 ₁₀	2.654 ₁₀	0.0	500.0				
25.0	1.460	3.191 ₁₀	3.191 ₁₀	0.0	500.0				
30.0	1.460	3.731 ₁₀	3.731 ₁₀	0.0	500.0				
35.0	1.460	4.267 ₁₀	4.267 ₁₀	0.0	500.0				
40.0	1.460	4.794 ₁₀	4.794 ₁₀	0.0	500.0				

Summation of all |CR(cal)-CR(exp)|: 2.639₁₀ 1

Summation of all (CR(cal)-CR(exp))/2: 9.311₁₀ 1

