To the Graduate Council:

I am submitting herewith a thesis written by Richard N. Bryan entitled "Discrete-Time Regulator for a Digital Heat Control System." I recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Electrical Engineering.

Robert E. Bodenheimer, Major Professor

We have read this thesis and recommend its acceptance.

Accepted for the Council:

Vice Chancellor
Graduate Studies and Research

Thesis
.B
.B713
cop. 2

DISCRETE-TIME REGULATOR FOR A DIGITAL HEAT CONTROL SYSTEM

A Thesis

Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Richard N. Bryan

March 1980

111619

ACKNOWLEDGEMENTS

The author is grateful for the opportunity to do the research leading to this thesis provided by Dr. Robert E. Bodenheimer and by Oak Ridge National Laboratory. Also, many thanks are extended to Dr. J. D. Birdwell, for his patient assistance and helpful suggestions.

ABSTRACT

This study describes the design of an optimal discrete-time regulator used to maintain the temperature in an irradiation experiment performed at Oak Ridge National Laboratory. A review of the mathematical basis of the state-variable regulator is followed by the development of the dynamical equations which describe the heater system. System-modeling experiments are explained, and simulation studies of the controller's operation are reviewed. Finally, the implementation of the controller on a dedicated minicomputer is presented.

TABLE OF CONTENTS

CHAPTER		PAGE
I.	INTRODUCTION AND PROBLEM STATEMENT	1
II.	MATHEMATICAL REVIEW AND DYNAMICAL EQUATIONS	5
III.	SYSTEM MODELING EXPERIMENTS AND SIMULATIONS	9
IV.	DIGITAL IMPLEMENTATION	16
V.	CONCLUSIONS	24
REFERENCES		26
APPENDIXES		28
APPENDIX A.	SYMBOLS	29
APPENDIX B.	COMPUTER SIMULATIONS	30
VITA		50

LIST OF FIGURES

FIGURE		PAGE
1.	Cut-Away View of Pressure Vessel Simulator	2
2.	Steady-State Distribution	11
3.	System Step Response	13
4.	Regulated System	15
5.	Typical Output Connection	18
6.	One-Cycle Timing Diagram	20
7.	One-Period Timing Diagram	21
8.	Overall System Diagram	23

CHAPTER I

INTRODUCTION AND PROBLEM STATEMENT

One of the most important questions in nuclear reactor construction concerns the effects of neutron radiation on the structural properties of steel used in pressure vessel walls. When stressed, ferretic steels fail in two modes; at lower temperatures, in a brittle mode and at higher temperatures in a ductile mode. The temperature at which the change from brittle failure to ductile behavior occurs is raised by neutron radiation, up to levels that could be expected during a reactor cold start (Carpenter, 1964).

The extent of the embrittlement of the steel is dependent on the temperature during irradiation; lower temperatures produce increased damage. In order to prevent catastrophic failure, therefore, it is essential to know the precise embrittling effects on various steels at accurately known temperatures and neutron fluxes. At Oak Ridge National Laboratory (ORNL), an experiment was devised to produce this information.

Samples of several types of steel, in standard test sizes were assembled into sets approximately 10 in. square, by one in. thick. Four of these sets were irradiated at different distances from the core of the Oak Ridge Research Reactor (ORR).

The pressure vessel simulator (PVS) consisted of three samples sets, each set having electrically heated plates attached to its front and back sides. This simulator was enclosed in a helium-filled box and submerged in the ORR cooling water pool, facing the reactor core window (see Figure 1).

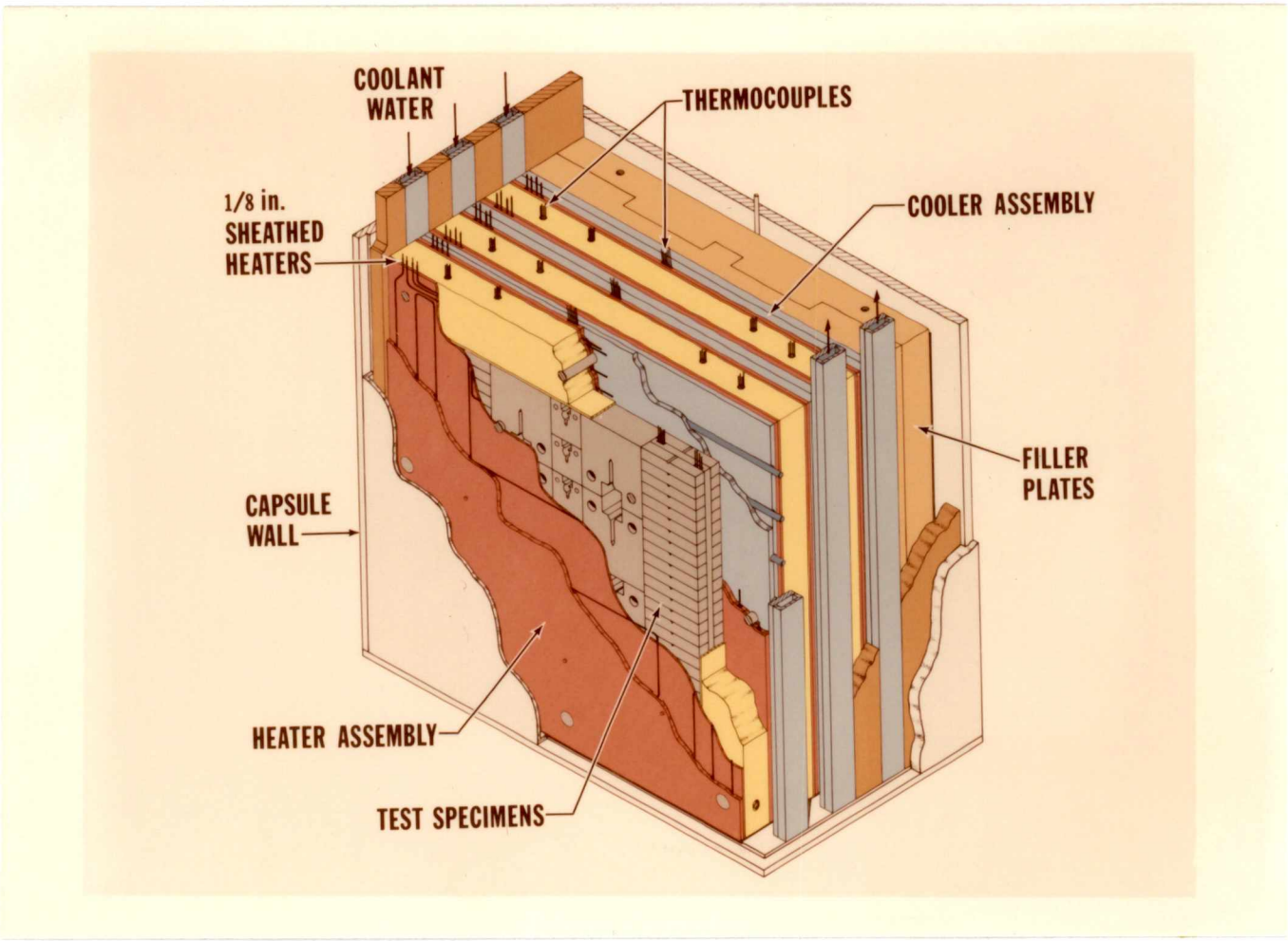


Figure 1. Cut-Away View of Pressure Vessel Simulator

Thermal isolation between the three sample set/heater units was provided by three cooler plates in which water flowed. The cooler plates also aided in removing heat generated inside the steel by gamma radiation.

The fourth sample set, with heater plates only, was placed nearer the core. This set was the surveillance simulator. Cooling for this set was provided by the reactor pool water only.

The data from this experiment was intended to show embrittling effects at 550°F; this was the set point for the control system. Variations over the distribution of temperatures of 10°F or less were required in order to correlate this experiment with data taken at other temperatures. A dedicated minicomputer system was chosen to maintain the temperature, and to provide periodic records of the temperature distribution. This data acquisition and control function was performed by a PDP 11T03 disk-based minicomputer with an IP300 input/output system.

Each sample set had 22 cromel-alumel thermocouples distributed throughout it. Twenty of these sensors were used for data reporting and control while the remainder were attached to alarm systems and supervisory monitors.

Each heater unit was a steel plate with 10 nichrome heating elements in five grooves on its surface. Two elements were placed in each groove and the paired elements were operated in parallel. The heaters were controlled by zero-crossing solid-state relays. The I/O system provided one-shot digital outputs to trigger the relays.

The purpose of the controller algorithm, therefore, was to read the temperature distribution and use these readings to adjust the heater power levels to create a uniform 550°F temperature in the simulators. The design procedures to develop the controller are covered in Chapters II and III. Chapter IV details the computer implementation of the algorithm.

CHAPTER II

MATHEMATICAL REVIEW AND DYNAMICAL EQUATIONS

Since the object of the controller was to maintain a specific temperature (550°F), the design of the controller was seen as a regulator design problem. If the system could be modeled by a linear, time-invariant state-variable description, the controller could be designed as a linear, time-invariant feedback system, and a simple digital implementation would result (Athans, 1966).

If one heater power level was abruptly increased, the temperature at any given thermocouple would increase exponentially. For the j-th thermocouple,

$$T = T_f(1 - e^{-\frac{t}{\tau}})$$

or,

$$\dot{T} = -\frac{1}{\tau} T + \frac{1}{\tau} T_f$$

Assuming a linear system,

$$T_f = \sum_{i=i}^{10} c_i \, p_i$$

where p is the input power.

With equations such as those above for each thermocouple, a state variable description of the system was formed. Defining

$$x_i = T_i - 550°F$$

$$a_{ii} = -\frac{1}{\tau_i}\ ;\ a_{ij} = 0,\ i \neq j$$

$$b_{ij} = \frac{1}{\tau_i}\ c_{ij}$$

$$u_i = p_i - p^*_i$$

where p* is the input required to produce 550°F, yielded

$$\dot{\underline{x}} = \underline{A}\,\underline{x} + \underline{B}\,\underline{u}$$

A discrete-time description of the system was required to implement the controller on a computer. With a sampling interval of Δt (Athans, 1964),

$$x(L+1) = \text{EXP}[\underline{A}\,\Delta t]\,\underline{X}(L) + \int_0^{\Delta t} \text{EXP}[\underline{A}\,\tau]\underline{B}\,\underline{u}\,d\,\tau$$

or,

$$\underline{x}(L+1) = \underline{F}\,\underline{x}(L) + \underline{D}\,\underline{u}$$

Designing the regulator required selecting a matrix so that

$$\underline{u} = -\underline{G}\,\underline{x}$$

and $\underline{G}$ was optimal with respect to some criterion. Defining a cost function,

$$J = \underline{x}^T(N)\,\underline{Q}\,\underline{x}(N) + \sum_{L=1}^{N} \underline{x}^T(L)\,\underline{Q}\,\underline{x}(L) + \underline{u}^T(L)\,\underline{R}\,\underline{u}(L)$$

produced an optimal value of $\underline{G}$ when the cost was minimized.

The cost, J, reflected two goals of the controller. First, the matrix $\underline{Q}$, the state-weighting matrix, showed the need of the controller to hold the temperatures near 550°F. Values of $\underline{x}$ (deviations from 550°F) are included in the cost quadratically; large errors are penalized by much greater cost values. Second, the input-weighting matrix, $\underline{R}$, reflected the fact that large values of input were to be avoided. By choosing values of $\underline{Q}$ much larger than $\underline{R}$, the controller would rapidly force the temperatures near 550°F with minimum cost overhead.

At some point L, along the state-space trajectory, as the state vector $\underline{x}$ is being forced to zero, a cost-to-go function v, can be defined as the cost associated with driving $\underline{x}$ to zero from L to the end of the interval N. That is,

$$v(L) = \underline{x}^T(L)\, \underline{K}\, \underline{x}(L)$$

The cost from L to N is also the sum of the costs at L plus the cost from L+1 to N. Since

$$\begin{aligned} v(L+1) &= [\underline{x}(L+1)]^T\, \underline{K}\, [x(L+1)] \\ &= [\underline{F}\, \underline{x}(L) + \underline{D}\, \underline{u}(L)]^T\, \underline{K}\, [\underline{F}\, \underline{x}(L) + \underline{D}\, \underline{u}(L)] \end{aligned}$$

Then, at L,

$$v = \underline{x}^T\, \underline{K}\, \underline{x} = \underline{x}^T\, \underline{Q}\, \underline{x} + \underline{u}^T\, \underline{R}\, \underline{u} + [\underline{F}\, \underline{x} + \underline{D}\, \underline{u}]^T\, \underline{K}\, [\underline{F}\, \underline{x} + \underline{D}\, \underline{u}]$$

Setting the derivative of v with respect to $\underline{u}$ to zero, and solving for $\underline{u}$, yields the value of input which minimizes the cost v.

$$\underline{u} = -[\underline{R} + \underline{D}^T \underline{K} \underline{D}]^{-1} \underline{D}^T \underline{K} \underline{F} \underline{x}$$

or,

$$\underline{G} = [\underline{R} + \underline{D}^T \underline{K} \underline{D}]^{-1} \underline{D}^T \underline{K} \underline{F}$$

Substituting this value of $\underline{u}$ into the cost equation for v (which must be true for all $\underline{x}$) yields

$$\underline{K} = \underline{F}^T \underline{K} \underline{F} - \underline{F}^T \underline{K} \underline{D} [\underline{R}^{-1} + \underline{D}^T \underline{K} \underline{D}]^{-1} \underline{D}^T \underline{K} \underline{F} + \underline{Q}$$

which is the nonlinear matrix (Ricatti) equation for K. Simulations of the system using various values for $\underline{Q}$ and $\underline{R}$ produced appropriate values of $\underline{G}$ for the system.

CHAPTER III

SYSTEM MODELING EXPERIMENTS AND SIMULATIONS

In order to define the system matrices, an experiment was performed on a sample set prototype. The sample set was approximately the same size and shape as the actual samples and was constructed of similar materials. Two heater plates and one cooler plate were attached to the sample set. This set of four plates was then placed in a helium-filled enclosure. The gas gap between the wall of the enclosure and the first heater plate was adjusted to the gap expected in the PVS experiment, and the area behind the cooler plate was filled with insulation. The enclosure was then placed in a water-filled barrel. This arrangement of gas gap, plates, and insulation provided a thermal environment similar to that expected in the front position of the PVS enclosure (see Figure 1, page 2).

The heater plates differed from the ones which would be used in the PVS capsule, in that the heater plate prototype had four heaters in each plate, while the final configuration had five. An additional heater was added to provide a more even heat distribution in the sample sets. The design of the controller was based on the four-heater configuration.

In the dynamical system equations,

$$\dot{\underline{x}} = \underline{A}\,\underline{x} + \underline{B}\,\underline{u}$$

the matrices $\underline{A}$ and $\underline{B}$ had to be defined through experiments on the prototype.

From the development in Chapter II, the $\underline{A}$ matrix is seen to be a diagonal matrix with the values $-1/\tau$ along the diagonal. Since the primary role of the controller was to maintain a steady-state distribution of temperatures, the time response of the regulator was very loosely specified, and only an approximate value for the time constant was required.

The experimental procedure was as follows. First, the sample set was heated to approximately 550°F, and the distribution of temperatures was trimmed to be as uniform as possible. Figure 2 shows the resultant temperature distribution. Next, one heater power level was reduced, and the transient behavior was recorded until a new steady-state was reached; this required about one hour.

Since the change in the distribution was due entirely to the change in a single heater, the ratio of the temperature change to the power change was the coefficient C for that point. For the i-th point,

$$\frac{\Delta T_i}{\Delta p_j} = c_{ij}$$

The steady-state temperature is a function of all the input power levels. Since only one input was changed, this operation corresponds to a partial derivative

$$\frac{\partial T_i}{\partial p_j} = c_{ij}$$

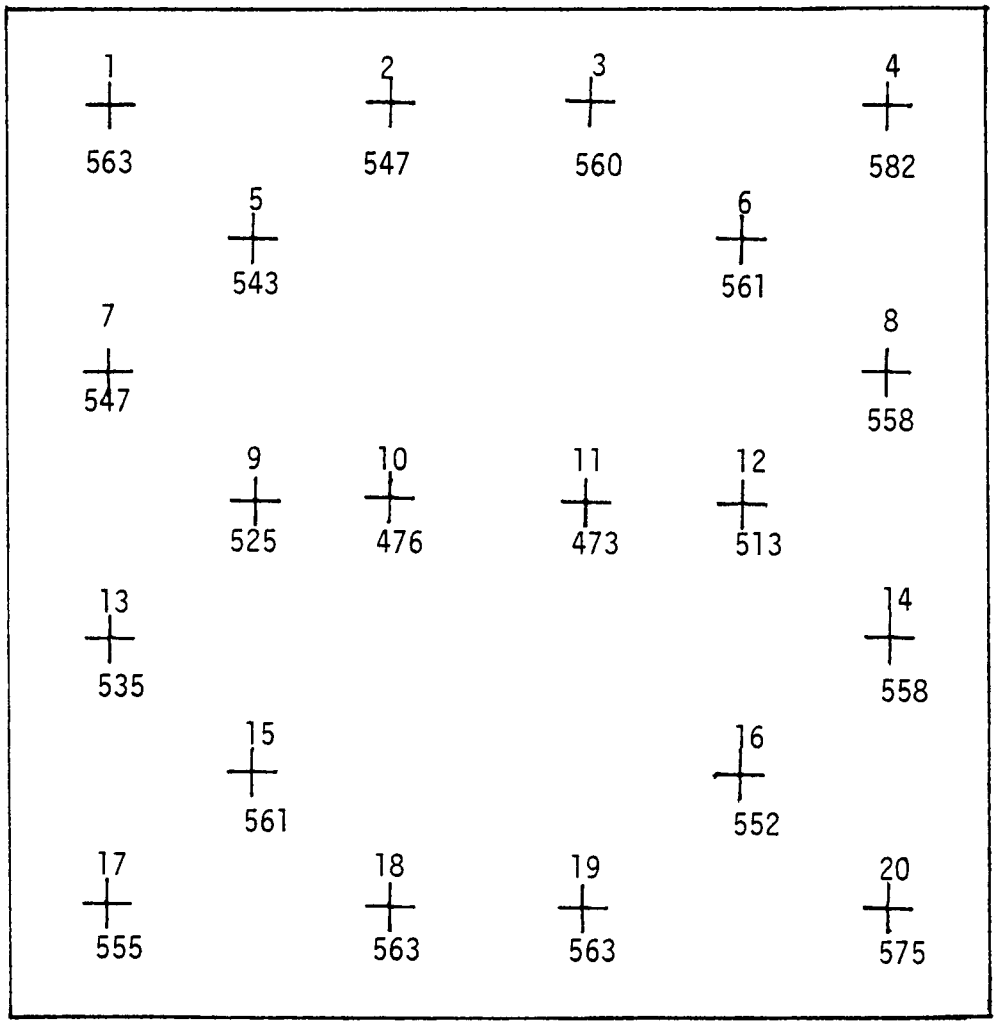


Figure 2. Steady-State Distribution

After the new temperature distribution was recorded, the heater was returned to its full power level, and a second heater was operated at reduced power. This operation produced another set of coefficients. This procedure was repeated for all eight heaters, and the coefficients C evaluated.

Data points were taken at equal intervals during the transient change in the system, after reducing the power level. The resulting set of points could be described by a linear relation

$$T(L+1) = \alpha_i \, T(L) + \alpha_o$$

where

$$\alpha_i = e^{-\Delta t/\tau}$$

Thus a least-squares fit could be found for α_i and τ could be found.

After having determined the dynamical equation matrices for the system, a computer simulation (Birdwell, 1979) of the system was performed, using constant input power levels starting from room temperature (see Figure 3). This plot was compared to the known behavior of the capsule under similar conditions and good agreement was found.

Having determined a mathematical model for the system, two additional matrices were to be found. These were $\underline{Q}$ and $\underline{R}$, the state- and input-weighting matrices for the cost function. The actual value of the cost "J" was unimportant, as long as it was minimal for the given system.

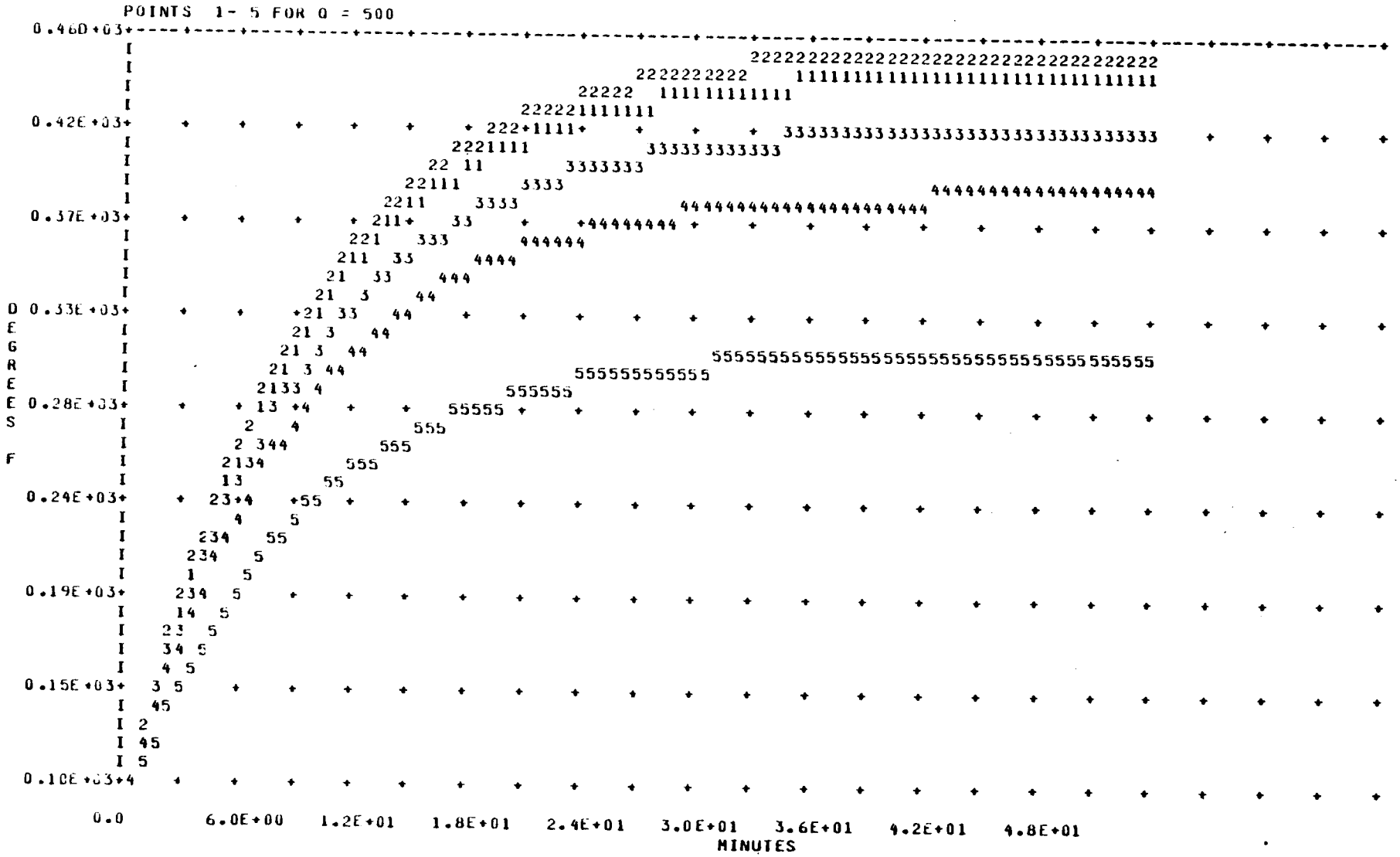


Figure 3. System Step Response

Therefore, one matrix could be fixed and the other adjusted for desired system behavior. Since all state errors should contribute equally to the cost, and all inputs are equally important, the two matrices were diagonal with equal values along their respective diagonals. The matrix $\underline{R}$ was chosen to be a unit diagonal matrix. By varying the value chosen for $\underline{Q}$ different regulators were designed and simulated, showing different system responses.

Increasing the value of $\underline{Q}$ so that the state errors were weighted more heavily, the system was forced to react faster, use larger input power, and achieve smaller steady-state errors. Therefore, the gain should be as large as possible, without requiring excessive input power.

Regulators were designed, through the Riccati equation, for values of $\underline{Q}$ of 1, 50, 500, and 5000. The system was then simulated using the regulators to control the inputs (see Figure 4). The value of 500 for $\underline{Q}$ gave good results; larger values caused large inputs to be required without producing improvement in the steady-state temperature distribution. Therefore the matrix $\underline{Q}$ was chosen to have the value 500 for each of its diagonal elements.

Having specified $\underline{A}$, $\underline{B}$, $\underline{Q}$, and $\underline{R}$ the regulator design was complete; the feedback matrix $\underline{G}$ could be calculated from the given matrices. The power inputs could be calculated from

$$\underline{u} = \underline{p}^* - \underline{G}\,\underline{x}$$

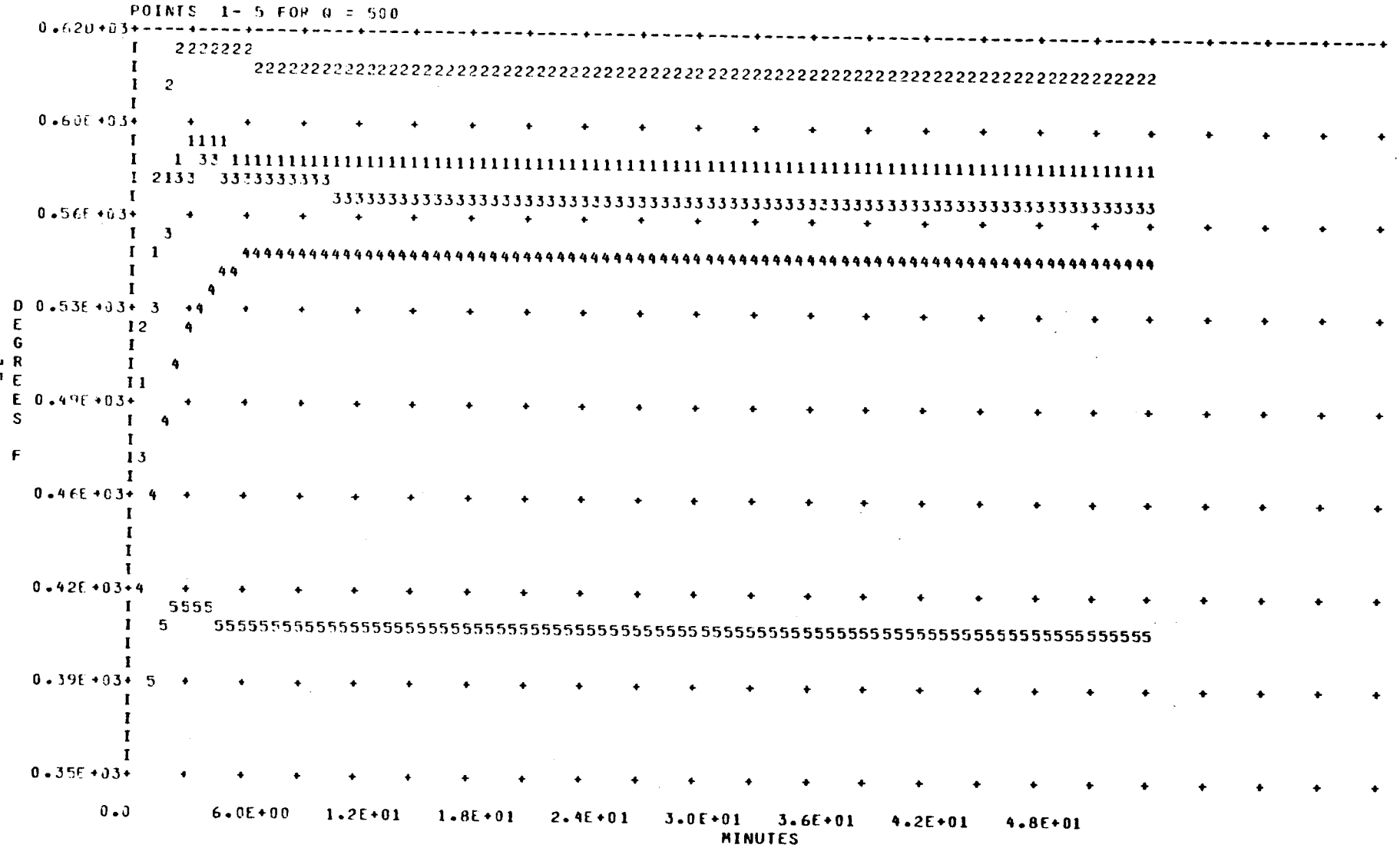


Figure 4. Regulated System

CHAPTER IV

DIGITAL IMPLEMENTATION

With the optimal feedback gains determined by the simulation studies, the controller was then implemented on the computer system. The control algorithm calculated the power to be applied to the heaters, based on the values of $\underline{p}^*$ and $\underline{x}$. A computer system configuration was chosen to acquire and report the necessary temperature data, calculate the power levels required, and control the heaters.

Data acquisition was accomplished through an IP300 peripheral subsystem interfaced to the 11T03 computer. A wide-range ADC (Analog to Digital Converter) provided readings of the thermocouple voltages through programmable-gain multiplexers. This voltage was proportional to the temperature difference between the capsule temperature and the ambient temperature. The ambient temperature was read by the ADC from the output of solid-state sensors within an isothermal enclosure which contained the thermocouple cold junctions. Thus, the capsule temperatures could be determined.

An output control system was now needed to provide linear control of the heaters. Phase-fired SCR (Silicon-Controlled Rectifier) control was not possible, due to the noise generated by the SCRS. Zero-crossing Solid-State Relays (SSRS) avoided the noise problem, but required the computer to fire the SSRS in an intermittent fashion to provide linear control of the heaters.

Digital one-shot outputs provided a method of firing the SSRS with a minimum of processor time overhead. A typical output connection is shown in Figure 5. The timing of the one-shot was set to approximately 1/60 second, so that one firing pulse would allow the 60 Hertz AC power to be applied to the heater for one cycle. An interrupt was generated in the computer system at 60 Hertz, and during each interrupt routine, the decision was made as to which heaters to fire and which to leave off.

This decision was made by a form of duty-cycle control. The maximum power available to each heater was determined from wattmeter inputs to the computer. The ratio of calculated power to maximum available power gave a duty-cycle percentage. During an interval of 100 cycles (at 60 Hertz) the SSR was fired on each cycle until the duty-cycle was reached, then left off until the end of the interval. In the next interval the cycle would be repeated. As an example, suppose the control routine indicated that 800 watts should be applied to a particular heater. If that heater's maximum power was 1000 watts, the duty-cycle should be 80 per cent. Therefore, the SSR would be triggered 80 times then left off for 20.

This interrupt structure also provided a method for timing the data acquisition routines. Over a period of five seconds, the sampling period, the computer must acquire all the data values, calculate the power levels and initialize the duty-cycle counters.

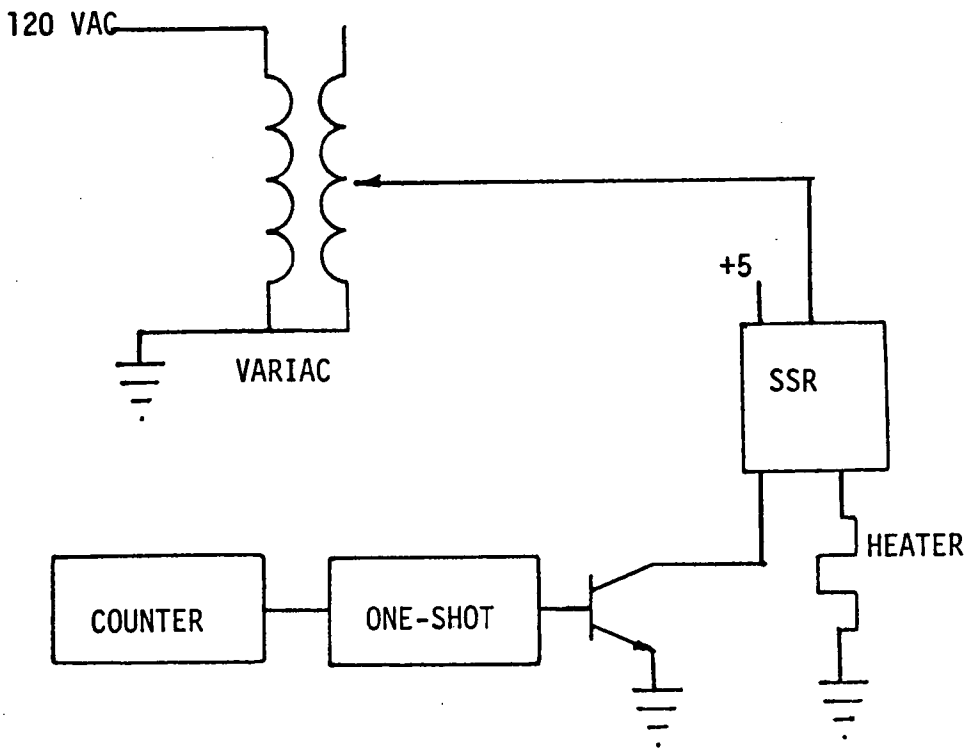


Figure 5. Typical Output Connection

The 300 cycle (five second) period was divided into three 100 cycle intervals. In each interval the firing routine proceeded as described above. During the first third of the period, however, data acquisition was performed as well.

On each interrupt one data value was read by the computer and a second conversion was started on the ADC. This second conversion was complete by the next interrupt, the ADC could be read and a third conversion started. This allowed the data from 100 points to be acquired with minimal processor overhead.

During the second third of the period the control calculations were performed. By timing the calculations this way, only current data was provided to the control algorithm. By the completion of the 300 cycles the new duty-cycle values were available to their respective counters, which were updated at interrupt number 299. In the next interrupt the index was reset and the process started again. Figures 6 and 7 show the timing for the system.

By using a 100-cycle interval, the control values were quantized into ranges which changed by one per cent of the maximum power. That is, changes in the input could be no smaller than that corresponding to one count in the duty-cycle counter. Longer intervals could have been used which would have resulted in smaller graduations of input levels. However, thermal cycling of the heaters, leading to early failure would occur if the heaters were switched on for long periods, then off for extended periods. Even though the time constants for the system (about 10 minutes) were so long, 100 cycles was chosen for the interval period to reduce the

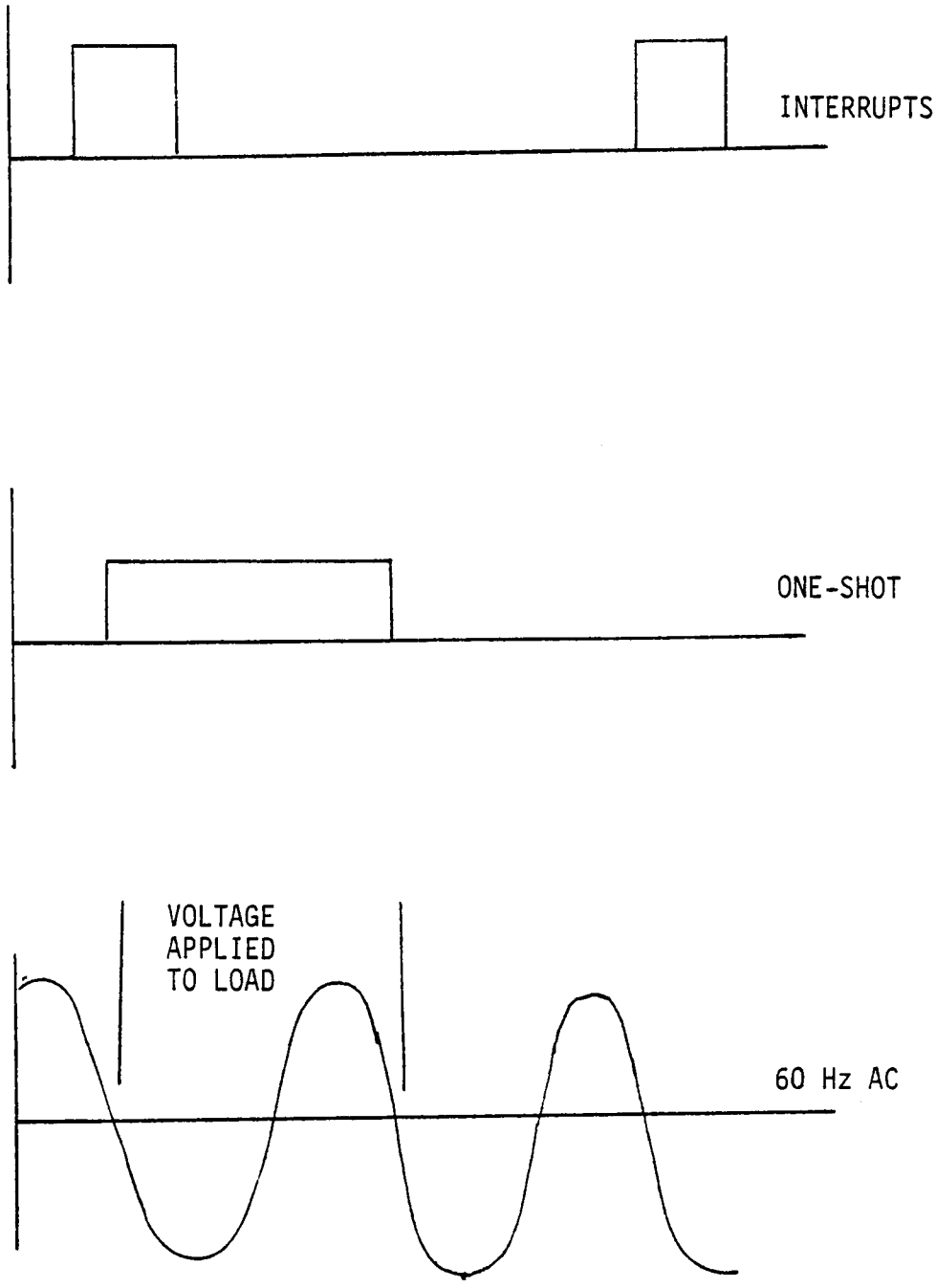


Figure 6. One-Cycle Timing Diagram

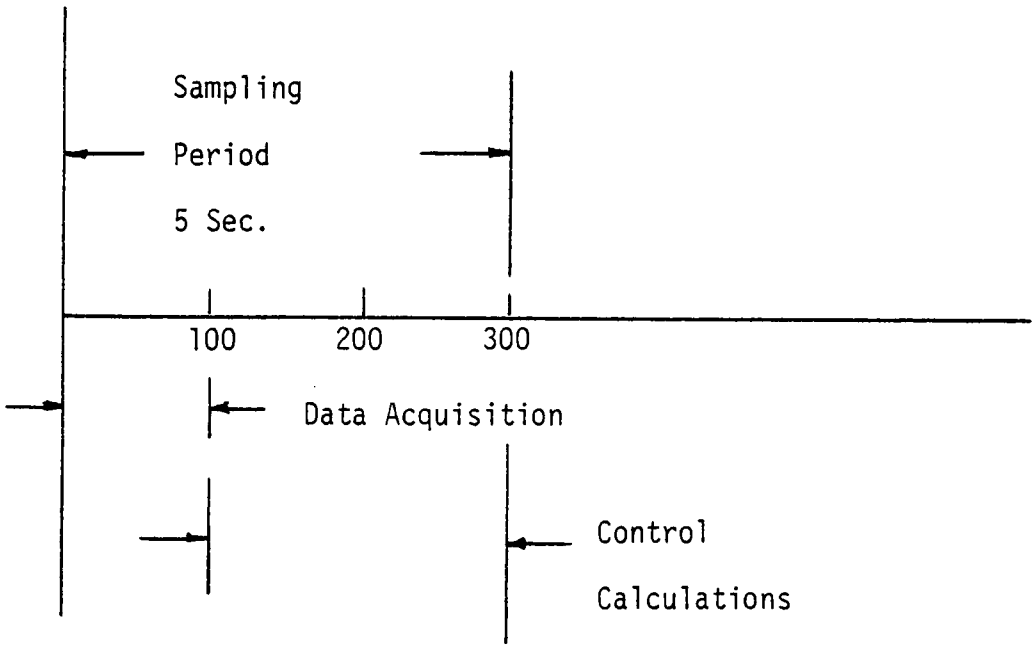


Figure 7. One-Period Timing Diagram

chance of heater failure. An overall diagram of the complete system is shown in Figure 8.

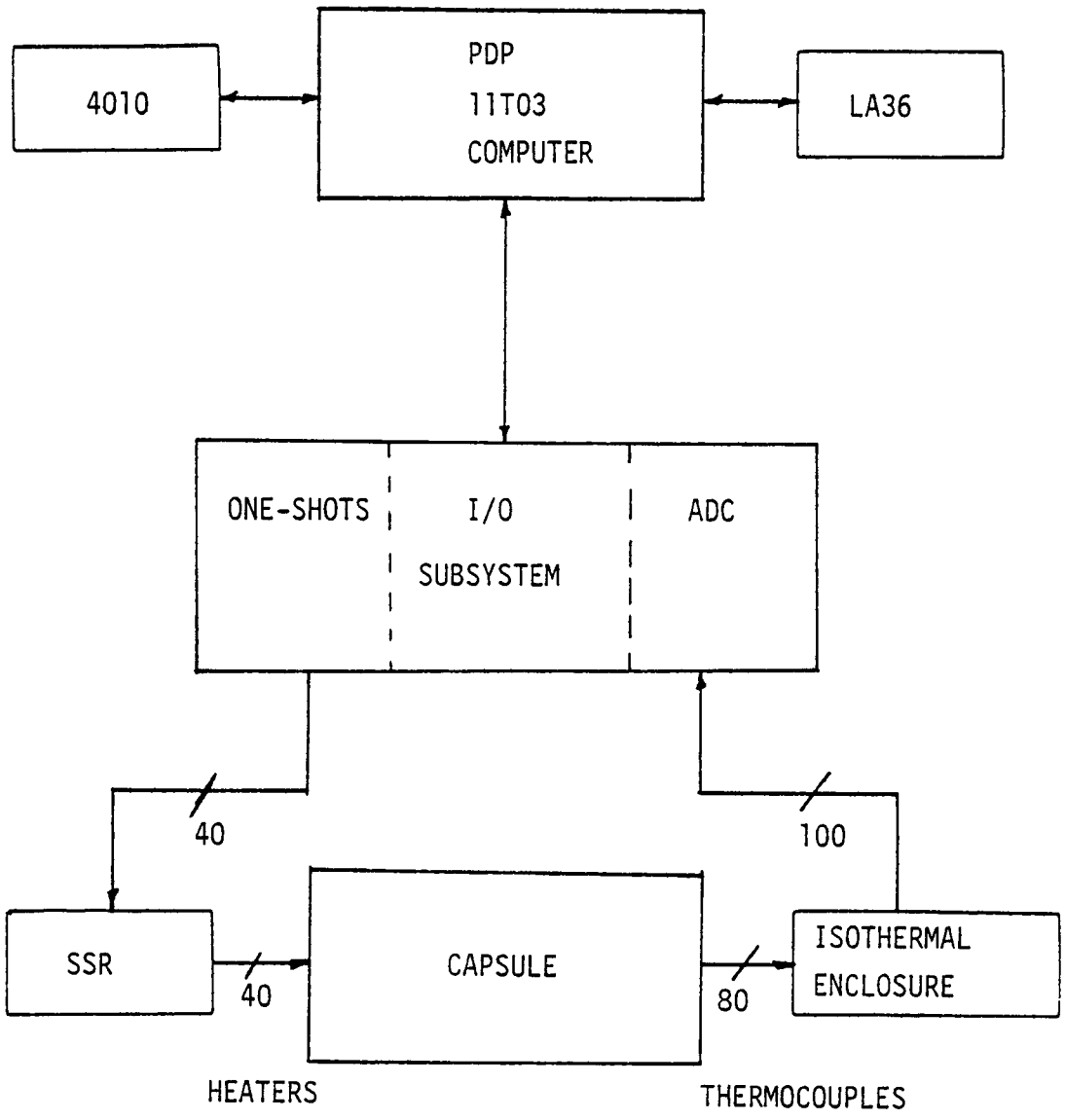


Figure 8. Overall System Diagram

CHAPTER V

CONCLUSIONS

This paper has described the design of a digital temperature controller, and the computer system which uses this control algorithm. Using a state-variable description of the system allowed a controller to be designed which was optimal with respect to a given criterion.

The system-modeling problem was solved by performing step-response experiments on a prototype of the sample sets. These results defined the dynamical system equations, and computer simulations based on this model defined the cost criterion.

The primary problem faced in implementing this controller was timing the one-shot outputs to provide linear control. The long time constants of the system allowed the controller to apply the 60-Hertz power in bursts. The duration of these bursts could be adjusted, in a pulse-width modulated form, to provide quantized linear control.

As of this writing, this control system has not been tested in its final configuration. These are several avenues to explore in case the system does not perform as well as the simulations indicate.

The control system is expected to be stable, regardless of other errors in the system. If any state in the system were unstable, the cost function j would be infinite, and no controller would produce a minimum. Instability due the discrete nature of the controller is most unlikely, since the sampling rate is so much shorter than the time constants for the system.

The most likely error to be expected is a distribution of temperatures in which some of the values are out of the range (530°F to 570°F) which are required by the experiment. One possible method to correct this type of error is to increase the weight given to the out-of-bound states in the cost matrix $\underline{Q}$. This action would force the feedback matrix $\underline{G}$ to place more emphasis on these states, and drive them closer to 550°F. The cost of such a change would be to allow other states to move farther away from the design value. A combination of increasing the weights of the largest deviations, and decreasing the weight of the smallest deviations might produce the most acceptable results.

An assumption throughout the controller design has been that the heaters attached to one sample plate will not affect the adjacent samples. One function of the cooler plate is to minimize such interaction. If this effect is noted, however, it might be possible to change the problem formulation to account for this behavior. The capsule could be modeled as a 60-state system with 30 inputs, for instance. Since the mathematical evaluation of such a large system is quite difficult, this alteration should be considered a last resort.

Overall, the key to the successful design of this controller lies in the accurate determination of the matrix $\underline{B}$, since this matrix describes the steady-state behavior of the system. Once it is evaluated by careful experimentation as described, the result should be an optimal, state-variable regulator.

REFERENCES

REFERENCES

Athans, Michael, and Peter L. Falb, Optimal Control (New York: McGraw-Hill, 1966), pp. 750-787.

Birdwell, J.D., Gilbert Jones, and Dennis Reed, "SYCOPACK," University of Tennessee, Knoxville, Tennessee, 1979.

Carpenter, G.F., N.R. Knopf, and E.S. Byron, "Anomalous Embrittling Effects Observed During Irradition Studies on Pressure Vessel Steels," Nuclear Science and Engineering, 19 (February, 1964), pp. 18-38.

APPENDIXES

APPENDIX A

SYMBOLS

a - scalar

$\underline{a}$ - vector

$\underline{A}$ - matrix

$\dot{\underline{x}}$ - superscript dot indicates derivative with respect to time

$\underline{F}^T$ - superscript "T" indicates the transpose of a matrix

$\underline{R}^{-1}$ - inverse

```
C
C*****************************************************
C
C                PROGRAM: RICSOL
C                AUTHOR: RICHARD N. BRYAN
C                DATE: OCTOBER, 1979
C
C                    THIS PROGRAM SOLVES THE DISCRETE-TIME
C                RICATTI EQUATION, FOR A GIVEN SET OF INPUT
C                MATRICES,AND COMPUTES AN OPTIMAL FEEDBACK
C                MATRIX FROM THIS SOLUTION. THE INPUT
C                MATRICES ARE OF THE FORM
C                          X' = AX + BU
C                WHERE 'A' IS A DIAGONAL 20 BY 20 MATRIX OF
C                TIME CONSTANTS,TAU. THE ELEMENTS OF THE
C                MATRIX B ARE OF THE FORM
C
C                          DELTA(X)
C                     ---------------  =  B
C                          DELTA(U)
C
C                THE SYSTEM MATRICES ARE THEN DISCRETIZED
C                FOR A 5 SECOND INTERVAL ( 0.0833 MIN ).
C                THE  COST FUNCTION WEIGHTING MATRICES
C                'Q' AND 'R' ARE FORMED, AND THE RICATTI
C                EQUATION IS SOLVED FOR THIS SYSTEM.
C                FINALLY, THE OPTIMAL FEEDBACK MATRIX
C                'G' IS CALCULATED.
C                    THE MATRICES WHICH ARE CALCULATED
C                ARE WRITTEN TO THE DISC INDETIFIED
C                BY THE VARIABLE 'KPUN'; IN ADDITION,
C                THE RICATTI SOLUTION AND THE FEEDBACK
C                MATRIX ARE PRINTED.
C                THE VALUES OF KPUN CURRENTLY USED ARE:
C                        7   -->  Q = 50
C                        10  -->  Q = 500
C                        11  -->  Q = 5000
C*****************************************************
C
      IMPLICIT REAL*8( A-H,O-Z)
      DIMENSION AINT(20,20), B(20,8), Q(20,20), R(8,8),
     1     T1(20,20),
     2    WORK(8), G(8,20),IPVT(8),
     3    AD(20,20), BD(20,8), TAU(20), Z4(8,20),
     5     TEP(8,8), Z3(20,8), RZ(40,40),RW(40,40),ER(40),
     6       EI(40),RWORK(20),ITYP(40),IRPVT(20)
      COMMON/INOU/ KIN,KPRT,KPUN,KPLT,KLOG
C
C
C*****************************************************
C               DEFINE I/O CONTROL
```

```
      KIN = 5
      KPRT = 6
      KPUN = 11
      KLOG = 0
C
C
C*************************************************************
C                INPUT SYSTEM MATRICES
C
      READ(KIN,100) (( B(I,J),J=1,8),I=1,20)
 100  FORMAT(8G8.6)
      READ(KIN,100) (TAU(I),I=1,20)
C
C
C*************************************************************
C
C                DISCRETIZE SYSTEM MATRICES AND FORM
C                'Q' AND 'R'.
C
      DO 200 I=1,20
            DO 200 J=1,8
                  B(I,J) = B(I,J)/TAU(I)
 200  CONTINUE
      DO 400 I=1,20
          DO 300 J=1,20
              AD(I,J) = 0.0D0
              AINT(I,J) = 0.0D0
              Q(I,J) = 0.0D0
 300      CONTINUE
          AD(I,I) = DEXP(-0.0833333D0/TAU(I) )
          AINT(I,I) = (1.0D0-AD(I,I))*TAU(I)
          Q(I,I) = 5000.0D0
 400  CONTINUE
      WRITE(KPUN,450) ((AD(I,J),J=1,20),I=1,20)
 450  FORMAT(1X,D14.7)
      DO 600 I=1,8
          DO 500 J=1,8
              R(I,J) = 0.0D0
 500      CONTINUE
          R(I,I) = 1.0D0
 600  CONTINUE
C
C
C*************************************************************
C
C                FORM MATRICES TO CALCULATE RICATTI
C                SOLUTION AND SOLVE.
C
      CALL MMUL(20,20,20,20,20,8,AINT,B,BD)
      WRITE(KPUN,450)  ((BD(I,J),J=1,8),I=1,20)
      CALL XYT(20,20,20,20,8,20,BD,BD,T1)
```

```
      CALL CPYMAT(20,20,20,20,AD,AINT)
      WRITE(KPRT,1550) Q(1,1)
 1550 FORMAT(1H1,T40,'SOLUTION TO RICATTI EQUATION ',
     1    'FOR Q =',1PD11.3,/)
      CALL RICDSD(40,20,20,20,20,40,RZ,RW,AINT,T1,Q,ER,EI,
     1     RWORK,ITYP,IRPVT)
      WRITE(KPRT,1500) ((Q(I,J),J=1,20),I=1,20)
      WRITE(KPUN,450)  ((Q(I,J),J=1,20),I=1,20)
C
C
C*********************************************************
C
C               CALCULATE 'G' MATRIX
C
      CALL SYMPRD(20,20,8,20,20,8,BD,Q,TEP,Z3)
      CALL MADD(8,8,8,8,8,R,TEP,R)
      CALL MINV(8,8,8,R,TEP,COND,WORK,IPVT)
      CALL XYT(8,20,8,8,8,20,TEP,BD,G)
      CALL MMUL(8,20,8,8,20,20,G,Q,Z4)
      CALL MMUL(8,20,8,8,20,20,Z4,AD,G)
      WRITE(KPRT,1400)
 1400 FORMAT('1         OPTIMAL FEEDBACK MATRIX G',/)
      WRITE(KPRT,1500) (( G(I,J),J=1,20),I=1,8 )
 1500 FORMAT( 2(1X,1P10D11.3/),/)
      WRITE(KPUN,450) ((G(I,J),J=1,20),I=1,8)
C
C
C*********************************************************
C
      STOP
      END
```

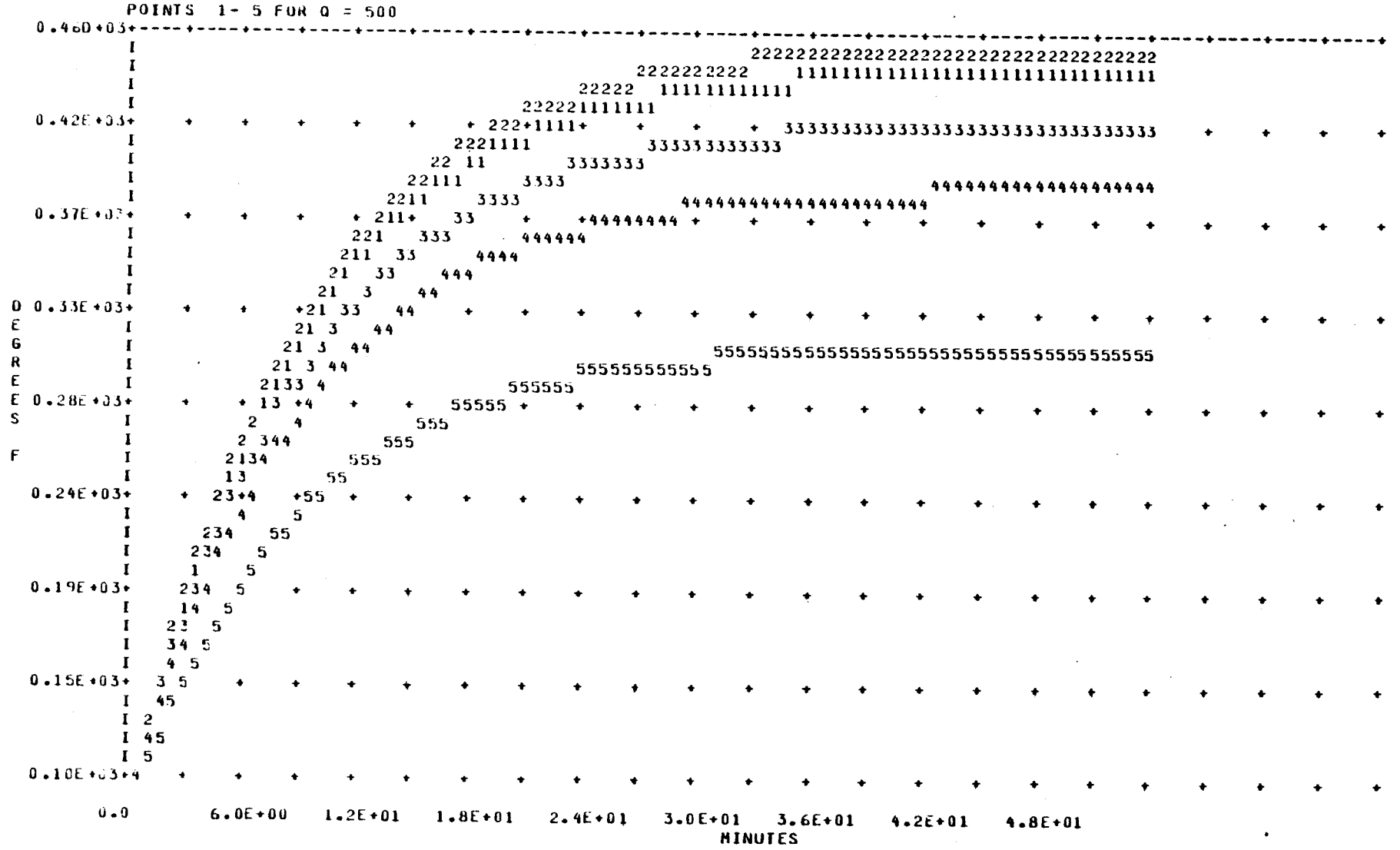
```
C
C
C                PROGRAM: SIMU
C                AUTHOR: RICHARD N. BRYAN
C                DATE: OCTOBER, 1979
C
C                    THE PURPOSE OF THIS PROGRAM IS TO
C                SIMULATE A LINEAR SYSTEM, USING BOTH
C                CONSTANT INPUTS AND USING A REGULATOR
C                TO CONTROL THE INPUTS. THE PROGRAM
C                'RICSOL' PROVIDES THE DISRETE-TIME
C                SYSTEM MATRICES, AS WELL AS THE
C                FEEDBACK MATRIX 'G'.
C                    FIRST, THE PROGRAM SIMULATES THE SYSTEM
C                WITH CONSTANT INPUTS, STARTING FROM 80
C                DEGREES F. THE INPUTS ARE IN THE VECTOR
C                'PSTAR', WHICH SHOULD RESULT IN A STEADY-
C                STATE TEMPERATURE DISTRIBUTION NEAR 550
C                DEGREES F BY THE END OF 1 HOUR. THE 20
C                TEMPERATURE POINTS ARE THEN PLOTTED FOR
C                THIS TIME INTERVAL, WITH 5 POINTS PER
C                PLOT. SECOND, THE REGULATOR IS STARTED,
C                USING THE FINAL DISTRIBUTION OF THE
C                UNREGULATED SYSTEM AS ITS INITIAL
C                STATE. THE INPUTS ARE CALCULATED FROM
C
C                     P = PSTAR - G( X-550 )
C
C                THE REGULATED SYSTEM IS THEN PLOTTED FOR
C                A 1-HOUR PERIOD. FINALLY, THE VALUE OF
C                PSTAR IS RESET BY THE AMOUNT -G( X-550 ).
C                A THIRD SET OF PLOTS IS PRODUCED FOR THIS
C                SYSTEM.
C
C
C
C
C                VARIABLE DEFINITIONS
C
C                AD - STATE MATRIX
C                BD - INPUT MATRIX
C                ISYM - PLOT CHARACTERS
C                S - RICATTI SOLUTION
C                G - FEEDBACK MATRIX
C                PSTAR - APPROXIMATE INPUT FOR 550 DEGREE
C                        DISTRIBUTION
C                PMAX - MAXIMUM POWER AVAILABLE TO EACH INPUT
C                BP - PRODUCT OF BD TIMES PSTAR
C                X - TEMPERATURES
C                Y - OUTPUT MATRIX
C                Z - TEMPORARY STORAGE
```

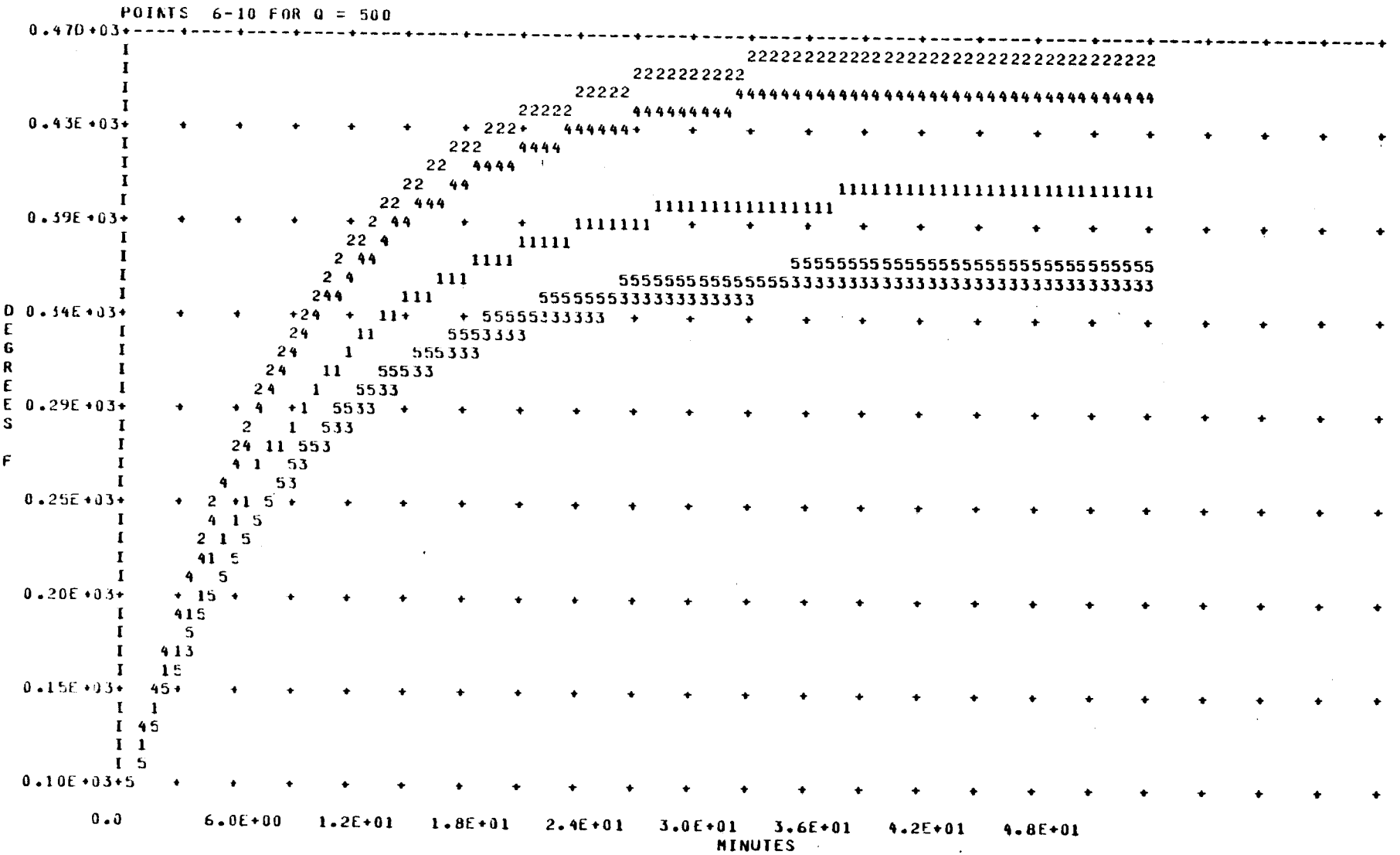
```
C                C - MATRIX WHICH SELECTS 5 STATES
C                     FOR PLOTTING
C                OUT - HOLDS 5 STATES FOR PLOTTING
C                YOUT - POINTS TO BE PLOTTED
C                DEL - DEVIATIONS OF TEMPERATURE FROM 550
C                PDEL - PRODUCT OF  G( X-550 )
C                P - INPUT
C                Z1 - TEMPORARY STORAGE
C                KIN - CARD READER INPUT NUMBER
C                KPRT - LINE PRINTER NUMBER
C                KPUN - DISC UNIT NUMBER
C                KPLT - PLOTTER NUMBER
C                KLOG - WHEN NONZERO, PRODUCES A HISTORY OF
C                       CALLS TO 'SYCOPACK'
C
C
C
C
C
      IMPLICIT REAL*8( A-H,O-Z)
      DIMENSION ISYM(5), AD(20,20), BD(20,8), S(20,20),
     1     G(8,20), PSTAR(8), PMAX(8), BP(20),
     2     X(20), Y(720,20), Z(20), C(20,5), OUT(720,5),
     3     YOUT(90,5), DEL(20), PDEL(8), P(8), Z1(20)
      COMMON/INOU/ KIN,KPRT,KPUN,KPLT,KLOG
C
C
C*********************************************************
C                DEFINE PLOT CHARACTERS
C
      DATA ISYM(1) /'1'/
      DATA ISYM(2) /'2'/
      DATA ISYM(3) /'3'/
      DATA ISYM(4) /'4'/
      DATA ISYM(5) /'5'/
C
C
C*********************************************************
C                DEFINE I/O CONTROL
C
      KIN = 5
      KPRT = 6
      KPUN = 11
C
C
C*********************************************************
C                READ IN SYSTEM MATRICES
C
      READ(KPUN,100) ((AD(I,J),J=1,20),I=1,20)
      READ(KPUN,100) ((BD(I,J),J=1,8),I=1,20)
      READ(KPUN,100) ((S(I,J),J=1,20),I=1,20)
```

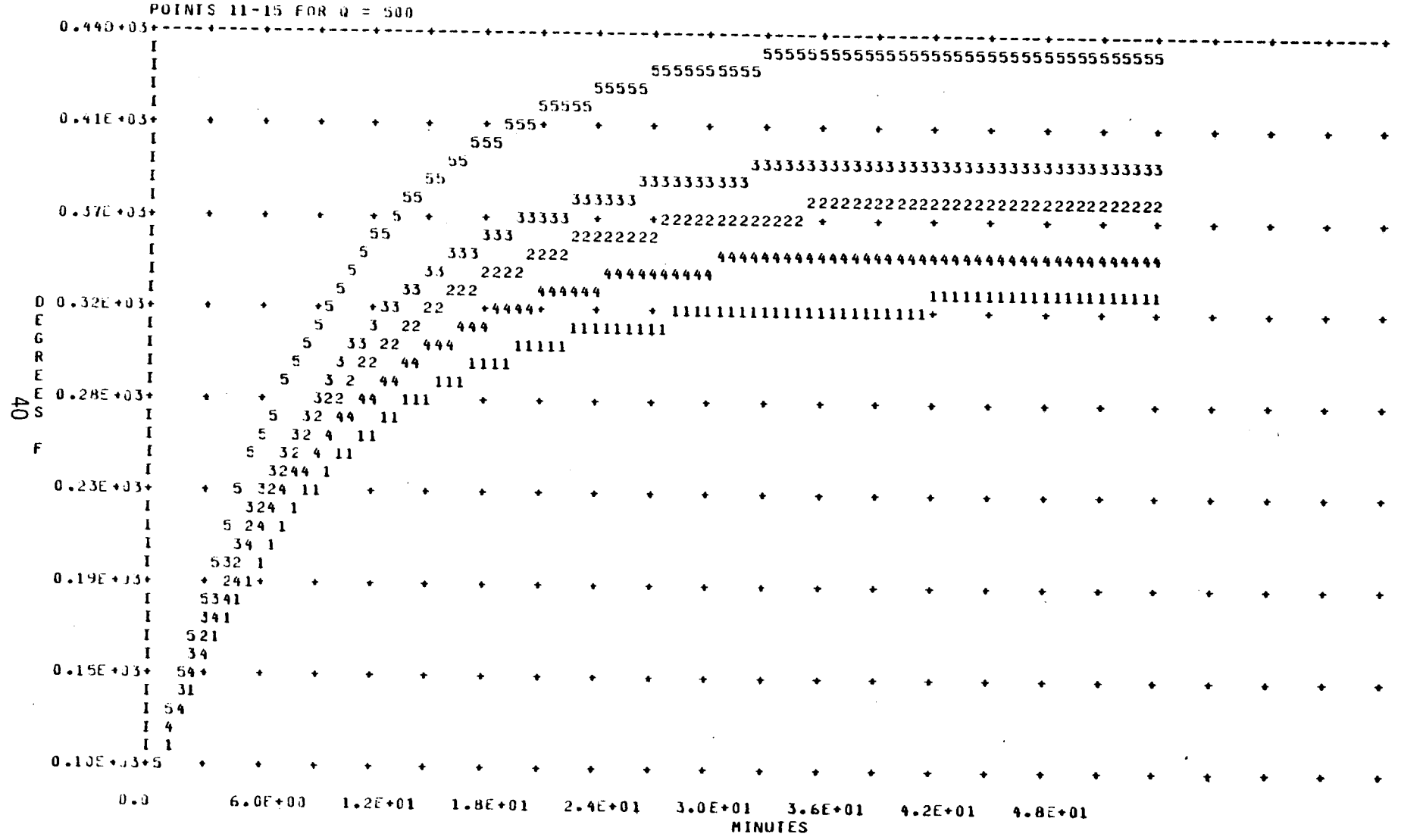
```
      READ(KPUN,100) ((G(I,J),J=1,20),I=1,8)
 100  FORMAT(1X,D14.7)
C
C
C*************************************************************
C                INITIALIZE VARIABLES
C
      READ(KIN,200) PSTAR
 200  FORMAT(8G8.6)
      DO 300 I=1,8
          PMAX(I) = 2500.0D0
 300  CONTINUE
      CALL MMUL(20,8,20,20,8,1,BD,PSTAR,BP)
      DO 400 I=1,20
          X(I) = 80.0D0
 400  CONTINUE
C
C
C*************************************************************
C                SIMULATE SYSTEM FOR 1 HOUR; WITH
C                SAMPLING INTERVAL OF 5 SECONDS, 720
C                POINTS.
C
      DO 600 I=1,720
          DO 500 J=1,20
              Y(I,J) = X(J)
 500      CONTINUE
          CALL MMUL(20,20,20,20,20,1,AD,X,Z)
          CALL MADD(20,20,20,20,1,Z,BP,X)
 600  CONTINUE
C
C
C*************************************************************
C                PLOT SYSTEM WITH 5 POINTS PER PLOT
C
      DO 1400 K=1,4
          DO 800 J=1,20
              DO 700 I=1,5
                  C(J,I) = 0.0D0
 700          CONTINUE
 800      CONTINUE
          DO 900 I=1,5
              C(5*(K-1)+I,I ) = 1.0D0
 900      CONTINUE
          CALL MMUL(720,20,720,720,20,5,Y,C,OUT)
          DO 1100 I=1,90
              DO 1000 J=1,5
                  YOUT(I,J) = OUT(8*I,J)
 1000         CONTINUE
 1100     CONTINUE
          CALL TLABEL('$MINUTES$')
```

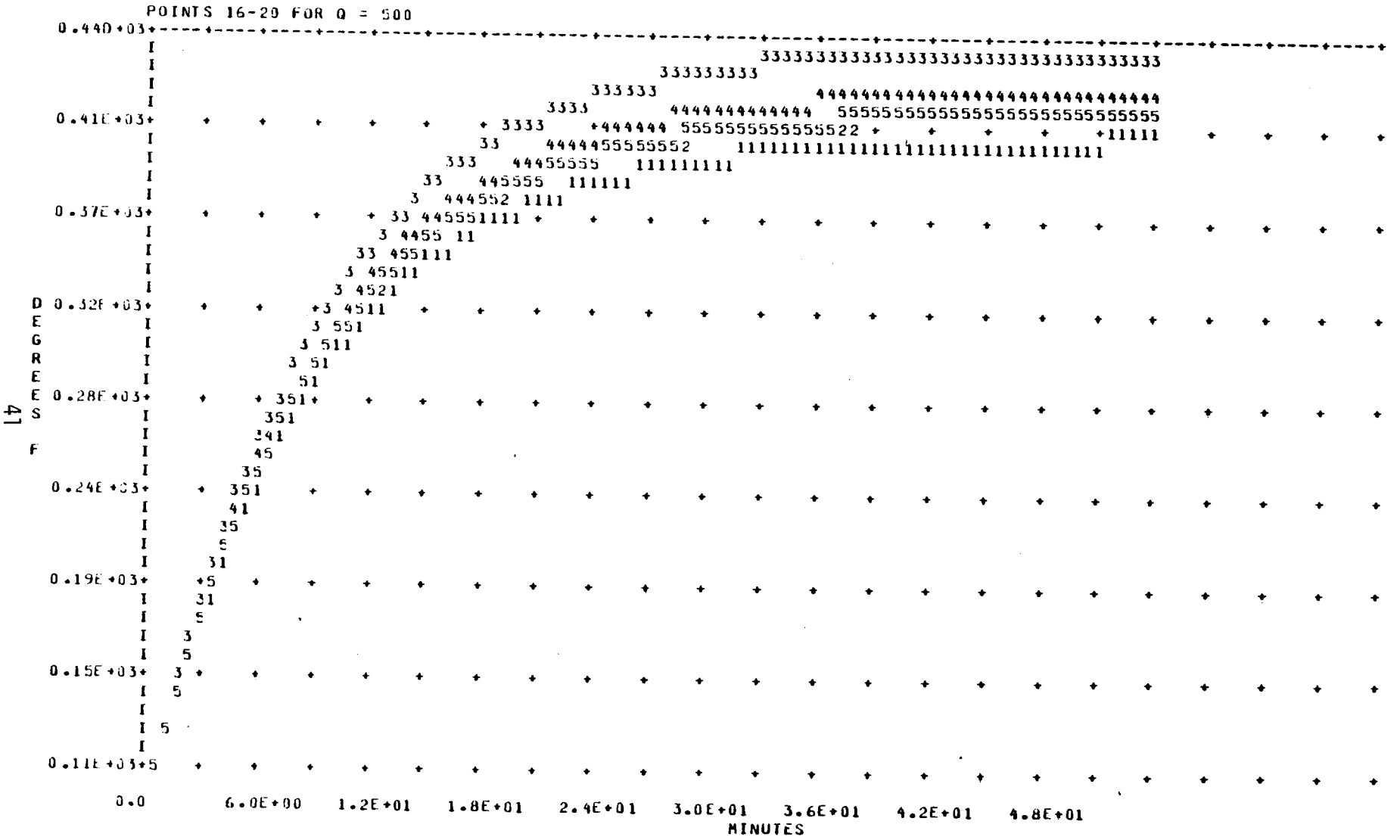
```
            CALL DLABEL('$DEGREES F$')
            CALL HPRPLT(YOUT,90,90,5,ISYM,0.6D0,40)
1400    CONTINUE
C
C
C*************************************************************
C               SIMULATE SYSTEM USING REGULATOR FOR
C               1 HOUR; RESET PSTAR ONCE (NUMBER OF
C               RESETS CONTROLLED BY VARIABLE L).
C
        DO 2600  L=1,2
        DO 1800 K=1,720
            DO 1500 I=1,20
                Y(K,I) = X(I)
1500        CONTINUE
            CALL  MMUL(20,20,20,20,20,1,AD,X,Z)
            DO 1600 I=1,20
                DEL(I) = X(I) - 550.0D0
1600        CONTINUE
            CALL MMUL(8,20,8,8,20,1,G,DEL,PDEL)
            CALL MSUB(8,8,8,8,1,PSTAR,PDEL,P)
             DO 1700 I=1,8
                IF( P(I) .GT. PMAX(I) ) P(I) = PMAX(I)
                IF(P(I) .LT. 500.0D0 ) P(I) = 500.0D0
1700        CONTINUE
            CALL MMUL(20,8,20,20,8,1,BD,P,Z1)
            CALL MADD(20,20,20,20,1,Z,Z1,X)
1800    CONTINUE
        DO 2400 K=1,4
            DO  1900 I=1,20
                DO 1900 J=1,5
                    C(I,J) = 0.0D0
1900        CONTINUE
            DO 2000 I=1,5
                C(5*(K-1)+I,I) = 1.0D0
2000        CONTINUE
            CALL MMUL(720,20,720,720,20,5,Y,C,OUT)
            DO 2200 I=1,90
                DO 2100 J=1,5
                    YOUT(I,J) = OUT(8*I,J)
2100            CONTINUE
2200        CONTINUE
            CALL DLABEL('$DEGREES F$')
            CALL TLABEL('$MINUTES$')
            CALL HPRPLT(YOUT,90,90,5,ISYM,0.6D0,40)
2400    CONTINUE
        DO 2500 M=1,8
            PSTAR(M) = PSTAR(M) - PDEL(M)
2500    CONTINUE
2600    CONTINUE
C
```

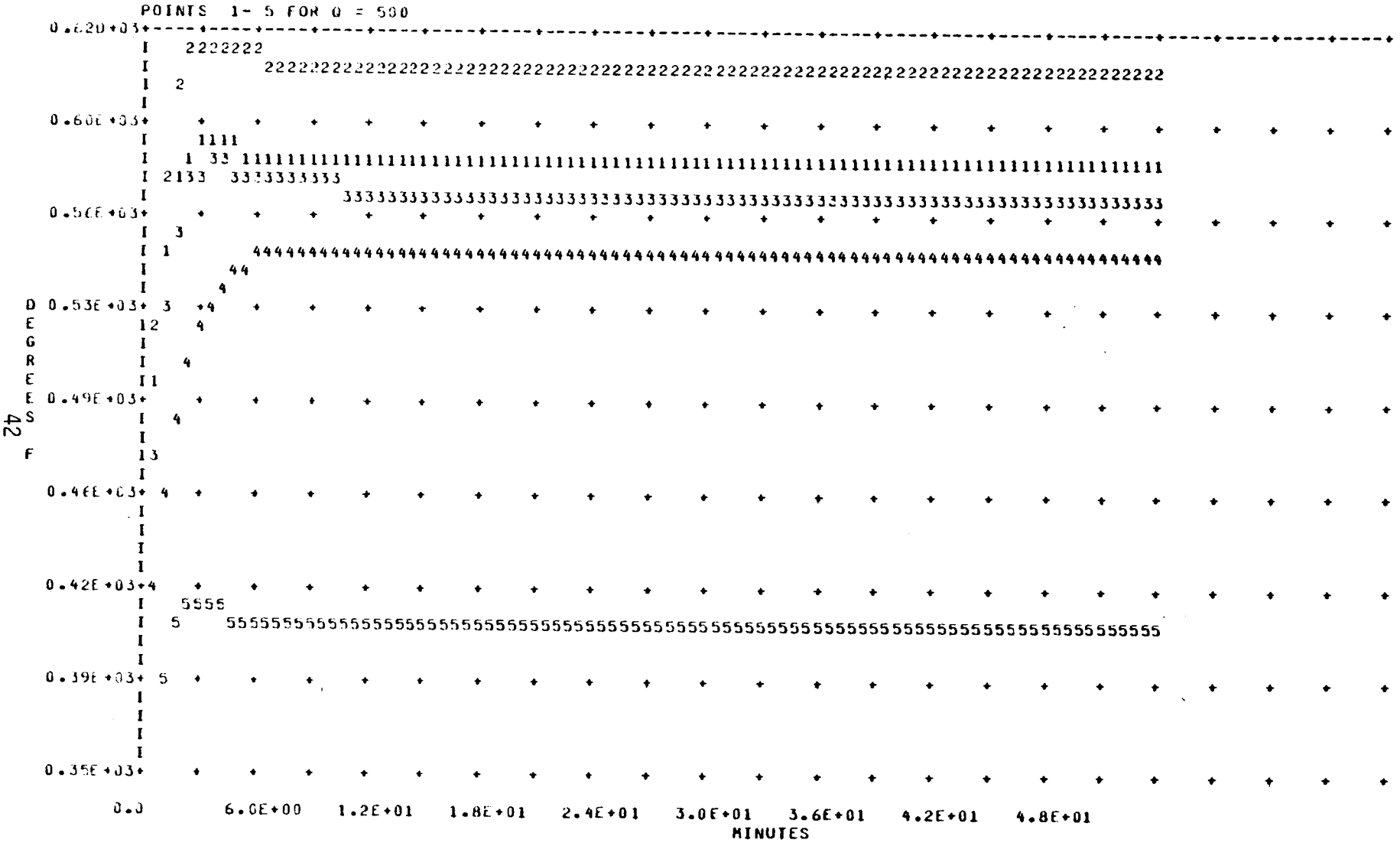
```
C
C**********************************************************
C
          STOP
          END
```

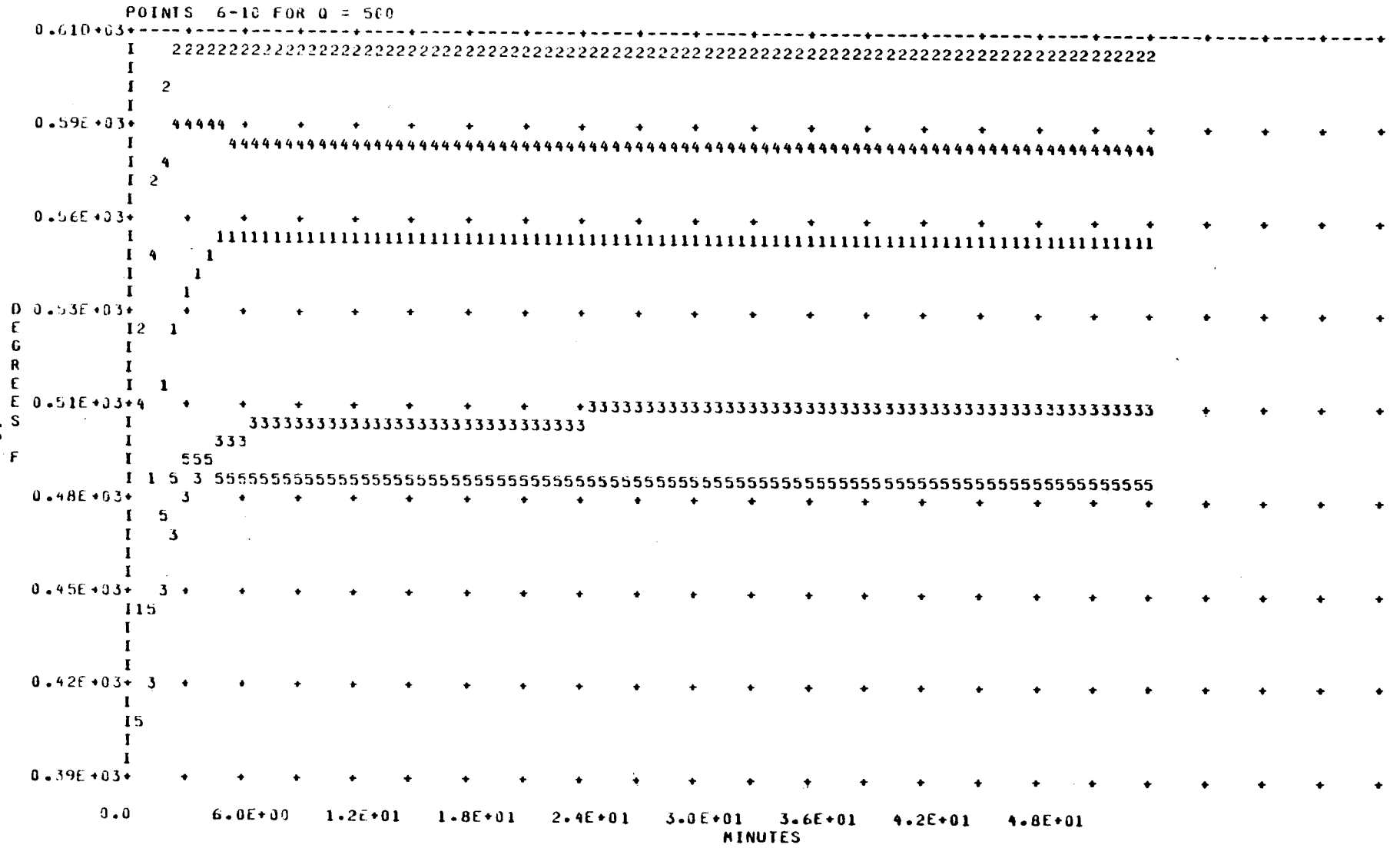

POINTS 1- 5 FOR Q = 500
0.46D+03
0.42E+03
0.37E+03
0.33E+03
0.28E+03
0.24E+03
0.19E+03
0.15E+03
0.10E+03
DEGREES F
0.0
6.0E+00
1.2E+01
1.8E+01
2.4E+01
3.0E+01
3.6E+01
4.2E+01
4.8E+01
MINUTES

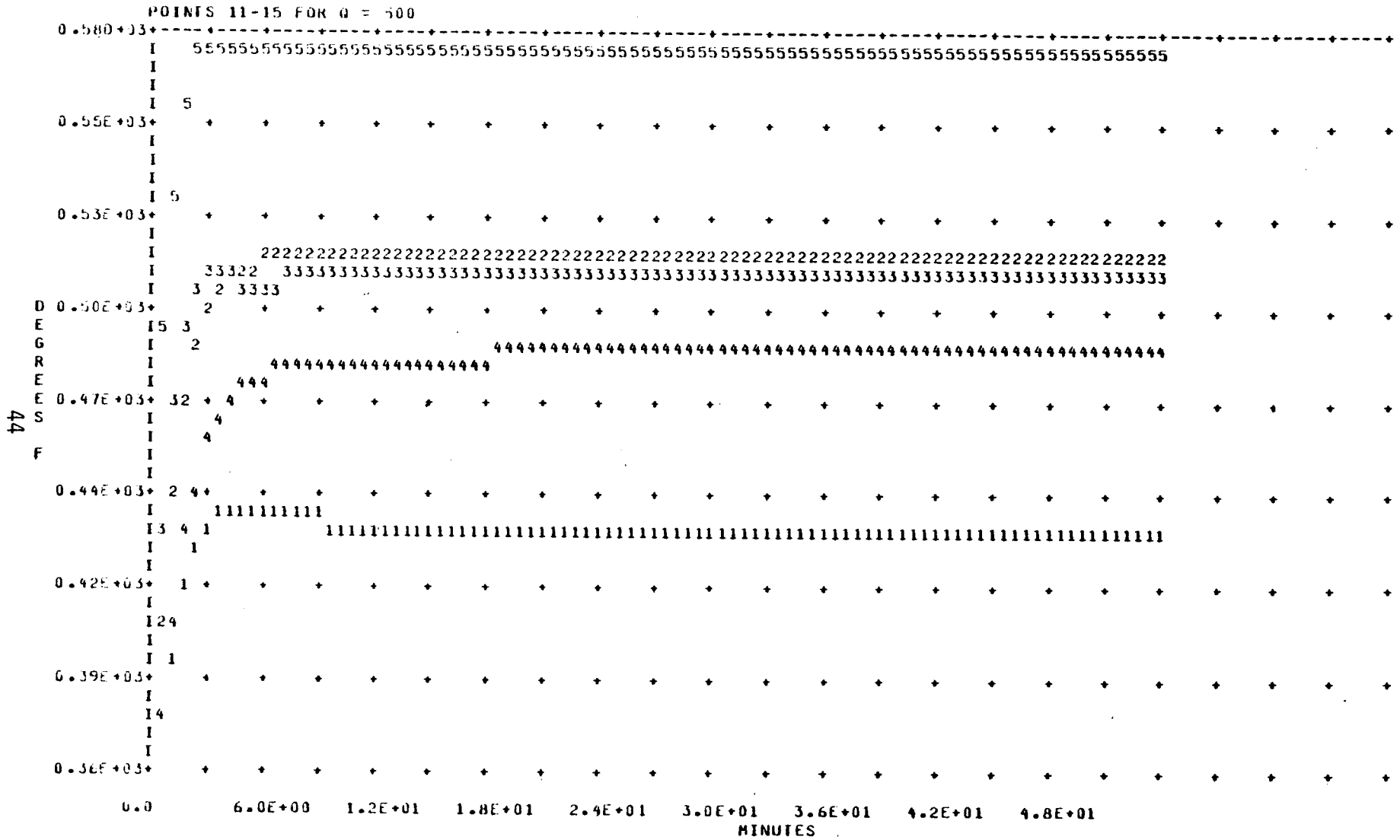

POINTS 6-10 FOR Q = 500
0.47D+03
0.43E+03
0.39E+03
0.34E+03
0.29E+03
0.25E+03
0.20E+03
0.15E+03
0.10E+03
DEGREES F
0.0
6.0E+00
1.2E+01
1.8E+01
2.4E+01
3.0E+01
3.6E+01
4.2E+01
4.8E+01
MINUTES

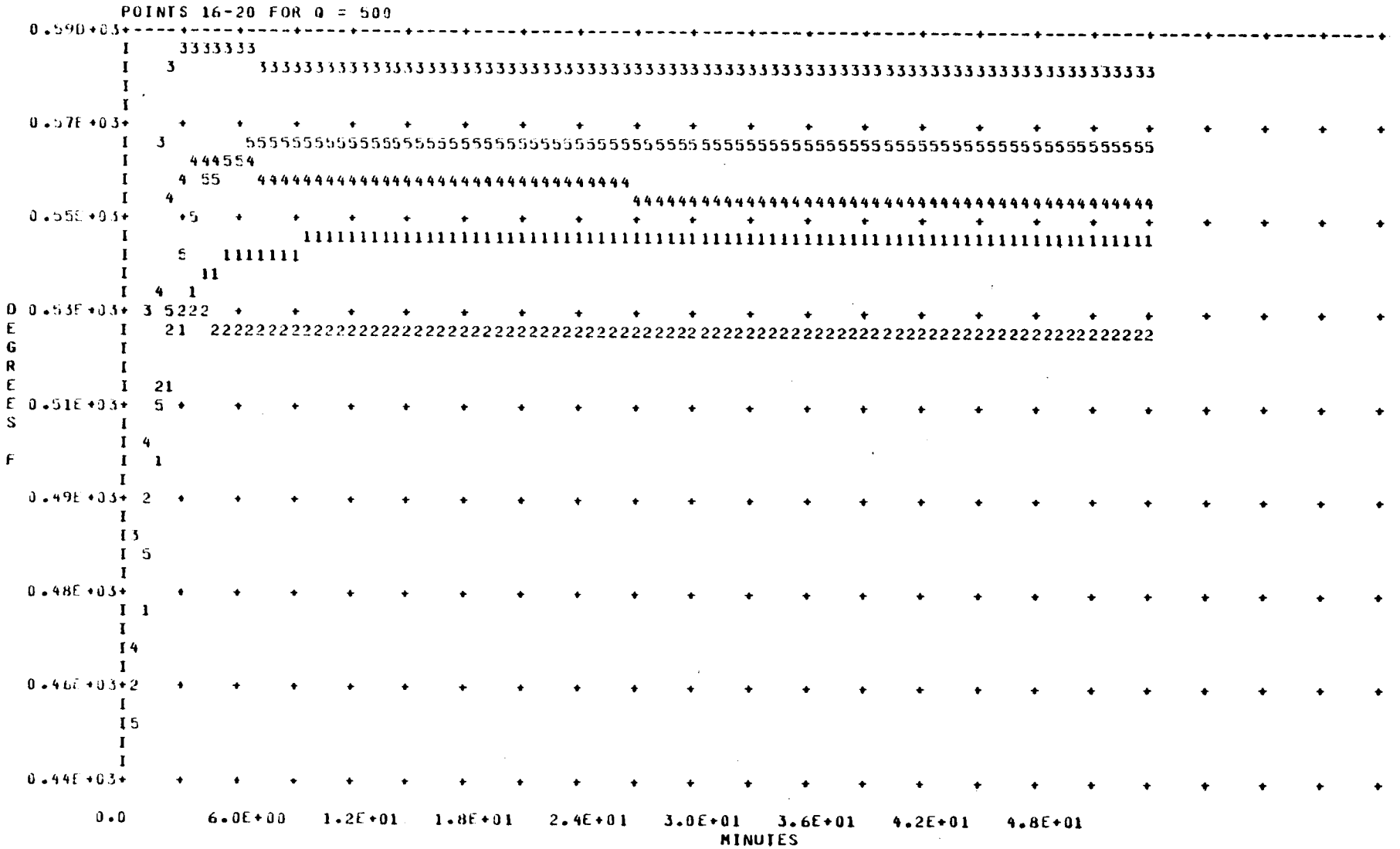

POINTS 11-15 FOR Q = 500
0.44E+03
0.41E+03
0.37E+03
0.32E+03
0.28E+03
0.23E+03
0.19E+03
0.15E+03
0.10E+03
DEGREES F
40
0.0
6.0E+00
1.2E+01
1.8E+01
2.4E+01
3.0E+01
3.6E+01
4.2E+01
4.8E+01
MINUTES

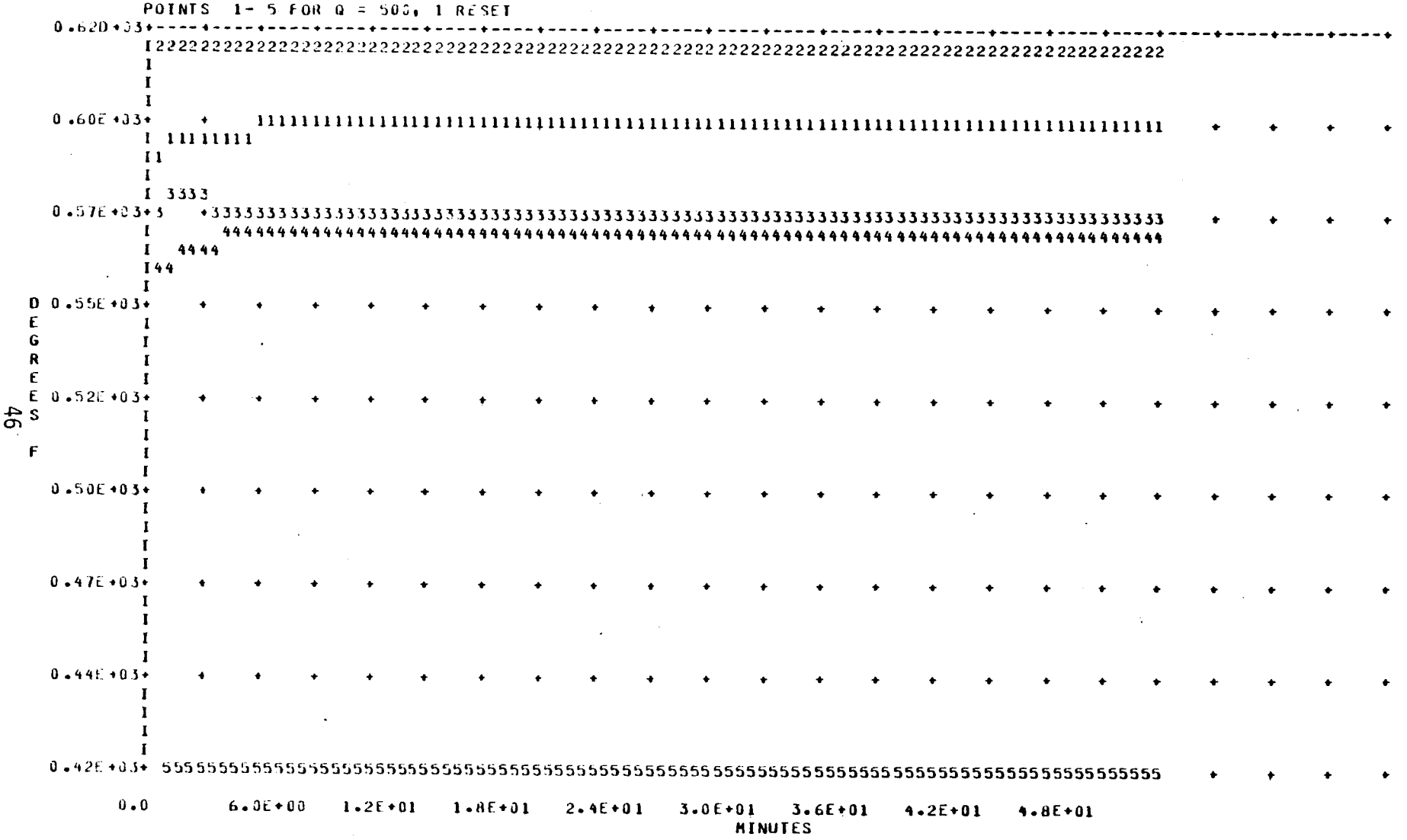

POINTS 16-20 FOR Q = 500
0.44E+03
0.41E+03
0.37E+03
0.32E+03
0.28E+03
0.24E+03
0.19E+03
0.15E+03
0.11E+03
DEGREES F
0.0
6.0E+00
1.2E+01
1.8E+01
2.4E+01
3.0E+01
3.6E+01
4.2E+01
4.8E+01
MINUTES

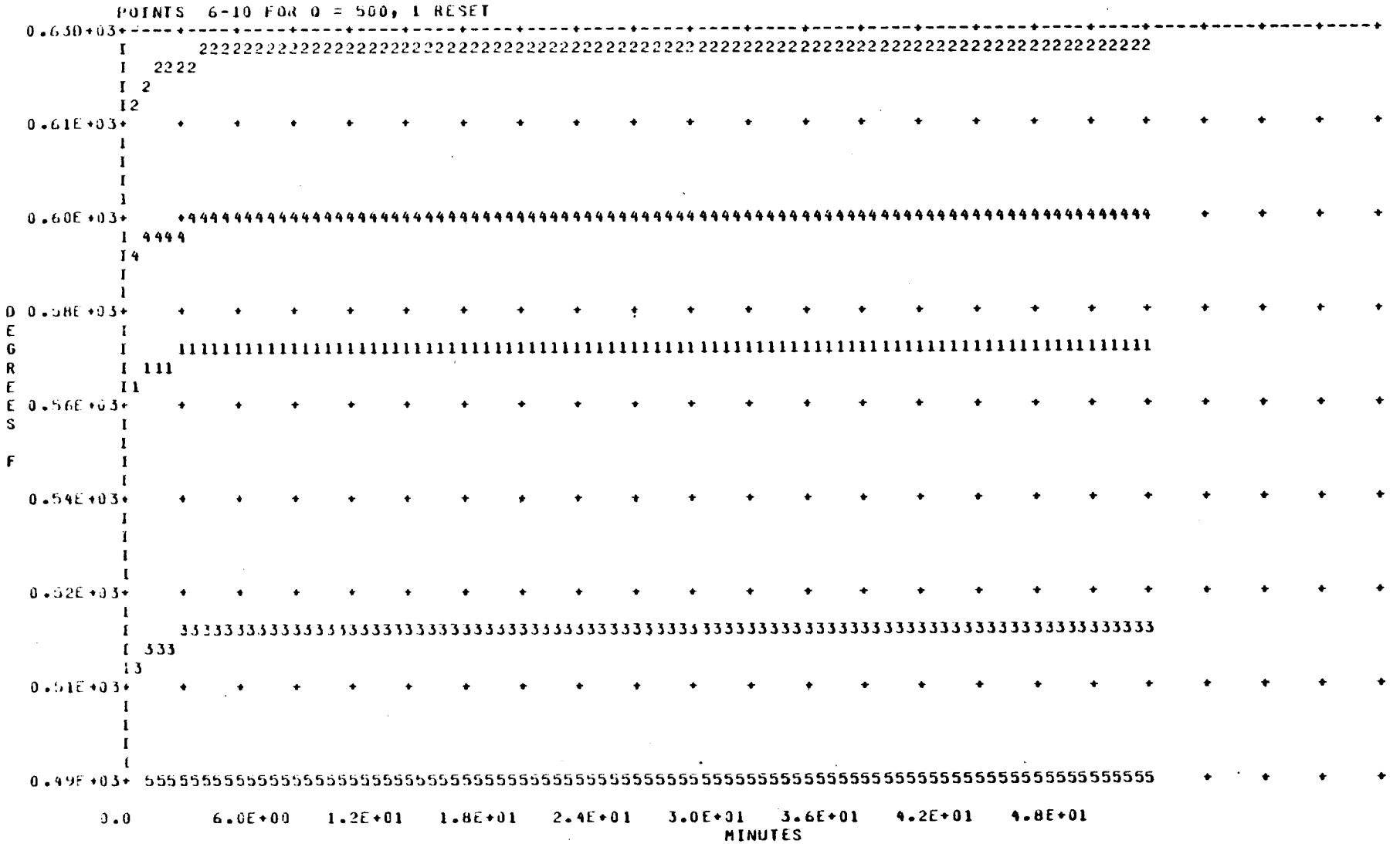

POINTS 1- 5 FOR Q = 500
0.62D+03
0.60E+03
0.58E+03
0.56E+03
0.53E+03
0.49E+03
0.46E+03
0.42E+03
0.39E+03
0.35E+03
DEGREES F
0.0
6.0E+00
1.2E+01
1.8E+01
2.4E+01
3.0E+01
3.6E+01
4.2E+01
4.8E+01
MINUTES

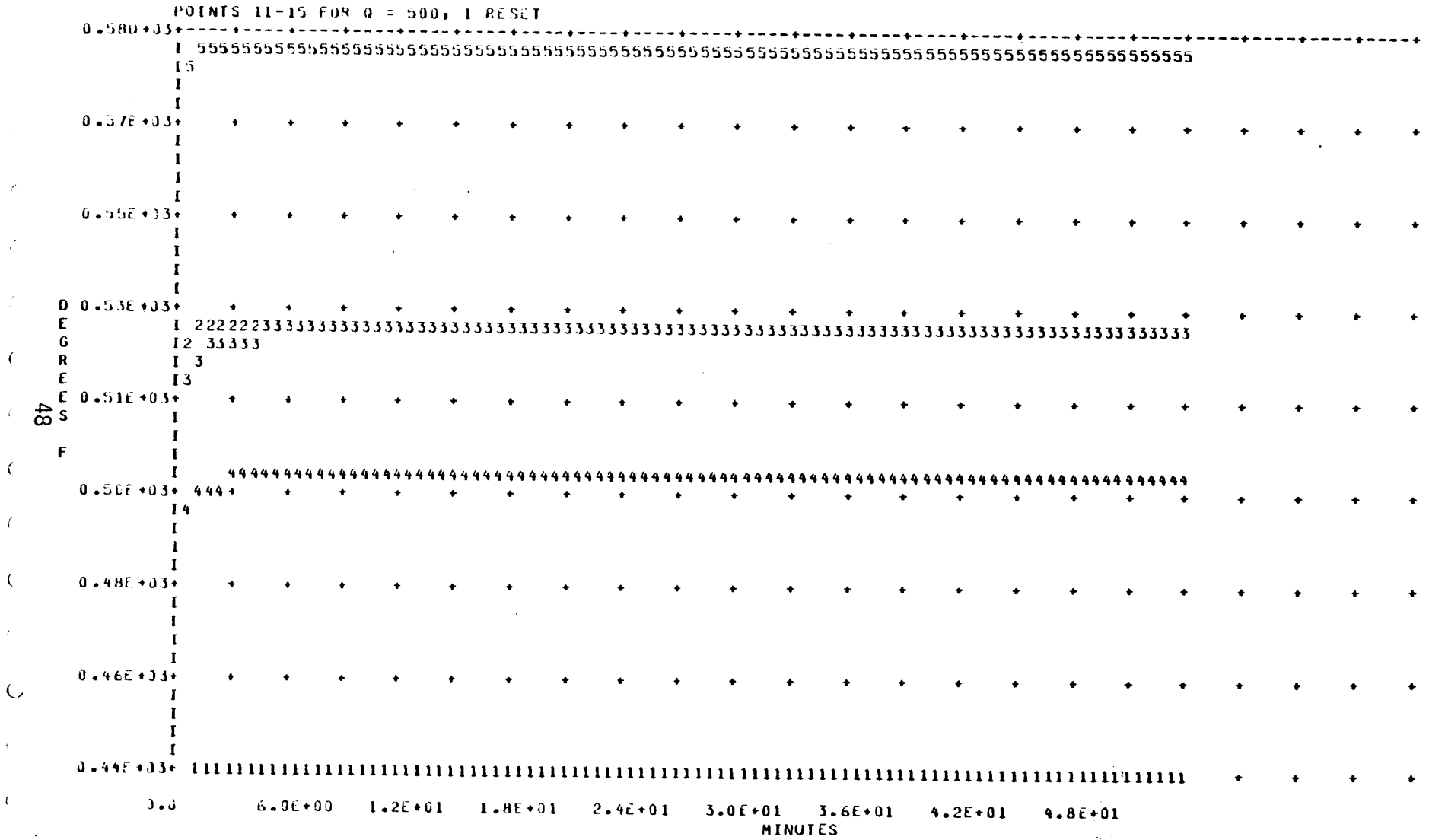

POINTS 6-10 FOR Q = 500
0.610+03
0.59E+03
0.56E+03
0.53E+03
0.51E+03
0.48E+03
0.45E+03
0.42E+03
0.39E+03
DEGREES F
0.0
6.0E+00
1.2E+01
1.8E+01
2.4E+01
3.0E+01
3.6E+01
4.2E+01
4.8E+01
MINUTES

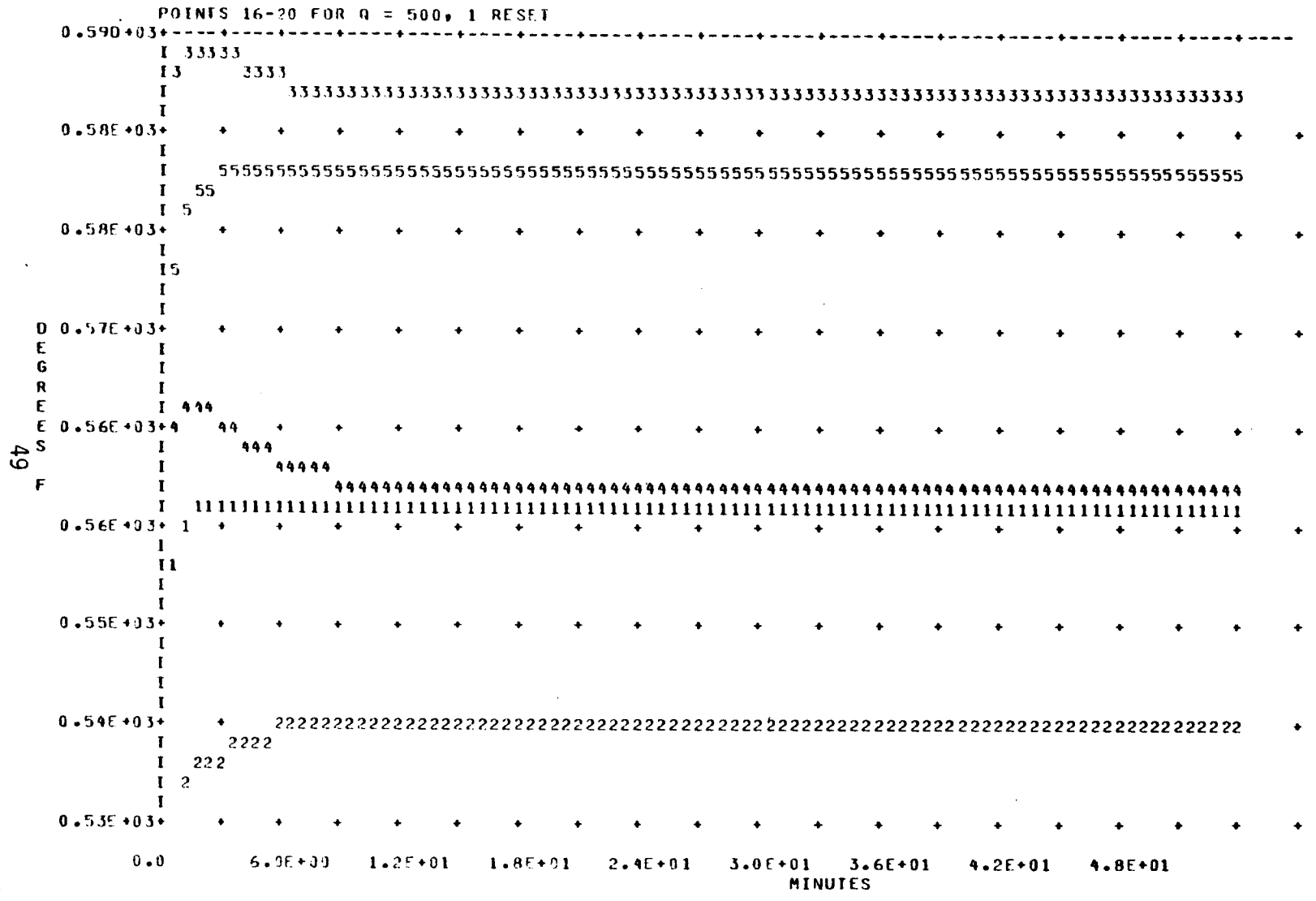

POINTS 11-15 FOR Q = 500
0.58E+03
0.55E+03
0.53E+03
0.50E+03
0.47E+03
0.44E+03
0.42E+03
0.39E+03
0.36E+03
DEGREES F
0.0
6.0E+00
1.2E+01
1.8E+01
2.4E+01
3.0E+01
3.6E+01
4.2E+01
4.8E+01
MINUTES


POINTS 16-20 FOR Q = 500
0.59E+03
0.57E+03
0.55E+03
0.53E+03
0.51E+03
0.49E+03
0.48E+03
0.46E+03
0.44E+03
DEGREES F
0.0
6.0E+00
1.2E+01
1.8E+01
2.4E+01
3.0E+01
3.6E+01
4.2E+01
4.8E+01
MINUTES


POINTS 1- 5 FOR Q = 500, 1 RESET
0.62D+03
0.60E+03
0.57E+03
0.55E+03
0.52E+03
0.50E+03
0.47E+03
0.44E+03
0.42E+03
DEGREES F
0.0
6.0E+00
1.2E+01
1.8E+01
2.4E+01
3.0E+01
3.6E+01
4.2E+01
4.8E+01
MINUTES


POINTS 6-10 FOR Q = 500, 1 RESET
0.63E+03
0.61E+03
0.60E+03
0.58E+03
0.56E+03
0.54E+03
0.52E+03
0.51E+03
0.49E+03
DEGREES F
3.0
6.0E+00
1.2E+01
1.8E+01
2.4E+01
3.0E+01
3.6E+01
4.2E+01
4.8E+01
MINUTES


POINTS 11-15 FOR Q = 500, 1 RESET
0.58D+03
0.57E+03
0.55E+03
0.53E+03
0.51E+03
0.50F+03
0.48E+03
0.46E+03
0.44E+03
DEGREES F
0.0
6.0E+00
1.2E+01
1.8E+01
2.4E+01
3.0E+01
3.6E+01
4.2E+01
4.8E+01
MINUTES


POINTS 16-20 FOR Q = 500, 1 RESET
0.590+03
0.58E+03
0.58E+03
0.57E+03
0.56E+03
0.56E+03
0.55E+03
0.54E+03
0.53E+03
DEGREES F
0.0
6.0E+00
1.2E+01
1.8E+01
2.4E+01
3.0E+01
3.6E+01
4.2E+01
4.8E+01
MINUTES

VITA

Richard N. Bryan was born in Birmingham, Alabama on April 2, 1949. He attended elementary schools in Gadsden, Alabama and Nashville, Tennessee and graduated from Maplewood High School in June, 1967. He received a Bachelor of Arts degree with a major in Physics from David Lipscombe College in August, 1972. After earning a Bachelor of Science degree from the University of Tennessee at Nashville, in August 1975, he was employed by Link Electric and Safety Control Company. In 1977, he enrolled in the Graduate School of the University of Tennessee at Knoxville, and in December of 1979 received the Master of Science degree, with a major in Electrical Engineering.