AEC
RESEARCH
and
DEVELOPMENT
REPORT

BNWL-348

2-

**INTERIM REPORT
ON CONTROL PROGRAM DEVELOPMENT**

J. R. KOSOROK

DECEMBER, 1966

<i>J. E. Panisko 30839 309</i>		<i>MAR 30 1970</i>

BATTELLE-NORTHWEST

BATTELLE MEMORIAL INSTITUTE / PACIFIC NORTHWEST LABORATORY

LEGAL NOTICE

This report was prepared as an account of Government sponsored work. Neither the United States, nor the Commission, nor any person acting on behalf of the Commission:

A. Makes any warranty or representation, expressed or implied, with respect to the accuracy, completeness, or usefulness of the information contained in this report, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or

B. Assumes any liabilities with respect to the use of, or for damages resulting from the use of any information, apparatus, method, or process disclosed in this report.

As used in the above, "person acting on behalf of the Commission" includes any employee or contractor of the Commission, or employee of such contractor, to the extent that such employee or contractor of the Commission, or employee of such contractor prepares, disseminates, or provides access to, any information pursuant to his employment or contract with the Commission, or his employment with such contractor.

PACIFIC NORTHWEST LABORATORY

RICHLAND, WASHINGTON

operated by

BATTELLE MEMORIAL INSTITUTE

for the

UNITED STATES ATOMIC ENERGY COMMISSION UNDER CONTRACT AT(45-1)-1830

PRINTED BY/ FOR THE U. S. ATOMIC ENERGY COMMISSION

3 3679 00060 3151

BNWL-348
UC-37, Instruments

INTERIM REPORT
ON CONTROL PROGRAM DEVELOPMENT

BY

J. R. Kosorok
Computers and Controls Section
Applied Physics and Electronics Department

DECEMBER, 1966

FIRST UNRESTRICTED
DISTRIBUTION MADE

JAN 24 '67

PACIFIC NORTHWEST LABORATORY
RICHLAND, WASHINGTON

Printed in USA. Price \$3.00. Available from the
Clearinghouse for Federal Scientific and Technical Information
National Bureau of Standards
U. S. Department of Commerce
Springfield, Virginia

INTERIM REPORT
ON CONTROL PROGRAM DEVELOPMENT

ABSTRACT

Hardware and software requirements of a successfully implemented direct digital control program for multiloop control are defined. Test results of the program with a simulated process show that sampling intervals of one-twelfth to one-fifth the effective process time constant give stable control for a process having an effective dead time approximately equal to the process time constant.

TABLE OF CONTENTS

LIST OF FIGURES	iv
INTRODUCTION	1
SUMMARY	2
DIGITAL CONTROL PROGRAM	2
Program Description.	3
Test Results	6
PROGRAM DEVELOPMENT	13
Process-Computer Communication	15
Operator-Computer Communication.	17
Controller Output Calculation	18
Response Criterion	20
ACKNOWLEDGEMENTS.	22
APPENDICES	
A. Program Operating Instructions	23
B. Listing of Controller Program	27
C. Program Extensions	57
D. The GE 412 Process Control Computer.	61
BIBLIOGRAPHY	67

LIST OF FIGURES

1.	Direct Digital Control of a Process	4
2.	Process Simulation Diagram	7
3.	Process Response to a Setpoint Change at Different Sampling Intervals	10
4.	Effects of Filtered and Unfiltered Noise on the Process Input	11
5.	Process Response Using MIAE Error Criterion	12
6.	Flow Chart for the Controller Program	14
7.	Flow Chart for Executive Control Routine	16
8.	Flow Chart for the Calculation Routine	19
9.	Error Function for the Calculation Routine	19
10.	Modified Integral of Absolute Error (MIAE)	21
11.	Operator Panel	23
12.	Flow Chart for Controller Program	28
13.	Possible Error Functions for the Calculation Routine	60

INTERIM REPORT
ON CONTROL PROGRAM DEVELOPMENT

INTRODUCTION

This report covers the design and use of a digital computer program that simulates three-mode (proportional-plus-integral-plus-derivative) controllers. The phase of the controller study project reported here yielded general guides for the design of digital controller programs, as well as useful rules for choosing sampling intervals that give stable control. All the tests were conducted with disturbances that were step changes in the setpoint. This study phase was concluded when the computer used to develop the controller program was moved to another location.

Future work will include further tests of the controller program using real-time dynamic simulations of FFTF processes. These tests are expected to provide information relating the effects on the controlled variables of disturbances which occur in different parts of the closed control loops.

The FFTF instrumentation prototype facility, presently under construction, will facilitate the evaluation of sampling interval and control method effects on critical FFTF processes. This prototype facility includes a PDP-7 digital computer, a Beckman 2133 analog computer and appropriate interfacing equipment. The programming methods described in this report will be adapted for use with the PDP-7 computer.

SUMMARY

A program developed for the GE 412 process computer controls any number of process loops with a direct digital control algorithm that is an approximation to a proportional plus-integral-plus derivative controller. For each process loop the operator can adjust each of the control parameters including a dead-band and integral term initial conditions. The operator can transfer from manual to computer control, or vice versa, at any time. The program evaluates an error criterion to aid the operator in optimizing a controller. The program is approximately 1100 words long, 270 of which contain the control algorithm. The rest of the program provides computer-process and computer-operator communication.

Although this report emphasizes the development of the control program, control tests were made using an analog computer simulation of a process. Stable control with setpoint step disturbances was achieved with sampling intervals of one-fifth to one-twelfth the effective process time constant, using a process dead time approximately equal to the effective first-order process time constant.

The listed references provide application and general background information on the subject of direct digital control.

The first section of this report, "Digital Control Program," describes the program in general terms and gives significant results from tests of the program. The second section, "Program Development," discusses the design of the program by detailing its four subroutines which allow the computer to communicate with the process and the operator, to calculate the controller outputs, and to evaluate the process response criterion. Four appendices give the additional program details of operating instructions, complete program listing, possible program extensions, and significant information about the GE 412 computer.

DIGITAL CONTROL PROGRAM

This section of the report describes the program in general terms and gives significant results from tests of the program.

PROGRAM DESCRIPTION

Three major problems in digital control that are solved with this program are: (1) The single process computer, which executes instructions sequentially, must control a number of parallel loops operating simultaneously. This requires the computer to synchronize its operation with the process so that calculated time intervals agree with actual elapsed time intervals. (2) The operator must be able to change the controlling action, that is, he must be able to "tune" the controller. (3) The computer must calculate the process control signals. This program solves these problems by providing communication between the analog process and digital computer, communication between the operator and computer, and computation of control values.

Figure 1 shows how these three functions are related. The communication paths from the process to the computer are via the multiplexed, analog-to-digital converter (ADC). The output from one of the process variable transducers that provide the control variables is switched to the ADC at the proper time under program control. Because the ADC is complex (and thus expensive), several inputs are switched to one ADC. The digital-to-analog converter (DAC) converts the digital control signals from the computer to analog signals; the holding device then connects these signals to the process. A zero-order hold, in which the last value is maintained until a new value is presented, is the most common hold. In the computer-process communication link, the control program reads the inputs and provides the outputs for the various loops at the proper time.

The operator, who is slow compared to the computer, communicates with the computer to change the proportional,

FIGURE 1. Direct Digital Control of a Process

integral, and derivative constants; the sampling intervals; and the setpoints. Since the operator may wish to modify the controller characteristics while the controller is operating, the program allows control and operator communication simultaneously. The characteristics for each loop can be changed independently of each other and of the other loops.

The Direct, Digital, Control (DDC) Algorithm in Figure 1 is a discrete approximation to a three-mode controller. The error is the difference between the setpoint and the process output. A control signal is the sum of three terms:

- Term 1 is proportional to the error
- Term 2 is proportional to the time integral of the error
- Term 3 is proportional to the time derivative of the error.

Since time is involved in two places in the calculation, the program must synchronize the computer operation with real time.

One of the main features of this controller program is its ability to control any number of process loops. All the controller parameters, except the sampling period, can be adjusted independently for each loop. With the program running in a GE 412 process control computer, 100 loops can be serviced every 20 sec. Even with its relatively slow cycle time of 10 μ sec, the 412 spends most of this time waiting for the electromechanical input-output multiplexer.

An example that shows the flexibility of digital control is the error criterion addition to the program. The operator may select a period of time for which the absolute value of the process error is integrated. By tuning the controller to minimize the error, he has a quantitative guide for optimizing the controller parameters. Other criteria may be used by simply modifying the program.

The ability to change the program, without changing the capital equipment, is an advantage when the process is highly complex. As the plant model is made more exact using operating experience, the control algorithm may be improved.

TEST RESULTS

All the operating experience with the controller program was obtained by controlling processes that were simulated on a small analog computer. The three different process simulations were:

1. A second order system with transport lag
2. A second order system
3. Two second order systems.

Figure 2 is a diagram of one simulation.

Much of the information about the effectiveness of the controller program was collected by a technician who operated it after a few hours of instruction. He quickly grasped the idea of the digital controller, because of his experience with analog controllers.

Two simulated process loops were controlled simultaneously to illustrate the multiloop control capability. Since two loops were stably controlled with no program-caused interaction, the program should be able to control any number of loops. In an application, the number of controlled loops would be limited by the number of ADC input points, the number of DAC's, the available memory space, and the minimum sampling interval.

Additional controller tests showed the effects of varying the sampling interval and of using low pass filtering on the derivative term. In all cases, the optimixing feature was used to adjust the controller variables.

1 FROM GE412 DAC AND ZERO ORDER HOLD τ_i TIME CONSTANT, OR LAG, IN SECONDS
TO STRIP CHART RECORDER

3 TO GE412 SAMPLER AND ADC

4 TO STRIP CHART RECORDER

FIGURE 2. Process Simulation Diagram

Sampling Interval

Controller tests show that sampling intervals of one-twelfth to one-fifth the minimum process time constant are adequate for feedback process control. As the sampling interval increases, there is an increase in output error and sensitivity to controller parameter changes. However, response time and stability remain relatively constant almost to the longest interval. Controller action rapidly deteriorates above one-fifth and shows slight improvement below one-twelfth.

The sampling interval was varied from 0.2 to above 2 sec. Since the process was simulated with two, 10 sec time constants and a 10 sec transport lag, the intervals correspond to sampling intervals of one-fiftieth to greater than one-fifth of a minimum time constant. Output error, response time, stability, and sensitivity to controller parameter changes were compared at the different sampling intervals. In all cases, the disturbance was a 10% input error step.

Output error was measured at 25 time constants from the input step and was defined as the difference, in percent of full scale, between the output and the setpoint. This output error was 0.1% for 0.2 sec sampling interval, 0.2% for a 0.4 sec interval, and 0.5% for a 2 sec sampling interval.

The closed-loop response time, which is the time for the process output to change from 10 to 90% of the input error step, remained almost constant for different sampling intervals (periods).

Closed loop stability was estimated by comparing the maximum overshoot and undershoot for different sampling rates. Stability was almost constant for the useable rates.

A qualitative indication of sensitivity is the degree of change in the modified integral of absolute error with a change of the proportional or integral constant. At shorter sampling intervals, the error integral was less sensitive to parameter changes than at longer intervals.

Figure 3 shows process response for sampling periods (T) of 0.2 sec and 2 sec. The response time is labeled "TR". Process responses for just these two periods are shown because the responses are similar for different intervals.

Filtering

Simple filtering was attempted to decrease the noise caused by differentiating. Ordinarily, in the difference approximation, the derivative is formed by dividing the difference between two errors at successive sampling instants by the sampling period. The differentiation noise was reduced by dividing the difference of errors separated by two or more sampling periods by the time between the errors.

This procedure reduced the noise, but the advantage from this was lost by incorrectly using the derivative term. Instead of calculating the derivative term every iteration it was calculated every other iteration; and the stored value was used when the derivative wasn't calculated. Not calculating the derivative every iteration nullified the filtering effect. Figure 4 shows the process input with noise filtering.

Optimizing

The controller was "tuned" until the Modified Integral of Absolute Error (MIAE) was minimized. A minimum MIAE was considered the optimum controller setting. Parts a and c of Figure 5 show optimum closed loop responses for three mode and proportional-plus-integral mode, respectively. Part b shows the response with three mode control before the controller was optimized. Notice that the MIAE in part b is more than twice.

FIGURE 3. Process Response to a Setpoint Step Change at Different Sampling Intervals

FIGURE 4. Effects of Filtered and Unfiltered Noise on the Process Input

FIGURE 5. Process Response Using MIAE Error Criterion

the MIAE of part a. The same process was used for the three parts of the figure. The error criterion provided a convenient method for rapidly tuning the controller. Other optimizing criteria could be evaluated with changes in programming.

PROGRAM DEVELOPMENT

The system program for digital control has three major functions and a fourth minor function:

- (1) It provides process-computer communication by responding to the priority interrupts from the elapsed time clock and the input-output multiplexer.
- (2) It provides operator-computer communication during program execution by accepting program modifications from peripheral equipment. The operator can also request that the program halt and allow manual control until he requests a return to the automatic mode.
- (3) It calculates the new process input, or controller output.
- (4) It determines an effectiveness of response criterion that is used to optimize the controller settings.

The flow chart for the program is in Figure 6. The broken lines following blocks 3, 8, and 11 indicate that the program jumps from a closed waiting loop to the subroutine associated with the interrupt. Since the multiplexer is much slower than the computer's central processor, the processor must wait until the multiplexer acknowledges its output or input commands. For example, several computer commands can be executed between the command to sample the process output (block 5) and the command to use this output (block 9). All the waiting time in this program could be used for data logging or other functions. The four program functions listed above correspond to the flow chart as follows:

FIGURE 6. Flow Chart for Controller Program

function 1 (process-computer communication) to blocks 1, 2, 3, 4, 5, 8, 10, 11, 16; function 2 (operator-computer communication) to blocks 6, 7, and 12; function 3 (calculation) to block 9; function 4 (response criterion) to blocks 14 and 15.

Each one of the four functions is mechanized as one or more subroutine of the control program and will be discussed in the following order: (1) Process-computer communication, (2) operator-computer communication, (3) controller output calculation, and (4) response criterion.

PROCESS-COMPUTER COMMUNICATION

Process-computer communication is the control program's main function. The unifying thread, or executive control, for the program emphasizes this. The executive control calls for process output samples, starts controller calculations, converts controller outputs, calls for optimizing calculations, and allows operator-computer communication while maintaining a strict timing schedule.

In Figure 7 the command execution sequence in automatic control is from block 6 through block 23 and then back to block 6. Blocks 1 through 5 are executed everytime the operator restarts the program after manual intervention.

The test in block 8 is included in the program to multiply the basic sampling interval range by ten. If one of the basic sampling intervals in the range from 0.2 to 0.8 sec is chosen, a controller calculation is made after each time interrupt. However, if the larger range is used, a controller calculation is made after every tenth time interrupt.

FIGURE 7. Flow Chart for Executive Control Routine

Blocks 13 and 14 introduce low pass filtering for the derivative action of the controller. The derivative is calculated by dividing the difference of two errors by the time between them. The operator can select times between errors of 1 to 15 sampling intervals. Program tests showed that this feature did reduce the noise amplification caused by differentiating. However, it can be made more effective by calculating the derivative for every sampling period. This would expand the memory requirements since, for each loop, the program would have to store as many past errors as periods between derivative calculations.

OPERATOR-COMPUTER COMMUNICATION

The program allows the operator to transfer from automatic to manual operation and then back to automatic operation or to tune the controller by modifying constants in the calculation routine. Since continuous process communication is of prime importance, the program allows operator communication only when it won't interfere with the process. However, the computer's response to the operator's changes seems to occur instantaneously.

Controller variables that the operator may change are

- Proportional constant
- Integral constant
- Derivative constant
- Sampling period
- Setpoint
- Initial condition on error integral
- Error dead-band
- Number of sampling periods for calculation of derivative term.

In addition, he may set the slope of the ramp function used in the response criterion calculation (see "Response Criterion") below.

CONTROLLER OUTPUT CALCULATION

Figure 8 shows the sequence in which the controller output calculations are performed. The units for the setpoint, process output, error, and controller output are percent of full scale.

Error, in percent, is in the setpoint minus the process output. The function in Figure 9 which is a part of the calculation program, transforms the actual error to that used in the succeeding calculations. The operator sets the percent width of the dead-band from zero to 100% of full scale. Since this function is programmed, it could be changed by simple program changes.

The program for calculating the controller output is a difference approximation to the following expression,

$$M = K_p e + K_p K_I \int e dt + M_0 + K_p K_D \frac{de}{dt}$$

where the constants and variables are defined as follows:

M = controller output, percent of full scale

K_p = proportional constant, dimensionless

e = error, percent of full scale

K_I = integral constant, sec^{-1}

t = time, sec

M_0 = initial condition, percent of full scale

K_D = derivative constant, sec.

This single routine is time-shared by all the control loops, with time as the only common variable. There was no interaction when the controller program was tested with two process loops.

In Figure 8, block 2 corresponds to the term $K_p e$; block 5 to the term $M_0 + K_p K_I \int e dt$, and block 7 to the term $K_p K_D \frac{de}{dt}$.

FIGURE 8. Flow Chart for the Calculation Routine

FIGURE 9. Error Function for the Calculation Routine

Each time one of the three terms is added to the controller output to update it, the magnitude of the output is clamped at a fixed percent of full scale. For most of the tests of the program, the clamp value was 100%. This corresponds to saturation in an analog controller. Unlike some analog controllers, however, the digital controller has a zero saturation recovery time. That is, as soon as the magnitude of the sum of the three terms is less than 100%, the controller comes out of saturation. If the integral were saturated at +100%, it would decrease at the instant a negative error was detected.

The time differential, dt , is approximated by a time difference. If the time difference is not known exactly, the program does not calculate the two terms containing time. An unknown time difference could occur immediately after the automatic mode is entered or immediately after the sampling interval is changed.

RESPONSE CRITERION

A modified integral of absolute error (MIAE) is calculated and displayed to the operator to aid him in optimizing controller settings. Figure 10 illustrates how the program calculates the MIAE. At time T_1 the operator changes the setpoint and the process response is from D to C along the curved line. The program generates the time function D-B-C with the slope of D-B provided by the operator. For example, the modified error at time T_2 is the difference between D-B-C and the process response. The modified error is integrated for a length of time, chosen by the operator, to obtain the MIAE. The operator may vary the proportional, integral, and derivative constants successively

until he obtains the minimum MIAE. A more common error criterion is the integral of absolute error (IAE)⁽¹⁾, which would yield the area A-B-C-D-A. Notice that the large area A-B-D with the IAE would make a variation in the process response almost negligible.

FIGURE 10. Modified Integral of Absolute Error (MIAE)

(1) S. C. Gupta and C. W. Ross. "Simulation Evaluation of a Digital Control System," *ISA Transactions*, pp. 271-279. July, 1964.

ACKNOWLEDGEMENTS

The author thanks C. D. Swanson, Applied Physics and Electronics Department, for his willingness to discuss process control, for his process simulations, and for his help in writing the report. L. D. Coffin, Mathematics Department, and E. Shimoda, Chemistry Department, were most cooperative in helping the author learn to program the GE 412. K. L. Chubb, Applied Physics and Electronics Department collected much of the test results for this report.

APPENDICES

APPENDIX A
PROGRAM OPERATING INSTRUCTIONS

The controller program is implemented in GE 412 machine language. In this form, the operator uses the 412 programming console and operator panel to communicate with the computer. These operating instructions describe the console and panel, including program parameters that can be changed, and explain how parameter changes may be fed into the program.

CONSOLE AND PANEL DESCRIPTION

The "Demand" key (a lighted push button) is the only key on the programming console that can be used when the program is running.

Figure 11 depicts those features of the operator panel that are used with this program. Each of the DATA, MODE, and POINT knobs may be set to any decimal digit. The MULT knob may be set to +1 or +10, only. The keys, which are internally lighted push buttons, labeled DATA INSERT and DATA SELECT, are set by the operator, but are reset by the program.

FIGURE 11. Operator Panel

The DATA knobs set the numerical values of the controller parameters. For example, the process output setpoint may be any value in the range of 0 to 100% with 0.1% steps. A number 501 on the knobs is 50.1%. The program reads the MULT knob only when the sampling interval is changed, and then only recognizes a +1 or +10.

MODE settings identify the variable whose value is set on the DATA knobs. For example, the setpoint is changed to 50.1% only if the MODE knobs are set to 8.

Since there may be any number of process loops, each loop is assigned a number starting from zero. If there were four loops, the POINT knobs would have to be set to 0, 1, 2, 3, in succession to make a setpoint change in all the loops.

After the operator fixes each of the DATA, MULT, MODE, and POINT knobs to the desired digit, he pushes the DATA INSERT KEY (DIK) to set it. When the light turns off, indicating the DIK is reset, the program is ready to accept more information from the operator panel. Setting the DATA SELECT KEY requests that the modified integral of absolute error be displayed.

CONTROLLER PARAMETERS

Any of the numbers 0 through 13 may be set on the MODE knobs to request changes in 14 different parameters or sets of parameters. Table I concisely defines the knob settings.

CONTROLLER TUNING

There are two procedures for inserting information into the controller program via the operator panel. One procedure is used when the program is started and before it is in the automatic mode; the second is used for the automatic control mode.

TABLE 1. *MODE, DATA, MULT, and POINT Settings for Controller Parameters*

MODE	DATA	MULT	POINT	COMMENTS
00	XXX.X	--	Any loop	Initial condition. 0 to 100%
01	XX.XX	--	Any loop	Proportional constant. 0 to 99.99
02	XXX.X	--	Any loop	One-half of dead-band. 0 to 100%
03	XXX.X	--	Any loop	Derivative constant. 0 to 999.9
04	OOXX	--	Any loop	Number of sampling intervals to calculate derivative. 1 to 15
05	X.XXX	--	Any loop	Integral constant. 0 to 9.999
06	OXXX	+1 or +10	--	Number of 3.2 msec clock pulses per sampling interval. 63 to 255
07	--	--	--	Initial condition set to last controller output for every loop
08	XXX.X	--	Any loop	Setpoint for process output. 0 to 100%
09	XXXX	--	Any loop	Clamp value for magnitude of the proportional term. 0 to 1600%
10	XXXX	--	Any loop	Clamp value for magnitude of derivative term. 0 to 1600%
11	XXXX	--	Any loop	Clamp value for magnitude of integral term. 0 to 1600%
12	XXXX	--	--	Number of process loops
13	OOXX	--	Any loop	Time, in seconds, to the 63% point of the ramp in the MIAE criterion. 15 to 50 sec

Starting Program

Before starting automatic control, the operator may use, as often as he wishes, any of the 14 MODE numbers for any loop. The operator requests this by setting the "Demand" key on the programming console. After the "Demand" key is set once, the program reads the knob settings from the operator panel every-time the DATA INSERT KEY (DIK) is set. When the "Demand" key is set a second time, the automatic controller action starts. If the operator does not wish to insert data initially, he may set the "Demand" key twice in succession.

Automatic Control

While the program is controlling, the controllers may still be modified. However, the operator knobs are read only during "waiting" time. Generally, the operator will notice little difference in the response to his request (made by setting the DIK), while the program is controlling or is in the start mode. The operator knobs should not be moved until the DIK is reset, which occurs when the light turns off. The operator may set the "Demand" key when he wishes to halt the controller action. To return to the controller action, the "Demand" key must then be set two more times.

A second way to halt the controller action is by setting the DATA SELECT key. In this case, the MIAE is calculated for a fixed length of time. After the time has elapsed, the MIAE and the calculation period are displayed in the "A" and "Q" registers, respectively. Setting the "Demand" key will then send the program to the point where the "Demand" key must be set two more times for automatic control.

APPENDIX B

LISTING OF CONTROLLER PROGRAM

Figure 12, which is a somewhat more detailed flow chart than Figure 6, and the program listing show how the various program functions are implemented in machine language. In the left column of the listing, machine commands are numbered sequentially, in octal digits, according to their locations in the computer's memory. Table II, below, relates blocks in Figure 12 to command addresses in the listing.

TABLE II. *Relation Between Blocks in the Flow Chart and Command Addresses in the Listing*

<u>NUMBER OF BLOCKS IN THE FLOW CHART</u>	<u>COMMAND ADDRESSES (OCTAL)</u>
1, 2, 3, 4, 5	2003 - 2212
6, 10, 13	2001 - 2002
7, 8, 9	2357 - 2421
11, 12	2277 - 2336 and 2540 - 3242
16, 17, 18, 19, 20 21, 22, 23	2337 - 2356 and 2430 - 2516
14, 15	3243 - 4200

FIGURE 12. Flow Chart for Controller Program

4002000	SLC /2000		02
	*	***** DUSMOR *****	04
	*	DIGITAL COMPUTER SIMULATION OF	05
	*	THREE MODE CONTROLLER	06
	*	**	07
02000	2602024	DUSMOR BRU STARTE	08
	*	THE PURPOSE OF OTMOD IS TO SET MO	10
	*	AS THE INITIAL OUTPUT FOR ALL	20
	*	THE CONTROLLERS	30
	*	PAUSE WILL BE USED BY ALL SEGMENTS	40
	*	*****	50
	*	PAUSE FOLLOWS	60
	*	*****	70
02001	2500012	PAUSE PAI	80
02002	2602001	BRU **1	90
	*	*****	100
	*	OTMOD FOLLOWS	110
	*	*****	120
02003	0623120	OTMOD LDX ZERO	130
02004	0023143	D1 LDA MO	140
02005	0323145	STA INTVAL	150
02006	2400005	SR4 5	160
02007	2510101	LCV 1	170
02010	0022250	LDA COPI	180
02011	2510103	LSC 1	190
02012	0002240	LDA CMTBL3+16	200
02013	0300014	STA 12	210
02014	2504000	NOP	220
02015	2602001	BRU PAUSE	230
02016	1420001	D2 INX 1	240
02017	0000001	LDA 1	250
02020	0202273	SUB NOCONT	260
02021	2514001	BPL 1	270
02022	2602141	BRU E11	280
	*	*****	290
02023	2602004	BRU D1	300
	*	*****	310
	*	STARTE ,START SEGMENT,FOLLOWS	320
	*	*****	330
02024	2500013	STARTE IAI	340
02025	2514013	BTC 1,1	350
02026	2504000	NOP	360
02027	2514015	BTC 1,3	370
02030	2504000	NOP	380
02031	2500015	SAI 2	390
02032	0002274	LDA TIM1	400
02033	2500017	LTC 1	410
		COUNTER 1 ELAPSES IN 3.2 MSEC OR LESS	420

02034	0002275	LDA TIM3	COUNTER 3 ELAPSES IN 33 TO 50 MSEC.	430
02035	2500021	LTC 3	ENOUGH TIME TO SERVICE12 INTERRUPTS	440
02036	0663120	LDX ZERO	3 CLEAR IR3	450
02037	0002253	LDA BRPSE	3 BRU TO PAUSE IN 8+IR3	460
02040	0360010	STA 8	3	470
02041	1460001	INX 1	3 TEST IF HAVE STORED BRU TO PAUSE IN	480
02042	0477764	BXL 12	ALL AUTO. INTPTS.	490
02043	2602040	BRU BK3E1	CLEAR CLOCK 3 OVERFLOW SEGMENT	500
02044	0002257	LDA BC03	CLEAR CLOCK 1 OVERFLOW SEGMENT	510
02045	0300011	STA 9	AFTER LEAVING THIS LOOP, ALL INTERRUPTS	520
02046	0002256	LDA BC01	WILL BE CLEARED	530
02047	0300010	STA 8	OVERFLOW WOULD TURN ON INTERRUPT AGAIN	540
02050	2602001	BRU PAUSE	OVERFLOW WOULD TURN ON INTERRUPT AGAIN	550
02051	2514013	BTC 1,1	NOW, WANT TO PUT BRU* IN ALL INTERRUPTS	560
02052	2504000	NOP	EXCEPT THE FIRST BIT OF PI REG.	562
02053	2602001	BRU PAUSE	3 CLEAR IR3	563
02054	2514015	BTC 1,3	2 BRU TO 9 I IR2	564
02055	2504000	NOP	3 THIS IS START OF LOOP TO ENTER BRU*	566
02056	2602057	BRU GOON	IN LOCATIONS 9 THRU 19	567
02057	2500013	IAI	3	568
02060	0663120	LDX ZERO	3 TEST IF SHOULD STAY IN LOOP OR NOT.	570
02061	0642254	LDA BRUNE	NOT ALL INTERRUPTS HAVE BRU*	580
02062	0000002	LDA 2	INT, LOC. 9 THRU 19 HAVE BRU*	590
02063	0360011	STA 9	CLOCK 1 INTERRUPT CAUSES BRU TO TIMINI	600
02064	1460001	INX 1	***** BRU* MAY BE INSERTED HERE *****	610
02065	1440001	INX 1	CMTBL3 CONTAINS SWITCH VALUES	620
02066	0477765	BXL 11	SW. PT. A3 SET TO A4	630
02067	2602062	BRU BK4E2	SW. PT. A6 SET TO A7	640
02070	0002255	LDA BTIMI	SW. PT. B1 SET TO B2	650
02071	0300010	STA 8	SW. PT. C1 SET TO C2	660
02072	2504000	NOP	SW. PT. E3 SET TO E4	670
02073	0002220	LDA CMTBL3	SW. PT. E5 SET TO E7	680
02074	0303257	STA A3		690
02075	0002221	LDA CMTBL3+1		700
02076	0303265	STA A6		710
02077	0002222	LDA CMTBL3+2		720
02100	0303614	STA B1		730
02101	0002223	LDA CMTBL3+3		740
02102	0303740	STA C1		750
02103	0002224	LDA CMTBL3+4		760
02104	0302123	STA E3		770
02105	0002225	LDA CMTBL3+5		780
02106	0302134	STA E5		790

02107	0002537		LDA RSTDSK		814
02110	2510103		LSC 1	RESET DATA SELECT KEY	816
02111	0002260		LDA BRUE18		818
02112	0300014		STA 12	GO TO E18 ON SCANNER INTERRUPT	820
02113	2602001		BRU PAUSE		822
02114	0003770	E18	LDA KNBTBL+5		824
02115	2510103		LSC 1	RESET DATA INSERT KEY	826
02116	0002276		LDA BRUE2		828
02117	0300014		STA 12	GO TO E2 ON SCANNER INTERRUPT	830
02120	2602001		BRU PAUSE		832
02121	2516017	E2	BRD 2	TEST DEMAND KEY	840
02122	2602134		BRU E5	DEMAND KEY NOT DEPRESSED	850
02123	2504000	E3	NOP	DEMAND KEY DEPRESSED, THIS IS A SW.PT.	860
		*		1ST TIME THRU THIS SHOULD BRU TO E4	870
02124	2602124		BRU *	THE PROGRAM SHOULD NEVER STOP HERE	880
02125	0002234	E4	LDA CMTBL3+12		890
02126	0302123		STA E3	E3 SHOULD NOW HAVE BRU TO E9	900
02127	2514025	E10	OCT 2514025	TEST DATA INSERT KEY ON OPERATOR PANEL	910
02130	2603243	E8	BRU RKPLA	D.I.K. IS SET, SO GO TO RKPLA	920
02131	0002235	E13	LDA CMTBL3+13	D.I.K. IS RESET, SO SET SW. E5	930
02132	0302134		STA E5	SWITCH AT E5 SET TO E6	940
02133	2602121		BRU E2		950
02134	2504000	E5	NOP	THIS IS A SW. PT.	960
02135	2602135		BRU *	SHOULD NEVER REACH THIS COMMAND	970
02136	2602127	E6	BRU E10		980
02137	2602121	E7	BRU E2		990
02140	2602003	E9	BRU OTHOD		1000
		*		*****	1010
		*	THE FINAL PHASE OF STARTE FOLLOWS		1020
		*	*****		1030
02141	0004071	E11	LDA NTIMCO	TIMER CONTROL WORD	1040
02142	2500017		LTC 1	TIMER 1 STARTED	1050
02143	0623120		LDX ZERO	1 INITIALIZE CONTROLLER COUNTER	1060
02144	0002226		LDA CMTBL3+6		1100
02145	0303257		STA A3	TO A5 AT SW. PT. A3	1110
02146	0002227		LDA CMTBL3+7		1120
02147	0303265		STA A6	TO A8 AT SW. PT. A6	1130
02150	0002230		LDA CMTBL3+8		1140
02151	0303614		STA B1	TO B3 AT SW. PT. B1	1150
02152	0002231		LDA CMTBL3+9		1160
02153	0303740		STA C1	TO C3 AT SW. PT. C1	1170
		*	THE ABOVE	4 CHANGE SWITCH COMMANDS ARE	1180
		*	FOR READING	3 KNOBS WHILE RUNNING	1190
02154	0762161		SPB SUB2	INITIALIZE TIME VARIABLES	1200
02155	0002245		LDA CMTBL3+21	SET KNOB SWITCH TO BRU TO RKPLA	1202
02156	0302355		STA KNBSW		1204
02157	2504000		NOP		1206
02160	2602001		BRU PAUSE	LEAVE STARTE	1208

02227	2603270	BRU A8	1490
02230	2603617	BRU B3	1500
02231	2603742	BRU C3	1510
02232	2602277	BRU ADCOF	1520
02233	2602121	BRU E2	1530
02234	2602140	BRU E9	1540
02235	2602136	BRU E6	1550
02236	2603251	BRU A2	1560
02237	2603740	BRU C1	1570
02240	2602016	BRU D2	1580
02241	2602337	BRU DACOG	1590
02242	2603244	BRU A1	1600
02243	2603274	BRU MK0DB	1610
02244	2603731	BRU RESDKC	1620
02245	2603243	BRU RKPLA	1630
02246	0061030	CIP0 OCT 0061030	1640
02247	0061230	CIP0 FOR CONT. 2	1642
02250	3034316	COPI OCT 3034316	1650
02251	3034216	COPI OCT 3034216	1652
		* VARIABLES AS BRU COMMANDS	1660
02252	2602057	BRGN BRU GOON	1670
02253	2602001	BRPSE BRU PAUSE	1680
02254	2600011	BRUNE BRU 9	1690
02255	2602357	BTIMI BRU TIMINI	1700
02256	2602051	BC01 BRU CLOV1	1702
02257	2602054	BC03 BRU CLOV3	1704
02260	2602114	BRUE18 BRU E18	1706
		* ***** VARIABLES IN TABLES *****	1710
02261	0000000	SCLRSL DEC 0	1712
	6000001	SLC +1	1714
02263	0000000	SCLSLP DEC 0	1720
	6000001	SLC +1	1722
02265	0000001	NSCLSL DEC 1 819	1730
	6000001	SLC +1	1732
		* SINGLE VARIABLES	1740
02267	0000000	SCLRI DEC 0	1742
02270	0000000	SCLINT DEC 0	1750
02271	0000001	NSCLIN DEC 1 819	1760
02272	0010074	TIMCOM OCT 0010074	1770
02273	0000001	NOCNT DEC 1 819	1772
02274	0010001	TIM1 OCT 0010001	1773
02275	0010003	TIM3 OCT 0010003	1774
02276	2602121	BRUE2 BRU E2	1780
		* * * * *	120
		* ADCOF FOLLOWS	130
		* * * * *	140
02277	2504000	ADCOF NOP	150
02300	2514021	BGNCLC BCO 1,1	160
		TEST FOR ADC OVERFLOW	

02301	2602305	BRU F1	ADC OVERFLOWED	170
02302	2510144	RCV 1	NO OVERFLOW	180
02303	2400002	SRA 2	INPUT,B17,TO B19 IN A	190
02304	2602306	BRU F2		200
02305	0002426	LDA ODEB19		210
02306	2410005	SLA 5	B19 TO B14	220
02307	0322517	STA COB14	1 PRESENT VALUE OF PROC. INPUT	222
02310	2400005	SRA 5	B14 TO B19	223
02311	2504532	NEG		224
02312	0122422	ADD STPROT	1 THETA1,B19,IN A	230
02313	0323133	STA THETA1	1	240
02314	2514001	BPL 1	TEST SIGN OF A	250
02315	2504532	NEG	NEG. OF THETA1	260
02316	0122424	ADD DBTHET	1 DIFF. BETWEEN THETA1 AND DBTHET	270
02317	2516001	BPL 2	TEST IF THETA1 OUTSIDE DEADBAND	280
02320	2602323	BRU F3	MAG. OF THETA1 OK	290
02321	2504002	LDZ	THETA1 IN DEADBAND	300
02322	0323133	STA THETA1	1 THETA1 IS NOW ZERO	310
02323	2602540	BRU INCALC		320
02324	0023137	LDA CONTVA	1 CONTVA IS B17 FROM INCALC	440
02325	0102427	ADD TOB19	ROUND	450
02326	2400002	SRA 2	CONTVA,B17,TO B19 AND ROUNDED	460
02327	0323137	STA CONTVA	1	470
02330	2510101	LCV 1	CONTROLLER OUTPUT VAL.	480
02331	0022250	LDA COPI	1	490
02332	2510103	LSC 1	SET CONT. OUTPUT TO NEW VAL.	500
02333	0002241	LDA CMTBL3+17		510
02334	0300014	STA 12	BRU TO DACOG ON SCANNER INTERRUPT	520
02335	2504000	NOP	***** BRU* CAN BE INSERTED HERE *****	530
02336	2602001	BRU PAUSE		540
			END OF ADCOF	550
			*****	590
			DACOG FOLLOWS	600
			*****	610
02337	0023131	DACOG LDA FLAG1	1 RESET FIRST TIME FLAG BIT	700
02340	2400001	SRA 1		710
02341	2410001	SLA 1		720
02342	0323131	STA FLAG1	1 BIT 19,FIRST TIME,IS RESET	730
02343	2602430	BRU MOG1	MODIF. TO CALC. MIAE	740
02344	0000001	LDA 1		750
02345	0202273	SUB NOCONT		760
02346	2516001	BPL 2	TEST IF HAVE RUN OUT OF CONTROLLERS	770
02347	2602351	BRU G2	NO	780
02350	0623120	LDX ZERO	1 CONTROLLER NO. ZEROED	790
02351	2514017	BRD 1	TEST DEMAND KEY ON PROGRAMMING CONSOLE	800
02352	2602024	BRU STARTE	DEMAND KEY WAS SET	810
02353	2504000	NOP	***** CAN INSERT BRU* HERE *****	820
02354	2514025	OCT 2514025	DEMAND KEY WAS RESET,TEST DIK	830

02355	2504000	KNBSW	NOP		D,I,K. SET. THIS IS A SW. PT.	840
02356	2602001		BRU PAUSE		D,I,K. RESET	850
		*		END OF	DACOG	860
		*		*****		870
		*		TIMINI	FOLLOWS	880
		*		*****		890
02357	2514013	TIMINI	BTC 1,1		RESET CLOCK OVERFLOW TO REMOVE INTER.	900
02360	2504000		NOP			910
02361	0002272		LDA TIMCOM			920
02362	2500017		LTC 1		CLOCK STARTED	930
02363	0023131		LDA FLAG1	1	WANT TO TEST BIT 19, FIRST TIME	940
02364	2516000		BEV 2			950
02365	2602371		BRU 16		FIRST TIME SET, DO NOT CALC. RATE	960
02366	0022261		LDA SCLRSL	1	FIRST TIME RESET, MAY CALC. RATE	1000
02367	2504112		SBO			1010
02370	0322261		STA SCLRSL	1	SCLRSL UPDATED	1020
02371	0002267	I6	LDA SCLRI			1030
02372	2504112		SBO		SCLRI UPDATED	1040
02373	0302267		STA SCLRI			1050
02374	2516002		BZE 2		TEST IF START OF SAMPLE PERIOD	1060
02375	2602351		BRU G2		NO	1070
02376	0022246		LDA CIPO	1	YES	1080
02377	2510103		LSC 1		READ INPUT	1090
02400	0022261	I2	LDA SCLRSL	1		1160
02401	2514002		BZE 1		TEST IF CAN CALC. SLOPE THIS TIME	1170
02402	2602407		BRU 13		YES	1180
02403	0023131		LDA FLAG1	1	NO	1190
02404	2202217		ANA BIRST		BIT 1, CALCULATE SLOPE, IS RESET	1210
02405	0323131		STA FLAG1	1		1220
02406	2602414		BRU 14			1230
02407	0022263	I3	LDA SCLSLP	1		1240
02410	0322261		STA SCLRSL	1	INITIALIZE RATE TIME INTERVAL SCALER	1250
02411	0002217		LDA BIRST			1260
02412	2504512		CPL		1 IN BIT 1,0 IN ALL OTHERS	1300
02413	2323131		ORY FLAG1	1	CALCULATE SLOPE, BIT 1, IS SET	1310
02414	0002270	I4	LDA SCLINT			1320
02415	0302267		STA SCLRI		INITIALIZE INTERVAL SCALER	1330
02416	0002232		LDA CMTBL3+10			1340
02417	0300014		STA 12		SET SCAN SWITCH TO ADCOF	1350
02420	2504000		NOP		***** CAN INSERT BRU* HERE *****	1360
02421	2602001		BRU PAUSE			1370
		*		END OF	TIMINI	1380
		*	VARIABLES	AND	CONSTANTS IN TABLES	1390
02422	0000372	STPROT	DEC 250 B19			1400
	6000001		SLC +1		CONT NO 2	1402
02424	0000005	DBTHET	DEC 5 B19			1410
	6000001		SLC +1		CONT NO 2	1412
		*		END OF	TABLES	1510

02426	0001777	ODEB19	DEC 1023 B19	CONSTANTS	1660
02427	0000002	T0B19	DEC 2 B19		1670
		*	END OF	CONSTANTS	1680
		*	*****	*****	1740
		*	MODIFICATION	TO CALCULATE MIAE	1750
		*	*****	*****	1760
02430	0003166	MOG1	LDA ITINT	ITINT IS B4	1770
02431	2400004		SRA 4	B4 TO B8	1780
02432	0122521		ADD TAU	1 UPDATE TAU,B8	1790
02433	2410001	G4	SLA 1	1 LFT SHFT GIVES OVERFLOW AFTER 128 SC	1800
		*		CHNG TIME BY CHANGING NO. OF SHIFTS	1810
		*		AT G4 AND G5	1820
02434	2514003		BOV 1	TEST FOR ELAPSED TIME	1830
02435	2602502		BRU G8	TIME ELAPSED	1840
02436	2400001	G5	SRA 1	1 RT SHFT RESTORES TIME FOR 128 SEC	1850
02437	0322521		STA TAU	1 NEW TIME,B8	1860
02440	2504006		MAQ		1870
02441	1522523		MPY A	1 A,B6,IS CONSTANT FOR RAMP	1880
02442	0122535		ADD OSTPNT	1 OLD SET POINT,B14	1890
02443	0322525		STA ETAU	1 ETAU,B14,IS FNC, OF TIME USED TO CALC	1900
		*		MIAE	1910
02444	0022527		LDA AMB	1 AMB,B10,IS NEW SET POINT MINUS OLD	1920
02445	2514001		BPL 1	TEST SIGN OF AMB	1930
02446	2602474		BRU G6	AMB IS POS	1940
02447	0022531		LDA NSTPRO	1 AMB IS NEG,NSTPRO,B19	1950
02450	2410005		SLA 5	B19 TO B14	1960
02451	2504532		NEG		1970
02452	0122525		ADD ETAU	1 FORM ETAU MINUS NSTPRO	1980
02453	2514001	G9	BPL 1	TEST SIGN OF DIFF BETWEEN ETAU,NSTPRO	1990
02454	2602500		BRU G7	DIFF IS POS	2000
02455	0022531		LDA NSTPRO	1 DIFF IS NEG,ETAU HAS PASSED NEW SET PT	2010
02456	2410005		SLA 5	B19 TO B14	2020
02457	0322525		STA ETAU	1 ETAU CLAMPED AT NEW SET POINT	2030
02460	0222517	G10	SUB COB14	1 MODIFIED ERROR,B14	2040
02461	2516001		BPL 2	TEST FOR SIGN OF MOD ERROR	2050
02462	2504532		NEG	ABS VAL OF MOD ERROR	2060
02463	2410004		SLA 4	B14 TO B10	2070
02464	2504006		MAQ		2080
02465	1503166		MPY ITINT	ABS VAL OF MOD ERROR TIMES DELTA TAU.	2090
		*		RESULT IS B14	2100
02466	0122533		ADD MIAE	1 INTEGRAL OF MOD ERROR UPDATED,B14	2110
02467	2514003		BOV 1	TEST MAG OF INTEGRAL	2120
02470	2602502		BRU G8	INTEGRAL TOO BIG	2130
02471	0322533		STA MIAE	1 INTEGRAL,B14,OK	2140
02472	1420001	G12	INX 1	1 UPDATE CONTROLLER NO.	2150
02473	2602344		BRU G13	GO BACK TO DACOG	2160
02474	0022531	G6	LDA NSTPRO	1 NEW SET POINT SMALLER THAN OLD,AMB POS	2170
					2180

02475	2410005	SLA 5	B19 TO B14	2190
02476	0222525	SUB ETAU	1 IF THIS RESULT IS NEG, THEN CLAMP ETAU	2200
02477	2602453	BRU G9	TEST FOR SIGN OF DIFF	2210
02500	0022525	LDA ETAU	1 ETAU NOT CLAMPED YET	2220
02501	2602460	BRU G10		2230
02502	2514026	OCT 2514026	TEST DATA SELECT KEY	2240
02503	2602505	BRU G11	DSK IS SET	2250
02504	2602472	BRU G12	DSK IS RESET	2260
02505	0022521	LDA TAU	1	2270
02506	2504006	MAQ	DISPLAY TAU IN Q REG,	2280
02507	0022533	LDA MIAE	1 DISPLAY MIAE IN A REG	2290
02510	2516017	BRD 2	DISPLAY UNTIL DEMAND KEY IS SET	2300
02511	2602510	BRU *-1	DEMAND KEY RESET	2310
02512	0002537	LDA RSTDsk	DEMAND KEY SET	2320
02513	2510103	LSC 1	RESET DSK	2330
02514	2516022	BSC 2,1	WAIT UNTIL DSK IS RESET	2340
02515	2602514	BRU *-1	DSK STILL SET	2350
02516	2602024	BRU STARTE	DSK IS NOW RESET	2360
			*****	2370
		WHEN	HALTS, TAU IS IN Q REG,	2380
		AND	MIAE IS IN A REG,	2390
		***** INDEXED	VARIABLES FOR MIAE *****	2400
02517	0042300	COB14 DEC 550 B14	CONT NO 2	2410
	6000001	SLC +1		2420
02521	0170000	DEC 30 B8	CONT NO 2	2430
	6000001	SLC +1		2440
02523	0103146	DEC 4.2 B6	CONT NO 2	2450
	6000001	SLC +1		2460
02525	0034100	DEC 450 B14	CONT NO 2	2470
	6000001	SLC +1		2480
02527	0144000	DEC 100 B10	CONT NO 2	2490
	6000001	SLC +1		2500
02531	0001046	DEC 550 B19	CONT NO 2	2510
	6000001	SLC +1		2520
02533	0000000	DEC 0 B14	CONT NO 2	2530
	6000001	SLC +1		2540
02535	0034100	DEC 450 B14	CONT NO 2	2550
	6000001	SLC +1		2560
			CONT NO 2	2570
		***** SCANNER COMMAND WORD FOR MIAE *****		2580
02537	3054010	RSTDsk OCT 3054010		2590
			END OF SEGMENT TO CALCULATE	2600
			MIAE	2610
			*****	2620
			INCALC, CALCULATION SEGMENT OF	2630
			CONTROLLER SIMULATION PROGRAM	2640
				10
				20

02540	0023133	INCALC	LDA	THETA1	1	40
02541	2514002	BZE	1			50
02542	2602567	BRU	ATOCO			60
02543	2410011	SLA	9	THETA1,B19,TO THETA1,B10,IN A		70
02544	2504006	MAQ				80
02545	1523135	MPY	KP	1 THETA1,B10,TIMES KP,B7 GIVES A,B17		90
02546	2411003	SLD	3	A AND Q,B17,TO A AND Q,B14		100
02547	2514003	BOV	1	OVERFLOW IF MAG,A GREATER THAN 16383		110
02550	2602560	BRU	STPVA	OVERFLOW BRANCH		120
02551	0323137	STA	CONTV	1 A,B14,TO CONTVA		130
02552	2516001	BPL	2	A,B14,IS PROP, TERM		140
02553	2602555	BRU	DIFF1	NEGATIVE BRANCH		150
02554	2504532	NEG		A WAS POS.		160
02555	0123141	ADD	MXPR	1 MXPR MINUS ABS. VAL. A		170
02556	2514004	BPL	1	ABS. VAL. A GREATER THAN MXPR IF A NEG		180
02557	2602570	BRU	PINCO	MAG, A OK, NEW CONTVA IN STORAGE		190
02560	0023133	LDA	THETA1	1 MAG, A TOO BIG		200
02561	2514001	BPL	1	TEST SIGN OF THETA1		210
02562	2602566	BRU	PLUS1	THETA1 POS.		220
02563	0023141	LDA	MXPR	1 THETA1 NEG.		230
02564	2504532	NEG		NEG. VAL. OF MXPR IN A		240
02565	2602567	BRU	ATOCO			250
02566	0031141	LDA	MXPR			260
02567	0323137	STA	CONTV	1 MXPR IS MAG, OF CONTVA		270
02570	0023135	LDA	KP	1		280
02571	0603120	LDX	ZERO	0 CLEAR IRO		292
02572	2410063	NOR	19	NORMALIZE KP		293
02573	0303171	STA	KPN	KP,NORMALIZED		294
02574	0000000	LDA	0			295
02575	0203127	SUB	TMLV	B VAL OF KPN		296
02576	0303172	STA	KPNBV			297
02577	0023131	LDA	FLAG1	1 WANT TO CHECK A FLAG BIT		298
02600	2514000	BEV	1	TEST IF FIRST TIME SET,SET IF EGL 1		300
02601	2602607	BRU	TSCS	FIRST TIME NOT SET,BIT 19 EGL 0		310
02602	2504002	LDZ				340
02603	0323147	STA	SLVALU	1 ZERO TO SLVALU		350
02604	0023133	LDA	THETA1	1		360
02605	0323151	STA	SLTH2	1 THETA1 TO SLTH2		370
02606	2603071	BRU	SAIVA	BYPASS DERIV, AND INTGRAL CALC.		380
02607	0023131	LDA	FLAG1	1 WANT TO CHECK A FLAG BIT		390
02610	2410301	LLC	1	BIT 1 TO BE CHECKED		400
02611	2514001	BPL	1	IF CALC, SLOPE,BIT1,IS SET,A IS NEG.		410
02612	2602735	BRU	SADVA	A WAS POS.,90 DO NOT CALC. DIRIV. TERM		420
02613	0023163	LDA	KD	1 BIT 1 EGL 1,90 CALC. DIRIV. TERM		430
02614	2514002	BZE	1	TEST IF KD IS ZERO		432
02615	2602626	BRU	DLRZ	KD AND ACCUM, ARE ZERO		434
02616	0023133	LDA	THETA1	1 KD NOT ZERO		436
02617	0023151	SUB	SLTH2	1 THETA1 MINUS SLTH2 IN A,B19		440

```

02620 0303165 STA DELERR
02621 0023133 LDA THETA1
02622 0323151 STA SLTH2
02623 0003165 LDA DELERR
02624 2516002 BZE 2
02625 2602630 BRU NODEL
02626 0323147 STA SLVALU
02627 2602735 BRU SADVA
02630 0603120 LDX ZERO
02631 2410063 NOR 19
02632 0303165 STA DELERR
02633 1703167 STX NORM
02634 0023153 LDA SLTIM
02635 0603120 LDX ZERO
02636 2410063 NOR 19
02637 0303170 STA TEMP
02640 0003167 LDA NORM
02641 2504032 ADO
02642 0200000 SUB 0
02643 0103126 ADD FFTN
02644 0303167 STA NORM
02645 2504002 LDZ
02646 2504006 MAQ
02647 0003165 LDA DELERR
02650 2400101 SRD 1
02651 1603170 DVB TEMP

* *
02652 0603120 LDX ZERO
02653 2410063 NOR 19
02654 0303170 STA TEMP
02655 0000000 LDA 0
02656 0103167 ADD NORM
02657 0203236 SUB NNTN
02660 0303167 STA NORM
02661 2504000 NOP
02662 0003171 LDA KPN
02663 2504006 MAQ
02664 1503170 MPY TEMP

*
02665 0603120 LDX ZERO
02666 2411063 DNO 19
02667 0303170 STA TEMP
02670 0000000 LDA 0
02671 0203236 SUB NNTN
02672 0103167 ADD NORM
02673 0103172 ADD KPNBV

DLRZ
MODEL
DELTA ERROR FOR THIS SLOPE PERIOD
1 NEW SLTH2 EQL OLD THETA1
TEST IF DELERR IS ZERO
DELERR,B19, IS NOT ZERO
1 SET SLVALU EQL ZERO
BYPASS DERIV. CALC.
0 SET I,R,0 TO ZERO
DELERR,B19,WAS IN A; IT IS NORMALIZED
DELERR,B UNKNOWN, STORED AT THIS STEP
0 B VAL, FOR DELERR IN NORM,B19
1 SLTIM,B4,IN A
0 SLTIM,BUNKOWN,IN A
B VAL FOR DELERR INCREASED 1 BEFORE DI
B VAL FOR DELERR BY SLTIM IN A
THIS MAKES A SMALLER THAN SLTIM,NOR.
DELERR DIV BY SLTIM, QUOTIENT IN A
CARDS NUMBERED 730 THRU 850 HAVE BEEN
REMOVED
0 CLEAR I,R, 0
QUO, NORMALIZED
NORM, QUO, IN TEMP
B VAL FOR NOR DELERR BY SLTIM
***** CAN INSERT BRU* HERE *****
KPN TO Q
KPN TIMES QUANT, DELERR,NOR,,BY SLTIM,
NOR,,NOR. IN A
0 CLEAR I,R, 0
A PLUS Q,NOR.
NOR, VAL, OF PROD. IN TEMP
B VAL, FOR LPAREN,KPN TIMES LPAREN,
DELERR,NOR,,BY SLTIM,NOR,,RPAREN,,NOR

```

02674	0303167	*	STA NORM	RPAREN,NOR, EQL PRODUCT IN A	1190
02675	0603120		LDX ZERO	0 CLEAR IRO	1200
02676	0023163		LDA KD	1	1210
02677	2410063		NOR 19		1220
02700	0303173		STA TEMP2	KD,NORM,,IN TEMP2	1230
02701	0000000		LDA 0	B VAL, FOR KD IN A	1250
02702	0203130		SUB NINE		1260
02703	0103167		ADD NOKM	B VAL, FOR KD,NOR,,TIMES PRODUCT IN A	1262
02704	0303167		STA NORM		1270
02705	1743241		STX SAVX2	2	1280
02706	1763242		STX SAVX3	3	1290
02707	0743174		SPB SUB1	2 GO TO SUBROUTINE SUB1 AND STORE THIS LOCATION IN IR2	1300
02710	0643241	*	LDX SAVX2	2	1310
02711	0663242		LDX SAVX3	3	1320
02712	0603240		LDX FLAG2	0 FLAG2 IS 1 IF MAG A TOO BIG,0 IF NOT	1330
02713	0517777		BXH 1	0 TEST VALUE OF IRO EQL FLAG2	1340
02714	2602725		BRU STDVA	FLAG2 IS 1, IF FLAG2 IS 0, GO TO NEXT COMMAND	1350
		*		CARDS NUMBERED 1410 THRU 1580 HAVE BEEN REMOVED	1380
		*			1390
		*			1400
02715	0323147	TMSL	STA SLVALU	1 SLVALU,B14,IN STORAGE	1590
02716	2514001		BPL 1	TEST SIGN OF A,A EQL. SLVALU,B14	1600
02717	2504532		NEG	FORM NEG, SLVALU	1610
02720	0303170		STA TEMP	TEMP CONTAINS NEG. ABS. VAL, SLVALU	1620
02721	0023155		LDA MXSL	1 THIS IS MAX, ALLOWABLE SLVALU	1630
02722	0103170		ADD TEMP	MXSL MINUS ABS. VAL. SLVALU IN A	1640
02723	2514001		BPL 1	TEST MAG, OF SLVALU	1650
02724	2602735		BRU SADVA	MAG. SLVALU OK	1660
02725	0003165	STDVA	LDA DELERR	MAG, SLVALU TOO BIG	1670
02726	2514001		BPL 1	TEST SIGN OF DELERR	1680
02727	2602733		BRU PLUS5	DELERR POS	1690
02730	0023155		LDA MXSL	1 DELERR NEG,	1700
02731	2504532		NEG	NEG, MXSL IN A	1710
02732	2602734		BRU DVAST		1720
02733	0023155	PLUS5	LDA MXSL	1	1730
02734	0323147	DVAST	STA SLVALU	1 SLVALU,B14,EITHER CONST OR CALC, VAL,	1740
02735	0023147	SADVA	LDA SLVALU	1 TWO PATHS TO SADVA	1750
02736	0123137		ADD CONTVA	1 SLVALU,B14,PLUS CONTVA,B14,IN A	1760
02737	2516003		BOV 2	TEST IF A OVERFLOWED	1770
02740	2602746		BRU SACVA	A DID NOT OVERFLOW	1780
02741	2516001		BPL 2	A OVERFLOWED,TEST FOR SIGN OF A	1790
02742	2602745		BRU PLUS2	BEFORE OVERFLOW,A WAS POS,	1800
02743	0003123		LDA MXCN	BEFORE OVERFLOW,A WAS NEG,	1810
02744	2602746		BRU SACVA	MAX, NEG, CONTVA IN A	1820
02745	0003122	PLUS2	LDA MXCP	MAX, POS, CONTVA IN A	1830
02746	0323137	SACVA	STA CONTVA	1 CONTVA,B14,EITHER CONST, OR CALC. VAL,	1840

02747	2504000	NOP	***** CAN INSERT BRU* HERE *****	1850
02750	0023161	LDA KI		1860
02751	2514002	BZE 1	TEST KI FOR 0	1870
02752	2603071	BRU SAIVA	BYPASS INTEGRAL CALC. IF KI EQL. 0	1880
02753	0603120	LDX ZERO	0 CLEAR IRO	1890
02754	2410063	NOR 19		1900
02755	0303170	STA TEMP	KI,NOR,,FROM A TO TEMP	1920
02756	0000000	LDA 0	B VAL. FOR KI,NOR,,IN A	1930
02757	0203126	SUB FFTN		1932
02760	0103172	ADD KPNBV		1940
02761	0303167	STA NORM	B VAL. FOR KI,NOR,,TIMES KPN IN NORM	1950
02762	0003170	LDA TEMP	KI,NOR,,IN A	1960
02763	2504006	MAQ	A TO Q	1970
02764	1503171	MPY KPN	KPN TIMES KI,NOR.	1980
02765	0603120	LDX ZERO	0 CLEAR IRO	1990
02766	2411063	DNO 19	KPN TIMES KI,NOR,,IN A	2000
02767	0303170	STA TEMP		2020
02770	0000000	LDA 0		2030
02771	0203236	SUB NNTN		2032
02772	0103167	ADD NORM	B VAL. OF NOR. PRODUCT2 IN A	2040
02773	0303167	STA NORM	***** CAN INSERT BRU* HERE *****	2050
02774	2504000	NOP		2052
02775	0003166	LDA ITINT		2060
02776	0603120	LDX ZERO	0 CLEAR IRO	2070
02777	2410063	NOR 19		2080
03000	0303173	STA TEMP2	ITINT,NOR,,IN TEMP2	2100
03001	0000000	LDA 0	B VAL. FOR ITINT,NOR,,IN A	2110
03002	0203126	SUB FFTN		2112
03003	0103167	ADD NORM	B VAL. FOR ITINT,NOR,TIMES PRODUCT2, NOR,,IN A	2120
03004	0303167	STA NORM		2130
03005	0003173	LDA TEMP2		2140
03006	2504006	MAQ		2150
03007	1503170	MPY TEMP	PRODUCT3 IN A AFTER EXEC. OF THIS STEP	2160
03010	0603120	LDX ZERO	0	2170
03011	2411063	DNO 19	PRODUCT3,NOR,,IN A	2180
03012	0303170	STA TEMP		2190
03013	0000000	LDA 0		2200
03014	0203236	SUB NNTN		2230
03015	0103167	ADD NORM		2232
03016	0303167	STA NORM	B VAL. FOR PRODUCT3,NOR,,IN NORM	2240
03017	0603120	LDX ZERO		2250
03020	0023133	LDA THETA1		2260
03021	2410063	NOR 19		2270
03022	0303173	STA TEMP2		2280
03023	0000000	LDA 0		2290
03024	0103167	ADD NORM	B VAL FOR THETA1,NOR,TIMES PRODUCT3, NOR,,IN A	2300
				2310
				2320

03025	0303167		STA NORM		2330
03026	1743241		STX SAVX2	2	2340
03027	1763242		STX SAVX3	3	2350
03030	0743174		SPB SUB1	2	2360
		*		GO TO SUBROUTINE SUB1 AND STORE THIS	2370
				LOCATION IN IR2	2380
03031	0643241		LDX SAVX2	2	2390
03032	0663242		LDX SAVX3	3	2400
03033	0603240		LDX FLAG2	0	2410
03034	0517777		BXH 1	0	2420
03035	2603037		BRU STTVA	FLAG2 IS 1	2430
03036	2603045		BRU SATVA	FLAG2 IS 0	2440
		*		CARDS NUMBERED 2460 THRU 2630 HAVE	2450
		*		BEEEN REMOVED	2460
03037	0023133	STTVA	LDA THETA1	1	2470
03040	2514001		BPL 1	TEST SIGN OF THETA1	2480
03041	2603044		BRU PLUS3	THETA1 IS POS,	2490
03042	0003123		LDA MXCN	THETA1 IS NEG,	2500
03043	2603045		BRU SATVA	A,B14, IS CONSTANT OR CALC. VALUE	2510
03044	0003122	PLUS3	LDA MXCP		2520
03045	0303170	SATVA	STA TEMP		2530
		*		PRESERVE THIS TERM FOR SIGN IF INTVAL	2540
				TOO BIG	2550
03046	0123145		ADD INTVAL	1	2560
03047	2514003		BOV 1	A,B14, PLUS INTVAL, B14, IN A AFTER THIS	2570
03050	2603061		BRU STIVA	TEST IF A OVERFLOWED	2580
03051	0323145		STA INTVAL		2590
03052	2514001		BPL 1	1	2600
03053	2504532		NEG	TEST SIGN OF A EQL INTVAL	2610
03054	0303173		STA TEMP2	A WAS POS., SO NEGATE IT	2620
03055	0023157		LDA MXIN	TEMP2 CONTAINS NEG, ABS. VAL OF INTVAL	2630
03056	0103173		ADD TEMP2	1	2640
03057	2514001		BPL 1	THIS IS MAX. ALLOW. INTVAL	2650
03060	2603071		BRU SAIVA	MXIN MINUS ABS. VAL INTVAL IN A	2660
03061	0003170	STIVA	LDA TEMP	TEST MAG. OF INTVAL	2670
03062	2514001		BPL 1	MAG. INTVAL OK	2680
03063	2603067		BRU PLUS4	TEST SIGN OF TEMP	2690
03064	0023157		LDA MXIN	TEMP POS,	2700
03065	2504532		NEG	1	2710
03066	2603070		BRU IVAST	TEMP NEG,	2720
03067	0023157	PLUS4	LDA MXIN	NEG, MXIN IN A	2730
03070	0323145	IVAST	STA INTVAL		2740
03071	0023145	SAIVA	LDA INTVAL	1	2750
03072	0123137		ADD CONTVA	INTVAL, B14, IS CONST OR CALC VAL	2760
03073	0323137		STA CONTVA	1	2770
03074	2514003		BOV 1	TWO PATHS TO SAIVA	2780
03075	2603103		BRU TNCO	1	2790
03076	2514001		BPL 1	INTVAL, B14, PLUS CONTVA, B14, IN A	2800
03077	2603106		BRU TCTB	1	2810
03100	2504002	ZCON	LDZ	TEST FOR OVERFLOW	2820
				A OVERFLOWED, MAG, CONTVA TOO BIG	2830
				TEST SIGN OF CONTVA	2840
				CONTVA POS,	2850
				0 IN A	2860

03101	0323137	STA	CONTVA	1 CONTVA SET TO 0	3030
03102	2603117	BRU	CLCOV		3040
03103	2514001	TNCO	BPL 1	TEST SIGN OF TOO BIG CONTVA EQL A	3050
03104	2603111	BRU	SATCO	CONTVA TOO BIG AND POS,	3060
03105	2603100	BRU	ZCON	CONTVA TOO BIG AND NEG,	3070
03106	0203124	TCTB	SUB OTHO	A CONTAINS POS, CONTVA	3080
03107	2516001	BPL	2	TEST MAG, OF CONTVA	3090
03110	2603114	BRU	ROCO	CONTVA,B14,MINUS 1001,B14,WAS NEG,	3100
03111	0003125	SATCO	LDA OBSN	CONTVA TOO BIG,SET TO 1000,B17	3110
03112	0323137	STA	CONTVA	1	3120
03113	2603117	BRU	CLCOV	CONTVA EQL 1000,B17	3130
03114	0023137	ROCO	LDA CONTVA	1 CONTVA,B14,TO A	3140
03115	2400003	SRA	3	CONTVA,B14,TO B17	3150
03116	0323137	STA	CONTVA	1 CONTVA NOW B17	3180
03117	2602324	CLCOV	BRU F4		3190
		*	THE	FOLLOWING	
		*		STATEMENTS	
		*		USED	
		*		IN	
		*		THIS	
		*		DEFINE	
		*		THE	
		*		PROGRAM	
03120	0000000	ZERO	DEC 0		3200
03121	0000022	EGHTN	DEC +18 B19		3210
03122	1777740	MXCP	DEC +16383 B14		3220
03123	2000040	MXCN	DEC -16383 B14		3230
03124	0076440	OTH0	DEC +1001 B14		3250
03125	0007640	OBSN	DEC +1000 B17		3260
03126	0000017	FFTN	DEC 15 B19		3270
03127	0000014	THLV	DEC 12 B19		3280
03130	0000011	NINE	DEC 9 B19		3282
		*	THE	FOLLOWING	
		*		STATEMENTS	
		*		INDEXED	
		*		BY	
		*		IR	
		*		AND	
		*		TEST	
		*		VALUES	
03131	1000001	FLAG1	OCT 1000001	1 IN BIT 19 IS FIRST TIME SET, BIT 1	3284
		*		EQL 1 IS CALCULATE SLOPE SET	3290
		*		CONT NO 2	3300
03133	6000001	THETA1	SLC +1		3310
	0000017		DEC 15 B19		3312
	6000001		SLC +1		3313
03135	0120000	KP	DEC 10 B7		3315
	6000001		SLC +1		3320
03137	0000000	CONTVA	DEC 0 B14		3322
	6000001		SLC +1		3330
03141	0076400	MXPR	DEC 1000 B14		3332
	6000001		SLC +1		3340
03143	0037200	MO	DEC 500 B14		3342
	6000001		SLC +1		3350
03145	0000000	INTVAL	DEC 0 B14		3352
	6000001		SLC +1		3360
03147	0000000	SLVALU	DEC 0 B14		3362
	6000001		SLC +1		3370
03151	0000000	SLTH2	DEC 0 B19		3372
	6000001		SLC +1		3382
				CONT NO 2	3390
				CONT NO 2	3392

03227	1400001	INX 1	0	3890
03230	1703240	STX FLAG2	0 FLAG2 IS NOW SET TO 1	3900
03231	2603234	BRU EXIT1		3910
03232	0603120	ENDS1 LDX ZERO	0	3920
03233	1703240	STX FLAG2	0 FLAG2 IS NOW SET TO 0,50 MAG OF A OK	3930
03234	2504000	EXIT1 NOP		3940
03235	2640001	BRU 1	2 RETURN TO MAIN PROGRAM SEGMENT	3950
			IN SUB1 AND INCALC	3960
03236	0000023	NNTN DEC 19 B19		3970
03237	0000016	FRTN DEC 14 B19		3982
			VARIABLES ADDED FOR SUB1	3990
03240	0000000	FLAG2 DEC 0		4000
03241	0000000	SAVX2 DEC 0		4010
03242	0000000	SAVX3 DEC 0		4020
			RKPLA READS THE VALUES SET IN THE	10
			OPERATOR KNOBS	20
			*****	30
03243	0643120	RKPLA LDX ZERO	2 IR2 TO INDEX KNOB PAIR	110
03244	0043763	A1 LDA KNBTL	2	120
03245	2510103	LSC 1	FIND KNOB PAIR L	130
03246	0002236	LDA CMTBL3+14		140
03247	0300014	STA 12	SET SCAN A.I, SW, TO A2	150
03250	2602001	BRU PAUSE		160
03251	2500023	A2 RDG	READ KNOB PAIR L INTO A	170
03252	0343771	STA KNBBCD	2	180
03253	1440001	INX 1	2 INCREMENT KNOB PAIR INDEX	190
03254	0457773	BXL 5	2 TEST IF HAVE READ ALL KNOBS...5 PAIRS	200
03255	2603257	BRU A3	HAVE NOT READ ALL KNOBS	210
03256	2603265	BRU A6	READ ALL KNOBS	220
03257	2504000	A3 NOP	THIS IS A SW, PT,	230
03260	2603244	A4 BRU A1		240
03261	0002242	A5 LDA CMTBL3+18		250
03262	0302355	STA KNBSW	SET SW, KNBSW TO A1	260
03263	2504000	NOP	***** CAN INSERT BRU* HERE *****	270
03264	2602001	BRU PAUSE		280
03265	2504000	A6 NOP	THIS IS A SW, PT,	290
03266	2504000	A7 NOP	***** CAN INSERT BRU* HERE *****	300
03267	2603274	BRU MKBDB		310
03270	0002243	A8 LDA CMTBL3+19		320
03271	0302355	STA KNBSW	KNBSW, A SW, PT., SET TO MKBDB	330
03272	2504000	NOP	***** CAN INSERT BRU* HERE *****	340
03273	2602001	BRU PAUSE		341
			*****	360
			MKBDB	370
			FOLLOWS	380
03274	1763242	MKBDB STX SAVX3	3 SAVE IR3	390
03275	0663120	LDX ZERO	3 CLEAR IR3	400
03276	0063771	B16 LDA KNBBCD	3	410

03277	2203744	ANA FNB4	SLSD,B4,IN A	420
03300	2504006	MAQ		430
03301	1503746	MPY TNB4	10 TIMES SLSD,B8	440
03302	0303170	STA TEMP		450
03303	0063771	LDA KN8BCD	3	460
03304	2203747	ANA FN88	LSD,B8,IN A	470
03305	0103170	ADD TEMP		480
03306	2400013	SRA 11	MODE OR PNT,B19	490
03307	3464066	XEC CMTBL4	3 STORE A	500
03310	0577777	BXH 1	3 TEST IF IR3 EQL 1	510
03311	2603314	BRU B17	IR3 EQL 1	520
03312	1460001	INX 1	3	530
03313	2603276	BRU B16	CONVERT AND STORE PNT NOW	540
03314	2504000	NOP		550
		B17		560
		*	*****	570
		*	FORM BCD DIGIT TABLE	580
		*	*****	590
03315	0003773	LDA KN8BCD*2		600
03316	2203744	ANA FN84	MSD,B4	610
03317	0304021	STA DIGTBL		620
03320	0003773	LDA KN8BCD*2		630
03321	2203747	ANA FN88	MSD,B8	640
03322	0304022	STA DIGTBL*1		650
03323	0003774	LDA KN8BCD*3		660
03324	2203744	ANA FN84	SLSD,B4	670
03325	2400010	SRA 8	SLSD,B12	680
03326	0304023	STA DIGTBL*2		690
03327	0003774	LDA KN8BCD*3		700
03330	2203747	ANA FN88	LSD,B8	710
03331	2400010	SRA 8	LSD,B16	720
03332	0304024	STA DIGTBL*3		730
		*	***** DIGIT TABLE FORMED *****	740
03333	0664072	LDX MODE	3	750
03334	2504000	NOP	***** CAN INSERT BRU* HERE *****	760
03335	3464025	XEC CMTBL5	3 GO TO RIGHT CONVERSION ROUTINE	770
		*	THE FOLLOWING SEGMENT CONVERTS BCD	780
		*	INTEGERS TO BINARY	790
03336	0663120	RGCNVT LDX ZERO	3 HAD SAVED IR3 BEFORE ENTERING RGCNVT	800
03337	2504002	LDZ		810
03340	0303170	STA TEMP	ZERO IN TEMP	820
03341	0003746	LDA TNB4		830
03342	2504006	MAQ		840
03343	1503170	MPY TEMP	10 TIMES TEMP IN A	850
03344	0164021	ADD DIGTBL	3	860
03345	0303170	STA TEMP		870
03346	1460001	INX 1	3	880
03347	0477774	BXL 4	3 TEST IF IR3 LESS THAN,OR EQL,3	890
03350	2603341	BRU B4	IR3 LESS THAN 4	

03351	2504000	NOP	***** CAN INSERT BRU* HERE *****	900
03352	0003775	LDA KN8BCD+4	IR3 GREATER THAN 3	910
03353	2203744	ANA FN84		920
03354	0304074	STA MULT	MULT KNOB SETTING	930
03355	0204075	SUB MLON		940
03356	2514002	BZE 1	TEST IF MULT EQL +1	950
03357	2603362	BRU B5	MULT EQL +1	960
03360	0003746	LDA TN84	MULT NOT EQL +1	970
03361	2603363	BRU B6	MULT EQL 10,B4	980
03362	0003753	LDA ONB4		990
03363	0304074	STA MULT	MULT EQL 1 OR 10,B4	1000
03364	0003750	LDA FRB19		1310
03365	0204072	SUB MODE		1320
03366	2514002	BZE 1	TEST IF MODE EQL 4	1030
03367	2604150	BRU B34	MODE EQL 4	1040
03370	0003751	LDA SXB19	MODE NOT EQL 4	1050
03371	0204072	SUB MODE		1060
03372	2516002	BZE 2	TEST IF MODE EQL 6	1070
03373	2603600	BRU B9	MODE NOT EQL 6	1080
03374	0004166	LDA BRUB36	MODE EQL 6	1090
03375	0303476	STA B33	SET SWITCH POINT	1092
03376	0003170	LDA TEMP	TEMP,B16	1094
03377	2410010	SLA 8	TEMP,B8,IN A	1100
03400	0304070	STA NCLCNT		1110
03401	2400013	SRA 11	TEMP,B8,TO TEMP,B19	1120
03402	0103752	ADD INSTB	TIMER CONTROL WORD NOW IN A	1130
03403	0304071	STA NTIMCO		1140
03404	0004074	LDA MULT		1150
03405	2410305	LLC 5	MULT,B4,TO MULT,B19	1160
03406	0302271	STA NSCLIN	B19	1170
03407	0004074	LDA MULT		1180
03410	2504006	MAQ		1190
03411	1504070	MPY NCLCNT	INTEGER,B12,IN A	1200
03412	1603754	DVD ODB10	RESULT IN A AND Q,B2	1210
03413	2504006	MAQ		1220
03414	1503755	MPY TPQB2	RESULT IN A AND Q,B4	1230
03415	1304100	DST PROD1	TEMP, STORAGE TO USE WHEN NOCONT CHNGD	1240
03416	2411044	DNO 4	ENTER HERE IN MODE NO. 12,NOCONT	1250
03417	1703167	STX NORM		1260
03420	0303170	STA TEMP		1270
03421	0002273	LDA NOCONT		1280
03422	2504000	NOP		1290
03423	2410063	NOR 19		1300
03424	2504006	MAQ		1310
03425	1503170	MPY TEMP		1320
03426	0303170	STA TEMP	PRODUCT,UNKNOWN B	1330
03427	0000000	LDA 0		1340
03430	0103167	ADD NORM	B VAL FOR PRODUCT	1350

B5
B6

B38

0

47

BNWL-348

03431	0303167	STA NORM	1360
03432	0003750	LDA FRB19	1370
03433	0203167	SUB NORM	1380
03434	2516001	BPL 2	1390
03435	2603477	BRU BNEG1	1400
03436	0300003	STA 3	1410
03437	0003170	LDA TEMP	1420
03440	2460000	SRA 0	1430
03441	0304076	STA NITIN	1440
03442	0663120	LDX ZERO	1442
03443	1763762	STX LST3	1444
03444	0003756	LDA TIMSW1	1446
03445	0303530	STA B24	1448
03446	0303570	STA B25	1450
03447	0003760	LDA OP2	1452
03450	0303512	STA B29	1454
03451	0303523	STA B28	1456
03452	0303526	STA B30	1458
03453	0303537	STA B31	1460
03454	0303566	STA B15	1462
03455	2603507	BRU B8	1464
03456	0663762	LDX LST3	1466
03457	1460001	INX 1	1468
03460	1763762	STX LST3	1470
03461	0000003	LDA 3	1472
03462	0202273	SUB NOCONT	1474
03463	2516001	BPL 2	1476
03464	2603507	BRU B8	1478
03465	0003757	LDA TIMSW2	1480
03466	0303530	STA B24	1482
03467	0303570	STA B25	1484
03470	0003761	LDA OP1	1486
03471	0303512	STA B29	1488
03472	0303523	STA B28	1490
03473	0303526	STA B30	1492
03474	0303537	STA B31	1494
03475	0303566	STA B15	1496
03476	2504000	NOP	1498
03477	2504532	NEG	1500
03500	0300003	BNEG1	1502
03501	0003170	STA 3	1504
03502	2470000	LDA TEMP	1506
03503	2514003	SLA 0	1508
03504	2603504	BOV 1	1510
03505	0304076	BRU *	1512
03506	2603442	STA NITIN	1520
03507	0002271	BRU B32	1530
		LDA NSCLIN	1540
		BB	
		3	
		TEST IF NITIN TOO BIG	
		NITIN TOO BIG WILL STOP CONTROLLER	
		MAG, OK	
		3	
		NO. OF LEFT SHIFTS IN A	
		MODE 4 TO B35, 6 TO B36, 12 TO B37	
		B33 IS A SW PT, SET BY MODES 4, 6, 12,	
		CONDITION	
		TO ORIGINAL	
		B29, B28, B30, B31, B15	
		AND INSTRUCTIONS	
		B24 AND B25	
		POINTS	
		SWITCH	
		RETURN	
		ALL INTERVALS CHANGED	
		NOT ALL INTERVALS CHANGED	
		TEST IF ALL SLOPE INTERVALS CHANGED	
		DIFF. IS 0 IF NO MORE CONTROLLERS	
		3	
		3	
		INCREMENT CONTROLLER INDEX BY 1	
		3	
		3	
		VARIABLES	
		INTERVAL	
		SLOPE	
		IR3 INDEXES	
		FROM LST3.	
		IR3 SET	
		SLOPE INTERVALS	
		SET SWITCHES B24 AND B25 TO CHANGE ALL	
		3	
		3	
		INITIALIZE LST3	
		3	
		INITIALIZE IR3, NITIN HAS BEEN STORED	
		3	
		NITIN, B4, IS CALCULATED, GO ON	
		3	
		A HAS NITIN, B4	
		3	
		NO. OF RT. SHIFTS, TO GET B4, IN A	
		B VAL GREATER THAN 4	
		3	
		1410	
		1400	
		1390	
		1380	
		1370	
		1360	

03510	2410004		SLA 4	B19 TO B15	1550
03511	2504006		MAQ		1560
03512	0664073	B29	LDX PNT	3	1570
03513	1564050		MPY NSLPR	3 NSLPR IS B4, INDEXED BY PNT	1580
03514	0362265		STA NSCLSL	3 B19, INDEXED BY PNT	1590
03515	2504532		NEG		1600
03516	2410013		SLA 11	NEG. OF NSCLSL, B8	1610
03517	0103747		ADD FNBB		1620
03520	2514001		BPL 1	TEST IF NSCLSL GREATER THAN 15	1630
03521	2603531		BRU B11	NSCLSL OK	1640
03522	0002271	B13	LDA NSCLIN	NSCLSL TOO BIG	1650
03523	0664073	B28	LDX PNT	3	1660
03524	0362265		STA NSCLSL	3 INDEXED BY PNT, OR RUNS THRU ALL CONT,	1670
03525	0004076		LDA NITIN		1680
03526	0664073	B30	LDX PNT	3	1690
03527	0364052		STA NSLTI	3 INDEXED BY PNT, OR RUNS THRU ALL CONT,	1700
03530	2603476	B24	BRU B33	HAVE STORED CONT, RELATED VAR	1710
03531	0004076	B11	LDA NITIN		1720
03532	2504000		NOP		1730
03533	2410044		NOR 4		1740
03534	0303170		STA TEMP		1750
03535	1703167		STX NORM	0	1760
03536	2504000		NOP		1782
03537	0664073	B31	LDX PNT	3	1784
03540	0064050		LDA NSLPR	3 ASSOC. WITH CONTROLLER INDEXED BY PNT	1790
03541	2410044		NOR 4		1800
03542	2504006		MAQ		1810
03543	1503170		MPY TEMP		1820
03544	0303170		STA TEMP	NOR PROD. OF NSLPR AND NITIN	1830
03545	0000000		LDA 0		1840
03546	0103167		ADD NORM	B VAL OF PROD	1850
03547	2504532		NEG		1860
03550	0103750		ADD FRB19		1870
03551	2514001		BPL 1		1880
03552	2603562		BRU B14	MAG OF PROD OK, NO. OF RT. SHIFTS IN A	1890
03553	2504532		NEG	NO. OF LEFT SHIFTS	1900
03554	0300003		STA 3		1910
03555	0003170		LDA TEMP		1920
03556	2470000		SLA 0	3 PRODUCT, B4, IN A	1930
03557	2514003		BOV 1	TEST IF PROD. TOO BIG	1940
03560	2603522		BRU B13	PROD. TOO BIG	1950
03561	2603566		BRU B15	PROD IN A, MAG. OK	1960
03562	2504000	B14	NOP		1970
03563	0300003		STA 3		1980
03564	0403170		LDA TEMP		1990
03565	2460000		SRA 0	3	2000
03566	0664073	B15	LDX PNT	3	2010
03567	0364052		STA NSLTI	3	2020

03570	2603476	B25	BRU B33	HAVE STORED VAR, ASSOC. WITH CONT,	2030
03571	0003170	B7	LDA TEMP	TEMP,B16,ROUTED HERE BY MODE 4,	2040
03572	2410014		SLA 12		2050
03573	2514003		BOV 1	CHECK IF TEMP LESS THAN 16	2052
03574	2603522		BRU B13	NO	2053
03575	0664073		LDX PNT	3	2060
03576	0364050		STA NSLPR	3 NSLPR,B4,INDXD BY IR3	2070
03577	2603507		BRU B8		2080
03600	0003170	B9	LDA TEMP		2090
03601	2410001		SLA 1	TEMP,B16,TO B15	2100
03602	2504006		MAQ		2110
03603	1504074		MPY MULT		2120
03604	0304077		STA DATA	PROD,,B19	2130
		*	*** * END OF	RGCNVT * * * *	2140
03605	1743241	RSPARM	STX SAVX2	2	2150
03606	0664073		LDX PNT	3	2160
03607	0644072		LDX MODE	2	2170
03610	0004077		LDA DATA		2180
03611	3443776		XEC CMTRL6	2 DATA,IN A,STORED AS CORRECT VARIABLE	2190
		*		FOR CORRECT CONTROLLER	2200
03612	0643241	B23	LDX SAVX2	2 IR2 RESTORED	2210
03613	0663242		LDX SAVX3	3 IR3 RESTORED	2220
03614	2504000	B1	NOP	THIS IS A SWITCH POINT	2230
03615	2504000	B2	NOP	***** BRU* CAN BE INSERTED HERE *****	2240
03616	2603731		BRU RESDKC		2250
03617	2504000	B3	NOP	***** BRU* CAN BE PUT HERE *****	2260
03620	0002244		LDA CMTBL3*20		2270
03621	0302355		STA KNBSW	KNBSW SET TO RESDKC	2280
03622	2602001		BRU PAUSE		2290
		*		*****	2300
		*	MOEQCV	FOLLOWS	2310
		*		*****	2320
03623	0663120	MOEQCV	LDX ZERO	3 CLEAR IR3	2330
03624	0063137	B21	LDA CONTVA	3	2340
03625	2410005		SLA 5	B19 TO B14	2350
03626	0363143		STA MO	3 CONTVA,B14 TO MO,B14	2360
03627	1460001		INX 1	3	2370
03630	0000003		LDA 3		2380
03631	0202273		SUB NOCONT	A IS NEG, IF HAVE MORE CONTROLLERS	2390
03632	2516001		BPL 2	TEST SIGN OF A	2400
03633	2603624		BRU B21	MORE CONT. LEFT	2410
03634	0663242		LDX SAVX3	3 NO MORE CONT, EXIT MOEQCV	2420
03635	2603614		BRU B1	GO TO RESDKC	2430
		*		START OF KPBI	2440
03636	0004022	KPBI	LDA DIGTBL*1		2450
03637	2410002		SLA 2	SMSD,B8,TO B6	2460
03640	0304022		STA DIGTBL*1		2470
03641	0004023		LDA DIGTBL*2		2480

03642	2410005	SLA 5	SLSD,B12,TO B7	2490
03643	0304023	STA DIGTBL+2		2500
03644	0004024	LDA DIGTBL+3		2510
03645	2410011	SLA 9	LSD,B16,TO B7	2520
03646	0304024	STA DIGTBL+3		2530
03647	0003743	LDA SCKP1		2540
03650	0303170	STA TEMP	1 IN TEMP	2550
03651	2603701	BRU FYCNVT		2560
		* KDBI	START OF KDBI	2570
03652	0004022	LDA DIGTBL+1		2580
03653	2410002	SLA 2	SMSD,B8,TO B6	2590
03654	0304022	STA DIGTBL+1		2600
03655	0004023	LDA DIGTBL+2		2610
03656	2410003	SLA 3	SLSD,B12,TO B9	2620
03657	0304023	STA DIGTBL+2		2630
03660	0004024	LDA DIGTBL+3		2640
03661	2410006	SLA 6	LSD,B16,TO B10	2650
03662	0304024	STA DIGTBL+3		2660
03663	2504002	LDZ		2670
03664	0303170	STA TEMP	0 IN TEMP	2680
03665	2603701	BRU FYCNVT		2690
		* KIBI	START OF KIBI	2700
03666	0004022	LDA DIGTBL+1		2710
03667	2410004	SLA 4	SMSD,B8,TO B4	2720
03670	0304022	STA DIGTBL+1		2730
03671	0004023	LDA DIGTBL+2		2740
03672	2410010	SLA 8	SLSD,B12,TO B4	2750
03673	0304023	STA DIGTBL+2		2760
03674	0004024	LDA DIGTBL+3		2770
03675	2410014	SLA 12	LSD,B16,TO B4	2780
03676	0304024	STA DIGTBL+3		2790
03677	0003745	LDA SCK12		2800
03700	0303170	STA TEMP	2,319,IN TEMP	2810
		* FYCNVT	FOLLOWS	2820
03701	1743241	STX SAVX2	2 IR2 SAVED, RESTORED IN RSPARM	2830
03702	0643170	LDX TEMP	2 INITIALIZE IR2	2840
03703	2504002	LDZ		2850
03704	0303170	STA TEMP		2860
03705	0300003	STA 3	CLEAR IR3	2870
03706	0044060	LDA DECTBL	2	2880
03707	2504006	MAQ		2890
03710	1564021	MPY DIGTBL	3	2900
03711	0103170	ADD TEMP		2910
03712	0303170	STA TEMP	PRODUCT PLUS TEMP BACK IN TEMP	2920
03713	0477777	BXL 1	3 TEST IF IR3 IS 0	2930
03714	2603716	BRU B18	IR3 IS 0	2940
03715	2603720	BRU B19	IR3 IS NOT 0	2950
03716	2410001	SLA 1		2960

03717	0303170	STA TEMP	TEMP NOW HAS CORRECT B VAL	2970
03720	1460001	B19 INX 1	3	2980
03721	1440091	INX 1	2	2990
03722	0577774	BXH 4	3 IF IR3 IS 4, END OF CONVERSION	3000
03723	2603725	BRU B20	WIND UP FYCNVT	3010
03724	2603706	BRU B22	CONTINUE WITH CONVERSION	3020
03725	0003170	B20 LDA TEMP		3030
03726	0304077	STA DATA		3040
03727	2504000	NOP	***** BRU* MAY BE PUT HERE *****	3050
03730	2603605	BRU RSPARM		3060
		* START OF RESDKC,		3080
		* SEGMENT TO RESET DATA INSERT KEY		3090
03731	0003770	RESDKC LDA KNBTBL+5		3100
03732	2510103	LSC 1	SCANNER TO RESET DIK	3110
03733	0002237	LDA CMTBL3+15		3120
03734	0300014	STA 12	SET SCAN, INT. TO C1	3130
03735	0002245	LDA CMTBL3+21		3140
03736	0302355	STA KNBSW	SET KNBSW TO RKPLA	3142
03737	2602001	BRU PAUSE		3144
03740	2504000	C1 NOP	SWITCH POINT	3150
03741	2602131	C2 BRU E15	GO BACK TO STARTE SEGMENT	3160
03742	2602001	C3 BRU PAUSE		3170
		* END OF RESDKC		3180
		* *****		3190
		* CONSTANTS FOR MKBDB		3200
		* *****		3210
03743	0000001	SCKP1 DEC 1 B19		3230
03744	1700000	FNB4 DEC 15 B4		3240
03745	0000002	SCK12 DEC 2 B19		3250
03746	1200000	TNB4 DEC 10 B4		3260
03747	0074000	FNR8 DEC 15 B8		3270
03750	0000004	FRR19 DEC 4 B19		3280
03751	0000006	SXB19 DEC 6 B19		3290
03752	0010000	INSTB OCT 0010000		3300
03753	0100000	ONB4 DEC 1 B4		3310
03754	1750000	ODB10 DEC 1000 B10		3320
03755	1463146	TPOB2 DEC 3,2 B2		3330
03756	2603456	TIMSW1 BRU B26		3332
03757	2603476	TIMSW2 BRU B33		3334
03760	0663762	OP2 LDX LST3	3	3336
03761	0664073	OP1 LDX PNT	3	3338
03762	0000001	LST3 DEC 1 B19		3340
		* KNBTBL TO CONTAINS SCAN, COMMAND		3350
		* READ KNOBS		3360
03763	3044403	KNBTBL OCT 3044403	LEAST SIG, MODE PAIR	3370
03764	3044603	OCT 3044603	LEAST SIG, POINT PAIR	3380
03765	3044003	OCT 3044003	MOST SIG, DATA PAIR	3390
03766	3044103	OCT 3044103	LEAST SIG, DATA PAIR	3400

03767	3044503	OCT 3044503	MULTI DATA INSERT KEY	3410
03770	3044710	OCT 3044710	OF KNTBL	3420
			CONTAINS BCD OF	3430
			KNOB VALUES	3440
03771	0000000	KNBBD DEC 0	MODE	3450
03772	0000000	DEC 0	POINT	3460
03773	0024000	DEC 0 0024000	DATA	3480
03774	0000000	DEC 0	EQL 500	3490
03775	0200000	OCT 0200000	MULTI CODE FOR 1	3500
			OF KBBBD	3510
			DIRECTS PROPER STORAGE	3512
			OF DATA	3520
03776	2604102	CMTBL6 BRU STMO	MODE NO. 0	3530
03777	0363135	STA KP	1	3540
04000	0362424	STA DBTHET	2	3550
04001	0363163	STA KD	3	3560
04002	2604002	BRU *	4	3570
04003	0363161	STA KI	5	3580
04004	2604004	BRU *	6	3590
04005	2604005	BRU *	7	3600
04006	2604121	BRU STSTPT	8	3610
04007	2604105	BRU STMXPR	9	3620
04010	2604110	BRU STMXSL	10	3630
04011	2604113	BRU STMXIN	11	3640
04012	2604171	BRU STNOCO	12	3650
04013	2604116	BRU STTAUS	13	3660
04014	2604014	BRU *	14	3662
04015	2604015	BRU *	15	3663
04016	2604016	BRU *	16	3664
04017	2604017	BRU *	17	3665
04020	2604020	BRU *	18	3666
			OF CMTBL6	3667
			HAS DATA DIGITS IN BCD	3670
04021	0000000	DIGTBL DEC 0		3910
04022	0000000	DEC 0		3920
04023	0000000	DEC 0		3930
04024	0000000	DEC 0		3940
			OF DIGTBL	3950
			DIRECTS THE PROGRAM TO	3960
			RIGHT CONVERT ROUTINE	3970
			SET NO. 0	3980
04025	2603336	BRU RGCNVT	1	3990
04026	2603636	BRU KPBI	2	4000
04027	2603336	BRU RGCNVT	3	4010
04030	2603652	BRU KDBI	4	4020
04031	2603336	BRU RGCNVT	5	4030
04032	2603666	BRU KIBI	6	4040
04033	2603336	BRU RGCNVT		4050

04034	2603623	BRU MOEQCV	SET CONTVA EQL MO	7	4060
04035	2603336	BRU RGCNV	SET STPROT	8	4070
04036	2603336	BRU RGCNV	SET MAXPR	9	4080
04037	2603336	BRU RGCNV	SET MAXSL	10	4090
04040	2603336	BRU RGCNV	SET MAXIN	11	4100
04041	2603336	BRU RGCNV	SET NOCONT	12	4110
04042	2603336	BRU RGCNV	SET TAUS	13	4112
04043	2603336	BRU RGCNV	SPARE	14	4113
04044	2603336	BRU RGCNV	SPARE	15	4114
04045	2603336	BRU RGCNV	SPARE	16	4115
04046	2603336	BRU RGCNV	SPARE	17	4116
04047	2603336	BRU RGCNV	SPARE	18	4117
			VARIABLES IN TABLES, ASSOCIATED WITH		4120
			CONTROLLERS		4130
04050	0100000	NSLPR DEC 1 B4			4150
	6000001	SLC +1	CONI NO 2		4152
04052	0014223	NSLTI DEC ,192 B4			4160
	6000001	SLC +1	CONT NO 2		4162
04054	0050244	PS3B4 OCT 0050244	0,63,B4		4170
04055	0050244	OCT 0050244	CONT NO 2		4172
04056	0120000	TAUS DEC 20 B8			4180
	6000001	SLC +1	CONT NO 2		4182
			FOLLOWS, FOR CONVERTING		4190
			KP, KD, KI		4200
04060	1440000	DECTBL DEC 100 B7			4210
04061	1200000	DEC 10 B4			4220
04062	1000000	DEC 1 B1			4230
04063	0146314	DEC 0,1 B0			4240
04064	0012172	DEC 0,01 B0			4250
04065	0001014	DEC 0,001 B0			4260
			END OF DECTBL		4270
			*****		4280
			COMMANDS TO STORE MODE AND POINT		4290
			*****		4300
04066	0304072	CMTBL4 STA MODE			4310
04067	0304073	STA PNT			4320
			END OF CMTBL4		4330
			*****		4340
			VARIABLES FOR MKRDB		4350
			*****		4360
04070	0000000	NCLCNT DEC 0 B8			4370
04071	0000000	NTIMCO DEC 0			4380
04072	0000000	MODE DEC 0 B19			4430
04073	0000000	PNT DEC 0 B19			4450
04074	0000000	MULT DEC 0			4460
04075	0200000	MLON OCT 0200000	MULT CODE FOR +1		4470
04076	0000000	NITIN DEC 0 B4			4480
04077	0000000	DATA DEC 0			4490

04100	0000000	PROD1	SLC E	SET ADDRESS FOR DOUBLE STORE	4492
	6000001		DEC 0 B4	CONTENTS OF ACC,	4494
			SLC +1	PROD1, C 0	4496
		*	***** END OF	VARIABLES *****	4500
04102	2410005	STMO	SLA 5	MO, B19, TO B14	4510
04103	0363143		STA MO		4520
04104	2603612		BRU B23		4530
04105	2410005	STMXPR	SLA 5	MXPR, B19, TO B14	4540
04106	0363141		STA MXPR		4550
04107	2603612		BRU B23		4560
04110	2410005	STMXSL	SLA 5	MXSL, B19, TO B14	4570
04111	0363155		STA MXSL		4580
04112	2603612		BRU B23		4590
04113	2410005	STMXIN	SLA 5	MXIN, B19, TO B14	4600
04114	0363157		STA MXIN		4610
04115	2603612		BRU B23		4620
04116	2410013	STTAUS	SLA 11	TAUS, B19, TO B8	4630
04117	0364056		STA TAUS		4640
04120	2603612		BRU B23		4650
04121	0362531	STSTPT	STA NSTPRO	3 NEW SET POINT, B19	4660
04122	0262422		SUB STPROT	3 NEW SET POINT MINUS OLD, B19	4670
04123	2410011		SLA 9	DIFF, B10	4680
04124	0362527		STA AMB	3 B10	4690
04125	0062422		LDA STPROT	3 B19	4700
04126	2410005		SLA 5	B19 TO B14	4710
04127	0362525		STA ETAU	3 ETAU STARTS AT OLD SET POINT	4720
04130	0362535		STA OSTPNT	3 OSTPNT, B14	4730
04131	0062531		LDA NSTPRO	3	4740
04132	0362422		STA STPROT	3 NEW SET POINT NOW IN CONTROLLER	4750
04133	2504002		LDZ		4760
04134	2504006		MAQ		4770
04135	0064054		LDA PS3B4	3 CONSTANT TO CALC RAMP SLOPE, B4	4780
04136	1664056		DVD TAUS	3 RESULT IS B+4	4790
04137	2504006		MAQ		4800
04140	1562527		MPY AMB	3 RAMP CONSTANT, B6, IN A REG	4900
04141	2514003		BOV 1	TEST IF CONSTANT TOO BIG	5000
04142	2604142		BRU *	CONSTANT TOO BIG	5010
04143	0362523		STA A	3 A, B6	5020
04144	2504002		LDZ		5030
04145	0362521		STA TAU	3 SET INITIAL TAU TO ZERO	5040
04146	0362533		STA MIAE	3 SET INITIAL INTEGRAL VALUE TO ZERO	5050
04147	2603612		BRU B23		5060
		* * *	* * * SEGMENT FOR	MODE 4 * * * * *	5070
04150	0004153	B34	LDA BRUB35	SETTING UP FOR MODE 4	5080
04151	0303476		STA B33		5090
04152	2603571		BRU B7		5100
04153	2604154	BRUB35	BRU B35	MODE4 SWITCH VARIABLE	5110
04154	0003761	B35	LDA OP1	MODE 4, PNT HAS CONTROLLER NO.	5120

04155	0302162	STA E16			5130
04156	0002215	LDA BRUE15			5140
04157	0302175	STA E14		WANT TO CHANGE RATE TIME VARIABLES FOR	5150
		*		CONTROLLER INDEXED BY PNT	5160
04160	0762161	SPB SUB2	3	GO TO SUB2 AND USE FOR MODE 4	5170
04161	0002213	LDA RE16			5180
04162	0302162	STA E16		RESTORE SUB2	5190
04163	0002214	LDA RE14			5200
04164	0302175	STA E14		RESTORE SUB2	5210
04165	2603612	BRU B23		CHANGED TIME VARIABLES	5220
		* * * * * END OF MODE		4 SEGMENT * * * * *	5230
		* * * * * SEGMENT		FOR MODE 6 * * * * *	5240
04166	2604167	BRUB36 BRU B36			5250
04167	0762161	B36 SPB SUB2	3		5260
04170	2603612	BRU B23			5270
		* * * * * END OF		MODE 6 SEGMENT * * * * *	5280
		* * * * * SEGMENT		FOR MODE 12 * * * * *	5282
04171	0302273	STNOCO STA NOCONT			5290
04172	0004176	LDA BRUB37			5300
04173	0303476	STA B33		SET SWITCH POINT	5310
04174	1004100	DLD PROD1		PRODUCT RESULTING AFTER LAST CHANGE IN	5320
		*		SAMPLING INTERVAL	5330
04175	2603416	BRU B38		CHANGE ALL TIME VARIABLES	5340
04176	2604177	BRUB37 BRU B37			5350
04177	0762161	B37 SPB SUB2	3	SET UP FOR FIRST TIME	5360
04200	2603612	BRU B23		FINISHED CHANGES FOR MODE 12	5370
		* * * * * END OF		MODE 12 SEGMENT * * * * *	5372
	6102000	END DUSMOR			5380

APPENDIX C

PROGRAM EXTENSIONS

There were several techniques that were not included in the program either because of lack of time or because they developed while the program was being tested. Those that might be valuable for inclusion in a controller program cover

- Operator-computer communication
- Dead-band requests
- Real time synchronization
- Filtering
- Optimization
- "Bumpless" transfer
- Digital control algorithms.

OPERATOR-COMPUTER COMMUNICATION

As can be seen in the listing of the calculation routine in Appendix B (octal locations 2540 through 3242), certain constants are multiplied together everytime a controller output is calculated. For example, the term, $KP * KD / \Delta T$ is formed at each output calculation. If this product were formed and stored everytime that KP, KD, or ΔT was read in, then this would shorten the calculation time. In this example, everytime KP, KD, or ΔT was read in from the operator knobs, a new product would have to be formed and made available in the calculation routine. Shortening the calculation routine would shorten the delay between sampling the process output and generating a new process input.

The GE 412 requires six multiplexer scans to change one controller parameter in one process loop. When the sampling interval is around 0.2 sec, or ten times this, one scan is made every interval, and the parameter change takes about a second. If the sampling period is 0.8 sec then a parameter

change takes 4 sec. Reading the operator's panel could be speeded up by changing the program to perform scans as there is time available.

Another change that could be advantageous is logging on the typewriter all parameters that were changed using the operator knobs. If the time of day were kept by the monitor program, then all changes could be typed with the accompanying time. The program could be simplified more by using the typewriter instead of the operator panel to communicate all controller changes.

A final suggestion for changes in the controller tuning routine concerns the multiplier knob, MULT. Instead of using just a single knob, any number could be used to give a wider range of multiplier values. Since the multiplier is used only to modify the sampling interval, using different clock pulse rates could also give a greater time range.

DEAD-BAND REQUESTS

The present dead-band routine (see Figure 9) forms a step in the error when the dead-band is exceeded. Figure 13 shows variations of a different method for allowing a dead-band. The method of part (a) gives a smooth transition from the dead-band zone to the non-zero error zone, while the method of part (b) has the same transition, but gives a greater weight to errors of magnitude larger than E_1 .

REAL TIME SYNCHRONIZATION

Considerable thought went into devising the timing system, because timing errors accumulate in the integral mode. Operating experience with the program revealed that a simpler timing method could be implemented by using the priority interrupt to synchronize the computer to the asynchronous clock pulses. The short routine that is required to service the time interrupt

would cause negligible timing error. (Appendix D describes the automatic priority interrupts for the GE 412 computer). Another change in program timing could allow individually adjustable sampling intervals for each control loop.

FILTERING

The variable time intervals for the derivative calculation require some fairly involved "bookkeeping." The noise filtering can be improved and the program simplified by calculating the derivative every sampling interval. This would require a queue with a length equal to the number of sampling intervals between errors in the derivative calculation. For example, if the difference between every other error were divided by two sampling intervals, then there would have to be storage for the errors at the preceeding two sampling instants. After each derivative calculation, the present error would go to the end of the queue, the preceeding error would go to the front of the queue, and the error from two sampling intervals past would be lost. Thus, if no filtering were desired, the queue length would be one; if errors separated by two sampling intervals were used, the queue length would be two; etc. If more extensive filtering were desired, this could be added with a filter subroutine.

By suppressing the derivative calculation at a setpoint change, the large transient in the controller output at a setpoint change, such as in Figure 4, would be eliminated. Transients like this produce undesirable disturbances in the closed process loop.

OPTIMIZING

Controller optimization using a specific response criterion could be automated. In the modified integral of absolute error (MIAE) criterion, all the operator's responses could be programmed. In the MIAE criterion, the process has to be perturbed before each controller change.

BUMPLESS TRANSFER

Generally, the operator starts process loops manually, then he transfers to automatic control. Digital controllers can perform bumpless transfer from the manual to the automatic mode without operator intervention. The computer could sample the output signal from the controller in the manual mode and use it as the initial output in the automatic control mode.

DIGITAL CONTROL ALGORITHMS

Different control algorithms can be used for process control because the algorithms are all programmed subroutines. Digital controllers do not have to be difference approximations to standard analog controllers, but can be designed for specific control problems. As more is learned about a process, its digital controllers can be refined and improved.

Experience has shown that the control subroutine in the program described in this report can be simplified by using clamping values of 100%. This is a distinct advantage for the integral mode and causes no errors in the proportional and derivative modes.

FIGURE 13. Possible Error Functions for the Calculation Routine

APPENDIX D

THE GE 412 PROCESS CONTROL COMPUTER

This description of the 412 computer is included to further the understanding of the program, which was implemented in 412 machine language, and to aid future users of our 412.

COMPUTER DESCRIPTION

The General Electric 412 process control system consists of the arithmetic unit and memory, the automatic priority program interrupts, elapsed time counters, input-output multiplexer, and peripheral equipment for operator-computer communication.

Arithmetic Unit

The central processing unit is a single address, stored program, fixed point, binary digital computer. An important feature of the central processor for digital control of multiple loops is automatic address modification. The address portion of any of three specific memory locations may be automatically added to an instruction before it is executed to modify its address.

Automatic Interrupts

The automatic priority program interrupt enables the process system to immediately acknowledge the random occurrence of a critical condition by causing the appropriate command to be executed. Automatic program interrupt allows pulses from external sources (process sensing devices) and internal sources (completion signals from peripheral equipment or time overflow pulses) to cause automatic interruption of the program currently operating in the computer. The Program Interrupt Register is 12 bits long and is divided into three groups of four bits each. The program can select the first four, the first eight, or all twelve bits to cause program interruption; or it may inhibit

all interrupts. Interrupts occurring in an inhibited state are not lost, but are held in the interrupt register. When the register is enabled, interruption will occur. Therefore, a program using interrupts must enable the correct group of interrupts and provide a method to suppress the undesired interrupts.

When an interrupt occurs in an enabled group and interrupts are permitted, the following operations are performed by the processor's hardware. The instruction being executed is completed, and all further interrupts are inhibited until permitted by the program. The instruction counter and contents of the A register are saved. Bits of the interrupt register are examined sequentially in order of priority, starting with the highest priority, until the first bit causing the interruption is found. The command at the word associated with this bit is then executed.

Elapsed Time Counters

To measure time intervals, our GE 412 system has two elapsed time counters, which are 8 bits long for counting to 255. The three pulse periods available to the counters are 3.2 msec, 16.7 msec, and 1 sec. A control word fixes (for a particular counter) the interrupt status, the pulse period, and the number of pulses counted until overflow. The elapsed time counter overflow must be tested by a sense and branch command, and may feed the interrupt register. The elapsed time counters have a maximum probable positive error equal to the specified pulse period (3.2 msec, 16.7 msec, or 1 sec.)

Input-Output Multiplexer

The input-output multiplexer (called a scanner-distributor in the GE 412 system) provides computer-process communication. The scanner functions automatically after receiving an input or output command to perform one of the following:

- Converting analog signals to digital
- Converting digital signals to analog and holding
- Reading knobs on the operator panel
- Resetting keys on the operator panel.

The scanner sets its assigned bit in the interrupt register upon completing its operation. The scanner-distributor completes an operation in 50 msec when operating at maximum speed.

Peripheral Equipment

The one peripheral digital device that is used for this program, out of the many available, is the operator panel. Program commands direct the multiplexer (scanner-distributor) to read binary coded decimal digits from the operator knobs on the panel into the central processor's A register. The state (either set, or reset) of each of the two keys (lighted push - buttons) on the panel is determined by sense and branch commands. Each key is manually set, but must be reset by a multiplexer command.

GE 412 HARDWARE AND SOFTWARE DETAILS

Some details concerning the input-output multiplexer, priority interrupts, elapsed time scales, and operator panel are inaccurate or vague as described in the "GE 412 Programming Manual" and are developed further here.

Multiplexer

The digital-to-analog converters (DAC's) are supplied from a negative supply referenced to the GE 412 ground, and so supply a negative current to single ended loads. The maintenance manual erroneously shows this supply to be positive with respect to the 412 ground. The current range from the two outputs (OC2-J10 and OC2-J11) used in program tests was 0 to - 20 mA for a 0 to 1023 range to the converter register.

The Programming Manual gives an erroneous description for constructing the output control word that causes the DAC to function as a zero order hold. Current command words are listed in the maintenance print, "I/O Information Summary, Address and Connections, #43B952491." The command 303416 (octal) produces a continuous output from OC2-J10 until the output register is changed and the command is executed again. The construction of analog-to-digital control words is described correctly in the GE 412 Programming Manual.

The word lengths of the ADC output and DAC input are not clearly defined in the programmer's manual. In the fast scan mode, the ten significant bits of the ADC output are transferred to bits 8 through 17 of the A register to give a decimal range of 0 to 1023 with the binary point at the right of bit 17. The command to convert from binary to analog output (mnemonic LCV), transfers bits 10 through 19 of the A register to be converter register with the binary point at the right of bit 19.

The linearity of the DAC point, OC2-J11, was tested with a digital voltmeter. The voltmeter, a Cubic Model V-71P, has a four significant digit output and has a specified accuracy of $\pm 0.01\%$ of the reading, ± 1 digit. Table III gives the results of the linearity test. The first column of the table is percent of full scale, based on decimal 1000 as full scale; the second column is the octal representation of the first column with a binary point at the right of bit 19; and the third column is the voltmeter reading. The DAC for OC2-J11 is linear within 0.1% of full scale.

Priority Interrupt

The operator must set the register selector knob to "PI" to view the state of the 12 priority interrupt channels on the auxiliary display register.

Elapsed Time Scalers

After a time pulse scaler over-flows, the timer interrupt must be answered by clearing its bit in the interrupt register and resetting the scaler overflow bit. If a "Branch on Timer Overflow" instruction is not executed to clear and reset, a priority interrupt will continue to occur even though the scaler has stopped.

Operator Panel

The states of either the "Data Insert" or the "Data Select" key can be sensed in two machine cycle times with the proper machine language command. The scanner can also be directed to reset either of these keys. These four codes are not listed with the mnemonic operating codes in the programmer's manual, so they are listed below.

TABLE III. *Data to Show Linearity of DAC Channel*

<u>Percent of Full Scale</u>	<u>Octal Representation of Converter Register Word</u>	<u>Digital Voltmeter Reading, V</u>
0	0000	0
0.1	0001	0.009
0.2	0002	0.018
0.5	0005	0.044
0.8	0010	0.070
1.0	0012	0.089
2.0	0024	0.180
5.0	0062	0.450
8.0	0120	0.720
10.0	0144	0.900
20.0	0310	1.800
30.0	0454	2.706
40.0	0620	3.607
50.0	0764	4.509
60.0	1130	5.409
70.0	1274	5.309
80.0	1440	7.209
90.0	1604	8.108
100.0	1750	9.006

If the key is set, the next command is executed, but if the key is reset, the second command after the test code is executed.

If the key is set, the next command is executed, but if the key is reset, the second command after the test code is executed.

```
Reset "Data Insert" key          3044710 (octal)
Reset "Data Select" key         3054010 (octal)
```

BIBLIOGRAPHY

The following references give general background information about direct digital control. In addition, the GE 412 Programming Manual, gives a rather complete description of the GE 412 Computer at Pacific Northwest Laboratory.

1. J. C. Belz, G. J. Kirk, and P. S. Radcliffe. "Application of the Prodac 50 System to D.D.C." 1965 IEEE International Convention Record, part 3, p 102.
2. J. W. Bernard and J. F. Cashen. "Direct Digital Control -- Questions That Must Be Answered" 19th Annual ISA Conference. October 1965.
3. J. W. Bernard and J. W. Wnykowski. "Direct Digital Control Experience in a Chemical Process," ISA Journal, vol 12, no. 12, pp. 43-47. December 1965.
4. H. S. Drewry and J. R. Howard. "For Successful Computing Control Start Simple" Control Engineering, p 76, August 1965.
5. GE 412 Programming Manual, General Electric Company, Arizona
6. A. L. Giusti, R. E. Otto, T. J. Williams. "Direct Digital Control," Control Engineering, vol 9, no. 6. June 1962.
7. L. S. Gomolak. "Direct Digital Control in Industry," Electronics, vol 37, no. 26. July 1964.
8. S. C. Gupta and C. W. Ross. "Simulation Evaluation of a Digital Control System," ISA Transactions, vol 11, no. 7, pp 271-279. July 1964.
9. H. W. Mergler and K. H. Hubbard. "Digital Control for the Single Loop," Control Engineering, vol 12, no. 2, pp 61-64. February 1965.
10. R. A. Morley and C. M. Cundall. "The Ferranti System and Experience with Direct Digital Control," Process Control and Automation, vol 12, no. 7, pp 289-295. July 1965.
11. C. W. Ross. "Control of a Process with Analyzer in the Feedback Loop," ISA Transactions, vol 10, no. 1, p 69. January 1963.
12. J. H. Wilkinson. Rounding Errors in Algebraic Processes, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1963.
13. G. V. Woodley. "Programming Aspects of Direct Digital Control," 20th Annual ISA Conference and Exhibit. October 1965.

DISTRIBUTIONNumber
of Copies

3	<u>Argonne National Laboratory</u> LMFBR Program Office Senior RDT Site Rep R. A. Jaross
53	<u>Atomic Energy Commission, Washington, D. C.</u> <u>Division of Reactor Development & Technology</u> J. J. Morabito - Headquarters (25) M. Shaw, Director, RDT Asst Dir for Nuclear Safety Analysis & Evaluation Br, RDT:NS Environmental & Sanitary Engineering Br, RDT:NS Research & Development Br, RDT:NS Asst Dir for Plant Engineering:RDT Applications & Facilities Br, RDT:PE Components Br, RDT:PE Instrumentation & Control Br, RDT:PE Systems Engineering Br, RDT:PE Asst Dir for Program Analysis:RDT Asst Dir for Project Management:RDT Liquid Metals Projects Br, RDT:PM FFTF Project Manager RDT:PM (3) Asst Dir for Reactor Engineering Control Mechanisms Branch RDT:RE Core Design Branch, RDT:RE Fuel Fabrications Branch, RDT:RE Fuel Handling Branch, RDT:RE Reactor Vessels Branch, RDT:RE Asst Dir for Reactor Technology Chemistry & Chemical Separations Br, RDT:RT (2) Fuels & Materials Branch, RDT:RT Reactor Physics Branch, RDT:RT Special Technology Branch, RDT:RT
2	<u>Atomics International</u> L. E. Glasgow Senior RDT Site Rep
1	<u>Atomic Power Development Associates</u> Senior RDT Site Rep

Number
of Copies

296	<u>Division of Technical Information Extension</u>
47	<u>Division of Technical Information Extension</u> UK/U.S. Fast Reactor Exchange (12) EURATOM/U.S. Fast Reactor Exchange (35)
5	<u>Douglas United Nuclear</u> T. W. Ambrose W. Dalos P. D. Gross G. F. Owsley O. C. Schroeder
2	<u>General Atomics Division</u> D. Coburn
4	<u>General Electric Company, Richland</u> R. L. Dickeman D. W. Leiby M. C. Leverett C. F. Poor
1	<u>General Electric Company, San Jose</u> Advanced Products Operation Senior RDT Site Rep
7	<u>General Electric Company, Sunnyvale</u> Bertram Wolfe Dr. K. Cohen, General Manager, APO (3) Dr. J. H. Field APO (3)
1	<u>General Electric Company, Vallecitos</u> Dr. L. Wilkinson
2	<u>Idaho Nuclear Corporation</u> D. R. de Boisbland Senior RDT Site Rep
3	<u>Isochem</u> H. C. Rathvon R. E. Tomlinson R. A. Watrous

Number
of Copies

15	<u>Richland Operations Office</u> P. G. Holsted, Senior RDT Site Rep-PNL (10) L. R. Lucas C. L. Robinson, Acting Director, Research & Development Division (2) R. K. Sharp Technical Information Library
1	<u>San Francisco Operations Office</u> J. B. Radcliffe, Director
1	<u>Stanford University</u> Dr. R. Shet
1	<u>Westinghouse Electric Corporation,</u> <u>Pittsburgh, Pennsylvania</u> Dr. J.C.R. Kelly
127	<u>Battelle-Northwest</u> D. T. Aase F. W. Albaugh E. R. Astley J. M. Atwood R. D. Benham (5) R. A. Bennett J. R. Boldt C. L. Boyd D. C. Boyd C. L. Brown W. L. Bunch C. A. Burgess W. A. Burns C. P. Cabell R. L. Carter T. T. Claudson L. D. Coffin P. D. Cohn C. R. Cole D. L. Condotta R. F. Corlett G. J. Crandall T. J. Crocker R. E. Dahl G. M. Dalen J. M. Davidson (2) J. L. Deichman

Number
of Copies

R. D. Dierks
D. R. Doman
W. D. Draper
G. E. Driver
R. V. Dulin
L. M. Finch
J. K. Flickinger
C. D. Flowers/M. T. Jakub
G. F. Garlick
L. H. Gerhardstein
S. M. Gill
C. L. Goss
E. W. Greenwell
V. W. Gustafson
J. E. Hansen
H. Harty
R. A. Harvey
J. W. Helm
R. J. Hennig
C. W. Higby
P. L. Hofmann
D. L. Hovorka
H. G. Johnson
J. E. Kaveckis
W. S. Kelly
J. R. Kosorok (20)
R. L. Knecht
L. W. Lang
H. D. Lenkersdorfer
C. W. Lindenmeier
W. W. Little, Jr.
R. J. Lobsinger
M. K. Mahaffey
K. R. Merckx
W. B. McDonald
M. K. Miihlhollen
D. M. Nero
M. G. Patrick
L. T. Pedersen
R. E. Peterson
J. G. Rake
D. E. Rasmussen
F. G. Richards
W. E. Roake

Number
of Copies

C. N. Rockwood
D. P. Schively
F. H. Shadel
D. W. Shannon
B. O. Shaver
D. E. Simpson
W. G. Spear
C. C. Steele
G. H. Strong
C. D. Swanson
G. R. Taylor
G. G. Thieme
J. P. Thomas
G. K. Tingey
K. G. Toyoda
R. A. Walker
J. H. Westsik
R. G. Wheeler
R. D. Widrig
N. G. Wittenbrock
M. R. Wood
D. C. Worlton
J. M. Yatabe
H. H. Yoshikawa
Technical Information Files (5)
Technical Publications (2)