DOE/MC/21153-1870
(DE85013604)
Distribution Category UC-92a

Further Development of a Fracture Model for Lenticular Gas Sands

Final Report

By
I.D. Palmer
C.T. Luiskutty

Work Performed Under Contract No.: DE-AC21-84MC21153

For
U.S. Department of Energy
Office of Fossil Energy
Morgantown Energy Technology Center
P.O. Box 880
Morgantown, West Virginia 26505

By
Oral Roberts University
7777 South Lewis
Tulsa, Oklahoma 74171

April 1985

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

This report has been reproduced directly from the best available copy.

Available from the National Technical Information Service, U. S. Department of Commerce, Springfield, Virginia 22161.

Price: Printed Copy A05
Microfiche A01

Codes are used for pricing all publications. The code is determined by the number of pages in the publication. Information pertaining to the pricing codes can be found in the current issues of the following publications, which are generally available in most libraries: *Energy Research Abstracts (ERA); Government Reports Announcements* and Index *(GRA* and I); *Scientific and Technical Abstract Reports (STAR);* and publication NTIS-PR-360 available from NTIS at the above address.

Abstract

The existing model for hydraulic fracture propagation is modified to incorporate a correction for the effect of vertical flow. The vertical flow component should be taken into consideration when the stress contrasts between the payzone and the bounding zones are not very high. To solve for fracture expansion in this situation, first solutions for a line crack expanding through uniform stress region, and then for a line crack expanding through a medium with stress contrast are obtained. The second solution is a relation between the pressure at the center of a vertical section, the half height of the section, and the rate of vertical expansion in that section. The solution represents a first-order correction for the effect of finite fluid flow in the vertical direction. The manner in which this solution is adopted in the simulator is described. The simulator is restricted to the case of symmetric stress contrast, and for a Newtonian fluid flow with no leakoff. The correction is an approximate one under the assumption that the vertical expansion is slow, and this gives an upper limit for the wellbore height.

When the vertical flow is taken into consideration, the height growth will be smaller than the one without the vertical flow. To illustrate the effect of the correction term, we have shown the results obtained for a set of parameters with and without the correction. We have also compared our results with those obtained from the P3DH model developed at MIT. In the one case we compared, we found discrepancies, and explained the reasons for these. After compensating for these discrepancies, we find reasonable agreement between the results of the ORU and MIT models.

Table of Contents

Page

Introduction..... 1

Solution for line crack expanding through uniform stress region.... 3

Solution for line crack expanding through medium with stress contrast..... 4

The fracture simulator..... 8

Results..... 8

Comparison with MIT fracture model..... 10

Conclusion..... 17

Nomenclature..... 18

Acknowledgement..... 19

References..... 19

Appendix A - The asymptotic limit of slow vertical expansion..... 24

Appendix B - Fortran code and user instructions for vertically symmetric simulator with 1D vertical fluid flow in addition to 1D horizontal fluid flow..... 26

Sample Output File..... 32

List of Figures

Page

Figure 1 Fracture geometry 2

Figure 2 Fracture height as a function of penetration from the wellbore (with and without the correction for vertical flow) 9

Figure 3 Pressure profile within the fracture (with and without the correction for vertical flow) 11

Figure 4 Width profile for the vertical section at the wellbore 12

Figure 5 Fracture height for the MIT parameters (Table 4) with q = 2.5 bpm and width modified by a factor of 1.8 15

Figure 6 Width profile for the ORU adapted case with q = 2.5 bpm 16

List of Tables

Page

Table 1 Pumping and formation parameters used to illustrate the correction for finite flow rates in the vertical direction 21

Table 2 Parameter values at the onset of the asymptotic limit for a penny crack 21

Table 3 Values of critical radius as a function of fluid viscosity 22

Table 4 Pumping and formation parameters used to compare the ORU and MIT simulators for lower stress contrasts 22

Table 5 Comparison between ORU and MIT models, for parameters in Table 3 23

I. Introduction

Recently developed hydraulic fracture simulators (vertical fracture) allow for a fracture height which varies with distance from the wellbore. In the simplest of these,[1-5] the predominant fluid flow is taken as 1D flow along the payzone direction, and is based on the assumption that the fracture length/height ratio is large ($\gtrsim 4$). These simulators most commonly treat the height confinement due to greater minimum in-situ stresses above and below the payzone.

In two of these simulators[3-5] the vertical flow is neglected in computing the rate of fracture expansion in the vertical direction, making the problem much simpler. In the other simulator,[1,2] vertical flow is incorporated, and the resulting modeling analysis is considerably more complicated. We give here a method for providing a correction for the effect of vertical flow on our results, which previously did not include it. Our approach is quite different from the other model[1,2] which does include the effect of vertical flow, and much simpler, and therefore provides an independent assessment of the effect. This is important because, while the simulators without the vertical flow component[3-5] work well when stress contrasts between pay and bounding zones are high ($\gtrsim$ several hundred psi), their solution diverges at some lower value of stress contrast. On the other hand, the simulator[1,2] which incorporates the vertical flow is more likely to work in situations of low stress contrasts.

The correction for finite vertical flow is made in the following way. The geometry of the fracture is shown in Figure 1. The fractue length is divided into a number of intervals (this is the approach of each of this kind of simulator[1-5]), and each vertical section is regarded as independent of its neighbors, i.e, the section is a 2D or line crack. The vertical expansion of

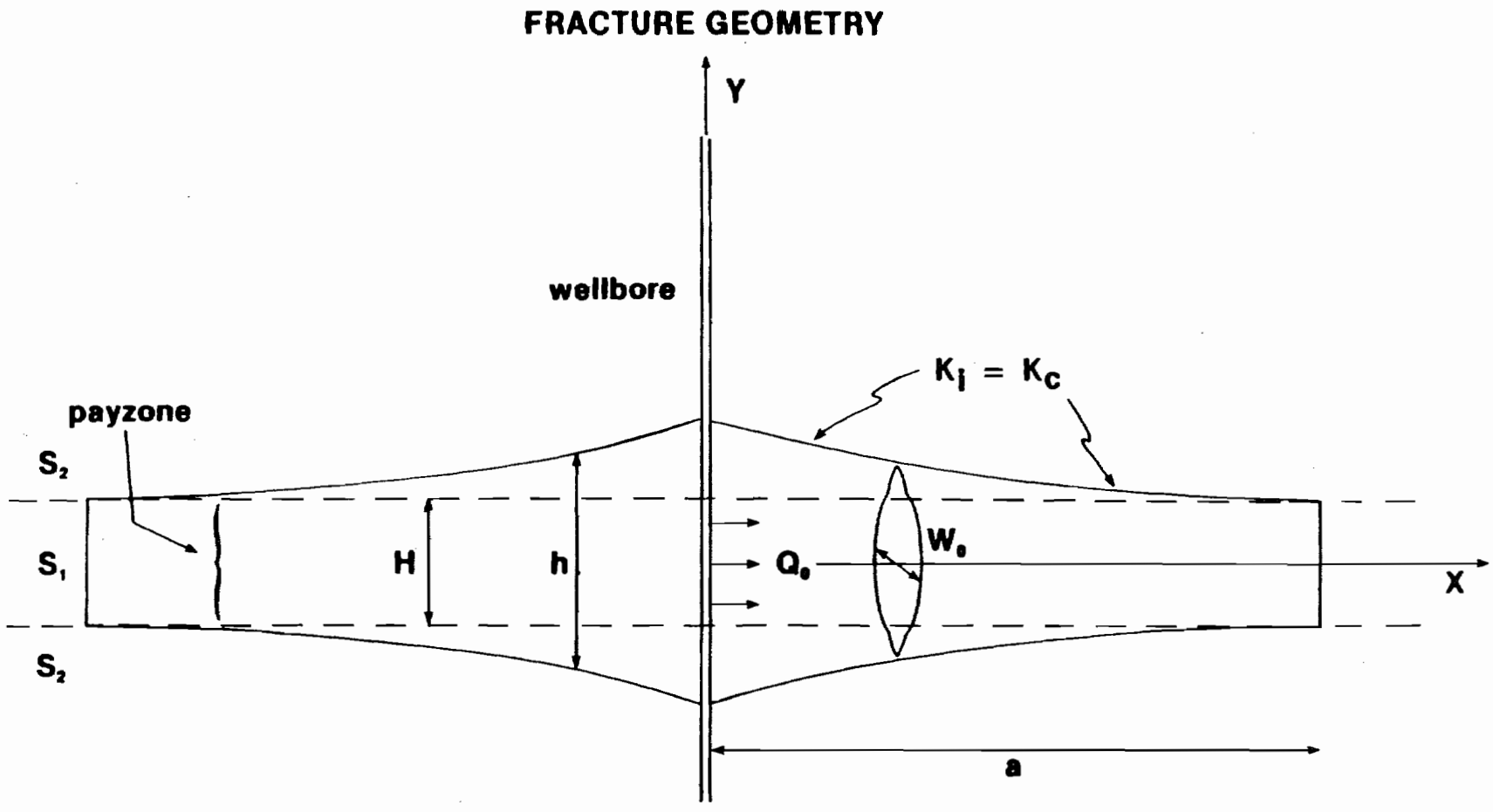

Figure 1. Fracture Geometry

any section from the payzone into a higher stress zone is equated to that of an "equivalent" section expanding through a uniform stress zone, whose solution we have determined. The equivalent line crack will thus be elliptical in shape.

2. Solution for line crack expanding through uniform stress region

The derivation follows the method given for a penny-shaped crack in Abe et al.[6] The results relevant to our discussion are given here.

If the rate of expansion is zero, of course we have the static result

$$\frac{p}{S} = \frac{K_c}{S\sqrt{\pi \ell}} + 1 \qquad (1)$$

and the fluid pressure in the crack is constant. (S is the minimum in situ stress). If the rate of expansion is not zero, but is slow, the region inaccessible to the fluid near the tip of the propagating fracture is very tiny. In this case, the average pressure in the crack, $\bar{p}$, can be given by (1), but p(0), the pressure at the center of the crack, will be greater than $\bar{p}$. Analysis shows that, for this asymptotic limit of slow vertical expansion

$$\frac{p_L(0)}{S} = \frac{\bar{p}_L}{S} + \frac{q_D}{(\bar{p}_L/S-1)^3}$$

$$\cdot \left[1 - \frac{2}{\pi} - 4\,(\ln 2-1)\left(\frac{1-\nu^2}{E}\right)(\bar{p}_L-S)\,\frac{\ell}{q(0)}\left(\frac{d\ell}{dt}\right)_L\right] \qquad (2)$$

$$\text{with } \frac{\bar{p}_L}{S} = \frac{Kc}{S\sqrt{\pi \ell}} + 1 \qquad (3)$$

$$\text{and } q_D = \frac{3\mu q(0)E^3}{16S^4\ell^2(1-\nu^2)^3} \qquad (4)$$

The meaning of the parameters is given at the end of the paper. The subscript L refers to the asymptotic limit. This equation indicates that $p_L(0)$ is greater than $\bar{p}_L$, as it must be, by an amount that increases with the rate of crack expansion, $d\ell/dt$.

Previously, we solved for fracture growth[4,5] by integrating the equation for pressure drop, dp/dx, from the fracture tip back to the wellbore. Knowing the pressure at a point x, we used (1), or rather an extension of it to the case of a payzone bounded by higher stress zones, to compute ℓ, the fracture height at that point. Our approach now is to take the pressure at x to be $p_L(0)$, and to solve (2) to find ℓ. Of course, we have to know $d\ell/dt$, the rate of height growth at this point x, and this is obtained by solving for two fractures closely spaced in time. The method is iterative, since $(d\ell/dt)_L$ depends on ℓ and vice versa.

3. Solution for line crack expanding through medium with stress contrast

In the actual situation, a vertical section expands from a region of minimum in-situ stress S_1 into a region of higher stress, S_2. Our approximate solution for such an expanding line crack is based on modifications to equations (2)-(4) as follows:

(i) Define an "equivalent" line crack of elliptical shape such that it has the same average width as the actual line crack. We do this by assuming that the entire crack lies in a uniform stress region, S_1, and by adjusting the elastic modulus E to E_q. Nolte[3] has shown that, for the same stress contrast above and below the payzone (i.e., vertically symmetric fracture),

$$E_q = sE \simeq \frac{1.2hE}{H} \qquad (5)$$

for $S_2 > S_1$, $E_q > E$; s is the stiffness multiplier.

(ii) Modify equations (2) and (4) by replacing S by S_1 and E by E_q. That is, when a stress contrast exists, we assume that the relation between $p_L(0)$ and $\bar{p}_L$ is the same as it is for the equivalent line crack. Since the difference between these two pressures is basically determined by the fluid flow, we are assuming that, to first-order, the vertical fluid flow behavior in the "pinched-in" section is the same as in the equivalent elliptical section.

(iii) Replace equation (3) by the corresponding zero-flow result in the case that a stress contrast does exist. This comes from the crack expansion criterion $K_i = K_c$.

The results are for non-zero stress contrast:

$$\frac{p_L(0)}{S_1} = \frac{\bar{p}_L}{S_1} + \frac{q_D}{(\bar{p}_L/S_1-1)^3}\left[1 - \frac{2}{\pi} - 4\,(\ln 2-1)\left(\frac{1-\nu^2}{E_q}\right)(\bar{p}_L-S_1)\,\frac{\ell}{q(0)}\left(\frac{d\ell}{dt}\right)_L\right] \quad (6)$$

$$\frac{\bar{p}_L}{S_1} - 1 = \frac{K_c}{S_1\sqrt{\pi\ell}} + \frac{2}{\pi}\,\frac{S_2-S_1}{S_1}\cos^{-1}\left(\frac{H}{2\ell}\right) \quad (7)$$

$$\text{with } q_D = \frac{3\mu q(0)E_q^{\,3}}{16\,S_1^{\,4}\ell^2(1-\nu^2)^3} \text{ and } E_q = \frac{1.2hE}{H} \quad (8)$$

The problem now becomes: given $p_L(0)$ at the center of a vertical section of the overall fracture (at some point x lying between fracture tip and wellbore), to solve (6) and (7) for fracture half height, ℓ, in that section.

We need first to know q(0), the flow rate into the half crack of the vertical section. This may be related to $d\ell/dt$ through the equation of mass conservation. for an elliptical line crack in a uniform stress region S_1 with an equivalent modulus E_q the width is given by

$$w(y) = w(0)\sqrt{1 - \frac{y^2}{\ell^2}}$$

where y is the coordinate along the length of the crack and w(0) is the maximum width, given by $w(0) = 4(1-\nu^2)(\bar{p}_L-S_1)\ \ell/E_q$. The cross-sectional area of the half-crack is then

$$A \equiv \frac{\pi\ \ell\ w(0)}{4} = \frac{\pi(1-\nu^2)\ell^2\ (\bar{p}_L-S_1)}{E_q}$$

Mass conservation applied to the vertical half-crack (in the absence of leakoff) is: inflow rate = rate of change of volume. Applying this to unit length along the payzone axis gives

$$q(0) = \frac{dA}{dt}$$

where q(0) is the flow rate per unit fracture length up into a vertical half crack (i.e., the 1D component of flow in the vertical direction). This gives

$$\frac{E_q\ q(0)}{\pi(1-\nu^2)} = \ell^2\ \frac{\partial}{\partial t}\ (\bar{p}_L - S_1) + 2\ell\ (\bar{p}_L - S_1)\ \left(\frac{d\ell}{dt}\right)_L \qquad (9)$$

This result applies to the equivalent line crack.

Now (9) contains $\partial/\partial t\ (\bar{p}_L-S_1)$ which can be obtained from equation (7). Thus (7) and (9) can be solved simultaneously to obtain q(0) as a function of $(d\ell/dt)_L$, and this relationship will then be used to solve (6) and (7). From (7)

$$\ell^2\ \frac{\partial}{\partial t}\ (\bar{p}_L-S_1) = \left(\frac{d\ell}{dt}\right)_L \left[\frac{H(S_2-S_1)}{\pi\sqrt{1-(H/2\ell)^2}} - \frac{K_c}{2}\sqrt{\frac{\ell}{\pi}}\right]$$

Combining this with (9) gives

$$\frac{E_q\ q(0)}{\pi(1-\nu^2)} = \left(\frac{d\ell}{dt}\right)_L \left[2\ell(\bar{p}_L-S_1) - \frac{K_c}{2}\sqrt{\frac{\ell}{\pi}} + \frac{H(S_2-S_1)}{\pi\sqrt{1-(H/2\ell)^2}}\right]$$

and substituting $(\bar{p}_L - S_1)$ from (7) leads to

$$\left(\frac{d\ell}{dt}\right)_L = \frac{E_q \; q(0)}{\pi(1-\nu^2)f} \tag{11}$$

$$\text{where } f = \frac{3}{2} K_c \sqrt{\frac{\ell}{\pi}} + \frac{4\ell}{\pi} (S_2 - S_1)\cos^{-1}\left(\frac{H}{2\ell}\right) + \frac{H(S_2 - S_1)}{\pi\sqrt{1-(H/2\ell)^2}} \tag{12}$$

This is the desired relation between $q(0)$ and $(d\ell/dt)_L$ needed to solve (6) and (7) which we do next. By substituting (8) and (11) into (6), we find

$$p_L(0) - S_1 = (\bar{p}_L - S_1) + \frac{3\pi}{16} \frac{\mu}{\ell^2} \frac{E_q^{\;2}}{(1-\nu^2)^2} \frac{f(d\ell/dt)_L}{(\bar{p}_L - S_1)^3} \cdot \left[.3634 + \frac{.3907\; \ell(\bar{p}_L - S_1)}{f}\right] \tag{13}$$

with $\bar{p}_L - S_1$ given by (7), f given by (12), and E_q given by (5). This is the equation to be solved for fracture half height, ℓ, in a vertical section, given $p_L(0)$, the pressure at the center of the section. The equation has been derived only for Newtonian flow, and leakoff has been neglected. It also only applies in the asymptotic limit of slow vertical expansion.

This solution for a line crack expanding through a medium containing a stress contrast is basically a relation between the pressure at the center of a vertical section, $p_L(0)$, the half-height of the section, ℓ, and the rate of vertical expansion, $(d\ell/dt)_L$, in that section. The solution represents a first-order correction for the effect of finite fluid flow in the vertical direction. Such flow manifests itself by a difference between $p_L(0)$ and $\bar{p}_L$, which can be seen in equation (13). When $(d\ell/dt)_L$ is zero, $P_L(0) = \bar{P}_L$, which was the basis of our fracture height calculation in previous simulators.[4,5]

This new solution for vertical expansion now needs to be coupled to the 1D fluid flow along the payzone to complete the simulator.

4. The fracture simulator

The formulation of the fracture simulator is similar to those previously described.[4,5] The predominant (1D) flow along the payzone direction is treated by integrating the pressure gradient equation, dp/dx, from the fracture tip at $x = a$, where the pressure is specified, back to the wellbore. The integration is performed by dividing the fracture length into a large number of vertical sections, and using a Runge-Kutta technique. After the pressure, which is taken to be $p_L(0)$, is computed in the next section, the pressure-height relation (13) is satisfied by an iterative procedure to find the half-height, ℓ, in that section. (This is then used in the next section to compute the next pressure and so forth.) This requires a knowledge of $d\ell/dt$, which is obtained by solving for a second fracture whose length is incremented to $a + da$. Since $d\ell/dt$ depends on ℓ and vice versa, a second iteration is required to converge $d\ell/dt$, and thereby solve (13).

5. Results

The theory above is applicable for Newtonian flow when pump rate is constant, and when leakoff is neglected. Furthermore, we have assumed the vertical expansion is not too fast. Table 1 contains pumping and formation parameters that have been used to illustrate the correction for finite flow rates in the vertical direction. The viscosity is taken to be constant with length along the fracture.

Figure 2 shows the fracture height as a function of distance from the wellbore, both with the vertical flow correction, and without, for a fracture half-length of 400 ft. The wellbore half-height is reduced from 174 ft. to 101 ft. when the correction is included. The flow rate in the uncorrected

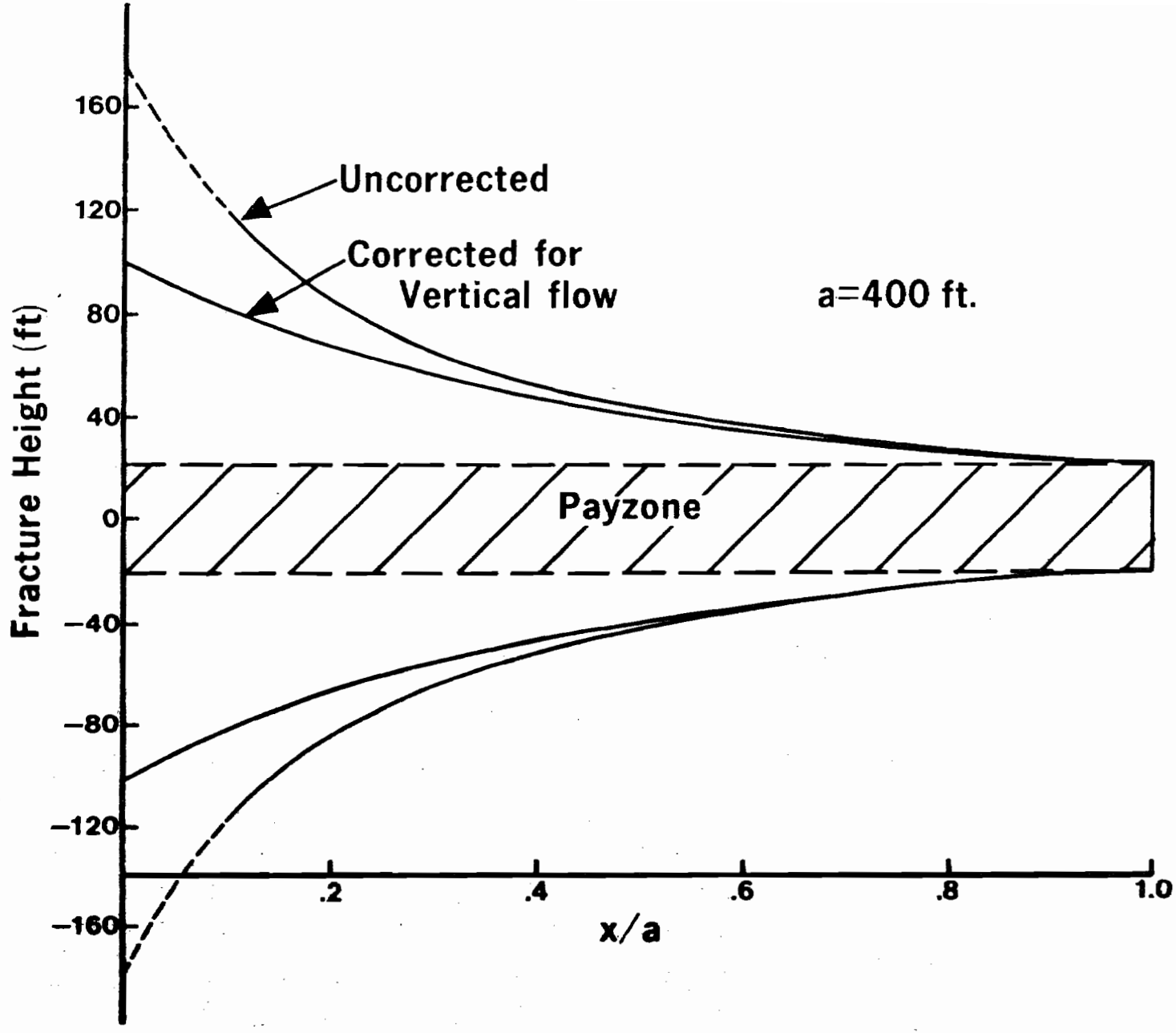

Figure 2. Fracture Height as a Function of Penetration From the Wellbore (With and Without the Correction for Vertical Flow)

case was taken to be the same as the converged flow rate in the corrected case.

Figure 3 gives the pressure profile within the fracture, both with and without the correction for finite flow rate. The difference is very slight. The pressure at the wellbore is found by integrating dp/dx from the tip back to the wellbore, and apparently the reduccd height of the channel (figure 2) makes only a marginal difference to the wellbore pressure in this case (the flow rate profile is the same in both cases).

To illustrate the proper width calculation we employ in the modeling, Figure 4 displays the width profile in a vertical section at the wellbore, for the case which includes the correction for finite vertical flow.

6. Comparison with MIT fracture model

The MIT model[1,2] superimposes on the predominant 1D flow along the payzone an auxiliary 1D flow in the vertical direction. Direct comparison with MIT results for low stress contrasts can be made by reference to their[2] Figures 5a, 5b. For comparison purposes, we have chosen the results corresponding to t = 20 min. Evidently the stress contrast for the high confinement case is 100 psi (by comparison with their Figures 3a-d), and we assume the other pumping and formation parameters are the same. These parameters are given in Table 4.

Comparison is now made between the MIT and adapted ORU simulators (the latter adapted for finite vertical flow), based on the parameters of Table 4. The results are given in Table 5, column 3. Although the times are similar, the ORU height is much greater, evidently because the net pressure is greater.

There are several possible reasons for the discrepancy, and we consider each of these in turn. Firstly, the width equation is different. In each

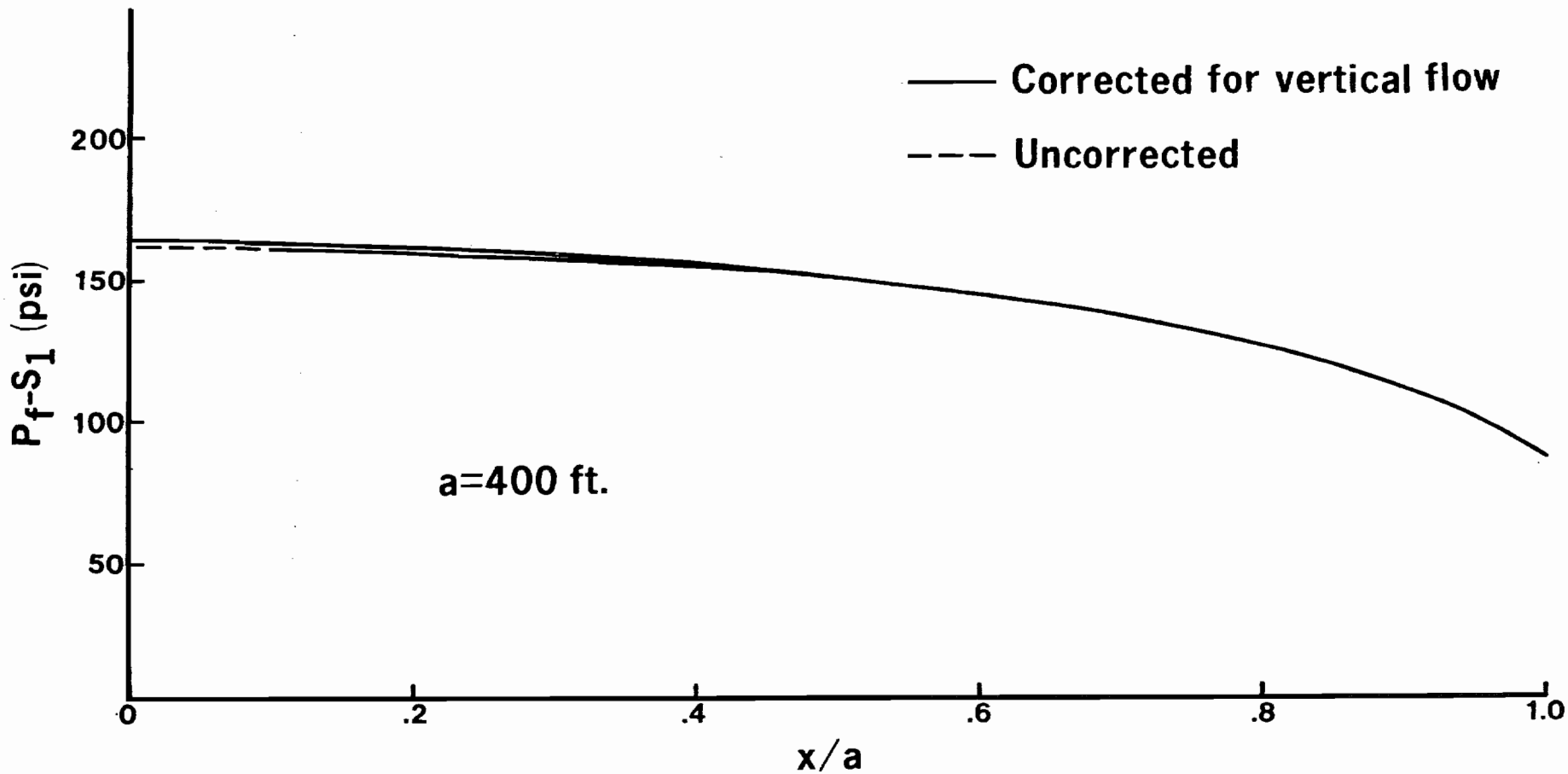

Figure 3. Pressure Profile Within the Fracture (With and Without the Correction for Vertical Flow)

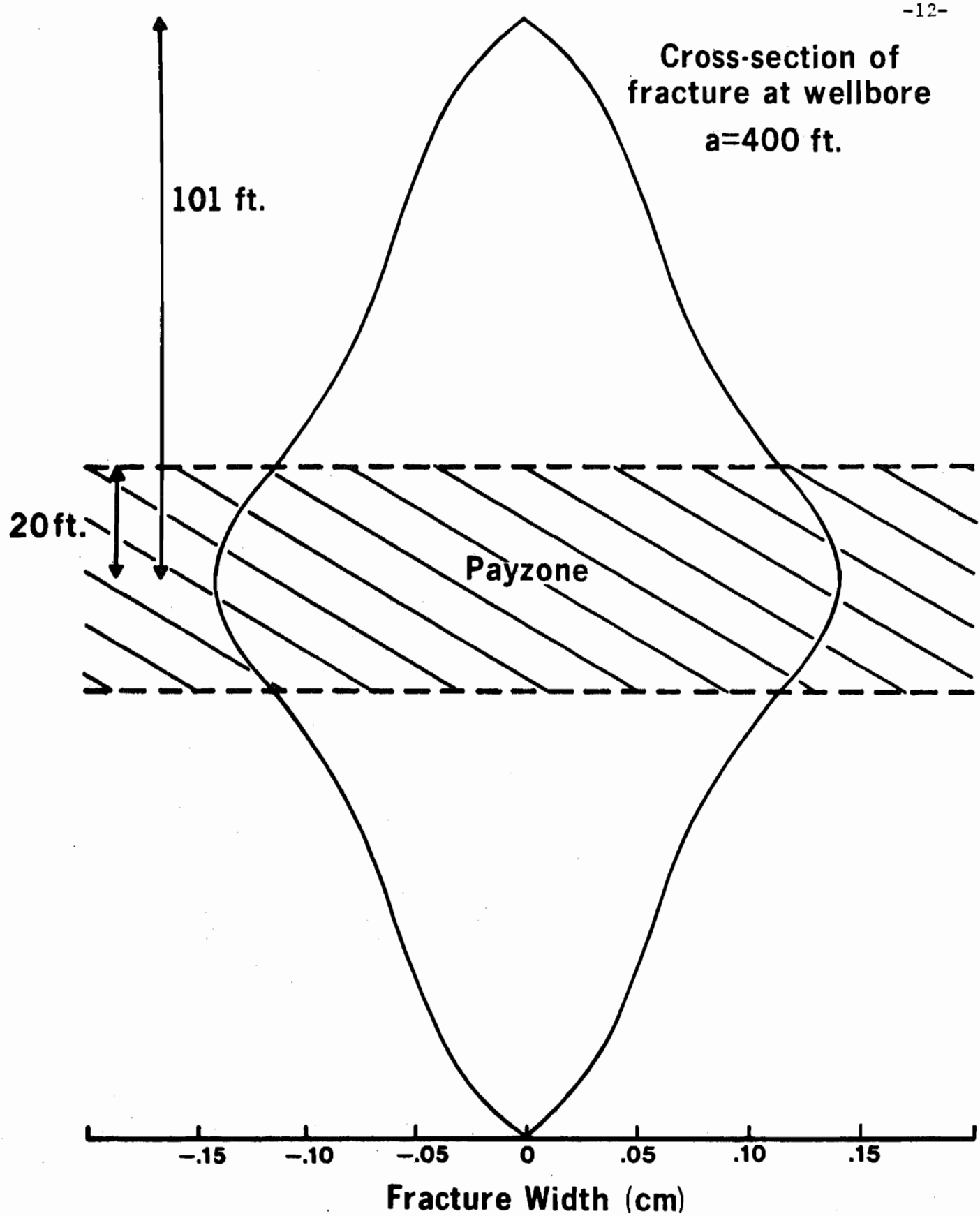

Figure 4. Width Profile for the Vertical Section at the Wellbore

vertical section our widths are derived from a formulation by England and Green[8], which have been shown to agree with independent formulations. Uniform fluid pressure is assumed. A survey has been made of the MIT widths in their Figure 3c[2], which can be compared with England and Green widths calculated from the net pressure and height corresponding to Figure 3c. For S_2-S_1 = 100 psi, their maximum widths in the wellbore section are 1.6 - 2.0 times greater than England and Green values. For 50 psi, the factor is 1.4-1.7. These discrepancies are not due to neglect of fracture perimeter shape, because they are in the wrong direction.

It has recently been established that the MIT widths disregard the increase in minimum in situ stress in the bounding zones.[9] It appears that the larger widths in the MIT model will produce a lower pressure at the wellbore, so this must account for some of the discrepancy in the results of Table 5.

We can compensate for the width discrepancy by artifically multiplying all ORU width equations by a factor of 1.8 (the average discrepancy for S_2-S_1 = 100 psi). When this is done the results, which also appear in Table 5, give better agreement between ORU and MIT models, but the width and the time are still off.

The discrepancy in times in Table 5 is suprising in view of the fact that the ORU fracture appears larger than the MIT fracture (leakoff is not included). To investigate this further, we estimate the fracture volume from the MIT dimensions given in figure 5a, b[2]. If the vertical sections of the t = 20 min fracture are approximated by ellipses, the total frature volume (2 wings) is computed to be 1.63×10^7 cm^3. Since each vertical section is expected to be "pinched in" somewhat by the higher stresses in the bounding zones, the geometric volume should be less than 1.63×10^7 cm^3. Now the

statements and definitions in the text[2] all indicate that $\dot{W}/p$ is the pump rate for just 1 wing, and this is given as 5 bpm (see Table 4). But this pump rate (doubled) for a time of 20 min gives 3.17×10^7 cm^3 for total fracture volume, which is at least a factor of two greater than the geometric volume estimated above. Thus it seems possible there is an internal inconsistency in these results, by a factor of around 2. Several other geometrical volumes can be estimated from the fracture dimensions given in Figure 3 of Clear et al.[2], by using the formula for the volume of an ellipsoid, $2\pi abw_o/3$. More inconsistencies similar to the above appear. It has recently been learned[9] that $\dot{W}/p$ is actually the total pump rate (both wings). Both the MIT corrections we have discussed (width and pump rate) will alter the comparison between MIT and ORU models in Table 5. Note that even though we compensated for the maximum width, the width profiles in a vertical section are still different. We therefore ran the model with q_o = 2.5 bpm and the results are given in the last column of table 5 and illustrated in figures 5 and 6. The agreement is surprisingly good in view of the fact that the MIT model assumes (1) $K_c = 0$ and (2) apparently an elliptical shape for each vertical cross section, whereas we use the shape computed for the stress contrast (figure 6).

That there is still some disagreement is not too surprising in view of the different relationships in the two models between the pressure in the center of a vertical section and the height of the section. In the MIT model, the authors have sought to calculate this relationship properly, and the modeling is very complicated. As described earlier, our approximation has quite a different basis. Furthermore, the MIT model omits fracture toughness altogether, whereas in the ORU model K_c has an integral role to play, and cannot be set to zero. Reducing K_c in the ORU model actually increases the height discrepancy between ORU and MIT in Table 5. On the other hand, an

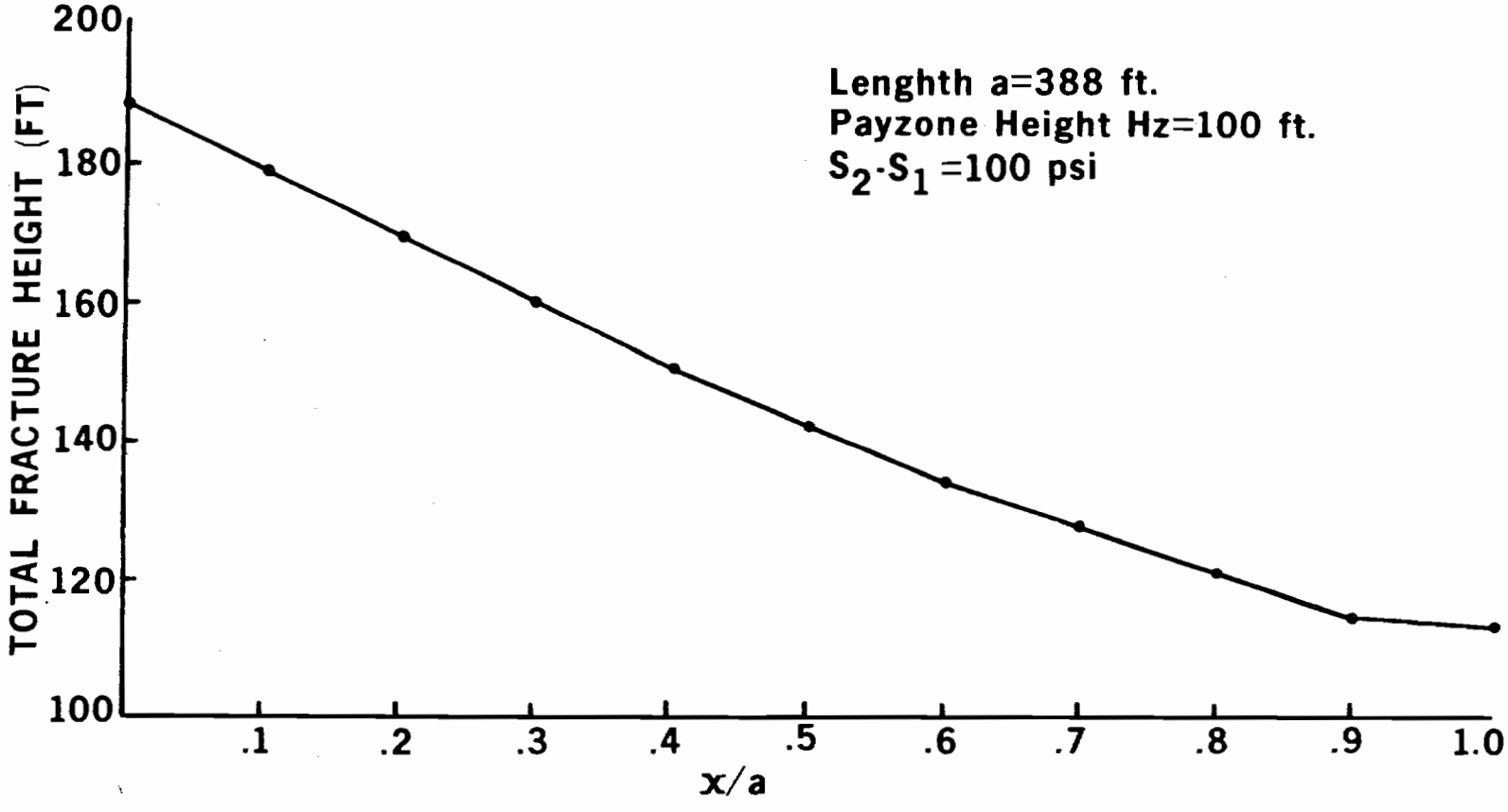

Figure 5. Fracture Height for the MIT Parameters (Table 4) With q = 2.5 bpm and Width Modified by a Factor of 1.8

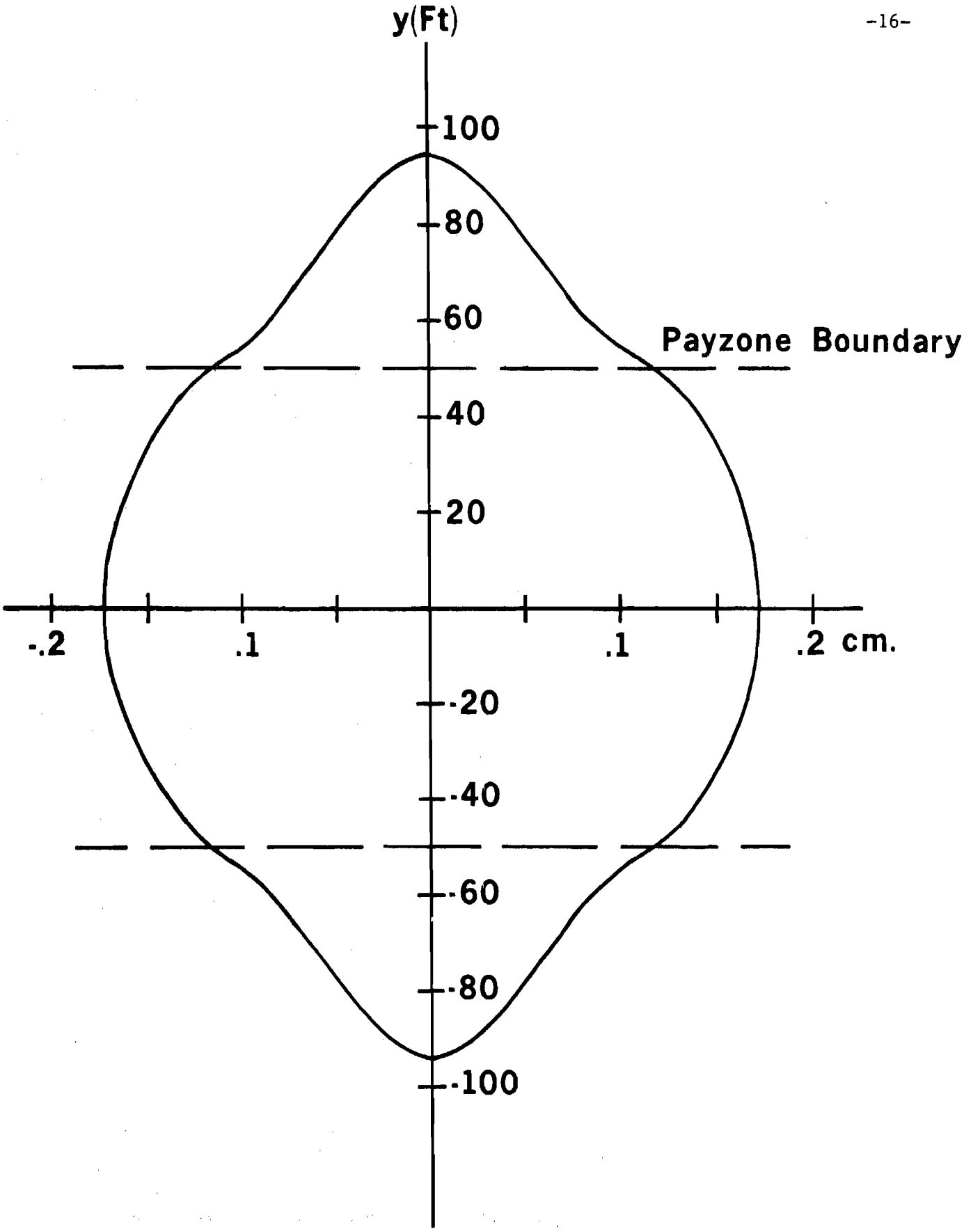

Figure 6. Width Profile for the ORU Adapted Case With q = 2.5 bpm

increase in K_c improves the height agreement but worsens the pressure and width discrepancy in Table 5. A persuasive argument that K_c cannot be neglected, at least in the case of a penny crack, has been given by Abe et al.[6]

7. Conclusion

The fracture model described herein calculates fracture height as a function of distance from the wellbore for the 3-layer case in which a payzone is bounded by zones of higher (but equal) minimum stresses. Variations in elastic modulus between zones are not considered, nor are variations in pump rate. The modeling discussed in this paper is appropriate under conditions that produce a highly elongated fracture, with length/height ratio greater than about 3.5.

The pseudo-3D model described herein assumes a highly elongated fracture inorder to satisfy the approximations made in determining the fracture width based on a series of 2D cracks in vertical sections. However, the model also accounts for a subsidiary 1D flow in the vertical direction, and therefore, at least from a fluid point of view, it should be more appropriate.

The effect of the vertical flow on the pressure profile in the fracture is minimal. However, the reduction in height can be substantial. In one case (see figures 2 and 3) the wellbore pressure with vertical flow is < 2% different from the case without vertical flow, but the wellbore heights differ by 70%.

The Fortran code for the fracture model is included in Appendix B, plus user instructions. A caution is inorder: converging the iterations on the flow rate, q(x), and on the fracture height (and height growth, dℓ/dt) are

rather sensitive. We have had difficulty sometimes in achieving an iterated result.

A comparison has been made with the MIT model, which also includes a vertical flow component. At a lower stress contrast, 100 psi, the ORU model predicts in one case a wellbore height which is greater than that of the MIT model by a factor of 2.5. One reason for the discrepancy is a disagreement in the width, with MIT widths greater by a factor of 1.8, due to their neglect of higher in situ stresses in the bounding[9] zones. When this adjustment is made in the ORU model, the agreement is better, but still not good. The lack of time agreement reflects partly an incorrect flow rate.[9] When this is corrected the agreement is quite good.

Nomenclature

a = fracture half-length

C = leakoff coefficient

E = Young's modulus

h = total fracture height

H = payzone height

K_i = stress intensity factor

K_c = fracture toughness

k = amplitude parameter for power law fluid

μ = fluid viscosity

ν = Poisson's ratio

n = exponent parameter for power law fluid

p_f = fluid presure

$p = p_f - S_1$ = net pressure

$q(x)$ = flow rate in x direction (one-wing)

S_1 = minimum in-situ stress in payzone

S_2 = minimum in-situ stress in bounding layers

t = time

$\tau(x)$ = time at which fracture reaches point x along the payzone

V_s = spurt loss coefficient

$w(y)$ = width of 2D crack in vetical section

w_o = maximum width of crack in vertical section

x = horizontal distance along payzone

y = vertical distance along 2D line crack

Acknowledgement

This work was partially supported by the U.S. Department of Energy and by Oral Roberts University. Grateful thanks are expressed to Lian Soh for excellent analysis and programming support. Special thanks to Dale Williams for his constant encouragement in this research.

References

(1) Settari, A., and M. P. Cleary, "Development and testing of a pseudo-three-dimensional model of hydraulic fracture geometry (P3DH)," 6th Symp. Reservoir Simulation of Soc. Pet. Engrs. of AIME, New Orleans, Feb. 1982.

(2) Cleary, M. P., R. G. Keck, and M. E. Mear, "Microcomputer models for the design of hydraulic fractures," SPE/DOE 11628, Symp. Low Permeability Gas Reservoirs (Soc. Pet. Engrs. of AIME), Denver, March 1983.

(3) Nolte, K. G., "Fracture design considerations based on pressure analysis," SPE 10911, Soc. Pet. Engrs. Coton Valley Symp., Tyler, Texas, 1982.

(4) Palmer, I. D., and H. B. Carroll, Jr., "Three-dimensional hydraulic fracture propagation in the presence of stress variations," Soc. Petr. Eng. J., 870-878, Dec. 1983.

(5) Palmer, I. D., and H. B. Caroll, Jr., "Numerical solution for height of elongated hydraulic fracture," SPE/DOE 11627, Symp. Low Permeability Gas Reservoirs (Soc. Petr. Engrs. of AIME), Denver, March 1983.

(6) Abe, H., T. Mura, and L. M. Keer, "Growth of a penny-shaped crack in hydraulic fracturing of rocks," J. Geophys. Res., 81, 29, 5335, 1976.

(7) Palmer, I. D., and H. R. Craig, "Modeling of asymmetric vertical growth in elongated hydraulic fractures and application to first MWX stimulation," SPE/DOE/GRI 12879, Procs. Unconventional Gas Recovery Symp., Pittsburgh, PA, May 1984.

(8) England, A. H., and A. E. Green, "Some two-dimensional punch and crack problems in classical elasticity," Proc. Cambridge Phil. Soc. 59, 489, 1963.

(9) Cleary, M. P., private communication, 1984.

(10) Palmer, I. D., and H. B. Carroll, Jr., "Numerical model of massive hydraulic fracture," Second Annual Report, DOE/BC/10215-27, 1983.

(11) Palmer, I. D., H. R. Craig, and C. T. Luiskutty, "Numerical model of massive hydraulic fracture," ORU/DOE Final Report (1985).

Table 1

Pumping and formation parameters used to illustrate the correction for finite flow rates in the vertical direction

$H = 40$ ft

$E = 3.34 \times 10^6$ psi

$\nu = .15$

$S_2 - S_1 = 150$ psi

$K_c = 1818$ lb in$^{-3/2}$

$C = 0$

$q(0) = 5$ bpm (1 wing)

$\mu(0) = 2$ cp

$a = 400$ ft (1 wing)

Table 2

Parameter values at the onset of the asymptotic limit for a penny crack[6]

Plot	$\frac{Cq_o}{S^4 R_o^2}$	$\frac{\pi}{\sqrt{2R_o}} \frac{K_c}{S}$	$\frac{R}{R_o}$
A	2.65×10^3	1.12	7×10^2
B	9.49×10^2	1.12	4×10^2
C	2.47×10^2	1.12	1.3×10^2
D	7.74×10	1.12	60
E	7.96×10^3	2.24	3.5×10^2
F	1.57×10	.447	1.3×10^2

Table 3

Values of critical radius as a function of fluid viscosity

μ (cp)	R_c (ft)
100	78
300	163
1000	364

Table 4

Pumping and formation parameters used to compare the ORU and MIT simulators for lower stress contrasts

$H = 100$ ft

$\bar{E} = 1.13 \times 10^6$ psi

$E = 4(1-\nu^2)\,\bar{E}$

$\nu = .1$

$S_2 - S_1 = 100$ psi

$C = 0$

$q(0) = 5$ bpm (1 wing)

$\mu(0) = 10$ cp

$a = 388$ ft

$h(a) = 113.6$ ft

$K_c = 1818$ lb/in$^{3/2}$ in ORU

Table 5

Comparison between ORU and MIT models, for parameters in Table 3

(lower stress contrasts)

	MIT	ORU adapted for finite vertical flow	ORU adapted plus all widths x 1.8	ORU adapted plus all widths x 1.8 with q(0)=2.5 bpm
wellbore half-height	105 ft	268 ft	120 ft	95 ft
wellbore pressure (above S_1)	86 psi	121 psi	104 psi	97 psi
wellbore width (maximum)	0.227 cm	0.440 cm	0.420 cm	0.344 cm
pumping time	20 min	22.3 min	15.5 min	24.1 min

APPENDIX A

The asymptotic limit of slow vertical expansion.

In our new simulator, which includes the effect of finite vertical flow, we assumed an asymptotic limit based on the theory and results of Abe et al.[6] This was discussed in the text, equations (1) → (4). In essence, for slow enough expansion rate, the average fluid pressure in the fracture is related to fracture toughness and fracture size by the standard equation for constant pressure. Furthermore, the near-tip region of fluid inaccessibility becomes so small that the pressure integration from crack tip to the crack center is much simplified.

The combination of parameters which corresponds to this asymptotic limit can be obtained from the departure from a straight line of the plot W_D in Figure 4 of Abe et al. These "departure points" are summarized in Table 2. The data are approximately fit by a relationship

$$\frac{Cq_o}{S^4 R_o^2} \left(\frac{R_o}{R}\right)^{3/2} \simeq 0.11 \left(\frac{\pi}{\sqrt{2R_o}} \frac{K_c}{S}\right)^3 \qquad (1)$$

in which the variables are defined in Abe et al.[6]

This is rearranged to give

$$\frac{R_c^{3/2}}{\mu} \simeq 0.1 \left(\frac{E}{1-\nu^2}\right)^3 \frac{Q_o}{S\, K_c^3} \qquad (2)$$

where R_c is the critical radius; beyond R_c the asymptotic limit is a good approximation for a penny crack. μ is fluid viscosity, E is Young's modulus, ν is Poisson's ratio, Q_o is inlet pump rate, S is minimum in-situ stress, and K_c is fracture toughness.

Typical MWX parameters[7] are $E = 4.8 \times 10^6$ psi, $\nu = .22$, $Q_o = 10$ bpm, $S = 6000$ psi[5], $K_c = 1200$ psi in$^{1/2}$. These give $R_c^{3/2}/\mu = 1.17 \times 10^5$ in cgs units. Table 3 shows that for typical fluid viscosities (100-300 cp), a penny fracture whose radius exceeds one or two hundred feet should be well approximated by the asymptotic limit.

It is difficult to translate this to the case of a vertical fracture which is elongated in the payzone direction. If one thinks of squashing a penny shape into an elongated shape, the reduced height of a vertical section (at the wellbore, say) would mean the asymptotic behavior is less likely. However, the corresponding reduced vertical component of flow in this section would make the asymptotic behavior more likely. It is actually a moot point, because we have not obtained a solution for vertical flow which is beyond the asymptotic limit. This solution is very difficult, and at this stage we think unwarranted. In view of this, we have to admit that our fracture height solution is still an upper limit, but nevertheless an improvement over the modeling without a correction for finite vertical flow.

APPENDIX B

Fortran code and user instructions for vertically symmetric simulator with 1D vertical fluid flow in addition to 1D horizontal fluid flow.

1. Program description

FLAMLEAK6B, a fracture simulator in which fracture height is self-determined, is a modification of FL5, published previously.[10] It is appropriate if the vertical fracture is highly elongated (length/height ratio ($\gtrsim$ 4), and if the minimum stresses in the two bounding zones are higher than in the payzone, and equal. Vertical lithostatic or hydrostatic gradients are not included. In addition to the 1D horizontal fluid flow, a 1D vertical fluid flow is included. Variations in elastic modulus from one zone to another are not included, nor are time variations in pumping rate or viscosity at the wellbore.

The program solves simultaneously the four equations for fracture expansion in reference 5 subject to the boundary condition that the fracture height at the tip of the long axis is approximately equal to the payzone height.

In this way it solves for 2 closely-spaced fracture of lengths a and a + da.

2. Program listing

Main program	Subroutines	Functions
FLAMLEAK6B	WIDTH	FUN
	RK2	DFX
	VLTQ	FFX
	QTEST	ASIN
	HCALC	ACOS
	DLDTCAL	PFUN

3. Operating mode

Interactive

4. Language

Fortran IV

5. Program developed on

Data General Eclipse minicomputer

6. Total CP time

About 2 min

7. Effect of step size on Runge-Ketta integration

The resolution in all 10 segments along the x-axis is determined by NDIV (normally 110). The resolution in the tip segment is enhanced by the factor MFAC (e.g., MFAC = 5 means steps will be 5 times closer together). If the stepts are not close enough in the first segment, the pressures along the fracture, given by PXA values, will be too similar and clearly erroneous. In this case, reduce the step size by increasing MFAC by a factor of 2. As a final check on the results, double NDIV, which reduces the step spacing along the entire x-axis. The results for wellbore half-height, b, and wellbore pressure (maximum PXA value) should not change significantly.

8. Too rapid height growth near the wellbore

For certain combinations of input parameters the program may abort because height grows too fast as the integration proceeds from the tip at x=a back to the wellbore. When this happens, heights and pressures will be printed out from the tip to some point between tip and wellbore, followed by a RUNTIME error. If the height is growing rapidly, this is almost certainly the reason for the error message. To check it, double the stress contrast S_2-S_1 and a successful simulation should result.

Parameters which can give this error include:

(i) too small S_2-S_1 (ii) too small H (iii) too high E

(iv) too large μ (iv) too large Q

9. Sample input data

The input data are given in the interactive mode. For MIT comparisons (tables 4 and 5), the input values are given below and at the head of the output file.

Sample input values:

2 - ITQ:	Selects the desired calculation; time and flow rate
1 - IWRITE:	Selects output
2E8 - KCB:	Fracture toughness constant (dynes/cm$^{3/2}$)
100 - HZ:	Payzone height (ft)
2.5 - Q:	One wing flow rate (BPM)
10 - RMU:	Viscosity (cp)
4.475E6 - EMOD:	Young's modulus (psi)
0.1 - RNU:	Poisson's ratio
1.8 - WIDCON:	Factor by which England and Green width is multipled (see text)
100 - S2MS1:	Stress contrast (psi)
388 - A:	Frac half length (ft)
10 - NUMWID:	No. of width calculations in each vertical section
1.0 - Q1:	Q(x) at x/a = 0.0
0.26 - Q2:	Q(x) at x/a = XA2
0.064 - Q3:	Q(x) at x/a = 1.0 Q_1, Q_2, Q_3 used to calculate the initial input flow rate variation
0.4 - XA2:	x/a corresponding to Q2.
1 -ICOMP:	Selects option to recompute and list all K_I's for checking purpose
5.0 - QPER:	Percentage convergence desired for Q(x)
0.01 - DIFCE:	Percentage convergence desired for half/height

1.0 - DLTPER:	Percentage convergence desired for dl/dt.
110 - NDIV:	No. of elements into which the frac length is divided
0.999 - XAZ:	x/a position where numerical integration begins
11 - N:	No. of segments the frac wing is divided into (output parameters available for each segment)
1 - IDLDT:	To input dl/dt values

0	DLDT(I):	dl/dt value for each segment; dl/dt = 0 for segment 1, avoids too large dl/dt which could give Runtime error in PFUN
1		
1		
1		
1		
1		
1		
1		
1		

1 - IDLDTZ:	Sets all DLDT(1) to zero
1 - NRES:	Flag to decide whether the resolution should be improved in the segment closest to the tip
5 - MFAC:	Multiplication factor used to increase resolution of tip segment
2 - IDEBUG:	Debug not desired
1.136 - HSTART	Frac height/payzone height at frac tip ($\gtrsim$ 1.0). (This is an arbitrary boundary condition, should be close to 1, but not too close because numerical convergence problems can occur.)

2 - IHST:	Signifies HSTART need not be changed
0.6 - ELR:	Initially guessed height value for iteration which solves for $K_i = K_c$, used in HCALC
1.0 - TSLOPE:	Slope of log a vs log t graph
1.0 - TCETP:	Intercept of log a - log t graph at t = 1
0 - LEAKC:	Leakoff coefficient (cgs)
0 - SP:	Spurt loss coefficient (cgs)
30 - IMP:	Indicate that no improvement intime discrepancy is desired
1 - NQTEST:	Compares input and output Q(x) in QTEST
20 - NUM:	Maximum number of iteration for Q(x)

10. Sample output data file

Essentially there are two fractures: Frac 1 with half-length a and Frac 2 with length 1.10a (e.g., 388 ft and 426.8 ft). For each of these fractures are given net pressure above payzone stress in psi (see 11 or 12 PXA values), half-height at the wellbore in ft(b), fracture full height in units of payzone height (see 11 or 12 H/HZ values), and width profile in each vertical section in cm (see 11 WIDTH values).

After each pair of fractures, we calculate flow rate profile (see OUTPUT QX), and this can be compared with the flow rate assumed to solve for the fractures (see INPUT QX). Fracture volume (1 wing, cm^3) for the first fracture is given by V/2 PUMPED FRAC1 (1 wing, cm^3 barrels and gallons). The time corresponding to the first frac is given by T1CHECK (sec., min.).

The program, as given here, does not iterate on length expansion because leakoff is not included (LEAKC = SP = 0). Therefore, dummy parameters 1, 1 are input for TSLOPE, TCEPT, the parameters which define

a(t), the length expansion rate. Because of this T(1) and T(2), the times computed from a(t) are meaningless and percentage differences between T1CHECK and T(1), and between T2CHECK and T(2) have no significance. The execution is manually terminated after the flow rate q(x) has converged to within a required percentage (e.g., 5%) or height growth rates DLDT (I) have converged to within 1% (in this example).

Despite the fact that leakoff is zero in this examle, the capability exists for its inclusion in the manner described previously[11].

```
X FLAMLEAK6B
DO YOU WISH TO CALCULATE 1=TIME, 2=TIME & Q, 3=NEITHER ?
2
INPUT:  IWRITE, KCB(CGS)(NOTE: KCB<>0 IN THIS PROG)
  HZ(ZONE WIDTH IN FT),  Q(BPM), RMU(CP)
1,2E8,100,2.5,10
IWRITE=    1      KCB=    0.200D  9 HZ=       100.0 Q=         2.5 MU=        10.0
 INPUT:  EMOD(YOUNG'S MODULUS, EG. 1.5E6 PSI), RNU(POISSON'S RATIO,EG. .1),
WIDCON(USUALLY 1-ENGLAND & GREEN)
4.475E6,.1,1.8
 YOUNG'S MODULUS=     0.447D  7 POISS.RATIO=        0.10 WICON= 1.80
INPUT: S2-S1(PSI),  A(FT)
100,388
S2-S1=       100.0    A=        388.0
ENTER NO. OF WIDTH CALCNS. IN EACH VERTICAL SECTION  USUALLY 10
10
PLEASE INPUT Q1,Q2,Q3,= VALUES OF QX AT XA=0,XA2,1.0
GOOD STARTING VALUES ARE 1.0,0.2, 0.1
1.0,.26,.064
PLEASE INPUT XA2 = XA VALUE CORRESPG. TO Q2
.4
Q1= 1.000Q2= 0.260Q3= 0.064
SLOPE =  0.327  0.327  0.327  0.327  0.327  0.327  0.327  1.850  1.850  1.850  1
.850
CEPT =  0.391  0.391  0.391  0.391  0.391  0.391  0.391  1.000  1.000  1.000  1.
000
 DO YOU WANT TO RE COMPUTE AND LIST ALL KI'S IN RANGE   0-1 FOR CHECKING    PURP
OSES?
 IF YES, HIT 1
1
INPUT PERCENT CONVERGENCE DESIRED FOR Q(X) USUALLY 20
5
Q(X) WILL CONVERGE TO          5. PERCENT
INPUT % CONVERGENCE DESIRED FOR HALF-HEIGHT ITERATION USUALLY 1
.01
HALF-HEIGHT CALCN WILL CONVERGE TO        0.010 %
INPUT % CONVERGENCE DESIRED FOR DL/DT ITERATION.
1
DL/DT CALCN WILL CONVERGE TO        1.000 %


INPUT: NDIV (MUST BE MULTIPLE OF N*(N-1) )
XAZ(EG. .999), N  (=11, NOT COMPLETELY PROGRAMMED FOR N=21 YET)
110,.999,11
NDIV=   110 XAZ=   0.999000 N=  11

HIT 1 IF YOU WANT TO INPUT DLDT VALUES
1
INPUT VALUES FOR DLDT(I), FOR I=1, 11
0,1,1,1,1,1,1,1,1,1,1

HIT 1 IF YOU WANT TO SET ALL DLDT(1) TO 0
1
 INPUT:  NRES,MFAC   (FOR USE IN RK2)
1,5
NRES =     1   MFAC =     5

HIT 1 IF YOU WANT TO PRINT ALL CALCNS FOR DEBUG
2
INPUT:  HSTART/HZ (CAN BE > OR < 1)
1.136
 HSTART/HZ =      1.1360
INITIAL PRESSURE AT X=A IS        70.8 PSI
 IF YOU WANT TO CHANGE HSTART,  HIT 1
2

INPUT ELR (USUALLY 0.6)
.6
STEP=     0.182D -2 XA=     0.999D  0 T1,T2,T3,T4=      0.681D -1    0.684D -1    0.6
84D -1    0.687D -1 Y=        0.48823013D  7
STEP=     0.182D -2 XA=     0.997D  0 T1,T2,T3,T4=      0.693D -1    0.696D -1    0.6
96D -1    0.699D -1 Y=        0.48831241D  7
STEP=     0.182D -2 XA=     0.995D  0 T1,T2,T3,T4=      0.704D -1    0.707D -1    0.7
07D -1    0.710D -1 Y=        0.48839604D  7
STEP=     0.182D -2 XA=     0.994D  0 T1,T2,T3,T4=      0.715D -1    0.719D -1    0.7
19D -1    0.722D -1 Y=        0.48848103D  7
STEP=     0.182D -2 XA=     0.992D  0 T1,T2,T3,T4=      0.727D -1    0.730D -1    0.7
30D -1    0.734D -1 Y=        0.48856741D  7
STEP=     0.182D -2 XA=     0.990D  0 T1,T2,T3,T4=      0.739D -1    0.742D -1    0.7
42D -1    0.745D -1 Y=        0.48865518D  7
C1,C2,C4 =      0.642D-11    0.439D  7  -0.300D 38STEP=    0.182D -2
```

```
PHI=  0.5809D  0 DELTAY=  0.1000D  0 H=  0.3251D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.184D  0 0.183D  0 0.180D  0 0.175D  0 0.168D  0 0.158D  0

0.146D  0 0.129D  0 0.107D  0 0.745D -1 0.000D  0
 I =  2 , AREA(I) =     459.0799
PHI=  0.5772D  0 DELTAY=  0.1000D  0 H=  0.3315D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.191D  0 0.190D  0 0.187D  0 0.182D  0 0.174D  0 0.164D  0

0.151D  0 0.133D  0 0.110D  0 0.742D -1 0.000D  0
 I =  3 , AREA(I) =     483.8557
PHI=  0.5717D  0 DELTAY=  0.1000D  0 H=  0.3394D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.199D  0 0.198D  0 0.194D  0 0.189D  0 0.181D  0 0.170D  0

0.156D  0 0.138D  0 0.113D  0 0.704D -1 0.000D  0
 I =  4 , AREA(I) =     512.1235
PHI=  0.5660D  0 DELTAY=  0.1000D  0 H=  0.3481D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.207D  0 0.206D  0 0.203D  0 0.197D  0 0.188D  0 0.177D  0

0.162D  0 0.142D  0 0.115D  0 0.672D -1 0.000D  0
 I =  5 , AREA(I) =     543.0619
PHI=  0.5600D  0 DELTAY=  0.1000D  0 H=  0.3575D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.216D  0 0.215D  0 0.211D  0 0.205D  0 0.196D  0 0.184D  0

0.168D  0 0.146D  0 0.116D  0 0.659D -1 0.000D  0
 I =  6 , AREA(I) =     577.0207
PHI=  0.5533D  0 DELTAY=  0.1000D  0 H=  0.3678D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.225D  0 0.224D  0 0.220D  0 0.213D  0 0.204D  0 0.191D  0

0.173D  0 0.150D  0 0.116D  0 0.652D -1 0.000D  0
 I =  7 , AREA(I) =     613.3836
PHI=  0.5434D  0 DELTAY=  0.1000D  0 H=  0.3815D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.236D  0 0.235D  0 0.231D  0 0.224D  0 0.213D  0 0.199D  0

0.180D  0 0.154D  0 0.111D  0 0.651D -1 0.000D  0
 I =  8 , AREA(I) =     660.3017
PHI=  0.5302D  0 DELTAY=  0.1000D  0 H=  0.4016D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.252D  0 0.250D  0 0.246D  0 0.238D  0 0.226D  0 0.210D  0

0.188D  0 0.157D  0 0.106D  0 0.654D -1 0.000D  0
 I =  9 , AREA(I) =     728.0583
PHI=  0.5143D  0 DELTAY=  0.1000D  0 H=  0.4273D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.270D  0 0.268D  0 0.263D  0 0.254D  0 0.240D  0 0.222D  0

0.196D  0 0.156D  0 0.105D  0 0.665D -1 0.000D  0
 I = 10 , AREA(I) =     814.7030
PHI=  0.4949D  0 DELTAY=  0.1000D  0 H=  0.4584D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.290D  0 0.288D  0 0.282D  0 0.271D  0 0.256D  0 0.233D  0

0.202D  0 0.147D  0 0.106D  0 0.681D -1 0.000D  0
 I = 11 , AREA(I) =     915.8950
 PXA(I):   0.488D  7   0.495D  7   0.508D  7   0.521D  7   0.534D  7   0.547D  7
 PXA(I):   0.560D  7   0.576D  7   0.595D  7   0.616D  7   0.636D  7

RXA(I):   0.999D  0   0.899D  0   0.799D  0   0.699D  0   0.599D  0   0.499D  0

RXA(I):   0.399D  0   0.299D  0   0.199D  0   0.990D -1  -0.100D -2
 ONE OF THE RXA'S MUST BE REAL CLOSE TO 0.    IS IT ????????
A=      388.0   B=        75.2
PXA: (IN PSI)
    70.81      71.85      73.63      75.51      77.43      79.35

    81.24      83.54      86.38      89.35      92.25
 ITERATION SUCCESSFUL ON KI
 Q=        2.5 RMU=        10.0H=        0.0 B=        75.2A =       388.0
 SHAPE OF FRAC H/HZ FROM XA=1 TO 0
     1.1361       1.0667       1.0875       1.1135       1.1419       1.1730

     1.2067       1.2516       1.3174       1.4020       1.5041
VOLUME(X) AS X=A TO 0 : (CGS)

.2715D  60.5576D  60.5889D  60.6239D  60.6623D  60.7039D  60.7531D  6

.8210D  60.9123D  60.1023D  70.0000D  0
 INPUT:  NRES,MFAC   (FOR USE IN RK2)
1,5
NRES =     1    MFAC =     5
```

```
HIT 1 IF YOU WANT TO PRINT ALL CALCNS FOR DEBUG
2
INPUT:  HSTART/HZ (CAN BE > OR < 1)
1.136
 HSTART/HZ =      1.1360
INITIAL PRESSURE AT X=A IS        70.8 PSI
 IF YOU WANT TO CHANGE HSTART,  HIT 1
2

INPUT ELR (USUALLY 0.6)
.6
STEP=     0.182D -2 XA=     0.999D  0 T1,T2,T3,T4=      0.681D -1    0.684D -1    0.6
84D -1    0.687D -1 Y=        0.48823822D  7
STEP=     0.182D -2 XA=     0.997D  0 T1,T2,T3,T4=      0.687D -1    0.690D -1    0.6
90D -1    0.693D -1 Y=        0.48832801D  7
STEP=     0.182D -2 XA=     0.995D  0 T1,T2,T3,T4=      0.693D -1    0.696D -1    0.6
96D -1    0.699D -1 Y=        0.48841854D  7
STEP=     0.182D -2 XA=     0.994D  0 T1,T2,T3,T4=      0.698D -1    0.701D -1    0.7
01D -1    0.705D -1 Y=        0.48850979D  7
STEP=     0.182D -2 XA=     0.992D  0 T1,T2,T3,T4=      0.704D -1    0.707D -1    0.7
07D -1    0.710D -1 Y=        0.48860177D  7
STEP=     0.182D -2 XA=     0.990D  0 T1,T2,T3,T4=      0.710D -1    0.713D -1    0.7
13D -1    0.716D -1 Y=        0.48869448D  7
C1,C2,C4 =      0.642D-11    0.439D  7  -0.300D 38STEP=    0.182D -2

PHI=  0.5636D  0 DELTAY=  0.1000D  0 H=  0.3490D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.191D  0 0.190D  0 0.187D  0 0.182D  0 0.174D  0 0.163D  0

0.149D  0 0.131D  0 0.105D  0 0.597D -1 0.000D  0
 I =  2 , AREA(I) =     500.8106
PHI=  0.5770D  0 DELTAY=  0.1000D  0 H=  0.3315D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.189D  0 0.188D  0 0.185D  0 0.180D  0 0.172D  0 0.162D  0

0.149D  0 0.132D  0 0.109D  0 0.734D -1 0.000D  0
 I =  3 , AREA(I) =     478.9472
PHI=  0.5740D  0 DELTAY=  0.1000D  0 H=  0.3364D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.196D  0 0.195D  0 0.192D  0 0.186D  0 0.178D  0 0.168D  0

0.154D  0 0.136D  0 0.112D  0 0.727D -1 0.000D  0
 I =  4 , AREA(I) =     501.7769
PHI=  0.5681D  0 DELTAY=  0.1000D  0 H=  0.3448D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.204D  0 0.203D  0 0.200D  0 0.194D  0 0.186D  0 0.175D  0

0.160D  0 0.141D  0 0.114D  0 0.680D -1 0.000D  0
 I =  5 , AREA(I) =     531.5932
PHI=  0.5623D  0 DELTAY=  0.1000D  0 H=  0.3539D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.213D  0 0.212D  0 0.208D  0 0.202D  0 0.193D  0 0.181D  0

0.166D  0 0.145D  0 0.116D  0 0.663D -1 0.000D  0
 I =  6 , AREA(I) =     564.3235
PHI=  0.5560D  0 DELTAY=  0.1000D  0 H=  0.3637D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.221D  0 0.220D  0 0.216D  0 0.210D  0 0.201D  0 0.188D  0

0.171D  0 0.149D  0 0.116D  0 0.654D -1 0.000D  0
 I =  7 , AREA(I) =     599.2255
PHI=  0.5487D  0 DELTAY=  0.1000D  0 H=  0.3746D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.231D  0 0.229D  0 0.225D  0 0.219D  0 0.209D  0 0.195D  0

0.177D  0 0.152D  0 0.115D  0 0.651D -1 0.000D  0
 I =  8 , AREA(I) =     637.0806
PHI=  0.5373D  0 DELTAY=  0.1000D  0 H=  0.3902D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.243D  0 0.242D  0 0.238D  0 0.230D  0 0.219D  0 0.204D  0

0.184D  0 0.156D  0 0.108D  0 0.652D -1 0.000D  0
 I =  9 , AREA(I) =     689.4978
PHI=  0.5242D  0 DELTAY=  0.1000D  0 H=  0.4114D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.259D  0 0.258D  0 0.253D  0 0.244D  0 0.232D  0 0.215D  0

0.191D  0 0.158D  0 0.105D  0 0.658D -1 0.000D  0
 I = 10 , AREA(I) =     761.5185
PHI=  0.5070D  0 DELTAY=  0.1000D  0 H=  0.4376D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.277D  0 0.275D  0 0.270D  0 0.260D  0 0.246D  0 0.226D  0

0.198D  0 0.152D  0 0.105D  0 0.670D -1 0.000D  0
 I = 11 , AREA(I) =     847.6884
PHI=  0.4895D  0 DELTAY=  0.1000D  0 H=  0.4688D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.296D  0 0.294D  0 0.288D  0 0.277D  0 0.260D  0 0.237D  0
```

```
0.203D  0 0.146D  0 0.106D  0 0.687D -1 0.000D  0
 I = 12 , AREA(I) =     950.5340
 PXA(I):   0.488D  7   0.493D  7   0.503D  7   0.516D  7   0.529D  7   42D  7
 PXA(I):   0.555D  7   0.568D  7   0.585D  7   0.604D  7   0.623D  7   0.642D  7

RXA(I):   0.999D  0   0.908D  0   0.817D  0   0.726D  0   0.635D  0   0.544D  0


RXA(I):   0.454D  0   0.363D  0   0.272D  0   0.181D  0   0.899D -1  -0.100D -2

 ONE OF THE RXA'S MUST BE REAL CLOSE TO 0.     IS IT ????????
A=      426.8  B=        76.9
PXA: (IN PSI)
    70.81      71.58      72.89      74.82      76.75      78.66

    80.53      82.42      84.85      87.60      90.40      93.10

 ITERATION SUCCESSFUL ON KI
 Q=        2.5 RMU=        10.0H=        0.0 B=        76.9A =       426.8
 SHAPE OF FRAC H/HZ FROM XA=1 TO 0
     1.1361      1.1449      1.0876      1.1036      1.1313      1.1612

     1.1934      1.2289      1.2802      1.3497      1.4359      1.5380

VOLUME(X) AS X=A TO 0 : (CGS)

.2961D  60.5793D  60.5799D  60.6110D  60.6480D  60.6880D  60.7310D  6


.7844D  60.8580D  60.9515D  60.1063D  7
PLEASE INPUT: TSLOPE,TCEPT,LEAKC,SP
TSLOPE=SLOPE OF LOG L(FT) VS LOG T(SEC) PLOT
TCEPT = INTERCEPT OF PLOT AT T=1SEC, IE LOG T=0. THE INTERCEPT IS THE LOG VALUE
LEAKC = LEAKOFF COEFFICIENT C IN CGS
SP = SPURT LOSS COEFFICIENT IN CGS
1,1,0,0
TSLOPE,TEXPONENT =     1.00000     1.00000  , TCEPT,TDIV =     1.00000    10.00000
  , LEAKC =    0.000000 SP=    0.000000
TAU(I) AS I=1 TO N+1
0.0000D  0   0.3880D  1   0.7760D  1   0.1164D  2   0.1552D  2   0.1940D  2

0.2328D  2   0.2716D  2   0.3104D  2   0.3492D  2   0.3880D  2   0.4268D  2

TOTLEAK1=  0.0000D  0, VTOT1=  0.6918D  7
TOTLEAK2=  0.0000D  0, VTOT2=  0.7791D  7

T(1)=  0.3880D  2, T(2)=  0.4268D  2
T(1),T(2) COME FROM RATE OF FRACTURE LENGTH EXPANSION A(T)
T1CHECK:(SEC,MIN) 0.1048D  4 0.1747D  2 T2CHECK:(SEC,MIN) 0.1180D  4 0.1967D  2
T1CHECK,T2CHECK= TIME CALCD FROM (FRAC VOL + LEAKOFF VOL)/ (INLET FLOW RATE)
THESE TIMES ARE TOTAL PUMPING TIMES.
DT (T2CHECK-T1CHECK)= 0.1323D  3 SEC.,   T(2)-T(1)= 0.3880D  1 SEC.
% DIFF. BETWEEN T1CHECK AND T(1) =       96.298246
% DIFF. BETWEEN T2CHECK AND T(2) =       96.384354

V/2 PUMPED (FRAC1) (CGS,BRL,GAL):  0.6918D  7 0.4367D  2 0.1828D  4
V/2 PUMPED (FRAC2) (CGS,BRL,GAL):  0.7791D  7 0.4918D  2 0.2058D  4

TO IMPROVE THE TIME DISCREPANCY WITHOUT PRINTING THE FLOW RATES, HIT 10
TO PRINT THE FLOW RATES AND IMPROVING THE TIME DISCREPANCY, HIT 20
TO PRINT THE FLOW RATES IF YOU ARE FINISHED WITH THE TIME DISCREPANCY, HIT 30
30
TSLOPE,TEXPONENT =     1.00000     1.00000  , TCEPT,TDIV =     1.00000    10.00000
  , LEAKC =    0.000000 SP=    0.000000

FRAC  1
LEAKRATE(X) AS X=0 TO A:
0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0
0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0

FRAC  2
LEAKRATE(X) AS X=0 TO A:
0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0
0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0
OUTPUT Q(X) AS X=0 TO A:
0.2500D  1   0.2385D  1   0.2273D  1   0.2167D  1   0.2077D  1   0.2000D  1

0.1926D  1   0.1857D  1   0.1794D  1   0.1730D  1   0.8480D  0
MAX.(QX OUTPUT, % DIF) = 0.1730D  1 615.74
INPUT QX, AS X=0 TO A:
0.2500D  1   0.2038D  1   0.1575D  1   0.1113D  1   0.6500D  0   0.5683D  0

0.4867D  0   0.4050D  0   0.3233D  0   0.2417D  0   0.1600D  0
HIT 1 IF YOU WANT TO CALL QTEST SUBROUTINE
1
INPUT MAX NUM OF ITERATION FOR QX :
20
 NUM = 20
```

```
SLOPE =
.273D  1 0.273D  1 0.274D  0 0.272D  0 0.289D  0 0.302D  0 0.315D  0 0.731D  0 0
.781D  0 0.800D  0 0.806D  0
CEPT =

.300D  1 0.300D  1 0.789D  0 0.788D  0 0.800D  0 0.808D  0 0.814D  0 0.981D  0 0
.996D  0 0.999D  0 0.100D  1
I,DL,DLDT(I):      1 0.134620D  2 0.000000D  0
I,DL,DLDT(I):      2 0.317939D  2 0.620182D  0
I,DL,DLDT(I):      3 0.245453D  2 0.592782D  0
I,DL,DLDT(I):      4 0.271850D  2 0.602761D  0
I,DL,DLDT(I):      5 0.294137D  2 0.611185D  0
I,DL,DLDT(I):      6 0.311371D  2 0.617700D  0
I,DL,DLDT(I):      7 0.337417D  2 0.627545D  0
I,DL,DLDT(I):      8 0.434770D  2 0.664345D  0
I,DL,DLDT(I):      9 0.492021D  2 0.685986D  0
I,DL,DLDT(I):     10 0.515758D  2 0.694959D  0
I,DL,DLDT(I):     11 0.516429D  2 0.695213D  0
MAX % DIF DLDT =     68.6960

HIT 1 IF YOU WANT TO PRINT ALL CALCNS FOR DEBUG
2
INPUT:  HSTART/HZ (CAN BE > OR < 1)
1.136
 HSTART/HZ =      1.1360
INITIAL PRESSURE AT X=A IS        70.8 PSI
 IF YOU WANT TO CHANGE HSTART,  HIT 1
2

INPUT ELR (USUALLY 0.6)
.6
STEP=     0.182D -2 XA=     0.999D  0 T1,T2,T3,T4=      0.289D  0    0.292D  0    0.2
92D  0    0.294D  0 Y=        0.48849436D  7
STEP=     0.182D -2 XA=     0.997D  0 T1,T2,T3,T4=      0.295D  0    0.298D  0    0.2
98D  0    0.300D  0 Y=        0.48884632D  7
STEP=     0.182D -2 XA=     0.995D  0 T1,T2,T3,T4=      0.301D  0    0.303D  0    0.3
03D  0    0.306D  0 Y=        0.48920495D  7
STEP=     0.182D -2 XA=     0.994D  0 T1,T2,T3,T4=      0.306D  0    0.309D  0    0.3
09D  0    0.312D  0 Y=        0.48957023D  7
STEP=     0.182D -2 XA=     0.992D  0 T1,T2,T3,T4=      0.312D  0    0.315D  0    0.3
15D  0    0.317D  0 Y=        0.48994217D  7
STEP=     0.182D -2 XA=     0.990D  0 T1,T2,T3,T4=      0.317D  0    0.320D  0    0.3
20D  0    0.323D  0 Y=        0.49032076D  7
C1,C2,C4 =      0.642D-11    0.439D  7  -0.300D 38STEP=    0.182D -2

PHI=  0.5657D  0 DELTAY=  0.1000D  0 H=  0.3476D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.200D  0 0.199D  0 0.196D  0 0.190D  0 0.182D  0 0.171D  0

0.156D  0 0.137D  0 0.110D  0 0.644D -1 0.000D  0
 I =  2 , AREA(I) =     523.2214
PHI=  0.5517D  0 DELTAY=  0.1000D  0 H=  0.3689D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.220D  0 0.219D  0 0.215D  0 0.209D  0 0.199D  0 0.187D  0

0.169D  0 0.146D  0 0.112D  0 0.628D -1 0.000D  0
 I =  3 , AREA(I) =     601.5215
PHI=  0.5360D  0 DELTAY=  0.1000D  0 H=  0.3906D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.238D  0 0.237D  0 0.232D  0 0.225D  0 0.214D  0 0.199D  0

0.179D  0 0.152D  0 0.105D  0 0.627D -1 0.000D  0
 I =  4 , AREA(I) =     673.9162
PHI=  0.5224D  0 DELTAY=  0.1000D  0 H=  0.4124D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.254D  0 0.253D  0 0.248D  0 0.239D  0 0.227D  0 0.210D  0

0.187D  0 0.154D  0 0.102D  0 0.634D -1 0.000D  0
 I =  5 , AREA(I) =     746.7114
PHI=  0.5080D  0 DELTAY=  0.1000D  0 H=  0.4346D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.269D  0 0.267D  0 0.262D  0 0.253D  0 0.239D  0 0.220D  0

0.193D  0 0.150D  0 0.102D  0 0.643D -1 0.000D  0
 I =  6 , AREA(I) =     818.6455
PHI=  0.4941D  0 DELTAY=  0.1000D  0 H=  0.4572D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.283D  0 0.281D  0 0.275D  0 0.265D  0 0.250D  0 0.228D  0

0.197D  0 0.143D  0 0.102D  0 0.654D -1 0.000D  0
 I =  7 , AREA(I) =     890.9212
PHI=  0.4823D  0 DELTAY=  0.1000D  0 H=  0.4798D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.297D  0 0.295D  0 0.288D  0 0.277D  0 0.260D  0 0.235D  0
```

```
0.198D  0 0.142D  0 0.103D  0 0.669D -1 0.000D  0
 I =  8 , AREA(I) =     966.5832
PHI=  0.4688D  0 DELTAY=  0.1000D  0 H=  0.5050D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.312D  0 0.309D  0 0.302D  0 0.289D  0 0.270D  0 0.242D  0
```



```
0.195D  0 0.142D  0 0.105D  0 0.685D -1 0.000D  0
 I =  9 , AREA(I) =    1049.2564
PHI=  0.4541D  0 DELTAY=  0.1000D  0 H=  0.5333D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.326D  0 0.324D  0 0.315D  0 0.301D  0 0.279D  0 0.247D  0

0.188D  0 0.143D  0 0.107D  0 0.702D -1 0.000D  0
 I = 10 , AREA(I) =    1140.0457
PHI=  0.4411D  0 DELTAY=  0.1000D  0 H=  0.5647D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.342D  0 0.339D  0 0.330D  0 0.313D  0 0.288D  0 0.250D  0

0.186D  0 0.145D  0 0.109D  0 0.722D -1 0.000D  0
 I = 11 , AREA(I) =    1244.3010
 PXA(I):   0.488D  7   0.516D  7   0.549D  7   0.573D  7   0.593D  7   0.610D  7
 PXA(I):   0.624D  7   0.637D  7   0.650D  7   0.662D  7   0.672D  7

RXA(I):   0.999D  0   0.899D  0   0.799D  0   0.699D  0   0.599D  0   0.499D  0


RXA(I):   0.399D  0   0.299D  0   0.199D  0   0.990D -1  -0.100D -2
 ONE OF THE RXA'S MUST BE REAL CLOSE TO 0.    IS IT ????????
A=      388.0  B=        92.6
PXA: (IN PSI)
    70.81      74.91      79.57      83.10      85.98      88.42

    90.53      92.46      94.27      95.96      97.53
 ITERATION SUCCESSFUL ON KI
 Q=        2.5 RMU=        10.0H=        0.0 B=        92.6A =       388.0
 SHAPE OF FRAC H/HZ FROM XA=1 TO 0
     1.1361      1.1403      1.2103      1.2816      1.3530      1.4258

     1.5000      1.5742      1.6570      1.7497      1.8527
VOLUME(X) AS X=A TO 0 : (CGS)

.3094D  60.6651D  60.7542D  60.8400D  60.9256D  60.1011D  70.1098D  7


.1192D  70.1295D  70.1410D  70.0000D  0

% DIF IN PXA(11),HP(11) WITH LAST FRAC1 :    5.4149   18.8178
 INPUT:  NRES,MFAC   (FOR USE IN RK2)
1,5
NRES =    1   MFAC =    5

HIT 1 IF YOU WANT TO PRINT ALL CALCNS FOR DEBUG
2
INPUT:  HSTART/HZ (CAN BE > OR < 1)
1.136
 HSTART/HZ =      1.1360
INITIAL PRESSURE AT X=A IS        70.8 PSI
 IF YOU WANT TO CHANGE HSTART,  HIT 1
2

INPUT ELR (USUALLY 0.6)
.6
STEP=     0.182D -2 XA=     0.999D  0 T1,T2,T3,T4=     0.289D  0   0.292D  0   0.2
92D  0    0.294D  0 Y=        0.48852888D  7
STEP=     0.182D -2 XA=     0.997D  0 T1,T2,T3,T4=     0.293D  0   0.296D  0   0.2
96D  0    0.299D  0 Y=        0.48891400D  7
STEP=     0.182D -2 XA=     0.995D  0 T1,T2,T3,T4=     0.297D  0   0.300D  0   0.3
00D  0    0.303D  0 Y=        0.48930435D  7
STEP=     0.182D -2 XA=     0.994D  0 T1,T2,T3,T4=     0.301D  0   0.304D  0   0.3
04D  0    0.307D  0 Y=        0.48969988D  7
STEP=     0.182D -2 XA=     0.992D  0 T1,T2,T3,T4=     0.305D  0   0.308D  0   0.3
08D  0    0.311D  0 Y=        0.49010050D  7
STEP=     0.182D -2 XA=     0.990D  0 T1,T2,T3,T4=     0.309D  0   0.312D  0   0.3
12D  0    0.314D  0 Y=        0.49050614D  7
C1,C2,C4 =      0.642D-11    0.439D  7  -0.300D 38STEP=   0.182D -2

PHI=  0.5565D  0 DELTAY=  0.1000D  0 H=  0.3597D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.202D  0 0.201D  0 0.198D  0 0.192D  0 0.183D  0 0.172D  0

0.156D  0 0.136D  0 0.107D  0 0.588D -1 0.000D  0
 I =  2 , AREA(I) =     541.0215
PHI=  0.5545D  0 DELTAY=  0.1000D  0 H=  0.3647D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.216D  0 0.215D  0 0.211D  0 0.205D  0 0.196D  0 0.183D  0
```

```
0.167D  0 0.145D  0 0.112D  0 0.628D -1 0.000D  0
 I =  3 , AREA(I) =     585.7410
PHI=  0.5392D  0 DELTAY=  0.1000D  0 H=  0.3859D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.234D  0 0.233D  0 0.229D  0 0.222D  0 0.211D  0 0.197D  0

0.177D  0 0.151D  0 0.106D  0 0.627D -1 0.000D  0
 I =  4 , AREA(I) =     658.2033
PHI=  0.5254D  0 DELTAY=  0.1000D  0 H=  0.4076D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.251D  0 0.249D  0 0.244D  0 0.236D  0 0.224D  0 0.208D  0

0.186D  0 0.154D  0 0.102D  0 0.632D -1 0.000D  0
 I =  5 , AREA(I) =     730.6134
PHI=  0.5116D  0 DELTAY=  0.1000D  0 H=  0.4294D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.266D  0 0.264D  0 0.259D  0 0.250D  0 0.236D  0 0.218D  0

0.192D  0 0.152D  0 0.102D  0 0.640D -1 0.000D  0
 I =  6 , AREA(I) =     802.3576
PHI=  0.4971D  0 DELTAY=  0.1000D  0 H=  0.4517D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.280D  0 0.278D  0 0.272D  0 0.262D  0 0.247D  0 0.226D  0

0.196D  0 0.144D  0 0.102D  0 0.651D -1 0.000D  0
 I =  7 , AREA(I) =     872.9150
PHI=  0.4849D  0 DELTAY=  0.1000D  0 H=  0.4746D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.294D  0 0.292D  0 0.285D  0 0.274D  0 0.257D  0 0.233D  0

0.198D  0 0.142D  0 0.103D  0 0.664D -1 0.000D  0
 I =  8 , AREA(I) =     947.6343
PHI=  0.4729D  0 DELTAY=  0.1000D  0 H=  0.4976D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.307D  0 0.305D  0 0.298D  0 0.285D  0 0.267D  0 0.240D  0

0.197D  0 0.142D  0 0.104D  0 0.679D -1 0.000D  0
 I =  9 , AREA(I) =    1024.1823
PHI=  0.4588D  0 DELTAY=  0.1000D  0 H=  0.5231D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.321D  0 0.318D  0 0.311D  0 0.297D  0 0.276D  0 0.245D  0

0.189D  0 0.143D  0 0.106D  0 0.696D -1 0.000D  0
 I = 10 , AREA(I) =    1105.9947
PHI=  0.4464D  0 DELTAY=  0.1000D  0 H=  0.5516D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.336D  0 0.333D  0 0.324D  0 0.308D  0 0.285D  0 0.249D  0

0.187D  0 0.144D  0 0.108D  0 0.714D -1 0.000D  0
 I = 11 , AREA(I) =    1200.5961
PHI=  0.4338D  0 DELTAY=  0.1000D  0 H=  0.5830D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.350D  0 0.347D  0 0.337D  0 0.320D  0 0.293D  0 0.250D  0

0.187D  0 0.146D  0 0.111D  0 0.733D -1 0.000D  0
 I = 12 , AREA(I) =    1305.5457
 PXA(I):   0.488D  7   0.512D  7   0.542D  7   0.568D  7   0.589D  7   0.606D  7
 PXA(I):   0.621D  7   0.634D  7   0.646D  7   0.657D  7   0.668D  7   0.678D  7

RXA(I):   0.999D  0   0.908D  0   0.817D  0   0.726D  0   0.635D  0   0.544D  0

RXA(I):   0.454D  0   0.363D  0   0.272D  0   0.181D  0   0.899D -1  -0.100D -2

 ONE OF THE RXA'S MUST BE REAL CLOSE TO 0.    IS IT ????????
A=      426.8   B=       95.6
PXA: (IN PSI)
    70.81      74.30      78.66      82.40      85.39      87.89

    90.04      91.94      93.70      95.36      96.91      98.35

 ITERATION SUCCESSFUL ON KI
 Q=        2.5 RMU=       10.0H=        0.0 B=       95.6A =       426.8
 SHAPE OF FRAC H/HZ FROM XA=1 TO 0
     1.1361      1.1801      1.1965      1.2660      1.3372      1.4088

     1.4821      1.5570      1.6324      1.7161      1.8096      1.9127

VOLUME(X) AS X=A TO 0 : (CGS)

.3199D  60.6663D  60.7356D  60.8212D  60.9065D  60.9906D  60.1077D  7

.1166D  70.1260D  70.1364D  70.1482D  7
PLEASE INPUT: TSLOPE,TCEPT,LEAKC,SP
TSLOPE=SLOPE OF LOG L(FT) VS LOG T(SEC) PLOT
```

```
TCEPT = INTERCEPT OF PLOT AT T=1SEC, IE LOG T=0. THE INTERCEPT IS THE LOG VALUE
LEAKC = LEAKOFF COEFFICIENT C IN CGS
SP = SPURT LOSS COEFFICIENT IN CGS
1,1.00,0,0
TSLOPE,TEXPONENT =     1.00000      1.00000   , TCEPT,TDIV =     1.00000     10.00000
  , LEAKC =    0.000000 SP=    0.000000
TAU(I) AS I=1 TO N+1
0.0000D  0    0.3880D  1    0.7760D  1    0.1164D  2    0.1552D  2          -39- 2

0.2328D  2    0.2716D  2    0.3104D  2    0.3492D  2    0.3880D  2    0.4268D  2

TOTLEAK1=  0.0000D  0, VTOT1=  0.9500D  7
TOTLEAK2=  0.0000D  0, VTOT2=  0.1079D  8

T(1)=  0.3880D  2, T(2)=  0.4268D  2
T(1),T(2) COME FROM RATE OF FRACTURE LENGTH EXPANSION A(T)
T1CHECK:(SEC,MIN) 0.1439D  4 0.2399D  2 T2CHECK:(SEC,MIN) 0.1635D  4 0.2724D  2
T1CHECK,T2CHECK= TIME CALCD FROM (FRAC VOL + LEAKOFF VOL)/ (INLET FLOW RATE)
THESE TIMES ARE TOTAL PUMPING TIMES.
DT (T2CHECK-T1CHECK)= 0.1951D  3 SEC.,    T(2)-T(1)= 0.3880D  1 SEC.
% DIFF. BETWEEN T1CHECK AND T(1) =       97.304417
% DIFF. BETWEEN T2CHECK AND T(2) =       97.388861

V/2 PUMPED (FRAC1) (CGS,BRL,GAL):  0.9500D  7 0.5997D  2 0.2510D  4
V/2 PUMPED (FRAC2) (CGS,BRL,GAL):  0.1079D  8 0.6811D  2 0.2850D  4

TO IMPROVE THE TIME DISCREPANCY WITHOUT PRINTING THE FLOW RATES, HIT 10
TO PRINT THE FLOW RATES AND IMPROVING THE TIME DISCREPANCY, HIT 20
TO PRINT THE FLOW RATES IF YOU ARE FINISHED WITH THE TIME DISCREPANCY, HIT 30
30
TSLOPE,TEXPONENT =     1.00000      1.00000   , TCEPT,TDIV =     1.00000     10.00000
  , LEAKC =    0.000000 SP=    0.000000

FRAC  1
LEAKRATE(X) AS X=0 TO A:
0.0000D  0    0.0000D  0    0.0000D  0    0.0000D  0    0.0000D  0    0.0000D  0
0.0000D  0    0.0000D  0    0.0000D  0    0.0000D  0

FRAC  2
LEAKRATE(X) AS X=0 TO A:
0.0000D  0    0.0000D  0    0.0000D  0    0.0000D  0    0.0000D  0    0.0000D  0
0.0000D  0    0.0000D  0    0.0000D  0    0.0000D  0    0.0000D  0
OUTPUT Q(X) AS X=0 TO A:
0.2500D  1    0.2360D  1    0.2226D  1    0.2094D  1    0.1963D  1    0.1836D  1

0.1710D  1    0.1581D  1    0.1451D  1    0.1314D  1    0.6210D  0
MAX.(QX OUTPUT, % DIF) = 0.1963D  1  14.10
INPUT QX, AS X=0 TO A:
0.2500D  1    0.2298D  1    0.2098D  1    0.1903D  1    0.1720D  1    0.1642D  1

0.1566D  1    0.1494D  1    0.1426D  1    0.1358D  1    0.6760D  0
HIT 1 IF YOU WANT TO CALL QTEST SUBROUTINE
1

SLOPE =

.276D  1 0.276D  1 0.479D  0 0.458D  0 0.459D  0 0.454D  0 0.461D  0 0.576D  0 0
.589D  0 0.604D  0 0.621D  0
CEPT =

.301D  1 0.301D  1 0.961D  0 0.944D  0 0.945D  0 0.942D  0 0.945D  0 0.992D  0 0
.995D  0 0.998D  0 0.100D  1
I,DL,DLDT(I):      1 0.670753D  2 0.000000D  0
I,DL,DLDT(I):      2 0.857065D  2 0.529690D  0
I,DL,DLDT(I):      3 0.849655D  2 0.514091D  0
I,DL,DLDT(I):      4 0.848142D  2 0.518693D  0
I,DL,DLDT(I):      5 0.850754D  2 0.523574D  0
I,DL,DLDT(I):      6 0.857483D  2 0.528556D  0
I,DL,DLDT(I):      7 0.868533D  2 0.536309D  0
I,DL,DLDT(I):      8 0.887133D  2 0.559475D  0
I,DL,DLDT(I):      9 0.901018D  2 0.573854D  0
I,DL,DLDT(I):     10 0.913202D  2 0.581462D  0
I,DL,DLDT(I):     11 0.914538D  2 0.581931D  0
MAX % DIF DLDT =     19.5403

HIT 1 IF YOU WANT TO PRINT ALL CALCNS FOR DEBUG
2
INPUT:  HSTART/HZ (CAN BE > OR < 1)
1.136
 HSTART/HZ =      1.1360
INITIAL PRESSURE AT X=A IS        70.8 PSI
 IF YOU WANT TO CHANGE HSTART,  HIT 1
2

INPUT ELR (USUALLY 0.6)
.6
STEP=    0.182D -2 XA=    0.999D  0 T1,T2,T3,T4=     0.272D  0    0.274D  0    0.2
74D  0    0.277D  0 Y=       0.48847376D  7
```

```
STEP=     0.182D -2 XA=     0.997D  0 T1,T2,T3,T4=      0.277D  0    0.280D  0    0.2
80D  0    0.283D  0 Y=        0.48880499D  7
STEP=     0.182D -2 XA=     0.995D  0 T1,T2,T3,T4=      0.283D  0    0.286D  0    0.2
86D  0    0.288D  0 Y=        0.48914276D  7
STEP=     0.182D -2 XA=     0.994D  0 T1,T2,T3,T4=      0.288D  0    0.291D  )    0.2
91D  0    0.294D  0 Y=        0.48948707D  7
STEP=     0.182D -2 XA=     0.992D  0 T1,T2,T3,T4=      0.294D  0    (                0.2
97D  0    0.299D  0 Y=        0.48983789D  7
STEP=     0.182D -2 XA=     0.990D  0 T1,T2,T3,T4=      0.299D  0    0.302D  0    0.3
02D  0    0.305D  0 Y=        0.49019521D  7
C1,C2,C4 =      0.642D-11    0.439D  7  -0.300D 38STEP=    0.182D -2

PHI=  0.5646D  0 DELTAY=  0.1000D  0 H=  0.3489D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.200D  0 0.199D  0 0.195D  0 0.190D  0 0.182D  0 0.171D  0

0.156D  0 0.137D  0 0.110D  0 0.635D -1 0.000D  0
 I =  2 , AREA(I) =     524.2171
PHI=  0.5512D  0 DELTAY=  0.1000D  0 H=  0.3694D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.220D  0 0.218D  0 0.215D  0 0.208D  0 0.199D  0 0.186D  0

0.169D  0 0.146D  0 0.112D  0 0.622D -1 0.000D  0
 I =  3 , AREA(I) =     600.0093
PHI=  0.5352D  0 DELTAY=  0.1000D  0 H=  0.3916D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.238D  0 0.236D  0 0.232D  0 0.225D  0 0.214D  0 0.199D  0

0.179D  0 0.151D  0 0.104D  0 0.622D -1 0.000D  0
 I =  4 , AREA(I) =     673.5994
PHI=  0.5208D  0 DELTAY=  0.1000D  0 H=  0.4145D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.254D  0 0.253D  0 0.248D  0 0.240D  0 0.227D  0 0.210D  0

0.187D  0 0.153D  0 0.101D  0 0.629D -1 0.000D  0
 I =  5 , AREA(I) =     749.6840
PHI=  0.5049D  0 DELTAY=  0.1000D  0 H=  0.4383D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.270D  0 0.268D  0 0.263D  0 0.254D  0 0.240D  0 0.220D  0

0.193D  0 0.146D  0 0.101D  0 0.639D -1 0.000D  0
 I =  6 , AREA(I) =     825.5002
PHI=  0.4906D  0 DELTAY=  0.1000D  0 H=  0.4630D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.286D  0 0.284D  0 0.277D  0 0.267D  0 0.251D  0 0.229D  0

0.196D  0 0.142D  0 0.102D  0 0.652D -1 0.000D  0
 I =  7 , AREA(I) =     905.1037
PHI=  0.4774D  0 DELTAY=  0.1000D  0 H=  0.4883D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.300D  0 0.298D  0 0.291D  0 0.279D  0 0.262D  0 0.236D  0

0.197D  0 0.141D  0 0.103D  0 0.667D -1 0.000D  0
 I =  8 , AREA(I) =     988.2060
PHI=  0.4619D  0 DELTAY=  0.1000D  0 H=  0.5155D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.315D  0 0.313D  0 0.305D  0 0.292D  0 0.272D  0 0.242D  0

0.189D  0 0.141D  0 0.105D  0 0.683D -1 0.000D  0
 I =  9 , AREA(I) =    1074.2416
PHI=  0.4485D  0 DELTAY=  0.1000D  0 H=  0.5450D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.330D  0 0.327D  0 0.319D  0 0.304D  0 0.281D  0 0.246D  0

0.185D  0 0.142D  0 0.107D  0 0.701D -1 0.000D  0
 I = 10 , AREA(I) =    1169.9648
PHI=  0.4356D  0 DELTAY=  0.1000D  0 H=  0.5768D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.345D  0 0.342D  0 0.333D  0 0.316D  0 0.290D  0 0.248D  0

0.185D  0 0.144D  0 0.109D  0 0.720D -1 0.000D  0
 I = 11 , AREA(I) =    1275.0324
 PXA(I):   0.488D  7   0.515D  7   0.546D  7   0.571D  7   0.592D  7   0.610D  7
 PXA(I):   0.625D  7   0.639D  7   0.651D  7   0.663D  7   0.673D  7

RXA(I):   0.999D  0   0.899D  0   0.799D  0   0.699D  0   0.599D  0   0.499D  0


RXA(I):   0.399D  0   0.299D  0   0.199D  0   0.990D -1  -0.100D -2
 ONE OF THE RXA'S MUST BE REAL CLOSE TO 0.    IS IT ????????
A=      388.0  B=        94.6
PXA: (IN PSI)
    70.81      74.70      79.27      82.87      85.86      88.42

    90.66      92.65      94.46      96.10      97.60
 ITERATION SUCCESSFUL ON KI
 Q=         2.5 RMU=        10.0H=         0.0 B=        94.6A =       388.0
```

```
 SHAPE OF FRAC H/HZ FROM XA=1 TO 0
     1.1361       1.1447       1.2120       1.2847       1.3598       1.4378

     1.5191       1.6022       1.6914       1.7880       1.8925
VOLUME(X) AS X=A TO 0 : (CGS)

.3100D  60.6648D  60.7531D  60.8416D  60.9314D  60.1023D  70.1120D  7

.1220D  70.1327D  70.1446D  70.0000D  0

% DIF IN PXA(11),HP(11) WITH LAST FRAC1 :     0.0680     2.1044
 INPUT:  NRES,MFAC    (FOR USE IN RK2)
1,5
NRES =    1    MFAC =     5

HIT 1 IF YOU WANT TO PRINT ALL CALCNS FOR DEBUG
2
INPUT:  HSTART/HZ (CAN BE > OR < 1)
1.136
 HSTART/HZ =      1.1360
INITIAL PRESSURE AT X=A IS        70.8 PSI
 IF YOU WANT TO CHANGE HSTART,  HIT 1
2

INPUT ELR (USUALLY 0.6)
.6
STEP=     0.182D -2 XA=     0.999D  0 T1,T2,T3,T4=     0.272D  0   0.274D  0   0.2
74D  0    0.277D  0 Y=        0.48850621D  7
STEP=     0.182D -2 XA=     0.997D  0 T1,T2,T3,T4=     0.276D  0   0.279D  0   0.2
79D  0    0.281D  0 Y=        0.48886892D  7
STEP=     0.182D -2 XA=     0.995D  0 T1,T2,T3,T4=     0.280D  0   0.283D  0   0.2
83D  0    0.286D  0 Y=        0.48923712D  7
STEP=     0.182D -2 XA=     0.994D  0 T1,T2,T3,T4=     0.285D  0   0.287D  0   0.2
87D  0    0.290D  0 Y=        0.48961075D  7
STEP=     0.182D -2 XA=     0.992D  0 T1,T2,T3,T4=     0.289D  0   0.291D  0   0.2
91D  0    0.294D  0 Y=        0.48998974D  7
STEP=     0.182D -2 XA=     0.990D  0 T1,T2,T3,T4=     0.293D  0   0.295D  0   0.2
95D  0    0.298D  0 Y=        0.49037402D  7
C1,C2,C4 =      0.642D-11    0.439D  7  -0.300D 38STEP=    0.182D -2

PHI=  0.5569D  0 DELTAY=  0.1000D  0 H=  0.3592D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.202D  0 0.201D  0 0.197D  0 0.191D  0 0.183D  0 0.171D  0

0.156D  0 0.136D  0 0.107D  0 0.589D -1 0.000D  0
 I =  2 , AREA(I) =     539.1833
PHI=  0.5535D  0 DELTAY=  0.1000D  0 H=  0.3659D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.216D  0 0.215D  0 0.211D  0 0.205D  0 0.196D  0 0.183D  0

0.167D  0 0.144D  0 0.112D  0 0.622D -1 0.000D  0
 I =  3 , AREA(I) =     586.6163
PHI=  0.5381D  0 DELTAY=  0.1000D  0 H=  0.3871D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.234D  0 0.233D  0 0.229D  0 0.221D  0 0.211D  0 0.197D  0

0.177D  0 0.150D  0 0.105D  0 0.622D -1 0.000D  0
 I =  4 , AREA(I) =     658.9969
PHI=  0.5238D  0 DELTAY=  0.1000D  0 H=  0.4097D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.251D  0 0.250D  0 0.245D  0 0.237D  0 0.225D  0 0.208D  0

0.185D  0 0.153D  0 0.101D  0 0.627D -1 0.000D  0
 I =  5 , AREA(I) =     734.1280
PHI=  0.5088D  0 DELTAY=  0.1000D  0 H=  0.4330D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.267D  0 0.265D  0 0.260D  0 0.251D  0 0.237D  0 0.218D  0

0.192D  0 0.150D  0 0.101D  0 0.637D -1 0.000D  0
 I =  6 , AREA(I) =     809.6819
PHI=  0.4937D  0 DELTAY=  0.1000D  0 H=  0.4572D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.282D  0 0.280D  0 0.274D  0 0.264D  0 0.248D  0 0.227D  0

0.196D  0 0.142D  0 0.101D  0 0.649D -1 0.000D  0
 I =  7 , AREA(I) =     886.1395
PHI=  0.4805D  0 DELTAY=  0.1000D  0 H=  0.4823D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.297D  0 0.295D  0 0.288D  0 0.276D  0 0.259D  0 0.234D  0

0.197D  0 0.141D  0 0.102D  0 0.662D -1 0.000D  0
 I =  8 , AREA(I) =     967.7634
PHI=  0.4663D  0 DELTAY=  0.1000D  0 H=  0.5079D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.311D  0 0.309D  0 0.301D  0 0.288D  0 0.269D  0 0.240D  0
```

```
0.192D  0 0.141D  0 0.104D  0 0.678D -1 0.000D  0
 I =  9 , AREA(I) =    1050.0360
PHI=  0.4526D  0 DELTAY=  0.1000D  0 H=  0.5354D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.325D  0 0.323D  0 0.314D  0 0.300D  0 0.278D  0 0.245D  0

0.186D  0 0.142D  0 0.106D  0 0.695D -1 0.000D  0
 I = 10 , AREA(I) =    1138.2587
PHI=  0.4403D  0 DELTAY=  0.1000D  0 H=  0.5650D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.340D  0 0.337D  0 0.328D  0 0.311D  0 0.287D  0 0.248D  0
```



```
0.185D  0 0.144D  0 0.108D  0 0.713D -1 0.000D  0
 I = 11 , AREA(I) =    1236.0152
PHI=  0.4273D  0 DELTAY=  0.1000D  0 H=  0.5969D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.355D  0 0.351D  0 0.341D  0 0.323D  0 0.294D  0 0.246D  0

0.185D  0 0.146D  0 0.110D  0 0.732D -1 0.000D  0
 I = 12 , AREA(I) =    1341.1556
 PXA(I):   0.488D  7   0.511D  7   0.541D  7   0.567D  7   0.588D  7   0.606D  7
 PXA(I):   0.622D  7   0.635D  7   0.648D  7   0.659D  7   0.669D  7   0.679D  7

RXA(I):   0.999D  0   0.908D  0   0.817D  0   0.726D  0   0.635D  0   0.544D  0

RXA(I):   0.454D  0   0.363D  0   0.272D  0   0.181D  0   0.899D -1  -0.100D -2

 ONE OF THE RXA'S MUST BE REAL CLOSE TO 0.    IS IT ????????
A=      426.8  B=        97.9
PXA: (IN PSI)
    70.81      74.19      78.47      82.23      85.30      87.90

    90.16      92.15      93.95      95.58      97.08      98.43

 ITERATION SUCCESSFUL ON KI
 Q=        2.5 RMU=        10.0H=        0.0 B=        97.9A =       426.8
 SHAPE OF FRAC H/HZ FROM XA=1 TO 0
     1.1361      1.1784      1.2006      1.2701      1.3443      1.4207

     1.5001      1.5825      1.6665      1.7566      1.8537      1.9584

VOLUME(X) AS X=A TO 0 : (CGS)

.3188D  60.6657D  60.7365D  60.8238D  60.9129D  60.1003D  70.1096D  7

.1193D  70.1294D  70.1404D  70.1524D  7
PLEASE INPUT: TSLOPE,TCEPT,LEAKC,SP
TSLOPE=SLOPE OF LOG L(FT) VS LOG T(SEC) PLOT
TCEPT = INTERCEPT OF PLOT AT T=1SEC, IE LOG T=0. THE INTERCEPT IS THE LOG VALUE
LEAKC = LEAKOFF COEFFICIENT C IN CGS
SP = SPURT LOSS COEFFICIENT IN CGS
1,1,0,0
TSLOPE,TEXPONENT =    1.00000     1.00000  , TCEPT,TDIV =    1.00000    10.00000
  , LEAKC =    0.000000 SP=    0.000000
TAU(I) AS I=1 TO N+1
0.0000D  0   0.3880D  1   0.7760D  1   0.1164D  2   0.1552D  2   0.1940D  2

0.2328D  2   0.2716D  2   0.3104D  2   0.3492D  2   0.3880D  2   0.4268D  2

TOTLEAK1=  0.0000D  0, VTOT1=  0.9636D  7
TOTLEAK2=  0.0000D  0, VTOT2=  0.1097D  8

T(1)=  0.3880D  2, T(2)=  0.4268D  2
T(1),T(2) COME FROM RATE OF FRACTURE LENGTH EXPANSION A(T)
T1CHECK:(SEC,MIN) 0.1460D  4 0.2433D  2 T2CHECK:(SEC,MIN) 0.1662D  4 0.2771D  2
T1CHECK,T2CHECK= TIME CALCD FROM (FRAC VOL + LEAKOFF VOL)/ (INLET FLOW RATE)
THESE TIMES ARE TOTAL PUMPING TIMES.
DT (T2CHECK-T1CHECK)= 0.2024D  3 SEC.,    T(2)-T(1)= 0.3880D  1 SEC.
% DIFF. BETWEEN T1CHECK AND T(1) =       97.342485
% DIFF. BETWEEN T2CHECK AND T(2) =       97.432588

V/2 PUMPED (FRAC1) (CGS,BRL,GAL):  0.9636D  7 0.6083D  2 0.2546D  4
V/2 PUMPED (FRAC2) (CGS,BRL,GAL):  0.1097D  8 0.6927D  2 0.2899D  4

TO IMPROVE THE TIME DISCREPANCY WITHOUT PRINTING THE FLOW RATES, HIT 10
TO PRINT THE FLOW RATES AND IMPROVING THE TIME DISCREPANCY, HIT 20
TO PRINT THE FLOW RATES IF YOU ARE FINISHED WITH THE TIME DISCREPANCY, HIT 30
30
TSLOPE,TEXPONENT =    1.00000     1.00000  , TCEPT,TDIV =    1.00000    10.00000
  , LEAKC =    0.000000 SP=    0.000000

FRAC  1
LEAKRATE(X) AS X=0 TO A:
0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0
0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0
```

```
FRAC  2
LEAKRATE(X) AS X=0 TO A:
0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0
0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0
OUTPUT Q(X) AS X=0 TO A:
0.2500D  1   0.2354D  1   0.2210D  1   0.2070D  1   0.1933D  1   0.1796D  1

0.1663D  1   0.1529D  1   0.1397D  1   0.1263D  1   0.5968D  0
MAX.(QX OUTPUT, % DIF) = 0.1263D  1  -4.69
INPUT QX, AS X=0 TO A:
0.2500D  1   0.2345D  1   0.2194D  1   0.2047D  1   0.1902D  1   0.1787D  1

0.1674D  1   0.1559D  1   0.1444D  1   0.1325D  1   0.6347D  0
HIT 1 IF YOU WANT TO CALL QTEST SUBROUTINE
1

SLOPE =
.269D  1 0.269D  1 0.523D  0 0.511D  0 0.515D  0 0.514D  0 0.525D  0 0.557D  0 0
.565D  0 0.583D  0 0.594D  0
CEPT =

.293D  1 0.293D  1 0.982D  0 0.973D  0 0.975D  0 0.975D  0 0.980D  0 0.993D  0 0
.995D  0 0.999D  0 0.100D  1
I,DL,DLDT(I):      1 0.645627D  2 0.000000D  0
I,DL,DLDT(I):      2 0.851553D  2 0.475246D  0
I,DL,DLDT(I):      3 0.885673D  2 0.475877D  0
I,DL,DLDT(I):      4 0.907908D  2 0.483672D  0
I,DL,DLDT(I):      5 0.927932D  2 0.491060D  0
I,DL,DLDT(I):      6 0.948220D  2 0.498564D  0
I,DL,DLDT(I):      7 0.966138D  2 0.506868D  0
I,DL,DLDT(I):      8 0.979950D  2 0.521863D  0
I,DL,DLDT(I):      9 0.993487D  2 0.532397D  0
I,DL,DLDT(I):     10 0.100107D  3 0.538076D  0
I,DL,DLDT(I):     11 0.100359D  3 0.538933D  0
MAX % DIF DLDT =     11.4558

HIT 1 IF YOU WANT TO PRINT ALL CALCNS FOR DEBUG
2
INPUT:  HSTART/HZ (CAN BE > OR < 1)  1.136
 HSTART/HZ =      1.1360
INITIAL PRESSURE AT X=A IS        70.8 PSI
 IF YOU WANT TO CHANGE HSTART,  HIT 1
3

INPUT ELR (USUALLY 0.6)
.6
STEP=     0.182D -2 XA=     0.999D  0 T1,T2,T3,T4=     0.260D  0   0.262D  0   0.2
62D  0    0.265D  0 Y=       0.48845933D  7
STEP=     0.182D -2 XA=     0.997D  0 T1,T2,T3,T4=     0.265D  0   0.268D  0   0.2
68D  0    0.270D  0 Y=       0.48877585D  7
STEP=     0.182D -2 XA=     0.995D  0 T1,T2,T3,T4=     0.270D  0   0.273D  0   0.2
73D  0    0.276D  0 Y=       0.48909865D  7
STEP=     0.182D -2 XA=     0.994D  0 T1,T2,T3,T4=     0.276D  0   0.278D  0   0.2
78D  0    0.281D  0 Y=       0.48942772D  7
STEP=     0.182D -2 XA=     0.992D  0 T1,T2,T3,T4=     0.281D  0   0.284D  0   0.2
84D  0    0.286D  0 Y=       0.48976302D  7
STEP=     0.182D -2 XA=     0.990D  0 T1,T2,T3,T4=     0.286D  0   0.289D  0   0.2
89D  0    0.291D  0 Y=       0.49010456D  7
C1,C2,C4 =      0.642D-11    0.439D  7  -0.300D 38STEP=   0.182D -2

PHI=  0.5641D  0 DELTAY=  0.1000D  0 H=  0.3495D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.200D  0 0.198D  0 0.195D  0 0.190D  0 0.181D  0 0.170D  0

0.156D  0 0.136D  0 0.109D  0 0.630D -1 0.000D  0
 I =  2 , AREA(I) =     524.0111
PHI=  0.5515D  0 DELTAY=  0.1000D  0 H=  0.3689D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.219D  0 0.217D  0 0.214D  0 0.207D  0 0.198D  0 0.185D  0

0.168D  0 0.145D  0 0.111D  0 0.620D -1 0.000D  0
 I =  3 , AREA(I) =     596.2969
PHI=  0.5358D  0 DELTAY=  0.1000D  0 H=  0.3904D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.236D  0 0.235D  0 0.231D  0 0.223D  0 0.213D  0 0.198D  0

0.178D  0 0.151D  0 0.104D  0 0.620D -1 0.000D  0
 I =  4 , AREA(I) =     668.0929
PHI=  0.5217D  0 DELTAY=  0.1000D  0 H=  0.4130D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.253D  0 0.251D  0 0.246D  0 0.238D  0 0.226D  0 0.209D  0

0.186D  0 0.153D  0 0.101D  0 0.626D -1 0.000D  0
 I =  5 , AREA(I) =     743.1826
PHI=  0.5059D  0 DELTAY=  0.1000D  0 H=  0.4366D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.269D  0 0.267D  0 0.262D  0 0.252D  0 0.238D  0 0.219D  0
```

```
0.192D  0 0.147D  0 0.101D  0 0.637D -1 0.000D  0
 I =  6 , AREA(I) =     818.9809
PHI=  0.4913D  0 DELTAY=  0.1000D  0 H=  0.4615D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.284D  0 0.282D  0 0.276D  0 0.266D  0 0.250D  0 0.228D  0

0.196D  0 0.141D  0 0.101D  0 0.649D -1 0.000D  0
 I =  7 , AREA(I) =     898.7269
PHI=  0.4777D  0 DELTAY=  0.1000D  0 H=  0.4875D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.299D  0 0.297D  0 0.290D  0 0.278D  0 0.261D  0 0.235D  0

0.196D  0 0.140D  0 0.102D  0 0.664D -1 0.000D  0
 I =  8 , AREA(I) =     983.4449
PHI=  0.4619D  0 DELTAY=  0.1000D  0 H=  0.5152D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.314D  0 0.312D  0 0.304D  0 0.291D  0 0.271D  0 0.241D  0

0.189D  0 0.141D  0 0.104D  0 0.680D -1 0.000D  0
 I =  9 , AREA(I) =    1070.5095
PHI=  0.4483D  0 DELTAY=  0.1000D  0 H=  0.5450D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.329D  0 0.327D  0 0.318D  0 0.303D  0 0.280D  0 0.246D  0

0.185D  0 0.142D  0 0.106D  0 0.697D -1 0.000D  0
 I = 10 , AREA(I) =    1166.7303
PHI=  0.4353D  0 DELTAY=  0.1000D  0 H=  0.5770D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.345D  0 0.341D  0 0.332D  0 0.315D  0 0.289D  0 0.247D  0

0.184D  0 0.144D  0 0.108D  0 0.716D -1 0.000D  0
 I = 11 , AREA(I) =    1271.7084
 PXA(I):   0.488D  7   0.514D  7   0.544D  7   0.569D  7   0.590D  7   0.608D  7
 PXA(I):   0.623D  7   0.637D  7   0.650D  7   0.661D  7   0.672D  7

RXA(I):   0.999D  0   0.899D  0   0.799D  0   0.699D  0   0.599D  0   0.499D  0


RXA(I):   0.399D  0   0.299D  0   0.199D  0   0.990D -1  -0.100D -2
 ONE OF THE RXA'S MUST BE REAL CLOSE TO 0.    IS IT ????????
A=      388.0  B=       94.6
PXA: (IN PSI)
    70.81      74.52      78.96      82.53      85.54      88.14

    90.42      92.45      94.27      95.93      97.42
 ITERATION SUCCESSFUL ON KI
 Q=        2.5 RMU=       10.0H=        0.0 B=       94.6A =       388.0
 SHAPE OF FRAC H/HZ FROM XA=1 TO 0
     1.1361      1.1467      1.2102      1.2808      1.3548      1.4324

     1.5142      1.5995      1.6904      1.7880      1.8929
VOLUME(X) AS X=A TO 0 : (CGS)

.3099D  60.6625D  60.7476D  60.8345D  60.9237D  60.1016D  70.1113D  7


.1215D  70.1323D  70.1442D  70.0000D  0

% DIF IN PXA(11),HP(11) WITH LAST FRAC1 :   -0.1797    0.0200
 INPUT:  NRES,MFAC   (FOR USE IN RK2)
1,5
NRES =    1   MFAC =    5

HIT 1 IF YOU WANT TO PRINT ALL CALCNS FOR DEBUG
2
INPUT:  HSTART/HZ (CAN BE > OR < 1)
1.136
 HSTART/HZ =      1.1360
INITIAL PRESSURE AT X=A IS       70.8 PSI
 IF YOU WANT TO CHANGE HSTART,  HIT 1
2

 INPUT ELR (USUALLY 0.6)
 .6
STEP=    0.182D -2 XA=    0.999D  0 T1,T2,T3,T4=     0.260D  0   0.262D  0   0.2
62D  0   0.265D  0 Y=       0.48849034D  7
STEP=    0.182D -2 XA=    0.997D  0 T1,T2,T3,T4=     0.264D  0   0.267D  0   0.2
67D  0   0.269D  0 Y=       0.48883710D  7
STEP=    0.182D -2 XA=    0.995D  0 T1,T2,T3,T4=     0.268D  0   0.271D  0   0.2
71D  0   0.273D  0 Y=       0.48918929D  7
STEP=    0.182D -2 XA=    0.994D  0 T1,T2,T3,T4=     0.272D  0   0.275D  0   0.2
75D  0   0.277D  0 Y=       0.48954686D  7
STEP=    0.182D -2 XA=    0.992D  0 T1,T2,T3,T4=     0.276D  0   0.279D  0   0.2
79D  0   0.281D  0 Y=       0.48990974D  7
STEP=    0.182D -2 XA=    0.990D  0 T1,T2,T3,T4=     0.280D  0   0.283D  0   0.2
83D  0   0.286D  0 Y=       0.49027785D  7
C1,C2,C4 =      0.642D-11   0.439D  7  -0.300D 38STEP=   0.182D -2
```

```
PHI=  0.5572D  0 DELTAY=  0.1000D  0 H=  0.3587D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.201D  0 0.200D  0 0.197D  0 0.191D  0 0.182D  0 0.171D  0

0.156D  0 0.133D  0 0.106D  0 0.589D -1 0.000D  0
 I =  2 , AREA(I) =     537.3723
PHI=  0.5533D  0 DELTAY=  0.1000D  0 H=  0.3661D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.216D  0 0.214D  0 0.211D  0 0.204D  0 0.195D  0 0.183D  0

0.166D  0 0.144D  0 0.111D  0 0.618D -1 0.000D  0
 I =  3 , AREA(I) =     584.9075
PHI=  0.5385D  0 DELTAY=  0.1000D  0 H=  0.3864D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.233D  0 0.232D  0 0.228D  0 0.220D  0 0.210D  0 0.196D  0

0.176D  0 0.150D  0 0.105D  0 0.619D -1 0.000D  0
 I =  4 , AREA(I) =     654.7324
PHI=  0.5244D  0 DELTAY=  0.1000D  0 H=  0.4085D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.250D  0 0.248D  0 0.243D  0 0.235D  0 0.223D  0 0.207D  0

0.185D  0 0.153D  0 0.101D  0 0.625D -1 0.000D  0
 I =  5 , AREA(I) =     728.4385
PHI=  0.5097D  0 DELTAY=  0.1000D  0 H=  0.4316D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.266D  0 0.264D  0 0.258D  0 0.249D  0 0.236D  0 0.217D  0

0.191D  0 0.150D  0 0.101D  0 0.634D -1 0.000D  0
 I =  6 , AREA(I) =     803.5801
PHI=  0.4944D  0 DELTAY=  0.1000D  0 H=  0.4557D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.281D  0 0.279D  0 0.273D  0 0.263D  0 0.247D  0 0.226D  0

0.195D  0 0.142D  0 0.101D  0 0.646D -1 0.000D  0
 I =  7 , AREA(I) =     879.7792
PHI=  0.4810D  0 DELTAY=  0.1000D  0 H=  0.4811D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.296D  0 0.293D  0 0.287D  0 0.275D  0 0.258D  0 0.234D  0

0.197D  0 0.140D  0 0.102D  0 0.659D -1 0.000D  0
 I =  8 , AREA(I) =     962.0321
PHI=  0.4665D  0 DELTAY=  0.1000D  0 H=  0.5074D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.310D  0 0.308D  0 0.300D  0 0.287D  0 0.268D  0 0.240D  0

0.192D  0 0.140D  0 0.104D  0 0.675D -1 0.000D  0
 I =  9 , AREA(I) =    1046.1199
PHI=  0.4524D  0 DELTAY=  0.1000D  0 H=  0.5354D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.325D  0 0.322D  0 0.314D  0 0.299D  0 0.277D  0 0.245D  0

0.185D  0 0.141D  0 0.105D  0 0.691D -1 0.000D  0
 I = 10 , AREA(I) =    1135.1932
PHI=  0.4400D  0 DELTAY=  0.1000D  0 H=  0.5653D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.339D  0 0.336D  0 0.327D  0 0.311D  0 0.286D  0 0.247D  0

0.184D  0 0.143D  0 0.107D  0 0.709D -1 0.000D  0
 I = 11 , AREA(I) =    1233.5334
PHI=  0.4269D  0 DELTAY=  0.1000D  0 H=  0.5974D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.354D  0 0.350D  0 0.340D  0 0.322D  0 0.294D  0 0.246D  0

0.184D  0 0.145D  0 0.110D  0 0.728D -1 0.000D  0
 I = 12 , AREA(I) =    1338.5794
 PXA(I):   0.488D  7   0.511D  7   0.539D  7   0.565D  7   0.586D  7   0.604D  7
 PXA(I):   0.620D  7   0.634D  7   0.646D  7   0.658D  7   0.668D  7   0.677D  7

RXA(I):   0.999D  0   0.908D  0   0.817D  0   0.726D  0   0.635D  0   0.544D  0


RXA(I):   0.454D  0   0.363D  0   0.272D  0   0.181D  0   0.899D -1  -0.100D -2

 ONE OF THE RXA'S MUST BE REAL CLOSE TO 0.    IS IT ????????
A=       426.8  B=        98.0
PXA: (IN PSI)
    70.81      74.07      78.23      81.92      84.99      87.62

    89.91      91.95      93.77      95.41      96.91      98.27

 ITERATION SUCCESSFUL ON KI
 Q=          2.5 RMU=        10.0H=         0.0 B=        98.0A =       426.8
 SHAPE OF FRAC H/HZ FROM XA=1 TO 0
     1.1361       1.1768       1.2011       1.2677       1.3403       1.4159

     1.4952       1.5784       1.6647       1.7565       1.8548       1.9599
```

```
VOLUME(X) AS X=A TO 0 : (CGS)

.3178D  60.6636D  60.7330D  60.8179D  60.9059D  60.9954D  60.1089D  7


.1187D  70.1290D  70.1401D  70.1521D  7                                  -46-
PLEASE INPUT: TSLOPE,TCEPT,LEAKC,SP
TSLOPE=SLOPE OF LOG L(FT) VS LOG T(SEC) PLOT
TCEPT = INTERCEPT OF PLOT AT T=1SEC, IE LOG T=0. THE INTERCEPT IS THE LOG VALUE
LEAKC = LEAKOFF COEFFICIENT C IN CGS
SP = SPURT LOSS COEFFICIENT IN CGS
1,1,0,0
TSLOPE,TEXPONENT =     1.00000     1.00000  , TCEPT,TDIV =     1.00000     10.00000
  , LEAKC =    0.000000 SP=    0.000000
TAU(I) AS I=1 TO N+1
0.0000D  0    0.3880D  1    0.7760D  1    0.1164D  2    0.1552D  2    0.1940D  2

0.2328D  2    0.2716D  2    0.3104D  2    0.3492D  2    0.3880D  2    0.4268D  2

TOTLEAK1=  0.0000D  0, VTOT1=  0.9586D  7
TOTLEAK2=  0.0000D  0, VTOT2=  0.1092D  8

T(1)=  0.3880D  2, T(2)=  0.4268D  2
T(1),T(2) COME FROM RATE OF FRACTURE LENGTH EXPANSION A(T)
T1CHECK:(SEC,MIN) 0.1452D  4 0.2421D  2 T2CHECK:(SEC,MIN) 0.1655D  4 0.2758D  2
T1CHECK,T2CHECK= TIME CALCD FROM (FRAC VOL + LEAKOFF VOL)/ (INLET FLOW RATE)
THESE TIMES ARE TOTAL PUMPING TIMES.
DT (T2CHECK-T1CHECK)= 0.2023D  3 SEC.,   T(2)-T(1)= 0.3880D  1 SEC.
% DIFF. BETWEEN T1CHECK AND T(1) =       97.328646
% DIFF. BETWEEN T2CHECK AND T(2) =       97.420796

V/2 PUMPED (FRAC1) (CGS,BRL,GAL):  0.9586D  7 0.6052D  2 0.2533D  4
V/2 PUMPED (FRAC2) (CGS,BRL,GAL):  0.1092D  8 0.6895D  2 0.2885D  4

TO IMPROVE THE TIME DISCREPANCY WITHOUT PRINTING THE FLOW RATES, HIT 10
TO PRINT THE FLOW RATES AND IMPROVING THE TIME DISCREPANCY, HIT 20
TO PRINT THE FLOW RATES IF YOU ARE FINISHED WITH THE TIME DISCREPANCY, HIT 30
30
TSLOPE,TEXPONENT =     1.00000     1.00000  , TCEPT,TDIV =     1.00000     10.00000
  , LEAKC =    0.000000 SP=    0.000000

FRAC  1
LEAKRATE(X) AS X=0 TO A:
0.0000D  0    0.0000D  0    0.0000D  0    0.0000D  0    0.0000D  0    0.0000D  0
0.0000D  0    0.0000D  0    0.0000D  0    0.0000D  0

FRAC  2
LEAKRATE(X) AS X=0 TO A:
0.0000D  0    0.0000D  0    0.0000D  0    0.0000D  0    0.0000D  0    0.0000D  0
0.0000D  0    0.0000D  0    0.0000D  0    0.0000D  0    0.0000D  0
OUTPUT Q(X) AS X=0 TO A:
0.2500D  1    0.2352D  1    0.2206D  1    0.2065D  1    0.1926D  1    0.1789D  1

0.1654D  1    0.1521D  1    0.1389D  1    0.1257D  1    0.5949D  0
MAX.(QX OUTPUT, % DIF) = 0.1257D  1  -1.64
INPUT QX, AS X=0 TO A:
0.2500D  1    0.2351D  1    0.2206D  1    0.2064D  1    0.1925D  1    0.1794D  1

0.1665D  1    0.1537D  1    0.1409D  1    0.1278D  1    0.6063D  0
HIT 1 IF YOU WANT TO CALL QTEST SUBROUTINE
1

FLOW RATE CONVERGES TO WITHIN          5. PERCENT AFTER    3 ITERATIONS

SLOPE =

.266D  1 0.266D  1 0.527D  0 0.522D  0 0.530D  0 0.531D  0 0.543D  0 0.558D  0 0
.564D  0 0.582D  0 0.592D  0
CEPT =

.290D  1 0.290D  1 0.979D  0 0.976D  0 0.981D  0 0.981D  0 0.988D  0 0.993D  0 0
.995D  0 0.999D  0 0.100D  1
I,DL,DLDT(I):      1 0.621002D  2 0.000000D  0
I,DL,DLDT(I):      2 0.828674D  2 0.442409D  0
I,DL,DLDT(I):      3 0.875959D  2 0.454410D  0
I,DL,DLDT(I):      4 0.905549D  2 0.465620D  0
I,DL,DLDT(I):      5 0.931472D  2 0.475720D  0
I,DL,DLDT(I):      6 0.956074D  2 0.485551D  0
I,DL,DLDT(I):      7 0.977959D  2 0.495112D  0
I,DL,DLDT(I):      8 0.993244D  2 0.506387D  0
I,DL,DLDT(I):      9 0.100785D  3 0.515264D  0
I,DL,DLDT(I):     10 0.101674D  3 0.520299D  0
I,DL,DLDT(I):     11 0.102000D  3 0.521533D  0
MAX % DIF DLDT =      7.4224
```

```
HIT 1 IF YOU WANT TO PRINT ALL CALCNS FOR DEBUG
2
INPUT:  HSTART/HZ (CAN BE > OR < 1)
1.136
 HSTART/HZ =      1.1360
INITIAL PRESSURE AT X=A IS        70.8 PSI
 IF YOU WANT TO CHANGE HSTART,  HIT 1
2

INPUT ELR (USUALLY 0.6)
.6
STEP=    0.182D -2 XA=     0.999D  0 T1,T2,T3,T4=     0.256D  0   0.259D  0   0.2
59D  0   0.261D  0 Y=        0.48845498D  7
STEP=    0.182D -2 XA=     0.997D  0 T1,T2,T3,T4=     0.261D  0   0.264D  0   0.2
64D  0   0.266D  0 Y=        0.48876702D  7
STEP=    0.182D -2 XA=     0.995D  0 T1,T2,T3,T4=     0.266D  0   0.269D  0   0.2
69D  0   0.272D  0 Y=        0.48908521D  7
STEP=    0.182D -2 XA=     0.994D  0 T1,T2,T3,T4=     0.272D  0   0.274D  0   0.2
74D  0   0.277D  0 Y=        0.48940951D  7
STEP=    0.182D -2 XA=     0.992D  0 T1,T2,T3,T4=     0.277D  0   0.279D  0   0.2
79D  0   0.282D  0 Y=        0.48973992D  7
STEP=    0.182D -2 XA=     0.990D  0 T1,T2,T3,T4=     0.282D  0   0.285D  0   0.2
85D  0   0.287D  0 Y=        0.49007641D  7
C1,C2,C4 =      0.642D-11    0.439D  7  -0.300D 38STEP=   0.182D -2

PHI=  0.5637D  0 DELTAY=  0.1000D  0 H=  0.3500D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.200D  0 0.198D  0 0.195D  0 0.190D  0 0.181D  0 0.170D  0

0.156D  0 0.136D  0 0.109D  0 0.627D -1 0.000D  0
 I =  2 , AREA(I) =     524.4313
PHI=  0.5515D  0 DELTAY=  0.1000D  0 H=  0.3688D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.218D  0 0.217D  0 0.213D  0 0.207D  0 0.197D  0 0.185D  0

0.168D  0 0.145D  0 0.111D  0 0.619D -1 0.000D  0
 I =  3 , AREA(I) =     595.0042
PHI=  0.5361D  0 DELTAY=  0.1000D  0 H=  0.3900D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.236D  0 0.234D  0 0.230D  0 0.223D  0 0.212D  0 0.197D  0

0.178D  0 0.150D  0 0.104D  0 0.619D -1 0.000D  0
 I =  4 , AREA(I) =     665.8039
PHI=  0.5221D  0 DELTAY=  0.1000D  0 H=  0.4123D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.252D  0 0.251D  0 0.246D  0 0.237D  0 0.225D  0 0.209D  0

0.186D  0 0.152D  0 0.101D  0 0.625D -1 0.000D  0
 I =  5 , AREA(I) =     740.1764
PHI=  0.5066D  0 DELTAY=  0.1000D  0 H=  0.4357D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.268D  0 0.266D  0 0.261D  0 0.252D  0 0.238D  0 0.219D  0

0.192D  0 0.147D  0 0.100D  0 0.635D -1 0.000D  0
 I =  6 , AREA(I) =     815.7210
PHI=  0.4918D  0 DELTAY=  0.1000D  0 H=  0.4606D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.284D  0 0.282D  0 0.275D  0 0.265D  0 0.249D  0 0.227D  0

0.196D  0 0.141D  0 0.101D  0 0.648D -1 0.000D  0
 I =  7 , AREA(I) =     895.0022
PHI=  0.4780D  0 DELTAY=  0.1000D  0 H=  0.4868D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.299D  0 0.296D  0 0.290D  0 0.278D  0 0.260D  0 0.235D  0

0.196D  0 0.140D  0 0.102D  0 0.662D -1 0.000D  0
 I =  8 , AREA(I) =     980.0996
PHI=  0.4621D  0 DELTAY=  0.1000D  0 H=  0.5146D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.314D  0 0.311D  0 0.304D  0 0.290D  0 0.270D  0 0.241D  0

0.189D  0 0.140D  0 0.104D  0 0.678D -1 0.000D  0
 I =  9 , AREA(I) =    1067.4133
PHI=  0.4484D  0 DELTAY=  0.1000D  0 H=  0.5444D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.329D  0 0.326D  0 0.318D  0 0.303D  0 0.280D  0 0.245D  0

0.184D  0 0.142D  0 0.106D  0 0.696D -1 0.000D  0
 I = 10 , AREA(I) =    1163.6060
PHI=  0.4354D  0 DELTAY=  0.1000D  0 H=  0.5764D  4
 WIDTH, BROKEN DOWN IN    10 INCREMENTS
 0.344D  0 0.341D  0 0.331D  0 0.314D  0 0.288D  0 0.247D  0

0.184D  0 0.143D  0 0.108D  0 0.714D -1 0.000D  0
 I = 11 , AREA(I) =    1268.4403
 PXA(I):   0.488D  7   0.513D  7   0.543D  7   0.568D  7   0.589D  7   0.607D  7
 PXA(I):   0.622D  7   0.637D  7   0.649D  7   0.661D  7   0.671D  7
```

```
RXA(I):   0.999D  0   0.899D  0   0.799D  0   0.699D  0   0.599D  0   0.499D  0

RXA(I):   0.399D  0   0.299D  0   0.199D  0   0.990D -1  -0.100D -2
 ONE OF THE RXA'S MUST BE REAL CLOSE TO 0.    IS IT ????????
A=       388.0  B=        94.6
PXA: (IN PSI)
    70.81      74.46      78.82      82.38      85.39      88.00
                                                                          -48-
    90.29      92.34      94.17      95.83      97.33
 ITERATION SUCCESSFUL ON KI
 Q=      2.5 RMU=        10.0H=        0.0 B=        94.6A =       388.0
 SHAPE OF FRAC H/HZ FROM XA=1 TO 0
     1.1361      1.1483      1.2099      1.2794      1.3525      1.4296

     1.5111      1.5970      1.6883      1.7862      1.8912
VOLUME(X) AS X=A TO 0 : (CGS)

.3101D  60.6619D  60.7455D  60.8314D  60.9200D  60.1012D  70.1109D  7

.1211D  70.1319D  70.1438D  70.0000D  0

% DIF IN PXA(11),HP(11) WITH LAST FRAC1 :   -0.0987   -0.0936
 INPUT:  NRES,MFAC   (FOR USE IN RK2)
1,5
NRES =    1   MFAC =    5

HIT 1 IF YOU WANT TO PRINT ALL CALCNS FOR DEBUG
2
INPUT:  HSTART/HZ (CAN BE > OR < 1)
1.136
 HSTART/HZ =      1.1360
INITIAL PRESSURE AT X=A IS       70.8 PSI
 IF YOU WANT TO CHANGE HSTART,  HIT 1
2

INPUT ELR (USUALLY 0.6)
.6
STEP=     0.182D -2 XA=     0.999D  0 T1,T2,T3,T4=      0.256D  0    0.259D  0    0.2
59D  0    0.261D  0 Y=        0.48848556D  7
STEP=     0.182D -2 XA=     0.997D  0 T1,T2,T3,T4=      0.260D  0    0.263D  0    0.2
63D  0    0.265D  0 Y=        0.48882750D  7
STEP=     0.182D -2 XA=     0.995D  0 T1,T2,T3,T4=      0.264D  0    0.267D  0    0.2
67D  0    0.270D  0 Y=        0.48917484D  7
STEP=     0.182D -2 XA=     0.994D  0 T1,T2,T3,T4=      0.269D  0    0.271D  0    0.2
71D  0    0.274D  0 Y=        0.48952752D  7
STEP=     0.182D -2 XA=     0.992D  0 T1,T2,T3,T4=      0.273D  0    0.275D  0    0.2
75D  0    0.278D  0 Y=        0.48988547D  7
STEP=     0.182D -2 XA=     0.990D  0 T1,T2,T3,T4=      0.277D  0    0.279D  0    0.2
79D  0    0.282D  0 Y=        0.49024863D  7
C1,C2,C4 =      0.642D-11   0.439D  7  -0.300D 38STEP=   0.182D -2

PHI=  0.5573D  0 DELTAY=  0.1000D  0 H=  0.3585D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.201D  0 0.200D  0 0.197D  0 0.191D  0 0.182D  0 0.171D  0

0.156D  0 0.135D  0 0.106D  0 0.589D -1 0.000D  0
 I =  2 , AREA(I) =     536.7886
PHI=  0.5530D  0 DELTAY=  0.1000D  0 H=  0.3664D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.215D  0 0.214D  0 0.210D  0 0.204D  0 0.195D  0 0.182D  0

0.166D  0 0.143D  0 0.111D  0 0.616D -1 0.000D  0
 I =  3 , AREA(I) =     584.7965
PHI=  0.5386D  0 DELTAY=  0.1000D  0 H=  0.3862D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.233D  0 0.231D  0 0.227D  0 0.220D  0 0.210D  0 0.195D  0

0.176D  0 0.150D  0 0.105D  0 0.618D -1 0.000D  0
 I =  4 , AREA(I) =     653.2036
PHI=  0.5247D  0 DELTAY=  0.1000D  0 H=  0.4080D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.249D  0 0.248D  0 0.243D  0 0.235D  0 0.223D  0 0.207D  0

0.184D  0 0.152D  0 0.101D  0 0.623D -1 0.000D  0
 I =  5 , AREA(I) =     725.9892
PHI=  0.5102D  0 DELTAY=  0.1000D  0 H=  0.4309D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.265D  0 0.263D  0 0.258D  0 0.249D  0 0.235D  0 0.217D  0

0.191D  0 0.150D  0 0.100D  0 0.633D -1 0.000D  0
 I =  6 , AREA(I) =     800.5948
PHI=  0.4948D  0 DELTAY=  0.1000D  0 H=  0.4549D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.280D  0 0.278D  0 0.272D  0 0.262D  0 0.247D  0 0.225D  0
```

```
0.195D  0 0.142D  0 0.101D  0 0.645D -1 0.000D  0
 I =  7 , AREA(I) =      876.2912
PHI=  0.4814D  0 DELTAY=  0.1000D  0 H=  0.4802D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.295D  0 0.293D  0 0.286D  0 0.275D  0 0.258D  0 0.233D  0

0.196D  0 0.140D  0 0.102D  0 0.658D -1 0.000D  0
 I =  8 , AREA(I) =      958.3481
PHI=  0.4669D  0 DELTAY=  0.1000D  0 H=  0.5067D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.310D  0 0.307D  0 0.300D  0 0.287D  0 0.268D  0 0.239D  0
```



```
0.192D  0 0.140D  0 0.103D  0 0.673D -1 0.000D  0
 I =  9 , AREA(I) =     1043.0239
PHI=  0.4526D  0 DELTAY=  0.1000D  0 H=  0.5348D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.324D  0 0.321D  0 0.313D  0 0.299D  0 0.277D  0 0.244D  0

0.185D  0 0.141D  0 0.105D  0 0.690D -1 0.000D  0
 I = 10 , AREA(I) =     1132.1340
PHI=  0.4400D  0 DELTAY=  0.1000D  0 H=  0.5649D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.339D  0 0.336D  0 0.326D  0 0.310D  0 0.285D  0 0.247D  0

0.184D  0 0.143D  0 0.107D  0 0.708D -1 0.000D  0
 I = 11 , AREA(I) =     1230.5325
PHI=  0.4270D  0 DELTAY=  0.1000D  0 H=  0.5969D  4
 WIDTH, BROKEN DOWN IN   10 INCREMENTS
 0.353D  0 0.350D  0 0.340D  0 0.322D  0 0.293D  0 0.245D  0

0.184D  0 0.145D  0 0.110D  0 0.726D -1 0.000D  0
 I = 12 , AREA(I) =     1335.4688
 PXA(I):   0.488D  7   0.510D  7   0.539D  7   0.564D  7   0.585D  7   0.603D  7
 PXA(I):   0.619D  7   0.633D  7   0.646D  7   0.657D  7   0.667D  7   0.677D  7

RXA(I):   0.999D  0   0.908D  0   0.817D  0   0.726D  0   0.635D  0   0.544D  0

RXA(I):   0.454D  0   0.363D  0   0.272D  0   0.181D  0   0.899D -1  -0.100D -2

 ONE OF THE RXA'S MUST BE REAL CLOSE TO 0.    IS IT ????????
A=       426.8  B=        97.9
PXA: (IN PSI)
    70.81      74.03      78.14      81.79      84.85      87.48

    89.78      91.83      93.66      95.31      96.82      98.18

 ITERATION SUCCESSFUL ON KI
 Q=        2.5 RMU=        10.0H=        0.0 B=        97.9A =       426.8
 SHAPE OF FRAC H/HZ FROM XA=1 TO 0
     1.1361      1.1763      1.2022      1.2671      1.3387      1.4136

     1.4923      1.5753      1.6624      1.7547      1.8532      1.9583

VOLUME(X) AS X=A TO 0 : (CGS)

.3174D  60.6632D  60.7320D  60.8155D  60.9027D  60.9916D  60.1085D  7


.1183D  70.1286D  70.1397D  70.1517D  7
PLEASE INPUT: TSLOPE,TCEPT,LEAKC,SP
TSLOPE=SLOPE OF LOG L(FT) VS LOG T(SEC) PLOT
TCEPT = INTERCEPT OF PLOT AT T=1SEC, IE LOG T=0. THE INTERCEPT IS THE LOG VALUE
LEAKC = LEAKOFF COEFFICIENT C IN CGS
SP = SPURT LOSS COEFFICIENT IN CGS
1,1,0,0
TSLOPE,TEXPONENT =     1.00000     1.00000  , TCEPT,TDIV =     1.00000    10.00000
  , LEAKC =    0.000000 SP=    0.000000
TAU(I) AS I=1 TO N+1
0.0000D  0   0.3880D  1   0.7760D  1   0.1164D  2   0.1552D  2   0.1940D  2

0.2328D  2   0.2716D  2   0.3104D  2   0.3492D  2   0.3880D  2   0.4268D  2

TOTLEAK1=  0.0000D  0, VTOT1=  0.9557D  7
TOTLEAK2=  0.0000D  0, VTOT2=  0.1089D  8

T(1)=  0.3880D  2, T(2)=  0.4268D  2
T(1),T(2) COME FROM RATE OF FRACTURE LENGTH EXPANSION A(T)
T1CHECK:(SEC,MIN) 0.1448D  4 0.2413D  2 T2CHECK:(SEC,MIN) 0.1650D  4 0.2750D  2
T1CHECK,T2CHECK= TIME CALCD FROM (FRAC VOL + LEAKOFF VOL)/ (INLET FLOW RATE)
THESE TIMES ARE TOTAL PUMPING TIMES.
DT (T2CHECK-T1CHECK)= 0.2021D  3 SEC.,    T(2)-T(1)= 0.3880D  1 SEC.
% DIFF. BETWEEN T1CHECK AND T(1) =       97.320596
% DIFF. BETWEEN T2CHECK AND T(2) =       97.413642

V/2 PUMPED (FRAC1) (CGS,BRL,GAL):  0.9557D  7 0.6034D  2 0.2525D  4
V/2 PUMPED (FRAC2) (CGS,BRL,GAL):  0.1089D  8 0.6876D  2 0.2877D  4
```

```
TO IMPROVE THE TIME DISCREPANCY WITHOUT PRINTING THE FLOW RATES, HIT 10
TO PRINT THE FLOW RATES AND IMPROVING THE TIME DISCREPANCY, HIT 20
TO PRINT THE FLOW RATES IF YOU ARE FINISHED WITH THE TIME DISCREPANCY, HIT 30
30
TSLOPE,TEXPONENT =     1.00000     1.00000   , TCEPT,TDIV =     1.00000     10.00000
  , LEAKC =    0.000000 SP=    0.000000

FRAC  1                                                                    -50-
LEAKRATE(X) AS X=0 TO A:
0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0
0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0

FRAC  2
LEAKRATE(X) AS X=0 TO A:
0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0
0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0   0.0000D  0
OUTPUT Q(X) AS X=0 TO A:
0.2500D  1   0.2352D  1   0.2206D  1   0.2064D  1   0.1924D  1   0.1787D  1

0.1653D  1   0.1519D  1   0.1388D  1   0.1257D  1   0.5949D  0
MAX.(QX OUTPUT, % DIF) = 0.1388D  1  -0.44
INPUT QX, AS X=0 TO A:
0.2500D  1   0.2352D  1   0.2206D  1   0.2065D  1   0.1926D  1   0.1790D  1

0.1657D  1   0.1525D  1   0.1394D  1   0.1262D  1   0.5977D  0
HIT 1 IF YOU WANT TO CALL QTEST SUBROUTINE
1

FLOW RATE CONVERGES TO WITHIN          5. PERCENT AFTER     4 ITERATIONS

SLOPE =

.265D  1 0.265D  1 0.526D  0 0.524D  0 0.533D  0 0.535D  0 0.548D  0 0.559D  0 0
.565D  0 0.583D  0 0.593D  0
CEPT =

.289D  1 0.289D  1 0.977D  0 0.975D  0 0.982D  0 0.983D  0 0.989D  0 0.994D  0 0
.995D  0 0.999D  0 0.100D  1
I,DL,DLDT(I):      1 0.613082D  2 0.000000D  0
I,DL,DLDT(I):      2 0.820542D  2 0.424195D  0
I,DL,DLDT(I):      3 0.872289D  2 0.442997D  0
I,DL,DLDT(I):      4 0.903465D  2 0.456314D  0
I,DL,DLDT(I):      5 0.930785D  2 0.468123D  0
I,DL,DLDT(I):      6 0.956267D  2 0.479342D  0
I,DL,DLDT(I):      7 0.979302D  2 0.489821D  0
I,DL,DLDT(I):      8 0.995708D  2 0.499517D  0
I,DL,DLDT(I):      9 0.101095D  3 0.507726D  0
I,DL,DLDT(I):     10 0.102056D  3 0.512621D  0
I,DL,DLDT(I):     11 0.102403D  3 0.514097D  0
MAX % DIF DLDT =      4.2939

HIT 1 IF YOU WANT TO PRINT ALL CALCNS FOR DEBUG
^C^B
*ABORT*
CONSOLE INTERRUPT
ERROR: FROM PROGRAM
X,FLAMLEAK6B
)
```

```
      COMPILER DOUBLE PRECISION
      COMPILER NOSTACK
C*********************************************************************
C*********************************************************************
C     FLAMLEAK6B - THIS PROGRAM SOLVES W(P) AND P(W) EQUATIONS
C                  FOR 1D FLUID FLOW AND ITERATES UNTIL KI = KCB
C                  ALONG PARIMETER. IT ALSO CALCULATES W(Y) PROPERLY,
C                  BUT ONLY CAN BE USED FOR SYMMETRIC STRESSES. LEAKOFF
C                  AND SPURT LOSS ARE INCLUDED.
C     SUBROUTINES : DWIDTH6B,DVLTQ6B2,DRK26B,QTEST6B,HCALC,DLDT
C     FUNCTIONS   : DFUNAM6B,PFUN,DFX,FFX,DASIN,DACOS
C                  SO THESE NEED TO BE LINKED.
C*********************************************************************
C*********************************************************************
C
      DIMENSION C(3),D(9),PXA(22),RXA(22),PXAP(22),VX(2,22),TIME(2)
      DIMENSION PSEC(22),KICHOICE(22,2),DIFCHOICE(22),HP(22),QX(22)
      DIMENSION AREAS(22),W(51)
      INTEGER WFLAG,FFLAG,KIFLAG
      REAL K,KSQ,KPSQ,KIA,KIB,KI1,KI2,KICHOICE,KDIF,M,L,KCA,KCB
      COMMON K,KSQ,KPSQ,KIA,KIB,KI1,KI2,KICHOICE,KDIF,M,KCA,
     C      DIFCHOICE
      COMMON/FNCTN/ FLP,FL,FFLAG
      COMMON/ABLOC/B,C2,HZ,C4,CONST,E,XAC,TOP,NFLOW,DELTA,HNEW,HOLD,
     C      XAOLD,XAINC,HP,PHI,FLAGPHI
      COMMON /SUBVTQ/ IFRAC,VX,TIME,AFAC,ITQ,Q,A,PI,S2MS1,WFLAG,PXA,
     C      N,EMOD,RNU,NODIV
      COMMON /CBLOC/ CEPT(22),SLOPE(22),WIDCON
      COMMON/KVAL/KCB,HSTART,ISEG,DIFCE,HSAV,IPR
      COMMON/QVAL/QX,INQX
      COMMON /SUBWIDTH/ W,AREAS,NUMWID,I,L,H,IBUG,SPXA11,SHP11
      COMMON/DLT/DLDT(12),SHP(12),DT
      COMMON/SUBFUN/RMU,SDLDT(12),DLTPER,IDEBUG,IIFLAG,IDLDT,IDLDTZ
      COMMON/PFUNB/S1RXA(11),S2RXA(12)
      REAL INQX(22)

      CALL OPEN(1,"@DATA",1,IER,128)
      CALL OPEN(2,"@LIST",2,IER,128)
      CALL OPEN(3,"FLAMLK6.OUT",2,IER,128)
      PI = 3.14159
      TOP=2./PI
      PHI=3.*PI/16.
      NUMEQ = 3
      FLAGPHI = 1
C      FLAG TO DETERMINE IF FIRST ACCURATE CALCN OF
C      PHI IN SR DWIDTH
C
C     INPUT VALUES:
C
      WRITE(2,590)
 590  FORMAT(" DO YOU WISH TO CALCULATE 1=TIME, 2=TIME & Q, 3=NEITHER ?")
      READ(1)ITQ
      WRITE(2,601)
 601  FORMAT(" INPUT:  IWRITE, KCB(CGS)(NOTE: KCB<>0 IN THIS PROG)"/
     C "    HZ(ZONE WIDTH IN FT),  Q(BPM), RMU(CP) ")
      READ(1) IWRITE,KCB,HZ,Q,RMU
      WRITE(2,602) IWRITE,     KCB,HZ,Q,RMU
 602  FORMAT(" IWRITE= ",I3,"      KCB= ", E11.3," HZ= ",F10.1,
     C " Q= ",F10.1," MU= ",F10.1)
      WRITE(2,761)
 761  FORMAT("  INPUT:  EMOD(YOUNG'S MODULUS, EG. 1.5E6 PSI), RNU("
     C "POISSON'S RATIO,EG. .1),"/" WIDCON(USUALLY 1-ENGLAND & GREEN)")
      READ(1) EMOD,RNU,WIDCON
      WRITE(2,762) EMOD,RNU,WIDCON
 762  FORMAT("  YOUNG'S MODULUS= ",E12.3," POISS.RATIO= ",F10.2,
     C " WICON=",F5.2)
      WRITE(2,603)
 603  FORMAT(" INPUT: S2-S1(PSI),  A(FT)")
      READ(1) S2MS1,L
      WRITE(2,604) S2MS1,L
 604  FORMAT(" S2-S1= ",F10.1,"   A=  ",F10.1)
```

```
      WRITE(2,63)
 63   FORMAT(" ENTER NO. OF WIDTH CALCNS. IN EACH VERTICAL SECTION
     C USUALLY 10")
      READ(1)NUMWID
      WRITE(3,64)NUMWID
 64   FORMAT(1X,I6," STEPS FOR WIDTH INTERVALS"/)

C
C     CALCULATE QX FROM PREVIOUS QX VALUES
C
      WRITE(2,450)
 450  FORMAT(" PLEASE INPUT Q1,Q2,Q3,= VALUES OF QX AT XA=0,XA2,1.0"/
     C " GOOD STARTING VALUES ARE 1.0,0.2, 0.1 ")
      READ(1)Q1,Q2,Q3
      WRITE(2,478)
478   FORMAT(" PLEASE INPUT XA2 = XA VALUE CORRESPG. TO Q2")
      READ(1) XA2

      XA1=0.
      XA3=1.0

C     INITIALIZE SLOPE,CEPT VALUES
      SLOP1 = (Q1 - Q2)/(XA2 - XA1)
      CEP1 = Q1 + SLOP1*XA1
      SLOP2 = (Q2 - Q3)/(XA3 -XA2)
      CEP2 = Q2 + SLOP2*XA2
      J2 = (1. - XA2) / .1  + 1
      J3 = J2 + 1
      DO 561 J=1,J2
      SLOPE(J) = SLOP2
 561  CEPT(J) = CEP2
      DO 562 J=J3,11
      SLOPE(J) = SLOP1
 562  CEPT(J) = CEP1
C     SLOPE(1),CEPT(1) CORRESPOND TO THE SEGMENT AT X=A
      WRITE(2,460) Q1,Q2,Q3,(SLOPE(J),J=1,11), (CEPT(J),J=1,11)
      WRITE(3,460) Q1,Q2,Q3,(SLOPE(J),J=1,11), (CEPT(J),J=1,11)
 460  FORMAT(" Q1=",F6.3,"Q2=",F6.3,"Q3=",F6.3/
     C " SLOPE = ",11(F6.3,1X)/" CEPT = ",11(F6.3,1X))

C     ??????????????????????????
C     THE ABOVE IS NOT PROGRAMMED FOR N=21, ONLY N=11
      WRITE(3,6661)HZ,Q,RMU,KC8,EMOD,RNU,S2MS1,L,WIDCON,DELTA
 6661 FORMAT(" FLAMOCO OUTPUT"//" INPUT VALUES: HZ,Q,RMU,K12,"
     C      "EMOD,RNU,S2-S1,L,WIDCON,DELTA"/3F10.1,2(E10.3,2X),/5F10.2)


      WRITE(2,588)
 588  FORMAT("  DO YOU WANT TO RE COMPUTE AND LIST ALL KI'S IN RANGE
     C U=1 FOR CHECKING    PURPOSES?"/"  IF YES, HIT 1 ")
      READ(1)  ICOMP
      WRITE(2,571)
 571  FORMAT(" INPUT PERCENT CONVERGENCE DESIRED FOR Q(X) USUALLY 20")
      READ(1)QPER
      WRITE(2,572)QPER
      WRITE(3,572)QPER
 572  FORMAT(" Q(X) WILL CONVERGE TO ",F10.0," PERCENT ")
      WRITE(2,573)
 573  FORMAT(" INPUT % CONVERGENCE DESIRED FOR HALF-HEIGHT ITERATION"
     C          " USUALLY 1")
      READ(1) DIFCE
      WRITE(2,574) DIFCE
 574  FORMAT(" HALF-HEIGHT CALCN WILL CONVERGE TO ",F12.3," % ")
      WRITE(2,575)
 575  FORMAT(" INPUT % CONVERGENCE DESIRED FOR DL/DT ITERATION.")
      READ(1) DLTPER
      WRITE(2,576) DLTPER
 576  FORMAT(" DL/DT CALCN WILL CONVERGE TO ",F12.3," % ")
```

```
C       CONVERT UNITS
        S2MS1 = 68940. * S2MS1
        HZ= 30.48 * HZ
        RP=HZ/2.
        EMOD=EMOD*68940.
        G=EMOD/(2.*(1.+RNU))
        W=W*2640.
        RMU=RMU*.01
        A=L*30.48
        C1=(1.-RNU) / G
        C4=-12.*W*RMU
        C4=C4/C1**3
        C2=S2MS1*2./PI
        DO 1 I=1,N
        DLDT(I) = 0.
 1      SDLDT(I) = 0.
        ITERD = 0
        IFRAC = 1
        ITERW = 0
        ICYCLE = 0
        IIFLAG = 0
        IYZ=0

        WRITE(2,611)
 611    FORMAT(//" INPUT: NDIV (MUST BE MULTIPLE OF N*(N-1) ) "/
     C  " XAZ(EG. .999),"
     C  " N  (=11, NOT COMPLETELY PROGRAMMED FOR N=21 YET)")
        READ(1) NDIV,XAZ,N
        WRITE(2,613) NDIV,XAZ,N
 613    FORMAT(   " NDIV= ",I5," XAZ= ",F10.6," N= ",I3)

C       INPUT DLDT VALUES IF USING CLEARY'S FIT
        WRITE(2,13)
 13     FORMAT(/" HIT 1 IF YOU WANT TO INPUT DLDT VALUES ")
        READ(1) IDLDT
        IF(IDLDT.NE.1) GOTO 14
           WRITE(2,131) N
 131       FORMAT(" INPUT VALUES FOR DLDT(I), FOR I=1,",I3)
           READ(1) (DLDT(I),I=1,N)
           WRITE(3,614) (DLDT(I),I=1,N)
 614       FORMAT(" INPUT DLDT(I=1,N) VALUES :"/6(F5.2)/)
           DO 132 I=1,N
 132          SDLDT(I)=DLDT(I)
        IF (DLDT(1).NE.0) GOTO 14

C       IDLDTZ=1 SETS ALL DLDT(1) TO 0 FOR ALL FRAC
C       THIS MEANS FOR FRAC1, DLDT=0 AT START OF FIRST SEGMENT FOR
C       ENTIRE SEGMENT. THIS AVOIDS TOO LARGE DLDT WHICH GIVES BOTH
C       FFUN NEGATIVE RUNTIME ERROR.
        WRITE(2,133)
 133    FORMAT(/" HIT 1 IF YOU WANT TO SET ALL DLDT(1) TO 0 ")
        READ(1) IDLDTZ

C       INITIALIZE S1RXA,S2RXA
 14     N1=N-1
        DO 188 I=1,N
           S1RXA(I) = XAZ - (I-1)*(1./N1)
 188       S2RXA(I) = XAZ - (I-1)*(1./N)

C       BEGIN NEW ITERATION

 9      ICYCLE=ICYCLE+1
        WRITE(2,591)
 591    FORMAT(" INPUT:  NRES,MFAC   (FOR USE IN RK2) ")
        READ(1) NRES,MFAC
        WRITE(2,592) NRES,MFAC
 592    FORMAT(" NRES = ",I4,"   MFAC = ",I4)

C       INPUT FOR H MODIFICATION
C       BEGIN NEW G(X) ITERATION
```

```
 93     IHST=0
        IBUG=0
C       PRINT OUT DETAIL CALCULATIONS FOR DEBUG?
        WRITE(2,400)
 400    FORMAT(/" HIT 1 IF YOU WANT TO PRINT ALL CALCNS FOR DEBUG ")
        READ(1) IDEBUG

C       TO RESET PHI

        PHI = 3.*PI/16.
        WRITE(2,6231)
 6231   FORMAT(" INPUT:  HSTART/HZ (CAN BE > OR < 1)")
        READ(1)  HSTART
        WRITE(2,6232)   HSTART
 6232   FORMAT(" HSTART/HZ = ",F10.4)
        HOLD=HSTART
        EL = HSTART * HZ/2.
        PBAR = KCB/DSQRT(PI*EL) + TOP*S2MS1*ACOS(HZ/(2.*EL))
        PXA(1) = PBAR
C       THIS IS OUR WAY OF FINDING PXA(1)
        YZ = PXA(1)
        PBARPSI = PBAR / 68940.
        WRITE(2,763) PBARPSI
 763    FORMAT(" INITIAL PRESSURE AT X=A IS ",F10.1," PSI ")

        IF (IFRAC.EQ.0)WRITE(3,6662)NRES,MFAC,NDIV,N,XAZ,YZP,DIFFAC,B
 6662   FORMAT(" NRES,MFAC,NDIV,N,XAZ,YZ,DIFFAC,B"/4I10,2F10.3/2F10.1)
        AFAC = 1. + 1./(N-1)
        IF(IFRAC.EQ.1)NTEMP=N
        NUMSEG=N-1
        IF(IFRAC.EQ.2) NTEMP=N+1
        IF(IFRAC.EQ.2) NUMSEG=N
        XAINC=1./FLOAT(NUMSEG)
        STEP=1./FLOAT(NDIV)

C       CALCULATE SOLN OF D.E. ARISING FROM  W(P) AND P(W) EQNS, USING
C       RUNGE - KUTTA
        KDIV= NDIV/NUMSEG
        ISTART=1
        H2= STEP/2.
        STEP=STEP/FLOAT(MFAC)
        H2=H2/FLOAT(MFAC)
        KDIV=KDIV*MFAC
C       THIS UPS THE NO OF STEPS IN 1ST SEGMENT  BY FACTOR MFAC
C       FOR MORE ACCURATE INTEGRATION NEAR XA=1.

        WRITE(2,92)
 92     FORMAT(" IF YOU WANT TO CHANGE HSTART,  HIT 1 ")
        READ(1) IHST
        IF(IHST.EQ.1) GO TO 93
        ISEG = 2

        CALL RK2( A,STEP,XAZ,YZ,NDIV,N,PXA,RXA,NRES,MFAC,IFRAC,H2,KDIV,ISEG)
        IF(IWRITE.EQ.1) WRITE(2,615) C1,C2,C4,STEP
 615    FORMAT(" C1,C2,C4 =  ",3E12.3,"STEP=",E12.3/)

C       TO CALCULATE NEW PHI

           I = ISEG
           H = HP(I)*HZ
           FLP = HZ/H
           L = H/2.

           CALL WIDTH

C       NOW RESET STEP, KDIV
        STEP=STEP*FLOAT(MFAC)
        H2=H2*FLOAT(MFAC)
        KDIV=KDIV/MFAC
```

```
      DO 336 ISEG=3,NTEMP
      CALL RK2(A,STEP,XAZ,YZ,NDIV,N,PXA,RXA,NRES,MFAC,IFRAC,P2,KDIV,ISEG)
C     TO CALCULATE NEW PHI

        I = ISEG
        H = HP(I)*HZ
        FLP = HZ/H
        L = H/2.

        CALL WIDTH

      IF(ISEG.EQ.NTEMP) GO TO 334
 336  CONTINUE

 334  B=HP(NTEMP) * HZ/2.
      IF(IWRITE.EQ.1)WRITE(2,620)(PXA(I), I=1,NTEMP)
 620  FORMAT(1X," PXA(I):",6E12.3/)
      IF(IWRITE.EQ.1)WRITE(2,630)(RXA(I), I=1,NTEMP)
 630  FORMAT(1X,/" RXA(I):",6E12.3/)
      IF(IWRITE.EQ.1) WRITE (2,634)
 634  FORMAT("  ONE OF THE RXA'S MUST BE REAL CLOSE TO 0.    IS IT ????????")

      AFT=A/30.48
      BFT=B/30.48
      DO 1111 I=1,NTEMP
1111  PXAP(I)=PXA(I)/68940.
      WRITE(2,500)AFT,BFT
 500  FORMAT(" A= ",F10.1,"  B= ",F10.1)
      WRITE(2,501)(PXAP(II),II=1,NTEMP)
 501  FORMAT(" PXA: (IN PSI)"/(6F10.2/))

      QB=Q/2640.
      SRMU=RMU/.01
      WRITE(2,627) QB,SRMU,HFT,BFT,AFT
 627  FORMAT(" ITERATION SUCCESSFUL ON KI          "/
     C "  Q= ",F10.1," RMU= ",F10.1,"H= ",F10.1," B= ",F10.1,
     C "A = ",F10.1)
      WRITE(2,337)  (HP(I),I=1,NTEMP)
 337  FORMAT(" SHAPE OF FRAC H/HZ FROM XA=1 TO 0 "/(6F12.4/))
      WRITE(3,337)  (HP(I),I=1,NTEMP)
      WRITE(3,501) (PXAP(II),II=1,NTEMP)
C     NOW SOLVE FRAC1 HEIGHTS FOR DLDT CALCNS
      IF (IFRAC.EQ.2) GOTO 777
      DO 778 I=1,N
 778  SHP(I) = HP(I)

C     CHECK WHICH CALCULATIONS ARE TO BE DONE - TIME, G, NEITHER?

 777  IF (ITG-2) 770,750,9
 750  CONTINUE

C     DIVIDE FRAC INTO N SEGMENTS AND CALC VOLUME OF EA SEGMENT.
C     CALL THIS FRAC1 THEN ITERATE FOR FRAC2 WHICH IS TO BE ONE
C     SEGMENT LONGER.

      IF (IFRAC.EQ.2) GO TO 770
      CALL VLTG
      ICYCLE = 0
      IF(IIFLAG.EQ.0) GOTO 769
         DIFPXA11=(PXA(11)-SPXA11)/PXA(11) * 100.
         DIFHP11 =(HP(11)-SHP11)/HP(11) * 100.
         WRITE(2,768) DIFPXA11,DIFHP11
 768     FORMAT(/" % DIF IN PXA(11),HP(11) WITH LAST FRAC1 :",2F10.4)
 769  SPXA11=PXA(11)
      SHP11=HP(11)
      GO TO 9

C     NOW CALC THE VOLUME OF EA OF THE N+1 SEGMENTS FOR FRAC2. FROM
C     SEGMENT VOL'S FOR FRACS 1&2 CALC FLOW RATES DOWN THE FRAC.
```

```
770     CONTINUE
        CALL VLTQ
        WRITE(2,100)
100     FORMAT(" HIT 1 IF YOU WANT TO CALL QTEST SUBROUTINE ")
        READ(1) NQTEST
        IF (NQTEST.NE.1) GOTO 101
        CALL QTEST (N,QPER,ITERQ)
101     CALL DLOTCAL (ITERQ)
        IIFLAG = IIFLAG + 1
        IFRAC=1
C       SET A TO THE ORIGINAL VALUE BEFORE STARTING THE NEXT ITERN.
        A = A/AFAC
        GO TO 93
        END
```

```
        COMPILER NOSTACK
        COMPILER DOUBLE PRECISION
C
C*********************************************************************
C       THIS SUBROUTINE PERFORMS RUNGE-KUTTA SOLUTION OF
C       DIFFERENTIAL EQN ARISING FROM W(P) AND P(W) EQNS.
C       WHERE W(P) IS CORRECTED 2D EQN. AND P(W) IS 1D
C       FLUID FLOW EQN.
C       IT'S CALLED FROM FLSHAH2
C       (DRK26B)
C*********************************************************************
        SUBROUTINE RK2(A,STEP,XAZ,YZ,NDIV,N,PXA,RXA,NRES,MFAC,IFRAC,H2,
     C                 KDIV,ISEGA)
        COMMON/ABLOC/B,C2,H2,C4,CONST,E,XAC,TOP,NFLOW,DELTA,HNEW,HOLD,
     C  XAOLD,XAINC,HP,PHI,FLAGPHI
        COMMON/KVAL/KCB,HSTART,ISEG,DIFCE,HSAV,IPR
        REAL KCB
        DIMENSION PXA(22),RXA(22),HP(22)

        IF(ISEG.GE.3) GO TO 20
        IPR=0
        RXA(1)=XAZ
        Y = YZ
        XA=XAZ
           DO 11 J=1,KDIV
              T1=-STEP*FUN(XA,Y,1)
              T2=-STEP*FUN(XA-H2,Y+T1/2.,2)
              T3=-STEP*FUN(XA-H2,Y+T2/2.,3)
              T4=-STEP*FUN(XA-STEP,Y+T3,4)
C             ALL STEPS ARE -VE COZ WE ARE INTEGRATING BACKWARDS ALONG
C             THE X AXIS  (DY=-DX * F(X,Y) )
              Y=Y+ A* (T1+2.*T2 + 2.*T3 + T4)/6.
C             THIS *A IS DONE HERE INSTEAD OF IN FUN, COZ ALL FUN( )
C             HAVE TO BE MULTIPLIED BY A.
              IF(IPR.GT. 5) GO TO 31
                 WRITE(2,55) STEP,XA,T1,T2,T3,T4,Y
 55              FORMAT(" STEP= ",E12.3," XA= ",E12.3," T1,T2,T3,T4=  ",
     C           4E12.3," Y= ",E20.8)
                 IPR=IPR+1
 31           CONTINUE
 11        XA=XA-STEP
           PXA(2) = Y
           HP(2) = HSAV
           RXA(2) = XA
```

```
      RETURN
20    CONTINUE
      ITEMP=ISEG-1
      XA=RXA(ITEMP)
      Y=PXA(ITEMP)
            DO 1 J=1,KDIV
               T1=-STEP*FUN(XA,Y,1)
               T2=-STEP*FUN(XA-H2,Y+T1/2.,2)
               T3=-STEP*FUN(XA-H2,Y+T2/2.,3)
               T4=-STEP*FUN(XA-STEP,Y+T3,4)
C              ALL STEPS ARE -VE COZ WE ARE INTEGRATING BACKWARDS ALONG
C              THE X AXIS  (DY=-DX * F(X,Y) )
               Y=Y+ A* (T1+2.*T2 + 2.*T3 + T4)/6.
C              THIS *A IS DONE HERE INSTEAD OF IN FUN, COZ ALL FUN( )
C              HAVE TO BE MULTIPLIED BY A.
               IF(IPR.GT. 5) GO TO 3
                  WRITE(2,55) STEP,XA,T1,T2,T3,T4,Y
                  IPR=IPR+1
3              CONTINUE
1           XA=XA-STEP
            PXA(ISEG) = Y
            HP(ISEG) = HSAV
         RXA(ISEG)= XA
      RETURN
      END
```

```
      COMPILER DOUBLE PRECISION
      COMPILER NOSTACK
C*****************************************************************
C     SUBROUTINE DWIDTH68, CREATED FROM MODIFICATION OF CLOSEWID10
C     ON 3/28/84 FOR NEWTONIAN FLOW IN FLAMLEAK68 PROG.
C     THIS SUBROUTINE CALCULATES WIDTH BY THE CLOSE METHOD AND
C     FROM THE WIDTH CALCULATES THE AREA.
C*****************************************************************
      SUBROUTINE WIDTH
      DIMENSION VX(2,22),T(2),AREAS(22),W(51),PXA(22),HP(22)
      REAL L
      INTEGER FFLAG,WFLAG
      COMMON/SUBVTU/IFRAC,VX,T,AFAC,ITU,U,A,PI,S2MS1,WFLAG,PXA,N,
     C      EMOD,RNU,NODIV
      COMMON/ABLOC/B,C2,HZ,C4,CONST,E,XAC,TOP,NFLOW,DELTA,HNEW,
     C      HOLD,XAOLD,XAINC,HP,PHI,FLAGPHI
      COMMON/SUBWIDTH/W,AREAS,NUMWID,I,L,H,IBUG,SPXA11,SHP11
      COMMON/FNCTN/FLP,FL,FFLAG
      COMMON/CBLOC/CEPT(22),SLOPE(22),WIDCON

C INITIALIZE VARIABLES BEFORE ENTERING W LOOP
      PFMS1= PXA(I)
      WINC = 1./NUMWID
      IENDNUMWID = NUMWID

C CALCULATE W FOR 'NUMWID' INTERVALS OF THE HALF-HEIGHT
      DO 200 K = 1,IENDNUMWID
      FL = WINC*(K-1)
      WP = (PI/2.* (PFMS1) - S2MS1*ACOS(FLP)) * DSQRT(1.-FL**2)

C TEST H, HZ RELATIONSHIP. NOTE: H IS ALWAYS >HZ.
      IF (H.GT.HZ)GO TO 19
         AREA2=0.0
         GO TO 20

 19   IF (FL - FLP) 10,20,30
```

```
 10     CONTINUE
C
C          CASE I: BELOW THE EDGE OF THE PAYZONE
C                  FL.LT.FLP
C
        GFL = DSQRT(FLP**2 - FL**2)
        GOTO 40
 20     CONTINUE
C
C IF FL=FLP, THEN SET FL=FLP+.001
C OTHERWISE WE GET A ZERO IN THE DENOM
C
        FL = FL + .001
 30     CONTINUE
C
C          CASE II: ABOVE THE EDGE OF THE PAYZONE
C                  FL.GE.FLP
C
        GFL = DSQRT(FL**2 - FLP**2)
 40     WA = FLP* DLOG((DSQRT(1.-FL**2)+DSQRT(1.-FLP**2))/GFL)
        WB = FL *DLOG((FLP*DSQRT(1.-FL**2)+FL*DSQRT(1.-FLP**2))/GFL)

        W(K) = (WP + (S2MS1) * (WA-WB))* 8.*(1-RNU**2)*L
     C     /(PI*EMOD)
        W(K) = W(K) * WIDCON
 200    CONTINUE
C
C       DEFINE TOP WIDTH TO BE 0.0
C
        KK = NUMWID + 1
        W(KK) = 0.0
C
C       CALCULATE THE TOTAL AREA.
C
        WTOT = 0.0
        DO 250 K=1,NUMWID
           KK = K + 1
           WTOT = WTOT + (W(K) + W(KK))/2.
 250    CONTINUE
        AREAS(I) = 2. * WINC * WTOT * L

C       CALCULATE PHI
        DELTAY = 1./NUMWID
        WNOT = W(1)
        NUMWP1=NUMWID+1
        SUM=0.5
        DO 275 K=2,NUMWP1
 275    SUM = SUM + (W(K)/WNOT)**3
        PHI = DELTAY*SUM

        IF(IBUG.EQ.1) GOTO 500
        WRITE(2,285) PHI,DELTAY,H
 285    FORMAT(" PHI=",E12.4," DELTAY=",E12.4," H=",E12.4)

C       CHANGE IENDNUMWID SO ALL NUMWID + 1 WIDTHS ARE PRINTED OUT
C
        IENDNUMWID = NUMWID + 1

        WRITE(2,300)NUMWID
        WRITE(3,300)NUMWID
 300    FORMAT(1X," WIDTH, BROKEN DOWN IN ",I4," INCREMENTS")
        WRITE(2,350)(W(J),J=1,IENDNUMWID)
        WRITE(3,350)(W(J),J=1,IENDNUMWID)
 350    FORMAT(1X,(6E10.3,1X)/)
        WRITE(2,400)I,AREAS(I)
        WRITE(3,400)I,AREAS(I)
 400    FORMAT(1X," I = ",I2," , AREA(I) =  ", F10.4)
 500    CONTINUE
        RETURN
        END
```

```
      COMPILER NOSTACK
      COMPILER DOUBLE PRECISION
C*****************************************************************
C     SUBROUTINE DVLTQ - CALC VOLUME, LEAKOFF, TIME, AND Q
C     LAST MODIFIED 7/25/84  (FOR FLAMLEAK6B).
C     -- DVLTQ6B1 + LEAKOFF VOLUME CORRECTION AND RESET ANY -VE
C        QX'S TO DV/DT FOR INPUT QX'S.
C
C*****************************************************************
      SUBROUTINE VLTQ
      DIMENSION VX(2,22),T(2),QX(22),AREAS(22),W(51),PXA(22),HP(22),
      DIMENSION TAU(21),TTOTLEAK(2),AVTAU(2),FX(2)
      INTEGER FFLAG,WFLAG
      REAL INQX(22),L,LEAKC,LEAKRATE(2,21),LEAKVOLUME(2,21)
      COMMON/SUBVTQ/IFRAC,VX,T,AFAC,ITQ,Q,A,PI,S2MS1,WFLAG,PXA,N,
     C      EMOD,RNU,NUDIV
      COMMON/ABLOC/B,C2,HZ,C4,CONST,E,XAC,TOP,NFLOW,DELTA,HNEW,HOLD,
     C      XAOLD,XAINC,HP,PHI,FLAGPHI
      COMMON/SUBWIDTH/W,AREAS,NUMWID,I,L,H,IBUG,SPXA11,SHP11
      COMMON/FNCTN/FLP,FL,FFLAG
      COMMON/CBLOC/CEPT(22),SLOPE(22),WIDCON
      COMMON/SUBLEAK/A1,VTOT1
      COMMON/QVAL/QX,INQX
      COMMON/DLT/DLDT(12),SHP(12),DT
      COMMON/DSLOP/TSLOPE

      IF (IFRAC.EQ.2) NUMSEG = N
      IF (IFRAC.EQ.2) GO TO 3
      NUMSEG = N-1

C     CALCULATE  AREA(FLP)

 3    XA = 1.0
      XAINCR=1./NUMSEG
      IF (IFRAC.EQ.1) AFAC = 1. + XAINCR

C     LOOP OVER THE VERTICAL SECTIONS CALCULATING AREAS
C     "AREAS(I)" IS THE AREA OF THE I'TH VERTICAL SECTION IN CGS

      AREAS(1) = 0.0
      IEND = NUMSEG + 1
      DO 100  I=2,IEND
      H = HP(I) * HZ
      FLP= HZ/H
         L = H/2.


C     CALL WIDTH SUBROUTINE TO CALCULATE WIDTH AND AREA FOR THIS SEGMENT
C     IBUG=1 COMMENTS OUT REDUNDANT WIDTH OUTPUTS
         IBUG=1
         CALL WIDTH
 100  CONTINUE
      WRITE(3,1500)
 1500 FORMAT(" AREAS(X) AS X=A TO 0 :")
      WRITE(3,2000) (AREAS(I),I=1,N)

C     NOW, CALCULATE VOLUME BY INTEGRATING THE AREA OVER X
C           V = 2 * INTEGRAL(A(X)DX) FROM X=0 TO X=A

      VTOT = 0.0
      DO 300  I = 1, NUMSEG
         II= I + 1
         VX(IFRAC,I) = XAINCR*A * (AREAS(I)+AREAS(II))/2.
 300     VTOT  = VTOT + VX(IFRAC,I)

C SAVE IFRAC = 1 VALUES FOR LATER USE IN LEAKOFF CALCULATIONS
      IF (IFRAC.NE.1) GOTO 350
         A1 = A/30.48
         VTOT1 = VTOT
      GOTO 360
 350  VTOT2=VTOT
 360  CONTINUE
```

```
C                                   Q IS THE FLOW RATE DOWN A 1-SIDED FRAC &
C                                   VTOT IS THE VOLUME OF A 1-SIDED FRAC
      WRITE(2,1900)
      WRITE (3,1900)
 1900 FORMAT(" VOLUME(x) AS X=A TO 0 : (CGS)")
      WRITE(3,2000)(VX(IFRAC,I),I=1,N)
      WRITE(2,2000)(VX(IFRAC,I),I=1,N)
 2000 FORMAT(7E10.4/)

C     CHECK IF Q CALC'NS ARE TO BE PERFORMED (ITQ=2)

      IF (ITQ.NE.2) GO TO 999

C     IF FRAC2 HAS NOT BEEN ITERATED FOR THEN ITERATE FOR FRAC2

      IF (IFRAC.EQ.2) GO TO 750
         A = AFAC * A
         IFRAC = IFRAC + 1
         GO TO 999
 750  CONTINUE

C     NOW CALC LEAKOFF RATE AND VOLUME FOR EACH SEGMENT
C
C INPUT VALUES FOR L(T) GRAPH AND LEAKOFF COEFFICIENT
      WRITE(2,4000)
 4000 FORMAT(" PLEASE INPUT: TSLOPE,TCEPT,LEAKC,SP"/
     C " TSLOPE=SLOPE OF LOG L(FT) VS LOG T(SEC) PLOT"/
     C " TCEPT = INTERCEPT OF PLOT AT T=1SEC, IE LOG T=0. THE"
     C " INTERCEPT IS THE LOG VALUE"/
     C " LEAKC = LEAKOFF COEFFICIENT C IN CGS"/
     C " SP = SPURT LOSS COEFFICIENT IN CGS")
      READ(1)TSLOPE,TCEPT,LEAKC,SP

C COMPUTE NEEDED VALUES FROM INPUT VALUES
      TEXPONENT = 1./TSLOPE; =1/N FOR A(T)=M*T*N
      TDIV = 10.**TCEPT; =M

      WRITE(2,4010)TSLOPE,TEXPONENT,TCEPT,TDIV,LEAKC,SP
 4010 FORMAT(" TSLOPE,TEXPONENT = ",2(F10.5,2X),", TCEPT,TDIV = ",
     C 2(F10.5,2X),", LEAKC = ",F10.6," SP= ",F10.6)

C CALC TIMES FROM L/T GRAPH
      NM1 = N - 1
      T(1) = (A1/TDIV)**TEXPONENT
      HL = A/30.48
      T(2) = (HL/TDIV)**TEXPONENT
C     CALC TAU TIMES. TAU = TIME IN SEC WHEN THE FRAC HITS MIDDLE
C     OF SEGMENT.
      DO 75 I=1,N
         HL = A1 * FLOAT(I-1)/(FLOAT(N)-1.)
         TAU(I) = (HL/TDIV)**TEXPONENT
 75   CONTINUE
      TAU(12) = T(2)
      NP1 = N + 1
      WRITE(2,4040)(TAU(I),I=1,NP1)
 4040 FORMAT(" TAU(I) AS I=1 TO N+1 ",/(6(E11.4,2X)/))

C CALC LEAKOFF RATE, LEAKOFF VOLUME
      C1 = A1 * 30.48 /(N-1)
      CC = C1 * LEAKC * HZ
      SPL = 2.*SP*HZ
      TC = 4.*HZ*LEAKC*TSLOPE
      DO 86 J=1,2
C        TOTLEAK=0.
         NX=NM1+(J-1)
         TX=T(J)
         AVTAU(J) = (TAU(I)**(1.-TSLOPE) + TAU(I+1)**(1.-TSLOPE))/2.
         SPT = 2.*SP*HZ*TSLOPE*TDIV/AVTAU(J)
         DO 85 I=1,NX
            R1 = DSQRT(TX-TAU(I))
            R2 = DSQRT(TX-TAU(I+1))
            LEAKRATE(J,I) = CC * (1./R1 + 1./R2)
 85      IF(I.EQ.NX) LEAKRATE(J,I)= TC*TDIV*R1/AVTAU(J)*30.48 + SPT
 86   CONTINUE
```

```
C IMPROVE TOTLEAK AS ON PAGE G21
      DO 111 J=1,2
         FX(J) = FFX(0.0,1.0,40,J)
C        FFX COMES FROM NUMERICAL INTEGRATION
 111  TTOTLEAK(J) = (TC*SQRT(T(J))*FX(1)+SPL)*A1*(1.+(J-1)/10.)*30.48
      WRITE(2,4050)TTOTLEAK(1),VTOT1,TTOTLEAK(2),VTOT2
 4050 FORMAT(" TOTLEAK1= ",E11.4,", VTOT1= ",E11.4/" TOTLEAK2= ",
     C E11.4,", VTOT2= ",E11.4)

C CALC TOTAL VOLUME,T1CHECK,T2CHECK
      TOTVOL1 = VTOT1 + TTOTLEAK(1)
      TOTVOL2 = VTOT2 + TTOTLEAK(2)
      T1CHECK = TOTVOL1/Q
      T2CHECK = TOTVOL2/Q
      VBRL1 = TOTVOL1/(2640.*60.)
      VBRL2 = TOTVOL2/(2640.*60.)
      VGAL1 = TOTVOL1/3785.
      VGAL2 = TOTVOL2/3785.
      T1MIN = T1CHECK/60.
      T2MIN = T2CHECK/60.
      DT = T2CHECK - T1CHECK
      T2MT1 = T(2)-T(1)
      WRITE(2,2500) T(1),T(2),T1CHECK,T1MIN,T2CHECK,T2MIN,DT,T2MT1
 2500 FORMAT(/" T(1)= ",E11.4," T(2)= ",E11.4/
     C " T(1),T(2) COME FROM RATE OF FRACTURE LENGTH EXPANSION A(T)"/
     C " T1CHECK:(SEC,MIN)",2E11.4," T2CHECK:(SEC,MIN)",2E11.4/
     C " T1CHECK,T2CHECK= TIME CALCD FROM (FRAC VOL + LEAKOFF VOL)/"
     C " (INLET FLOW RATE)"/
     C " THESE TIMES ARE TOTAL PUMPING TIMES."/
     C " DT (T2CHECK-T1CHECK)=",E11.4," SEC.,   T(2)-T(1)=",E11.4,
     C " SEC.")
      T1PER = 100. *(T1CHECK - T(1))/T1CHECK
      T2PER = 100. *(T2CHECK - T(2))/T2CHECK
      WRITE(2,2550) T1PER,T2PER
 2550 FORMAT(" % DIFF. BETWEEN T1CHECK AND T(1) = ",F14.6/
     C " % DIFF. BETWEEN T2CHECK AND T(2) = ",F14.6)
      WRITE(2,2551) TOTVOL1,VBRL1,VGAL1,TOTVOL2,VBRL2,VGAL2
 2551 FORMAT(/" V/2 PUMPED (FRAC1) (CGS,BRL,GAL): ",3E11.4/
     C " V/2 PUMPED (FRAC2) (CGS,BRL,GAL): ",3E11.4)

      WRITE(2,660)
 660  FORMAT(/" TO IMPROVE THE TIME DISCREPANCY WITHOUT PRINTING THE"
     C " FLOW RATES, HIT 10"/" TO PRINT THE FLOW RATES AND"
     C " IMPROVING THE TIME DISCREPANCY, HIT 20"/" TO PRINT THE FLOW"
     C " RATES IF YOU ARE FINISHED WITH THE TIME DISCREPANCY, HIT 30")
      READ(1) IMP
      IF (IMP.EQ.10) GOTO 749
      IF (IMP.NE.30) GOTO 87
         WRITE(2,4010)TSLOPE,TEXPONENT,TCEPT,TDIV,LEAKC,SP
C        WRITE(3,4035)HL
         WRITE(3,4040)(TAU(I),I=1,NK)
         WRITE(3,4050)TTOTLEAK(1),VTOT1,TTOTLEAK(2),VTOT2
         WRITE(3,2500) T(1),T(2),T1CHECK,T1MIN,T2CHECK,T2MIN,DT,T2MT1
         WRITE(3,2551) TOTVOL1,VBRL1,VGAL1,TOTVOL2,VBRL2,VGAL2
         WRITE(3,2550) T1PER,T2PER
 87   DO 88 J=1,2
         NX=NM1+(J-1)
         WRITE(2,2590) J
         IF(IMP.EQ.30) WRITE(3,2590) J
 2590    FORMAT(/" FRAC ",I2)
         WRITE(2,2600) (LEAKRATE(J,I),I=1,NX)
         IF(IMP.EQ.30) WRITE(3,2600) (LEAKRATE(J,I),I=1,NX)
 2600    FORMAT(" LEAKRATE(X) AS X=0 TO A: "/ (6(E11.4,2X)))
 88   CONTINUE

C     NOW CALC FLOW RATE DOWN EACH SEGMENT:OUTPUT Q(X) AS X=0 TO A

      QX(1) = Q
      IV1 = N-1
      IV2 = N
```

```
      DO 755  I=2,NM1
         M = I-1
         QX(I) = QX(M) - (VX(2,IV2)-VX(1,IV1))/DT - LEAKRATE(1,M)
         IV1 = IV1 - 1
755      IV2 = IV2 - 1

C IMPROVE QX(A) AS ON PAGE G21
      QC1 = TC/2.*A1*30.48*FX(2)/SQRT(T(1))
      QC2 = SPL*TSLOPE*A1*30.48/T(1)
      QX(11) = QX(1) - (VTOT2-VX(2,1)-VTOT1)/DT - QC1 - QC2

      DO 756  I=1,N
756      QX(I) = QX(I)/2640.
      WRITE(2,3000)  (QX(I),I=1,N)
      IF(IMP.EQ.30) WRITE(3,3000) (QX(I),I=1,N)
3000  FORMAT(" OUTPUT Q(X) AS X=0 TO A:"/ (6(E11.4,2X)/))

C RESET ANY -VE QX TO QXLAST
      QXLAST = 0.0
      NEG = 0
      DO 12 I=1,N
         IF(QX(I).GT.0.) GOTO 12
         NEG = 1
         QX(I) = QXLAST
12    CONTINUE
      IF(NEG.EQ.1) WRITE(2,13) QXLAST
13    FORMAT(/" ERROR ...... ERROR ...... ERROR ......"/
     C        " ERROR ...... ERROR ...... ERROR ......"/
     C  " FOR COMPUTING NEW INPUT QX, ANY -VE QX IS RESET TO :",F5.2/)
      IF(NEG.EQ.1.AND.IMP.EQ.30) WRITE(3,13) QXLAST

C REFORM AND WRITE OUT INPUT QX
      INQX(1) = Q/2640.
      QNOT = INQX(1)
      XA = 0.0
      XAINCR = 1./(N-1)

      DO 3100 I=2,N
      XA=XA+XAINCR
         ISE = 13 - I
         INQX(I) = QNOT*(-SLOPE(ISE)*XA + CEPT(ISE))
3100  CONTINUE

C PRINT MAX-QX OUTPUT AND ITS %DIF FROM INQX
      QMAX = 0.
      DO 758 I=1,N
         QXMQI = QX(I)-INQX(I)
         IF(ABS(QXMQI).LE.ABS(QMAX)) GOTO 758
            QMAX = QXMQI
            KMAX = I
758   CONTINUE
      QMAXPER = QMAX/INQX(KMAX) * 100.
      WRITE(2,757) QX(KMAX),QMAXPER
      IF(IMP.EQ.30) WRITE(3,757) QX(KMAX),QMAXPER
757   FORMAT(" MAX.(QX OUTPUT, % DIF) =",E11.4,F7.2)

      WRITE(2,3200)(INQX(I),I=1,N)
      IF(IMP.EQ.30) WRITE(3,3200)(INQX(I),I=1,N)
3200  FORMAT(" INPUT QX, AS X=0 TO A:"/(6(E11.4,2X)/))

      IF(IMP.EQ.30) GOTO 999
749   WRITE(2,661)
661   FORMAT(//"THIS IS A NEW ITERATION ON A(J)")
      GOTO 750
999   RETURN
      END
```

```
      COMPILER NOSTACK
      COMPILER DOUBLE PRECISION
C
C*****************************************************************
C     THIS FUNCTION SUBPROGRAM IS CALLED BY RK2 IN RUNGE-KUTTA
C     SOLUTION OF DIFF.EQN. ARISING FROM W(P) AND P(W) EQNS.
C     IN FLAMOCO
C*****************************************************************
      FUNCTION FUN(XA,Y,ICON)
      DIMENSION HP(22),QX(22),VX(2,22),TIME(2),PXA(22)
      COMMON/ABLOC/B,C2,HZ,C4,CONST,E,XAC,TOP,NFLOW,DELTA,HNEW,HOLD,
     C        XAOLD,XAINC,HP,PHI,FLAGPHI
      COMMON/CBLOC/CEPT(22),SLOPE(22),WIDCON
      COMMON/KVAL/KCB,HSTART,ISEG,DIFCE,HSAV,IPR
      COMMON/SUBVTQ/IFRAC,VX,TIME,AFAC,ITQ,Q,A,PI,S2MS1,WFLAG,PXA,N,
     C        EMOD,RNU,NODIV
      COMMON/SUBWIDTH/W(51),AREAS(22),NUMWID,I,L,H
      REAL KCB,L
      INTEGER WFLAG

      IF(IFRAC.EQ.1) M=ISEG
      IF(IFRAC.EQ.2) M=IFIX(12.-XA*10.)
      IF(M.GT.11)  M=11

C     THIS REMOVES PROBLEM WITH SMALL -VE XA
C     THIS IS NOT PROGRAMMED FOR N=21 , ONLY N=11 !!!!!!!!!!!!!!!!!
C     BELOW IS CASE WHEN H > HZ

C     HCALC IS NOT CALLED WHEN CALCULATING FRAC1 & FRAC2
C     HCALC IS CALLED ONLY ONCE AND THE REST OF CALCNS USES THE
C     SAME H VALUE
      IF (ICON.GE.2) GOTO 90
      CALL HCALC (ICON,Y,H,HZ,XA)

C     NOW SET FRAC HEIGHT FOR SEGMENT 1
 90   IF (IPR.EQ.0.) HP(1)=HSAV
      FP =HZ/H
      HS1=DSQRT(1. - FP*FP)
      F1B=KCB*DSQRT(TOP*H) + TOP*S2MS1*HZ*DLOG((1.+HS1)/FP)
      F1B=F1B*WIDCON
      F2 = -(C4/PHI)*(1.- Q)*(CEPT( M)    - SLOPE(M)    *XA)
C     THE (1.-XA)*(CEPT-SLOPE)  TERM IS X VARIATION FOR HML AND Q.
      FUN= - F2/(H*F1B**3)
      RETURN
      END
```

```
      COMPILER NOSTACK
      COMPILER DOUBLE PRECISION
      SUBROUTINE QTEST(N,QPER,ITERQ)
C
C*****************************************************************
C     THIS SUBROUTINE COMPARES INPUT QX AND OUTPUT QX, AND
C     AVERAGES THEM POINT BY POINT. THE RESULT IS FED BACK
C     AS A NEW INPUT QX AND THE FRACTURE RE-COMPUTED. CYCLE
C     IS REPEATED TIL INPUT AND OUTPUT QX CONVERGES.
C*****************************************************************

      COMMON/QVAL/QX,INQX
      COMMON/CBLOC/CEPT(22),SLOPE(22)
      DIMENSION QX(22)
      REAL INQX(22)
      IF(ITERQ.GE.1) GOTO 2
```

```
            WRITE(2,3)
 3          FORMAT(" INPUT MAX NUM OF ITERATION FOR QX :")
            READ(1) NUM
            WRITE(2,4) NUM
 4          FORMAT("  NUM =",I3)
 2       CONTINUE
         DO 10 J=1,N
         DIFQ =DABS(QX(J) - INQX(J)) / INQX(J) *100.
         IF(DIFQ.GT.QPER) GO TO 20
 10      CONTINUE
         WRITE(2,50)QPER,ITERQ
         WRITE(3,50)QPER,ITERQ
 50      FORMAT(/ " FLOW RATE CONVERGES TO WITHIN ",F10.0," PERCENT AFTER  ,
     C   I4," ITERATIONS ")
 20      DO 30 J=1,N
 30      QX(J) = (3.* QX(J) + INQX(J))/4.
C        THIS NEW QX IS AVERAGE OF INPUT AND OUTPUT QX.
         DO 80 J=2,N
         I = 13-J
         SLOPE(J) = (QX(I-1) - QX(I))/ (0.1 * QX(1))
 80      CEPT(J) = QX(I-1)/QX(1)  + SLOPE(J)*(1.0 - (J-1)*.1)
C        DIVIDING BY QX(1) NORMALIZES THE QX'S TO 1 AT THE WELLBORE

         SLOPE(1) = SLOPE(2)
         CEPT(1) = CEPT(2)
         WRITE(2,100) (SLOPE(J),J=1,N),(CEPT(J),J=1,N)
 100     FORMAT(/" SLOPE = "/11(E9.3,1X)/
     C   " CEPT = "/11(E9.3,1X))
         ITERQ = ITERQ + 1
         IF(ITERQ.GT.NUM) GO TO 60
         GO TO 70
 60      CONTINUE
         WRITE(2,90) ITERQ
         WRITE(3,90) ITERQ
 90      FORMAT(" QX ITERATIONS EXCEED ",I4,  SO PROGRAM STOPPED ")
         STOP
 70      RETURN
         END
```

```
         COMPILER NOSTACK
         COMPILER DOUBLE PRECISION
         SUBROUTINE HCALC(ICON,Y,H,HZ,XA)
C*****************************************************************
C   HCALC - THIS IS TO FIND A SOLUTION FOR CRACK HALF LENGTH L
C        FOR GIVEN PRESSURE P(0).  SEE PAGE(E14) IN NOTES "CORRECTION
C        FOR FASTER VERTICAL EXPANSION". THE SOLUTION IS BY ITERATION
C        ??????????????????
C        CAN WE SHORTEN THIS BY NOT CALLING FUN 4 TIMES IN EACH RK
C        CALCN?
C*****************************************************************
         COMMON/KVAL/KCB,HSTART,ISEG,DIFCE,HSAV,IPR
         REAL KCB
         COMMON/SUBFUN/RMU,SDLDT(12),DLTPER,IDEBUG,IIFLAG,IDLDT,IDLDTZ
         COMMON/ELMBLOC/ELM
         COMMON/SUBVTW/IFRAC,VX(2,22),TIME(2),AFAC,ITW,Q,A,FI,SEMS1,WFLAG,
     C            PXA(22),N,EMOD,RNU,NODIV
         INTEGER WFLAG

         IF(IPR.NE.0) GOTO 500
C           EL IS IN UNITS OF HZ
            ELL = 0.5
```

```
      IF(IOLDT.EQ.1) GOTO 3
         ELR = 0.6
         GOTO 510
C        INPUT VALUE FOR ELR TO PREVENT BLOW UP WHEN USING CLEARLY'S
C        FIT
 3       WRITE(2,4)
 4       FORMAT(/" INPUT ELR (USUALLY 0.6) ")
         READ(1) ELR
C        THIS ELR IS A GUESS. IT HAS TO BE GREAT ENOUGH THAT SOLUTION
C             LIES BETWEEN ELL,ELR
         GOTO 510
 500  ELL = .99*ELM
      IF(ELL.LE.0.5) ELL=0.5000001
      ELR = 1.1*ELM

 510  CONTINUE
C     METHOD OF FALSE POSITION SOLUTION FOR ELM
      DO 100 I=1,50
         T1 = PFUN(ELR,Y,XA)
         T2 = PFUN(ELL,Y,XA)
         ELMOLD = ELM
         ELM = (ELL*T1 - ELR*T2)/(T1-T2)
         IF (IFRAC.NE.1) GOTO 11
         IF(IPR.LT.5.AND.IDEBUG.EQ.1) WRITE(2,10)I,ELL,ELM,ELR,ELMOLD
 10      FORMAT(" I=",I4," ELL, ELM, ELR = ",3F10.6," ELMOLD= ",F10.6)
 11      T5 = PFUN(ELM,Y,XA)
         T6 = T2
         IF (T5*T6) 20,30,40
 20         ELR = ELM
            GOTO 50
 30         ELR = ELM
 40         ELL = ELM
 50      CONTINUE
         DIFF = T5/Y * 100.
C        % ERROR OF CONVERGENCE RELATIVE TO NET PRESSURE Y=P(0)-S1
         IF (IFRAC.NE.1) GOTO 90
            IF (IPR.LE.5.AND.IDEBUG.EQ.1) TYPE " T5= ",T5," %DIFCE= ",
     C        DIFF," DIFCE= ",DIFCE
 90      IF (ABS(DIFF).LT.DIFCE) GOTO 150
 100  CONTINUE
      WRITE(2,125)
 125  FORMAT(" NO SOLUTION TO CRACK HALF-HEIGHT AFTER 50 ITERATIONS ")
      STOP

 150  CONTINUE
C     NOW CALC H=2*L
      H = 2.* ELM * HZ
C     SAVE H FROM 1ST CALL IN FUN
      IF (ICON.EQ.1) HSAV = H/HZ
      RETURN
      END
```

```
      COMPILER NOSTACK
      COMPILER DOUBLE PRECISION
      FUNCTION PFUN(EL,Y,XA)
C*****************************************************************
C     THIS SUBPROGRAM CALCULATES THE FUNCTION
C     PFUN = (P(0) - S1) - (PBAR - S1) -3*PI/16 ... DL/DT ...
C     SEE PAGE E14 OF NOTES FOR FAST VERTICAL EXPANSIONS
C     NOTE THAT (P(0) - S1) = Y
C*****************************************************************
      DIMENSION PXA(22),VX(2,22),TIME(2),AREA(22),W(51)
      COMMON/KVAL/KCB,HSTART,ISEG,DIFCE,HSAV,IPR
      INTEGER WFLAG,ASEG
      REAL KCB,L,NU,MU
      COMMON/DLT/DLDT(12),SHP(12),DT
      COMMON/SUBWIDTH/W,AREA,NUMWID,I,L,H
      COMMON/SUBVTQ/IFRAC,VX,TIME,AFAC,ITQ,Q,A,PI,S2MS1,WFLAG,
     C          PXA,N,EMOD,NU,NODIV
      COMMON/ABLOC/B,C2,HZ,C4,CONST,E,XAC,TOP,NFLOW,DELTA,HNEW,HOLD,
     C          XAOLD,XAINC,HP(22),PHI,FLAGPHI
      COMMON/SUBFUN/MU,SDLDT(12),DLTPER,IDEBUG,IIFLAG,IDLDT,IDLDTZ
      COMMON/PFUNB/S1RXA(11),S2RXA(12)
      COMMON/PFUNC/TDLBDT

C     USING TRIANGULATION TO EXTRAPOLATE DLBDT CORR.TO XA
C     AT FRAC2 ISEG=3 OF FRAC1 BECOMES ISEG=2 (FRAC2 HAS 11 SEG)
      ASEG=ISEG
      IF(IFRAC.NE.2) GOTO 5
      IF(ISEG.NE.2) GOTO 4
         DLBDT = DLDT(1)
         GOTO 9
 4    ASEG=ASEG-1
 5       KSEG = ASEG-1
      IF(IFRAC.EQ.2) GOTO 6
         D1 = S1RXA(KSEG)
         D2 = S1RXA(ASEG)
         GOTO 7
 6    D1 = S2RXA(ASEG)
      D2 = S2RXA(ISEG)
 7       DLBDT = (XA - D1)/(D2 - D1)*(DLDT(ASEG) - DLDT(KSEG))
     C          + DLDT(KSEG)
 9       IF (IDEBUG.EQ.1.AND.DLBDT.NE.TDLBDT) WRITE(2,8) DLBDT
 8       FORMAT(" DLBDT=",E13.6)

C     NEXT CALC Y(=PXA OR (P(0)-S1) IN LINEC9)
         F1 = 1.5*KCB*SQRT(HZ*EL/PI)
         F2 = 4.*HZ*EL/PI*S2MS1*ACOS(.5/EL)
         F3 = HZ*S2MS1/(PI*SQRT(1.-(.5/EL)**2))
      F  = F1 + F2 + F3
      IF(IFRAC.EQ.2) GOTO 15
         IF (IDEBUG.EQ.1.AND.IPR.LE.5) WRITE(2,10) F
 10      FORMAT(" F= ",E13.6)
C     TO CALC E
 15   EP = 2.4*EL*EMOD

      PBARMS1 = KCB/SQRT(PI*HZ*EL) + 2./PI*S2MS1*ACOS(.5/EL)
      IF(IDLDT.EQ.1) GOTO 200
      IF(IIFLAG.GE.1) GOTO 200
         GG = 0.
         GOTO 300
 200  GG =  3.*PI/16.*MU/(EL*HZ)**2*EP**2/(1.-NU**2)**2
     C        * F*DLBDT/(PBARMS1)**3*(.3634 + .3907*EL*HZ/F*PBARMS1)
 300  CONTINUE
      PFUN = Y - PBARMS1 - GG
      IF (IFRAC.EQ.2) GOTO 40
         IF(IDEBUG.EQ.1.AND.IPR.LE.5) WRITE(2,30) PBARMS1,PFUN
 30      FORMAT(" (PBAR - S1)= ",E13.6," PFUN= ",E13.6)
 40   CONTINUE
      TDLBDT=DLBDT
      RETURN
      END
```

```
      COMPILER DOUBLE PRECISION
      SUBROUTINE DLDTCAL(ITERD)
C
C*********************************************************************
C     THIS SUBROUTINE CALCULATES DLDT(IN CGS) FOR EACH VERTICAL
C     SECTION, AND THEN ITERATES TIL IT CONVERGES
C*********************************************************************
      DIMENSION DIFFT(12)
      COMMON/DLT/DLDT(12),SHP(12),DT
      COMMON/ABLOC/B,C2,HZ,C4,CONST,E,XAC,TOP,NFLOW,DELTA,HNEW,HOLD,
     C        XAOLD,XAINC,HP(22),PHI,FLAGPHI
      COMMON/SUBFUN/RMU,SDLDT(12),DLTPER,IDEBUG,IIFLAG,IDLDT,IDLDTZ

C DT WITH AND WITHOUT LEAKOFF IS CALCD IN DVLTQ6B

      DO 10 I=1,11
         DL = (HP(I+1) - SHP(I))*HZ/2.
C        HP IS H/HZ FOR FRAC2, SHP IS H/HZ FOR FRAC1
         IF(IIFLAG.GE.1) GOTO 4
         IF(IDLDT.EQ.1) GOTO 4
            DLDT(I) = DL/DT
 4          DLDT(I)=(DL/DT + SDLDT(I))/2.
         IF (I.EQ.1.AND.IDLDTZ.EQ.1) DLDT(1)=0.0
         WRITE(2,9) I,DL,DLDT(I)
 9       FORMAT(" I,DL,DLDT(I): ",I5,2E13.6)
 10   CONTINUE
      ITERD = ITERD + 1

C     TEST TO SEE IF DLDT HAS CONVERGED.
      DIFFTMAX=0.
      DO 20 J=1,11
         DIFFT(J) = ABS(DLDT(J) - SDLDT(J))/DLDT(J) * 100.
         IF(DIFFTMAX.LT.DIFFT(J)) DIFFTMAX=DIFFT(J)
 20   CONTINUE
      IF (IDEBUG.EQ.1) GOTO 25
         WRITE(2,21) DIFFTMAX
         WRITE(3,21) DIFFTMAX
 21      FORMAT(" MAX % DIF DLDT = ",F10.4)
         GOTO 29
 25         WRITE(2,26) (DIFFT(J),J=1,11)
 26         FORMAT(" DIFFT= ",F10.4)
            IF(DIFFTMAX.LE.DLTPER) WRITE(2,27) DLTPER,ITERD
 27         FORMAT(/" DL/DT CONVERGES TO WITHIN ",F10.0," PERCENT "
     C         "AFTER ",I4," ITERATIONS. ")
 29   IF (DIFFTMAX.GT.DLTPER) GOTO 30
      STOP

 30   CONTINUE
      DO 40 I=1,11
 40      SDLDT(I) = DLDT(I)
      RETURN
      END
```

```
      COMPILER DOUBLE PRECISION
      FUNCTION  DFX( FLVAR,I )
C*********************************************************************
C    DFX, RNUMINT INTEGRATES OVER THIS FUNCTION (FOR FLK10)
C*********************************************************************
      COMMON /DSLOP/ TSLOPE

      IF(I.EQ.1) DFX = FLVAR**(TSLOPE-1.) * SQRT(1.-FLVAR)
      IF(I.EQ.2) DFX = FLVAR**(TSLOPE-1.) / SQRT(1.-FLVAR)
      RETURN
      END
```

```
        COMPILER DOUBLE PRECISION
        FUNCTION FFX(A,B,NODIV,F)
C*****************************************************************
C    PERFORM THE NUMERICAL INTEGRATION OF WIDTH EQN USING THE
C    TRAPEZOIDAL METHOD TO INTEGRATE OVER F2 FROM A TO B (FOR FLK10)
C*****************************************************************
C
C    PERFORM NUMERICAL INTEGRATION

        INTEGER F
        AREA = 0.0
        WIDTH = (B-A) / NODIV

        DO 30 J = 1,NODIV
           XL = A + (J-1) * WIDTH
           IF(XL.EQ.0.0) XL=WIDTH/10.
           XR = A +    J    * WIDTH
           IF(F.EQ.1) GOTO 2
           IF(B.EQ.1.0.AND.J.EQ.NODIV) XR=1.- WIDTH/10.
 2         SMALLA = (WIDTH/2.) * (    DFX(XL,F) +  DFX(XR,F)   )
 30        AREA = AREA + SMALLA
        FFX = AREA
        RETURN
        END
```

```
        COMPILER DOUBLE PRECISION
        FUNCTION ASIN(YY)
C*****************************************************************
C    THIS IS USED TO COMPUTE ARCSIN FROM THE DATA GIVEN
C    IF YY IS POSITIVE, RETURNS 0-90
C    IF YY IS NEGATIVE, RETURNS 270-360
C*****************************************************************
        XX=SQRT(1.-YY**2)
        ASIN=ATAN2(YY,XX)
        RETURN
        END
```

```
        COMPILER DOUBLE PRECISION
        FUNCTION ACOS( XX )
C*****************************************************************
C    THIS IS USED TO COMPUTE ARCCOS FROM THE DATA GIVEN
C    IF XX IS POSITIVE, RETURNS 0-90
C    IF XX IS NEGATIVE, RETURNS 90-180
C*****************************************************************
        YY=SQRT(1.-XX**2)
        ACOS=ATAN2(YY,XX)
        RETURN
        END
```