

AN ABSTRACT OF THE THESIS OF

Alla J. M. Bakir for the degree of Master of Science in
Nuclear Engineering presented on May 19, 1988

Title: Development of a Personal Computer-based Reactivi-
ty Meter for a Research Reactor

Abstract approved: *Redacted for Privacy* _____
Stephen E. Binney

A personal computer with a Data Acquisition and Control Adapter interface has been used to monitor the power of the Oregon State University TRIGA Reactor, and from the time behavior of the reactor power the reactivity has been computed using the point kinetics equations and displayed on a digital read-out panel.

This technique has been verified with several measurements, including pulse reactivity, prompt jump and drop, and control rod calibration.

**Development of a Personal Computer-based
Reactivity Meter for a Research Reactor**

by

Alla J. M. Bakir

A THESIS

Submitted to

Oregon State University

**in partial fulfillment of
the requirements for the
degree of
Master of Science**

Completed May 19, 1988

Commencement June 1989

APPROVED:

Redacted for Privacy

Professor of Nuclear Engineering in charge of major

Redacted for Privacy

Head of department of Nuclear Engineering

Redacted for Privacy

Dean of Graduate School

Date thesis is presented

May 19, 1988

Typed by Sadie's Word Processing for Alla J. Mohammad Bakir

ACKNOWLEDGMENTS

I am very pleased that I have been able to contribute to this area of research. I would like to express my gratitude to my advisor, Dr. Stephen E. Binney, who motivated me to do my best. I am very thankful to my committee, Dr. Andrew C. Klein, professor Donald L. Amort, and Dr. Jonathan D. Istok. A special thanks goes to the reactor staff, Arthur David Hall, Harold Busby, and Terry V. Anderson, who helped me to do some experiments, and provided me with some information about the Oregon State Triga Reactor.

I dedicate this thesis to my parents and my niece Zinna.

TABLE OF CONTENTS

	<u>Page</u>
I. INTRODUCTION	1
II. THEORY OF REACTIVITY MEASUREMENT	4
II.1 Introduction	4
II.2 Background	4
II.3 Point Reactor Kinetics	5
II.4 Delayed Neutrons	6
II.5 Step Change in Reactivity	8
II.6 Derivation of Integrodifferential Reactor Kinetic Equations	12
III. EQUIPMENT	15
III.1 Introduction	15
III.2 Fission Chamber and Log-Channel Amplifier	15
III.3 Data Acquisition System	17
III.4 Data Acquisition and Control Adapter	20
III.5 Digital Read-out Panel	21
III.6 Personal Computer	22
IV. SOFTWARE DESCRIPTION	24
IV.1 Introduction	24
IV.2 DACA Functions	24
IV.3 Formulation of the Reactivity Program LOGGER	26
V. RESULTS	32
V.1 Introduction	32
V.2 Reactor Power Equation	32
V.3 Step Reactivity Change Experiments	37
A. \$0.20 Step Change in Reactivity	37
B. \$0.70 Step Change in Reactivity	39
V.4 Safety Rod Calibration Curve	44
V.5 Positive Reactivity Prompt Jump	47
V.6 Negative Reactivity Prompt Drop	51
VI. CONCLUSIONS AND RECOMMENDATIONS	53
REFERENCES	55
APPENDICES	
A. FISSION CHAMBER DETECTOR AND AMPLIFIER	56

	<u>Page</u>
B. IBM PERSONAL COMPUTER DATA ACQUISITION AND CONTROL ADAPTER (SOFTWARE AND CIRCUIT DIAGRAMS)	65
C. PROGRAMS	111

LIST OF FIGURES

<u>Figure</u>		<u>Page</u>
II.1	Mechanism of delayed neutron emission	8
III.1	Block diagram of reactivity meter system	16
III.2	Basic data acquisition system - block diagram	18
III.3	Typical data acquisition/sampled-data system	19
III.4	Digital read-out panel	23
V.1	ADC output code versus reactor power (whole range)	35
V.2	ADC output code vs. reactor power from 0.01 to 100 W	35
V.3	ADC output code vs. reactor power from 100 to 1000 W	36
V.4	ADC output code vs. reactor power from 1 kW to 1 MW	36
V.5	Power trace following step reactivity change of \$0.20	38
V.6	Computed reactivity following pulse reactivity change of \$0.20	38
V.7	Excess reactivity versus reactor period for OSTR	40
V.8	Power trace following step reactivity change of \$0.70	42
V.9	Computed reactivity following step reactivity change of \$0.70	42
V.10	Comparison of safety rod calibration curves	46
V.11	Power trace following a + \$0.40 step reactivity insertion	48
V.12	Associated reactivity jump for a + \$0.40 step insertion	48

<u>Figure</u>		<u>Page</u>
V.13	Time behavior of reactor power following step reactivity insertion of + β 0.40 on an expanded scale	50
V.14	Power trace following a - β 0.40 step reactivity insertion	52
V.15	Associated reactivity drop for a - β 0.40 step insertion	52

LIST OF TABLES

<u>Table</u>		<u>Page</u>
II.1	Characteristics of Delayed Fission Neutrons in Thermal Fission	7
IV.1	Delayed Neutron Data for ^{235}U	31
V.1	ADC Output Code Versus Reactor Power	33
V.2	Part of the Output of LOGGER for + \$0.70 Pulse	43
V.3	Reactivity Worth for Different Segments of the Safety Rod	45
V.4	Comparison Between Reactivity Meter Measurement and Period Method Measurement of the Reactivity Worth for the Safety Rod	45

Development of a Personal Computer-based Reactivity Meter for a Research Reactor

CHAPTER I

INTRODUCTION

A reactivity meter is a useful device for monitoring and control of a nuclear reactor. There are three different techniques to measure the reactivity of a nuclear reactor. First, there is a static technique in which a reactivity change to be measured is balanced by another known change in reactivity, where the reactor power (at criticality) is maintained constant. Second, there is the dynamic technique, in which a reactivity change is left unbalanced. Different methods can then be used to determine the reactivity. One of these methods is the asymptotic period method, in which the stable or asymptotic period is measured. Then the inhour equation is used to find the reactivity. This technique works well for positive periods, but the negative periods are dominated by the longest delayed neutron precursor decay and hence provide very low sensitivity to the negative reactivity. Some other methods for this technique are the rod drop, source jerk, rod oscillator, and pulsed neutron methods.

Finally, there is the kinetic technique, in which reactivity changes may be made continuously and the reac-

tivity at each instant determined by analyzing the variation of reactor power level using the point kinetics equations.

The history of reactivity meter use began in the early 1950's. The rod oscillator was one such method of this era. It was used to measure the reactivity of the CP-2 reactor as it was varied sinusoidally in time by oscillating the cadmium rod about its mean position.¹² Other measurements were made by reactor period method.

One modern and accurate method employs a kinetic reactivity measurement. This method utilizes a computer (digital or analog) to derive an instantaneous reactivity from the time behavior of the reactor power.⁵

The reactivity meter discussed in this thesis uses an analog to digital interface between a research reactor and personal computer. An IBM Data Acquisition and Control adapter (DACA) card inserted in a Leading Edge Model D personal computer monitors the power of the Oregon State University TRIGA Reactor (OSTR). The method of Tuttle⁵ has been modified to determine the reactivity of the reactor as a function of time.

The reactivity meter system is comprised of a fission chamber, wide range log channel amplifier, data acquisition and control adapter, digital read-out panel, and a Leading Edge Model D personal computer. The first two parts were already components of the OSTR.

Reactor power was measured by monitoring the log

power signal, where ten volts corresponds to the full steady state power of 1 MW. The DACA produces a code that is linearly proportional to the voltage monitored. A correlation equation for this code in terms of reactor power was determined and used. As soon as the reactor power history is determined, the associated reactivity can be calculated using the point kinetics equations.

One of the applications of the reactivity meter with the OSTR is the control rod calibration.

CHAPTER II

THEORY OF REACTIVITY MEASUREMENTS

II.1 Introduction

A personal computer reactivity meter converts power measurements into reactivity, using the point reactor-kinetics equations.

II.2 Background

The neutron flux is the total rate at which neutrons pass through a unit area regardless of the neutron direction [pg. 110 (1)]. The neutron flux (neutron/cm²-s) is defined as the neutron density (neutron/cm³) multiplied by average neutron velocity (cm/s).

The multiplication factor is defined as the ratio of the number of fission neutrons in one generation divided by the number of fission neutrons in the preceding generation [pg. 102 (2)]. In equation form this can be written as:

$$k = \frac{\text{Number of neutrons in one generation}}{\text{Number of neutrons in preceding one}} \quad (2.1)$$

If k is greater than 1, the reactor is called supercritical and the chain reaction increases with time. On the other hand, if k is less than 1, the reactor is subcritical and the chain reaction decreases with time.

Finally, if k is equal to 1, the reactor is critical and the chain reaction proceeds at a constant rate.

It is more convenient to measure the ratio of the deviation of the neutron multiplication factor from its critical value $k = 1$. [Pg. 239 in (1)]

This is known as the reactivity of the reactor, and is given the symbol ρ ; thus

$$\rho(t) = \frac{k(t) - 1}{k(t)} \quad (2.2)$$

II.3 Point Reactor Kinetics

The time behavior of the neutron population in a reactor core induced by changes in reactor multiplication is known as nuclear reactor kinetics [Pg. 233 (1)].

The one-speed diffusion equation can be used to derive the point reactor kinetics equations, which also can be derived by using the transport equation.

The point reactor kinetics equations are:

$$\frac{dn}{dt} = \left[\frac{\rho(t) - \beta}{\Lambda} \right] n(t) + \sum_{i=1}^6 \lambda_i C_i(t) + s \quad (2.3)$$

$$\frac{dC_i}{dt} = \frac{\beta_i}{\Lambda} n(t) - \lambda_i C_i(t) \quad \text{for } i=1,2,\dots,6 \quad (2.4)$$

where

$n(t)$ is the time-dependent neutron density
(neutrons/cm³)

$\rho(t)$ is the time-dependent reactivity

β is the total delayed neutron fraction

- $C_i(t)$ is the neutron precursor concentration of group i , (atoms/cm³)
- λ_i is the decay constant of precursor i , (s⁻¹)
- S is the independent neutron source, (neutrons/cm³-s)
- Λ is the neutron generation time, (s), and
- β_i is the i th group delayed neutron fraction.

II.4 Delayed Neutrons

There are two types of neutrons in a nuclear reactor, prompt and delayed. Most of the neutrons emitted in fission appear virtually at the instant of fission; these are prompt neutrons. A small fraction of the fission neutrons appear a relatively long time after the fission event; these are delayed neutrons. The time behavior of a reactor depends upon the various properties of these two types of neutrons [Pg. 267 in (2)].

The prompt neutrons form about 99.3% of the total neutrons and have an energy range of about 0.18 to 12 MeV, whereas the delayed neutrons form about 0.7% of the total neutrons and have an average energy of 0.5 MeV. Although only a very small fraction of the fission neutrons are delayed, these delayed neutrons are vital for the effective control of the fission chain reaction [pg. 6 in (1)].

Experimental studies of the rate of emission of the delayed neutrons have shown that these neutrons fall into six groups, each characterized by a definite exponential decay rate [Pg. 110 in (3)].

Table II.1 shows the average number of delayed neutrons per fission in each group, and the total fraction of fission neutrons for ^{233}U , ^{235}U and ^{239}Pu in thermal fission.

Table II.1 Characteristics of Delayed Fission Neutrons in Thermal Fission

Approximate Half-Life (seconds)	Number of Delayed Neutrons per Fission			Energy (MeV)
	U-233	U-235	Pu-239	
55	5.7×10^{-4}	5.2×10^{-4}	2.1×10^{-4}	0.25
23	19.7	34.6	18.2	0.46
6.2	16.6	31.0	12.9	0.41
2.3	18.4	62.4	19.9	0.45
0.61	3.4	18.2	5.2	0.41
0.23	2.2	6.6	2.7	-
Total delayed	0.0066	0.0158	0.0061	
Total fission				
neutrons	2.49	2.42	2.93	
Fraction delayed	0.0026	0.0065	0.0020	

[pg. 111 in (3)]

Most of the neutron-rich fission products undergo beta decay; in a few cases, however, the daughter is produced in an excited state with sufficient energy to make possible the emission of a neutron. It is in this manner that the delayed neutrons arise, the characteristic half-life being determined by that of the parent (or precursor) of the actual neutron emitter. [Pg. 111 in (3)]

For example the mechanism proposed to account for the 55-s group of delayed neutrons is shown in Figure II-1.

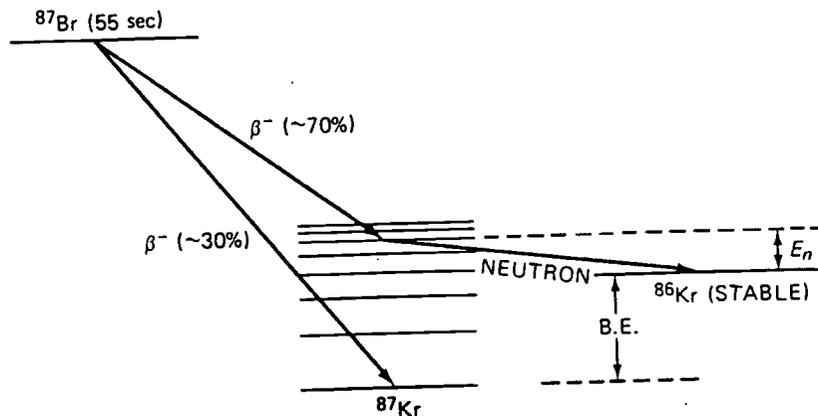


Figure II.1 Mechanism of delayed neutron emission.

[pg. 112 in (3)]

II.5 Step Change in Reactivity

A situation of interest is that a reactor operating in the critical state is subjected to a sudden (step) change in the effective multiplication factor (or reactivity). The resulting rate of change in the neutron density can then be readily evaluated by solving the linear, first order differential equations (2-3) and (2-4).

A common method for solving such equations is to use exponential trial functions, such as

$$n(t) = A e^{\omega t} \quad (2.5)$$

$$\text{and} \quad C_i(t) = B_i e^{\omega t} \quad \text{for } i=1,2,\dots,6 \quad (2.6)$$

where $n(t)$ and $C_i(t)$ are the neutron density and i^{th} precursor number density, respectively, at time t after

the reactivity is changed; the coefficients A , B_i , and ω are to be determined. The actual solutions are the sum of $x+1$ exponential terms, where x is the number of delayed neutron precursors.

From equations (2.4), (2.5), and (2.6), it is found that

$$B_i = \frac{\beta_i}{\Lambda (\omega + \lambda_i)} A \quad \text{for } i=1,2,\dots,6 \quad (2.7)$$

From this result, together with equations (2.5) and (2.6), equation (2.3) can be transformed into

$$\omega\Lambda = \rho + \sum_{i=1}^6 \left(\frac{\lambda_i \beta_i}{\omega + \lambda_i} - \beta_i \right) \quad (2.8)$$

where β has been replaced by the sum of the β_i terms and the source term has been neglected. It follows, therefore, that

$$\rho = \omega\Lambda + \sum_{i=1}^6 \frac{\omega\beta_i}{\omega + \lambda_i} \quad (2.9)$$

Equation (2.9) is the characteristic equation which relates the parameters ω to the properties of the reactor, as represented by the quantities ρ , Λ , β_i and λ_i . It is seen to be an algebraic equation of the seven possible values of ω . The variation of neutron density with time may then be expressed by a linear combination of seven terms of the form of equation (2.5), i.e.,

$$n(t) = A_0 e^{t\omega_0} + A_1 e^{t\omega_1} + \dots + A_6 e^{t\omega_6} \quad (2.10)$$

where w_0, w_1, \dots, w_6 are the seven roots of equation (2.9). The A_0, A_1, \dots, A_6 are constants determined by the initial conditions of the reactor in the steady state.

Some indication of the relative importance of the stable and transient terms in equation (2.10) may be obtained by considering the simple case in which there is assumed to be only one group of delayed neutrons. For a single (average) group of delayed neutrons, the variation of the neutron density with time is given by equation (2.10) as

$$n(t) = A_0 e^{t w_0} + A_1 e^{t w_1} \quad (2.11)$$

and the corresponding expression for the delayed-neutron precursor is

$$C(t) = B_0 e^{t w_0} + B_1 e^{t w_1} \quad (2.12)$$

The problem of immediate interest is to determine the constants, A_0, A_1, B_0 and B_1 . If $k \approx 1$ equations (2.4) and (2.9) for a single precursor are

$$\frac{dC}{dt} = \frac{\beta}{\ell} n - \lambda C \quad (2.13)$$

$$\rho = w \ell + \frac{w \beta}{w + \lambda} \quad (2.14)$$

where Λ is substituted by ℓ/k , ℓ is the prompt neutron life time, and λ is a decay constant of the one delayed neutron group, which is equal to

$$\lambda = \frac{\beta}{\sum_{i=1}^6 \frac{\beta_i}{\lambda_i}} \quad (2.15)$$

Equation (2.14) can be solved for ω . Provided $(\beta - \rho + \lambda l)^2 \gg 4(\lambda \rho l)$ and $|\rho| \ll \beta$, the two solutions of equation (2.14) are

$$\omega_0 \approx \frac{\lambda \rho}{\beta - \rho} \quad \text{and} \quad \omega_1 = \frac{-(\beta - \rho)}{l} \quad (2.16)$$

If n_0 is the neutron density in the critical state before the step change in the reactivity, i.e., at $t=0$, it follows from equation (2.11) that it is equal to $A_0 + A_1$. Similarly, the initial value of the precursor concentration C_0 is equal to $B_0 + B_1$ by equation (2.12). Furthermore, in the critical state dC/dt is zero, and so a relationship between C_0 and n_0 can be derived from equations (2.13) and (2.14). With the information now available, and using equation (2.16) for ω_0 and ω_1 , it is found, after neglecting certain terms of small magnitude, that

$$A_0 \approx \frac{\beta}{\beta - \rho} n_0 \quad \text{and} \quad A_1 \approx - \frac{\rho}{\beta - \rho} n_0 \quad (2.17)$$

The variation of the neutron density with time is consequently given by equations (2.11), (2.16), and (2.17) as

$$n \approx n_0 \left[\frac{\beta}{\beta - \rho} e^{\lambda \rho t / (\beta - \rho)} - \frac{\rho}{\beta - \rho} e^{-(\beta - \rho)t / l} \right] \quad (2.18)$$

where n is the neutron density at time t after the step change in reactivity.

The second term of equation (2.18) dies out in a very short time. Subsequently, the neutron density variation is given by the first term alone, namely,

$$n = n_0 \frac{\beta}{\beta - \rho} e^{\lambda \rho t / (\beta - \rho)} \quad (2.19)$$

For small times t , the exponential term is essentially unity; hence, if n/n_0 is replaced by P/P_0 , the prompt-jump power ratio is

$$\frac{P}{P_0} \approx \frac{\beta}{\beta - \rho} \quad (2.20)$$

A more exact treatment gives $(1-\rho)\beta/(\beta-\rho)$ for this ratio provided $\rho > \beta$; however, for nearly all cases of interest $1-\rho \approx 1$, and the result is then the same as equation (2.20). [pg 233 to 240 in (3)]

II.6 Derivation of Integrodifferential Reactor Kinetic Equations

With the use of equation (2.2), the point reactor kinetics equations (2.3) and (2.4) can be rewritten as follows:

$$\frac{dn}{dt} = \frac{n}{\Lambda} \left(1 - \frac{1}{k(t)} - \beta \right) + \sum_{i=1}^6 (\lambda_i C_i) + S \quad (2.21)$$

Rate of Change of Neutron Density	Net rate of Change of Neutrons Due to Absorption, Leakage, and Production of Prompt Neutrons	Delayed Neutron Production	Source
-----------------------------------	--	----------------------------	--------

$$\frac{dC_i}{dt} = \frac{n}{\Lambda} \beta_i - \lambda_i C_i \quad (2.22)$$

Rate of Change of the Precursor Group i with Respect to Time	Production Rate	Radioactive Decay Rate
--	-----------------	------------------------

Equation (2-22) may be integrated, subject to the condition that the precursor population was zero infinitely long ago.

$$C_i(t) = \int_{-\infty}^t \frac{n}{\Lambda} \beta_i e^{-\lambda_i(t-r)} dr \quad \text{for } i=1,2,\dots,6 \quad (2.23)$$

where r is a dummy variable of integration.

The time-dependent values of the precursor populations, C_i , may be entered in equation (2-21) to give a single integrodifferential equation:

$$\frac{dn}{dt} = \frac{n}{\ell} \left[k(1-\beta) - 1 \right] + \sum_{i=1}^6 \lambda_i \int_{-\infty}^t k \frac{n}{\ell} \beta_i e^{-\lambda_i(t-r)} dr + S \quad (2.24)$$

The dependence on the multiplication constant, k , has been made explicit by the relation:

$$\Lambda = \frac{\ell}{k} \quad (2.25)$$

Equation (2-24) can be solved directly for reactivity as:

$$\rho = \frac{k-1}{k} = \beta + \frac{1}{k} \left(\frac{\ell}{n} \right) \left[\frac{dn}{dt} - \sum_{i=1}^6 \lambda_i \int_{-\infty}^t k \frac{n}{\ell} \beta_i e^{-\lambda_i(t-r)} dr - S \right] \quad (2.26)$$

Expressing reactivity in units of the delayed neutron fraction, β , that is, in dollars ($\$1 = \beta$) and letting $a_i \beta = \beta_i$ equation (2.26) becomes:

$$\rho = 1 + \frac{1}{\beta k} \left(\frac{\ell}{n} \right) \left[\frac{dn}{dt} - \beta \sum_{i=1}^6 a_i \lambda_i \int_{-\infty}^t k \frac{n}{\ell} e^{-\lambda_i(t-r)} dr - S \right] \quad (2.27)$$

Equation (2-27) permits the determination of the instantaneous reactivity from a knowledge of the reactor power history $n(t)$. [pg. 17 to 19 in (5)].

CHAPTER III

EQUIPMENT

III.1 Introduction

The personal computer-based reactivity meter is comprised of four components: The fission chamber and log-channel amplifier of the OSTR, an IBM Data Acquisition and Control Adapter, a digital read-out panel, and a Leading Edge Model D personal computer.

The fission chamber measures the thermal neutron flux from the OSTR and outputs an electrical current that is proportional to the thermal neutron flux. The current is converted to voltage by the preamplifier. A value of ten volts represents the full power of the OSTR, which is 1 MW. This signal is converted into a digital signal by the Data Acquisition and Control Adapter to give a code that represents the power. This code is used to find reactivity by the program LOGGER (see Appendix C) and display the magnitude of the reactivity on the digital read-out panel. Figure III-1 shows a block diagram of the reactivity meter system.

III.2 Fission Chamber and Log-Channel Amplifier

The fission chamber is a Reuter Stokes Model RS-C3-2510-114 detector and is designed to detect thermal neutron fluxes from 1 to 2×10^{10} nv (nv = neutrons/cm²-s).

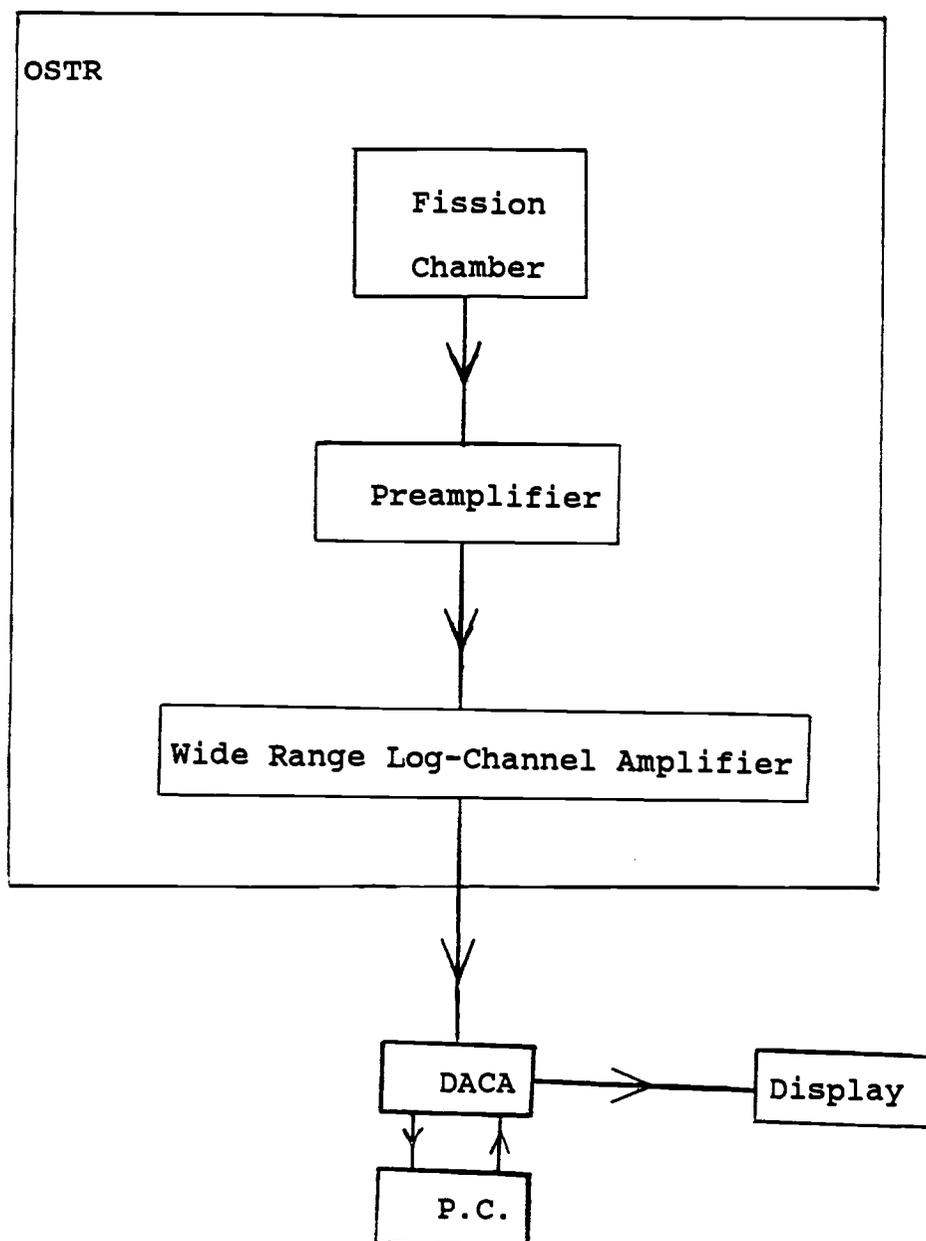


Figure III.1 Block diagram of reactivity meter system.

The fission chamber can be operated at any temperature up to 300°C. It has an operating voltage range from 400 to 700 volts. The current output is directly proportional to the thermal neutron flux, ϕ

$$I_n = k \phi$$

where $k = 1.4 \times 10^{-13}$ A/nv.

Basically, a fission chamber consists of a nitrogen filled cylindrical aluminum can containing several cylindrical electrodes coated with a thin layer of 90% enriched uranium. Thermal neutrons, incident on the chamber coating fission some of the ^{235}U atoms. The fission fragments then ionize the nitrogen in the chamber. The ions are collected and an electrical pulse is detected [pg. 32 (8)]. Specifications for the fission chamber are located in Appendix A.

The current signal out of the fission chamber is converted to a voltage signal by the pre-amplifier, and then amplified by the wide range log power channel amplifier to give a voltage signal proportional to the logarithm of the power of the reactor.

Some descriptions and specifications for the wide range log power channel are found in Appendix A.

III.3 Data Acquisition System

A Data Acquisition and Control Adapter (DACA) was used as an interface between the OSTR and a personal computer to monitor the reactor power which can be used to calculate the reactivity.

In general a Data Acquisition System (DAS) is defined as an electronic instrument, or system of interconnected electronic devices, dedicated to the measurement and

quantization of analog signals for digital analysis or processing. A graphic representation of where the DAS "fits in" is found in Figure III.2. Once a parameter to be measured is translated into the analog-electrical domain, the DAS performs this translation. In some cases the DAS simply records, or stores, the digital data until a more sophisticated system may be capable of analysis or further processing [pg. 1 and 2 in (6)].

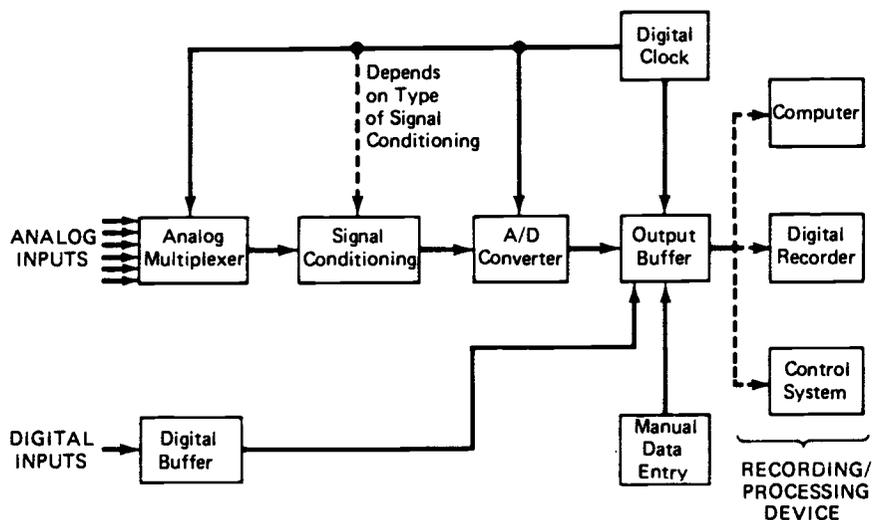


Figure III.2 Basic data acquisition system - block diagram. [pg. 2 in (6)]

Fundamental to the operation of a DAS is the concept of sampling, which may be defined as the act of measuring a continuous signal at discrete time intervals. In the case of the reactivity meter, this signal was the reactor power.

A typical data acquisition/sampled-data system is shown in Figure III.3. As the diagram indicates, four major processes are involved:

1. Sampling
2. Quantization
3. Digital processing, and
4. Recovery [pg. 11 in (6)].

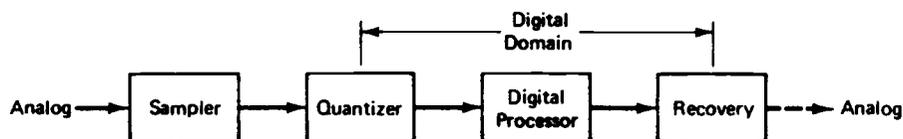


Figure III.3 Typical data acquisition/sampled-data sample [pg. 12 in (6)]

Monitoring a constantly changing quantity requires sampling at a rate slightly more than twice the highest frequency of the event. This is known as the sampling theorem. The sampling rate of the DACA used in this research can vary from 1 to about 2,000 samples/second and enables sampling of high frequency events, such as a pulse reactivity measurement.

The quantization process can be defined as the conversion of an input function that has values in a continuous range to digital output that has only discrete values [pg. 44 in (6)]. The Analog to Digital (A/D) converter is an example of a quantizer. A/D converters are usually categorized by the number of bits of resolution they

support; the greater the number of bits, the greater the number of discrete voltage levels that the converter, such as used in the DACA, can represent. For example, a 12-bit A/D converter, such as used in the DACA, can express 4096 different voltage levels.

The recovery function of the DAS consists of the reconstruction either of the original signal that was quantitized or a derivative of it. A Digital to Analog (D/A) converter is one example of a recovery device. In the case of this reactivity meter, the D/A function was not used.

III.4 Data Acquisition and Control Adapter

The IBM Personal Computer Data Acquisition and Control Adapter Model IBM 6323710 provides both analog and digital I/O capabilities. It is installed in a full-length expansion slot in a personal computer, and up to four may be installed in a system. The adapter provides:

1. Four analog input channels multiplexed into an analog-to-digital converter (ADC), with 12 bit resolution.
2. Two analog output channels, each having its own digital-to-analog converter (DAC), with 12 bit resolution.
3. A 16-bit digital input port.
4. A 16-bit digital output port.
5. A 32-bit timer.

6. A 16-bit, externally-clocked, timer/counter.
7. An expansion bus [pg. 1 in (7)].

The AD574AK 12-bit analog-to-digital converter is used to convert the analog signals (voltages) to digital values.

There are three switch-selectable ranges for the DACA:

1. -5 to +5 volts (bipolar)
2. -10 to +10 volts (bipolar)
3. 0 to +10 volts (unipolar)

Since the output signal of the log-power channel is in the range of 0-10 volts, which represents 0 to 1 MW of power, the 0 to +10V range of the DACA was used. More specification and a circuit diagram for the Data Acquisition and Control Adapter are found in Appendix B.

III.5 Digital Read-out Panel

As soon as the reactivity is calculated (using some special software functions discussed later in Chapter IV), its value is displayed on a digital read-out panel to provide the reactor operator with the value of reactivity. This device was designed and assembled by the author. The value of the reactivity was taken from the DACA as a binary-coded-decimal (BCD) number. Then the 7447 integrated circuit BCD-To-Seven Segment decoder/drivers were used, as shown in Figure III-4, to drive four IEE 1461E

displays from the BCD reactivity value. Bit 0 was used for the sign of the reactivity.

III.6 Personal Computer

A Leading Edge Model D personal computer was used to run the reactivity program (LOGGER). This personal computer has the following features:

1. Two built-in, 5 $\frac{1}{4}$ inch, double-density, double-sided floppy disk drives.
2. 640 kilobytes of random access memory (RAM) for program and work storage.
3. A parallel input/output (I/O) port for the printer.
4. A serial I/O port for either a printer or a communications device.
5. A built-in date and time-of-day circuit with 30 day battery back-up.
6. Four expansion slots [pg. 1 in (4)].

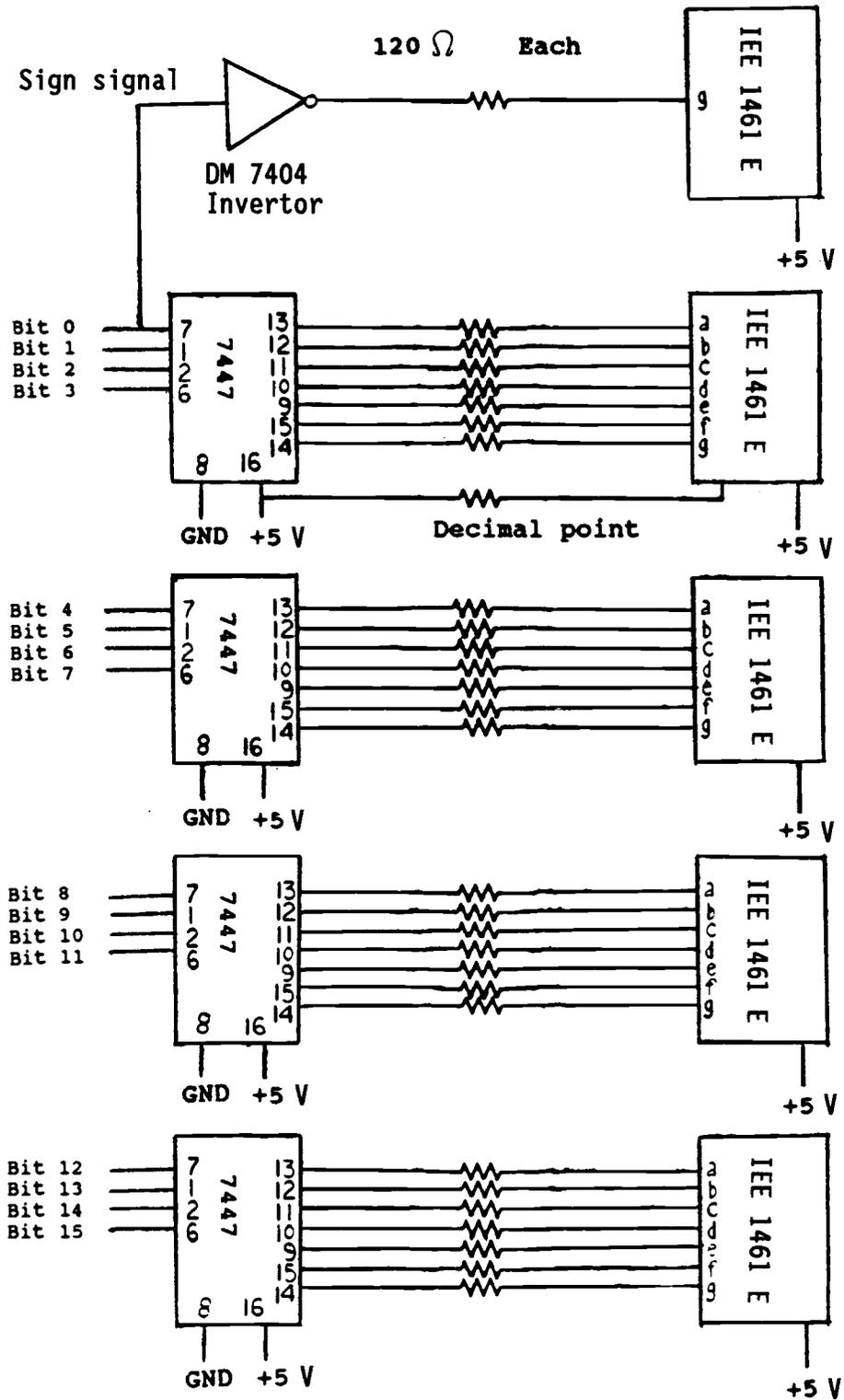


Figure III.4 Digital read-out panel.

CHAPTER IV

SOFTWARE DESCRIPTION

IV.1 Introduction

The reactivity program LOGGER has three major sections. The first section reads reactor power with the DACA; the next section solves the reactor point kinetics equations to find the reactivity; and finally the third section displays the value of the reactivity on a digital readout panel.

IV.2 DACA Functions

There are three types of functions used by the DACA:

1. Input Functions: Collect input data and move it to memory.
2. Output Functions: Move data from memory to an external device.
3. Utility Functions: Control counter/timers and program execution.

Each of these functions also fall into distinct classes according to the rate or frequency at which the adapter performs them:

- a. Simple Functions: Also called non-interactive functions, these execute only once.
- b. Multiple Functions: These iterative functions execute according to the number of times specified by a count argument entered

into the function. A rate argument governs the rate of execution.

- c. Scanning Function: Scans are sets of single inputs collected from a range of consecutively numbered channels.

Function names reflect the function type and device involved. The first set of letters of the function name is one of the following:

- A = analog device
- B = binary device
- Bit = a binary digit device
- C = counter device.

The interior two letters of the function name indicate input or output:

- IN = input function
- OU = output function

The last letters show the rate type:

- S = simple function
- M = multiple function
- SC = scanning function

For example, BINM is a multiple binary input function (Binary Input Multiple). AINS is a simple analog input (Analog Input Simple) function [Pg. 3.3 and 3.4 in (9)].

These functions can be used with compiled BASIC, Interpreted BASICA, Lattice C, or FORTRAN programs.

More instructions for these functions in FORTRAN language are found in Appendix B.

IV.3 Formulation of the Reactivity Program LOGGER:

The LOGGER program was developed by slightly modifying a program written by R.J. Tuttle.⁵ The equation for reactivity (2.27) is a continuous function of the neutron density, n . The power data, however, are measured as a discrete variable approximating the time behavior of n . Therefore, the continuous problem must be converted to a discrete problem that can be solved by means of a personal computer.

The measured power is proportional to an integral of the neutron absorption rate over a short time interval:

$$P(m) = A \int_{(m-1)\Delta t}^{m\Delta t=t} \frac{n}{\ell} dr \approx A \frac{n(t)}{\ell} \Delta t \quad (4.4)$$

where $P(m)$ = power at time $m\Delta t$

A = proportionality constant

m = discrete time interval index

$$\text{Therefore} \quad \frac{n(t)}{\ell} \approx \frac{P(m)}{A\Delta t} \quad (4.5)$$

The rate of change of the neutron density is approximated by the difference between successive power data:

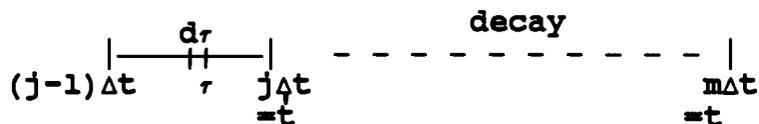
$$\frac{dn}{dt} \approx \frac{\ell}{A\Delta t} \left[\frac{P(m+1) - P(m)}{\Delta t} \right] \quad (4.6)$$

The convolution integral for the delayed neutron source in equation (2.27) is replaced by an initial value

based on the assumption that the power has been constant for a sufficient time such that the precursor populations are at their equilibrium values and can be approximated by a summation from time minus infinity to time t . This can be shown as:

$$\begin{aligned}
 \int_{-\infty}^t k \frac{n}{l} e^{-\lambda_i(t-\tau)} d\tau &= \int_{-\infty}^0 k \frac{n}{l} e^{-\lambda_i(t-\tau)} d\tau + \int_0^t k \frac{n}{l} e^{-\lambda_i(t-\tau)} d\tau \\
 &= \frac{1}{\lambda_i} k \frac{n}{l} e^{-\lambda_i(t-\tau)} \Big|_{-\infty}^0 + \int_0^t k \frac{n}{l} e^{-\lambda_i(t-\tau)} d\tau \\
 \int_{-\infty}^t k \frac{n}{l} e^{-\lambda_i(t-\tau)} d\tau &= \\
 \frac{1}{\lambda_i} k(0) \frac{n(0)}{l} e^{-\lambda_i t} + \sum_{j=1}^m \left[\int_{(j-1)\Delta t}^{j\Delta t} k \frac{n(t')}{l} e^{-\lambda_i(t'-\tau)} d\tau \right] e^{-\lambda_i(m-j)\Delta t}
 \end{aligned} \tag{4.7}$$

The term $e^{-\lambda_i(m-j)\Delta t}$ in equation (4.7) represents the decay of the precursor from time $j\Delta t$ to $m\Delta t$, $t=j\Delta t$, and $(j-1)\Delta t \leq \tau \leq j\Delta t$



The power in each time interval is treated as a two-term expansion series determined by the two adjoining power points, neglecting the change in k during the time interval;

$$k \frac{n}{l} (\vartheta) = k \frac{n}{l} \Big|_{\vartheta=0} + \vartheta \frac{d}{d\vartheta} \left(k \frac{n}{l} \right) \tag{4.8}$$

where ϑ is a dummy variable.

Equations (4.5), (4.6) and (4.8) can be used to evaluate the integral term on the right hand side of equation (4.7):

$$\int_{(j-1)\Delta t}^{j\Delta t} k \frac{n(\dot{t})}{\ell} e^{-\lambda_i(\dot{t}-\tau)} d\tau = \int_{(j-1)\Delta t}^{j\Delta t} k \frac{n(j)}{\ell} e^{-\lambda_i(\dot{t}-\tau)} d\tau$$

$$+ \int_{(j-1)\Delta t}^{j\Delta t} \tau \frac{d}{d\tau} \left(k \frac{n}{\ell} \right) e^{-\lambda_i(\dot{t}-\tau)} d\tau$$

If it is assumed that $k(j) \approx k(j-1)$, and $n(j) \approx n(j-1)$, then

$$\int_{(j-1)\Delta t}^{j\Delta t} k \frac{n(\dot{t})}{\ell} e^{-\lambda_i(\dot{t}-\tau)} d\tau \approx k(j) \frac{P(j)}{A\Delta t} \frac{1}{\lambda_i} \left[e^{-\lambda_i(\dot{t}-\tau)} \Big|_{(j-1)\Delta t}^{j\Delta t} \right]$$

$$+ \left[k(j) \left(\frac{1}{A\Delta t} \right) \left(\frac{P(j+1)-P(j)}{\Delta t} \right) \int_{(j-1)\Delta t}^{j\Delta t} \tau e^{-\lambda_i(\dot{t}-\tau)} d\tau \right]$$

The integral term on the right hand side can be evaluated by parts to get equation (4.9).

$$\text{So } \int_{(j-1)\Delta t}^{j\Delta t} k \frac{n(\dot{t})}{\ell} e^{-\lambda_i(\dot{t}-\tau)} d\tau \approx \frac{k(j)}{\lambda_i A \Delta t} \left(P(j) (1 - e^{-\lambda_i \Delta t}) + \right.$$

$$\left. \left[\frac{P(j+1)-P(j)}{\Delta t} \right] \left(\Delta t - \frac{1 - e^{-\lambda_i \Delta t}}{\lambda_i} \right) \right)$$

(4.9)

Substituting equations (4.6), (4.7) and (4.9) into

equation (2.27) we have:

$$\rho(m) = 1 + \frac{1}{\beta k(m)} \left(\frac{A\Delta t}{P(m)} \right) \left\{ \frac{\ell}{A\Delta t} \left[\frac{P(m+1) - P(m)}{\Delta t} \right] \right. \\ \left. - \beta \sum_{i=1}^6 a_i \lambda_i \left\{ \frac{1}{\lambda_i} \left[\frac{k(0)P(0)}{A\Delta t} e^{-\lambda_i m \Delta t} \right. \right. \right. \\ \left. \left. + \sum_{j=1}^m \frac{k(j)}{A\Delta t} \left(P(j) (1 - e^{-\lambda_i \Delta t}) + \left[\frac{P(j+1) - P(j)}{\Delta t} \right] \right) \right. \right. \\ \left. \left. \left(\Delta t - \frac{1 - e^{-\lambda_i \Delta t}}{\lambda_i} \right) e^{-\lambda_i (m-j) \Delta t} \right] \right\} - S \right\} \quad (4.10)$$

Simplifying and rearranging, this becomes:

$$\rho(m) = 1 + \left\{ \frac{\ell}{\beta A\Delta t} \left[P(m+1) - P(m) \right] - \sum_{i=1}^6 a_i \left\{ k(0)P(0) e^{-\lambda_i m \Delta t} \right. \right. \\ \left. \left. + \sum_{j=1}^m k(j) \left(P(j) (1 - e^{-\lambda_i \Delta t}) \right) \right. \right. \\ \left. \left. + \left[\frac{P(j+1) - P(j)}{\Delta t} \right] \left(\Delta t - \frac{1 - e^{-\lambda_i \Delta t}}{\lambda_i} \right) e^{-\lambda_i (m-j) \Delta t} \right\} - \frac{S}{\beta} \right\} \\ \left. \right/ (k(m)P(m)) \quad (4.11)$$

[Pages 20 to 21 in (5)]

The variable names used in the program LOGGER and their corresponding functions are:

$$N = m$$

$$R(N) = \rho(m)$$

$$BETA = \beta$$

$$K(N) = K(m)$$

$$L = \ell$$

$$P(N) = P(m)$$

$$DN = P(m+1) - P(m)$$

$$DT = \Delta T$$

$$A(I) = a_i$$

$$LAMBDA = \lambda_i$$

$$SUM = \sum_{i=1}^6 a_i \lambda_i \int_{-\infty}^{m\Delta t} k \frac{n}{\ell} e^{-\lambda_i(t-\tau)} d\tau$$

$$PBAR = [P(m+1) + P(m)]/2$$

$$AT = (\frac{\ell}{\beta})/\Delta t$$

$$DP = [P(m+1) - P(m)]/2$$

$$B(I) = e^{-\lambda_i \Delta t}$$

$$BD(I) = 1 - e^{-\lambda_i \Delta t}$$

$$D(I) = \frac{\lambda_i \Delta t - (1 - e^{-\lambda_i \Delta t})}{\lambda_i \Delta t}$$

$$PIN(N) = \frac{2\Delta t}{[P(m+1) + P(m)]}$$

$$SC = S/\beta$$

The values for constants are shown in Table IV.2.

Table IV.1 Delayed Neutron Data for ^{235}U

Group	Decay Constant λ_i (s^{-1})	Fraction β_i^*	$a_i = \frac{\beta_i}{\beta}$
1	0.0124	0.000215	0.033067
2	0.0305	0.001533	0.2190095
3	0.111	0.001372	0.195940
4	0.301	0.002765	0.394955
5	1.14	0.000805	0.115042
6	3.01	0.000294	0.041987

[Page 100 in (10)]

$\beta = 0.007$ for OSTR

$l = 0.0001$ s

$S = 0.0$ (assumed)

Program LOGGER is found in Appendix C.

*These values have been weighted by $\frac{0.007}{0.0065}$, so they are different from the reference values.

CHAPTER V

RESULTS

V.1 Introduction

Many experiments have been done to verify the performance of the program LOGGER. In order to provide more complete documentation, five examples will be analyzed here. The non-fission source was neglected during these measurements.

V.2 Reactor Power Equation

The program POWER was used to verify the ADC output code for different values of reactor power. Table V.1 shows the ADC output code results versus measured reactor output power.

Program CURVE was used to find the reactor power equation. Since the ADC output code is proportional to the logarithm of the reactor power, the logarithm of the reactor power was calculated and a linear regression was used to find an equation that is the form of

$$\text{Power} = 10^{(a + b C_p)}$$

where a and b are constants and

C_p is the ADC code associated with the power.

When the whole power range (0 to 1 MW) was fitted with one equation, the values for a and b were found to be

Table V.1

ADC Output Code Versus Reactor Power

<u>Code</u>	<u>Reactor Power</u>
955	18 mW
1040	30
1237	100
1432	300
1669	1 W
1839	3
2055	10
2249	30
2488	100
2544	120
2582	150
2604	180
2632	210
2665	240
2680	270
2702	300
2902	1 kW
3105	3
3305	10
3502	30
3718	100
3912	300
3998	500
4048	700
4092	900
4095	1 MW

$$a = -4.0641 \qquad b = 0.00244$$

and the correlation coefficient was equal to 0.9997, and so throughout Figure V.1 shows the relationship between reactor power (whole range) and the ADC output code.

Since the reactor amplifier circuit changes from a pulse rate circuit to a log ac (Campbell) circuit (see pg. 61) when the power changes between 100 and 300 W, the whole power range was divided into three sections. For each range section, an equation was found. Figures V.2, V.3, and V.4 show the relationship between the reactor power and the ADC output code for the three power ranges of 0.01 to 100 W, 100 W to 1 kW, and 1 kW to 1 MW, respectively.

The equations for these three power ranges are:

$$1. \text{ Power} = 10^{(-4.06015 + 0.00245 C_p)} \qquad (5.1)$$

$$\text{for } 0 \leq C_p \leq 2488 \qquad (0 \text{ W} \leq P \leq 100 \text{ W})$$

This equation has a correlation coefficient of 0.9997.

$$2. \text{ Power} = 10^{(-3.81625 + 0.00233 C_p)} \qquad (5.2)$$

$$\text{for } 2488 < C_p < 2902 \qquad (100 \text{ W} < P < 1 \text{ kW})$$

This equation has a correlation coefficient of 0.9951.

$$3. \text{ Power} = 10^{(-4.26878 + 0.0025 C_p)} \qquad (5.3)$$

$$\text{for } 2902 \leq C_p \leq 4095 \qquad (1 \text{ kW} \leq P \leq 1 \text{ MW})$$

This equation has a correlation coefficient of 0.9998. Listings of programs POWER and CURVE are found in Appendix C.

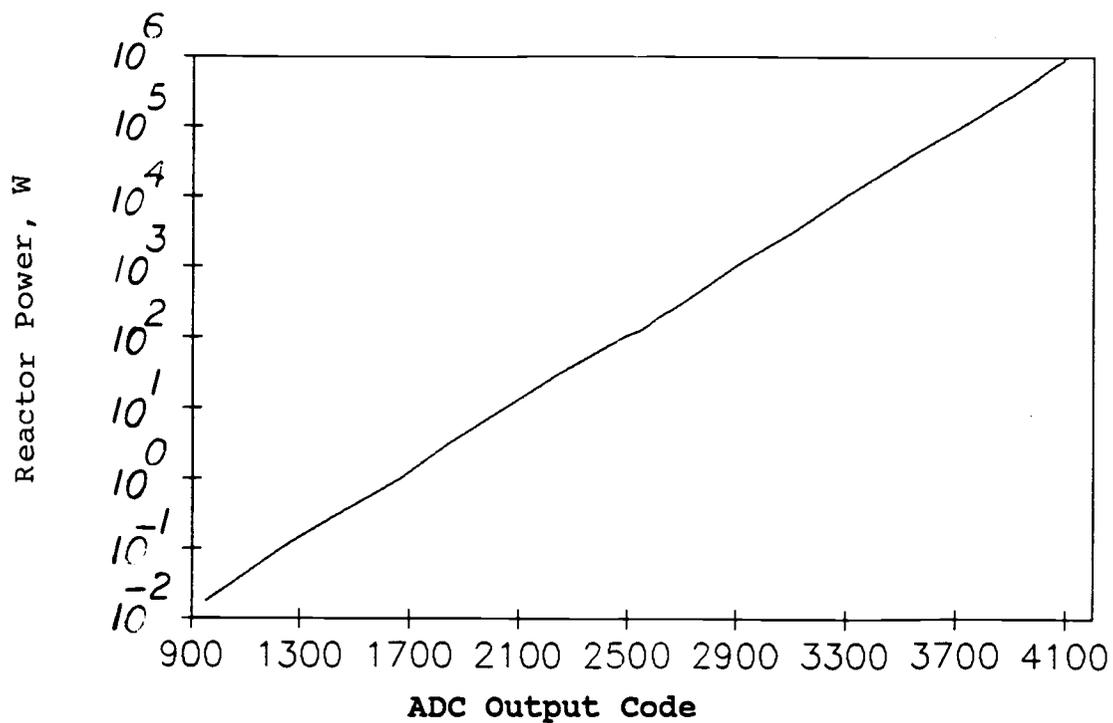


Figure V.1 ADC output code versus reactor power (whole range)

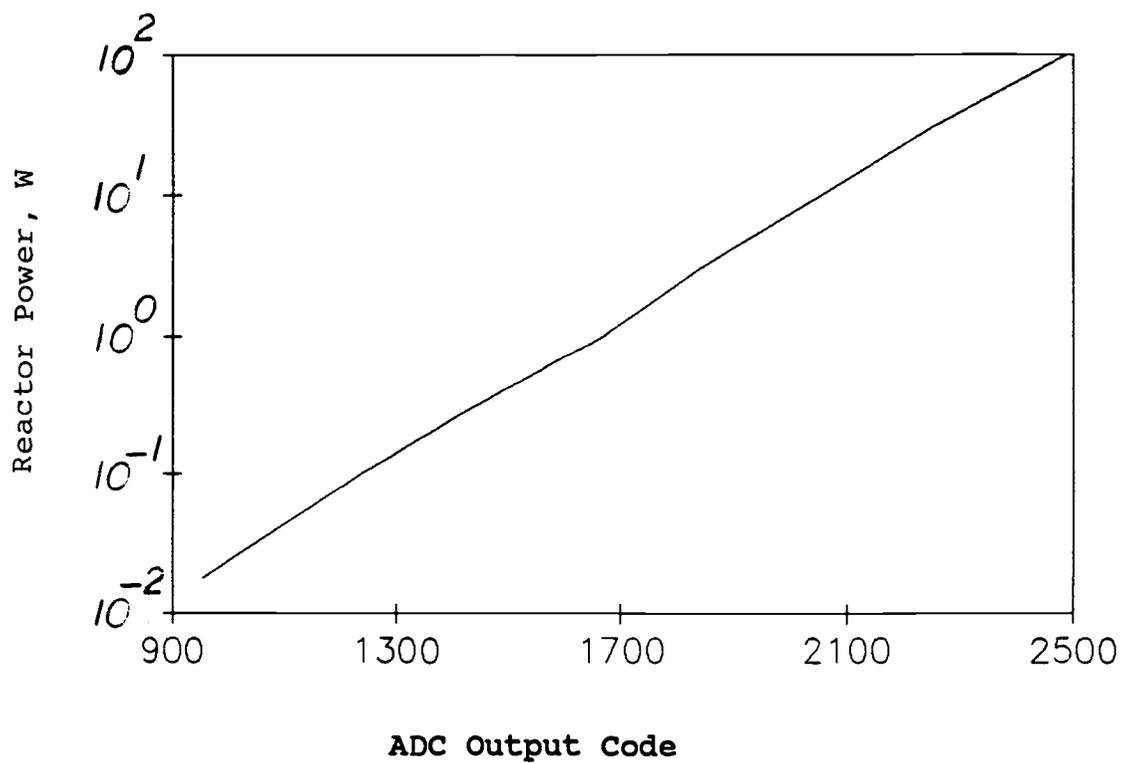


Figure V.2 ADC output code vs. reactor power from 0.01 to 100 W.

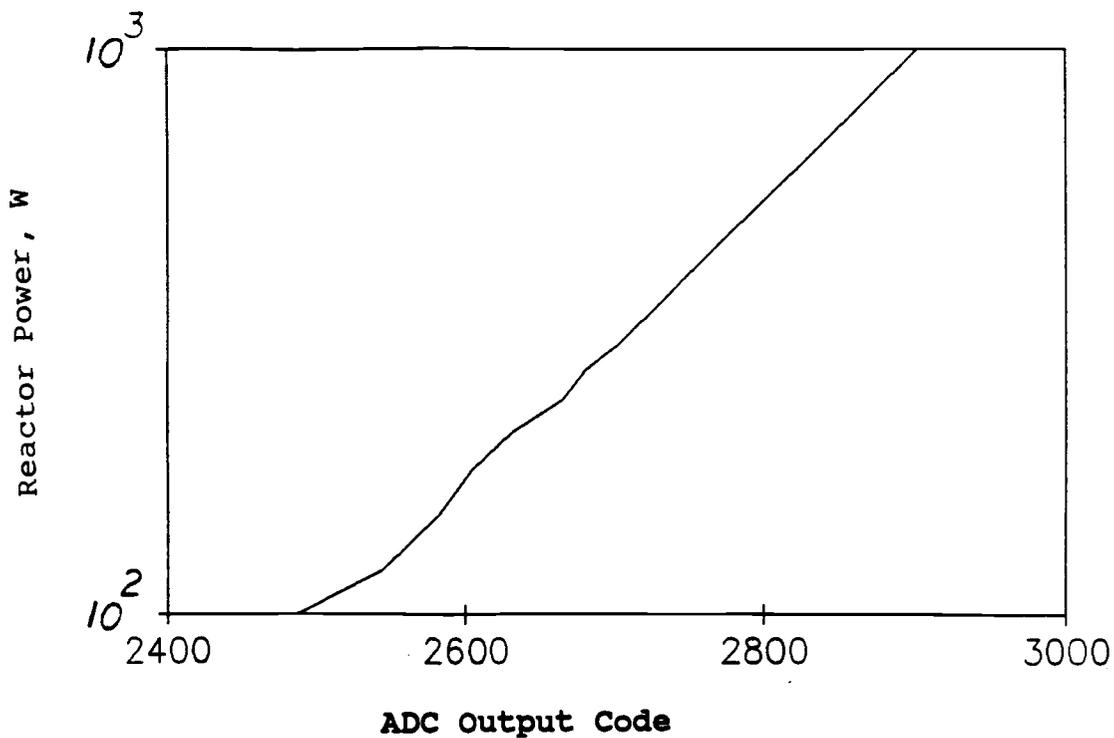


Figure V.3 ADC output code vs. reactor power from 100 to 1000 W.

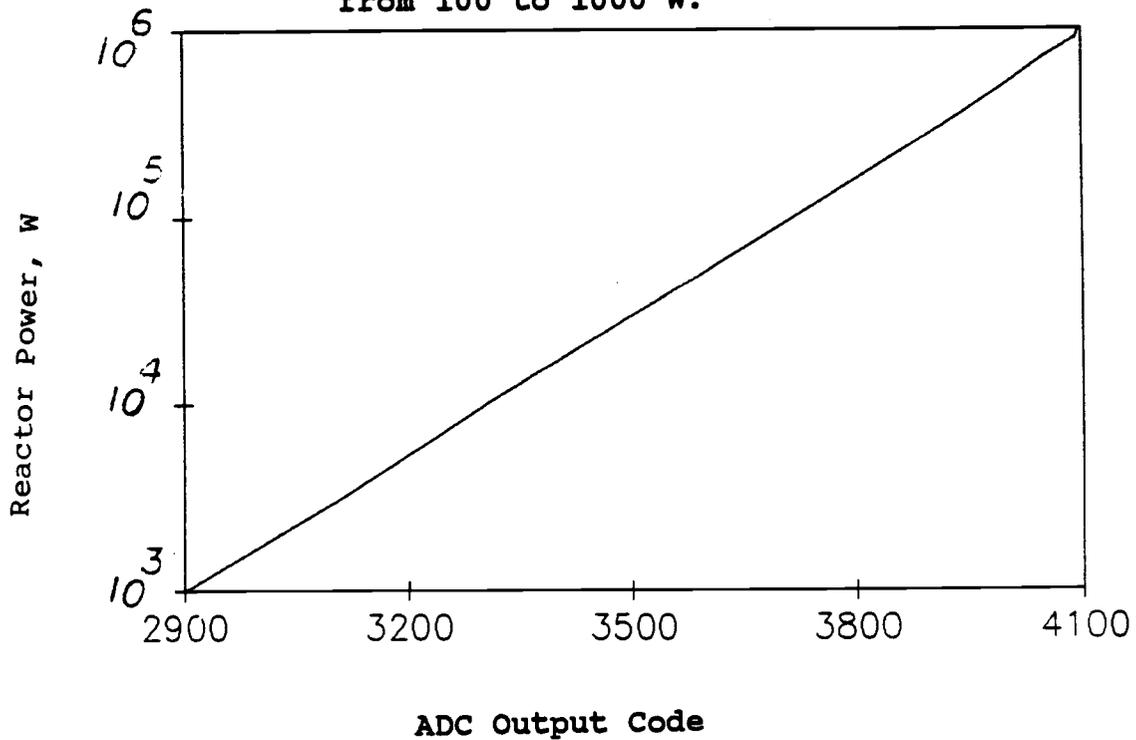


Figure V.4 ADC output code vs. reactor power from 1 kW to 1 MW

V.3 Step Reactivity Change Experiments

A. \$0.20 Step Change in Reactivity

Initially, the reactor power was held at 5 W with the control rods at the positions shown below:

<u>Transient</u>	<u>Safety</u>	<u>Shim</u>	<u>Regulating</u>
47 (in)	475	485	463

The transient rod was moved to position 162 to add (promptly) 20 cents of reactivity in the square wave mode. Power data were measured using a rate of 1.0 sample/s and a count of 100 samples to follow the power for 100 s.

$$\text{Total sampling time} = \frac{\text{Count}}{\text{Rate}} \quad (5.4)$$

Figure V.5 shows the power trace and Figure V.6 shows the corresponding reactivity as a function of time. To analyze this data, two power measurements were chosen at 40 s and 60 s intervals after the reactivity change to ensure that all negative roots had died out, and the power was increasing by the reactor period term and following the equation:

$$P_1 = P_0 e^{\Delta t/T} \quad (5.5)$$

where

P_0 = power at time t_0 , W

P_1 = power at time t_1 , W

Δt = time difference, s

T = reactor period, s

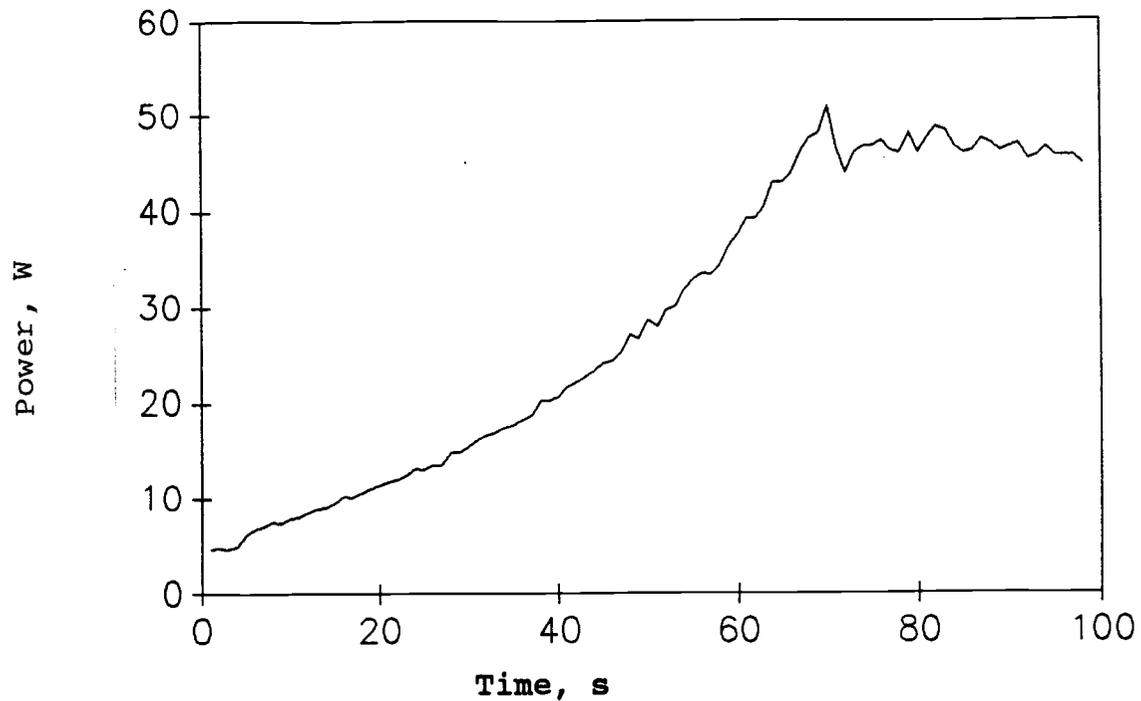


Figure V.5 Power trace following step reactivity change of $\$0.20$.

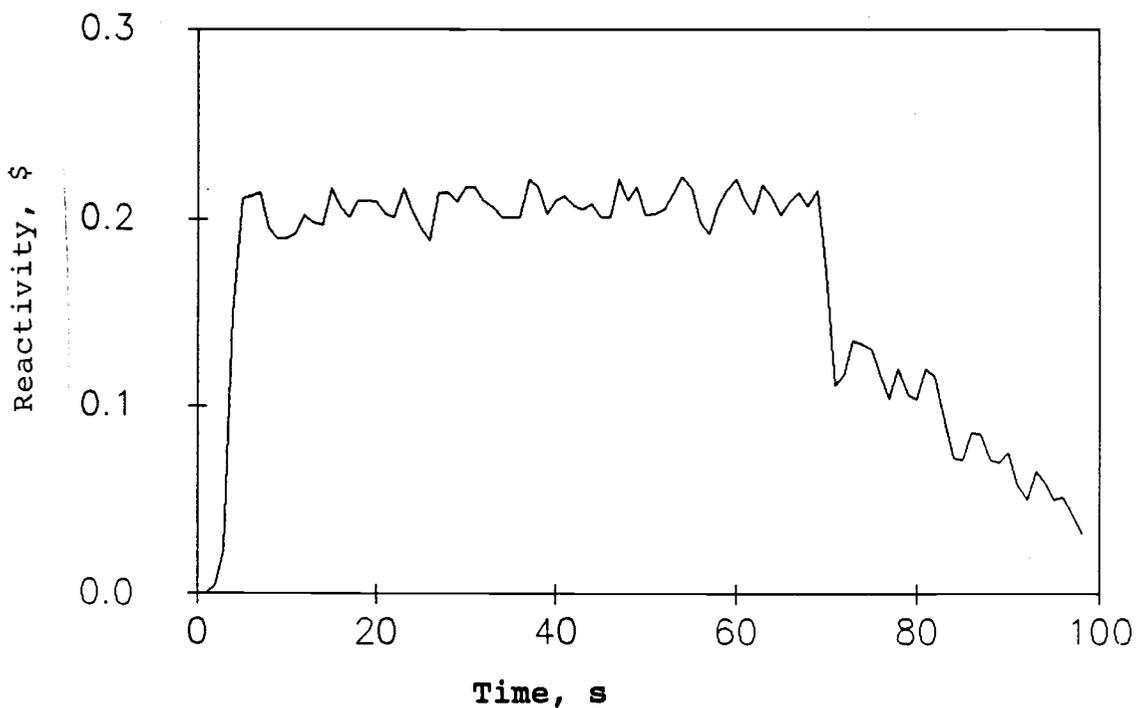


Figure V.6 Computed reactivity following step reactivity change of $\$0.20$.

For this situation,

$$P_0 = 20.55 \text{ W} \quad \text{at } t = 40 \text{ s}$$

$$P_1 = 37.57 \text{ W} \quad \text{at } t = 60 \text{ s}$$

$$\text{So } 37.57 = 20.55 e^{20/T}$$

Solving this equation for T yields a period of

$$T = 33.15 \text{ s}$$

Figure V.7, which represents the reactivity versus reactor period for the OSTR, gives a reactivity of about \$0.21 for a period of 33.158 s. This value of reactivity compares well with the predicted value from LOGGER (5% difference). This is evidence that the reactivities measured with the reactivity meter are consistent with theory.

For the reactivity measurement noise was observed. Signals usually contain noise, which is much wider in its bandwidth than the frequencies of the signals. A pre-sampling or "aliasing" filter prior to sampling can be used to limit the bandwidth of this noise. Also this experiment was performed without keeping the power constant for 1 minute prior to the measurement to permit establishment of the equilibrium precursor concentration.

B. \$0.70 Step Change in Reactivity

Another step reactivity experiment was performed using \$0.70. In this experiment the reactor power was first held at 5 W for 1 minute by setting the control rods

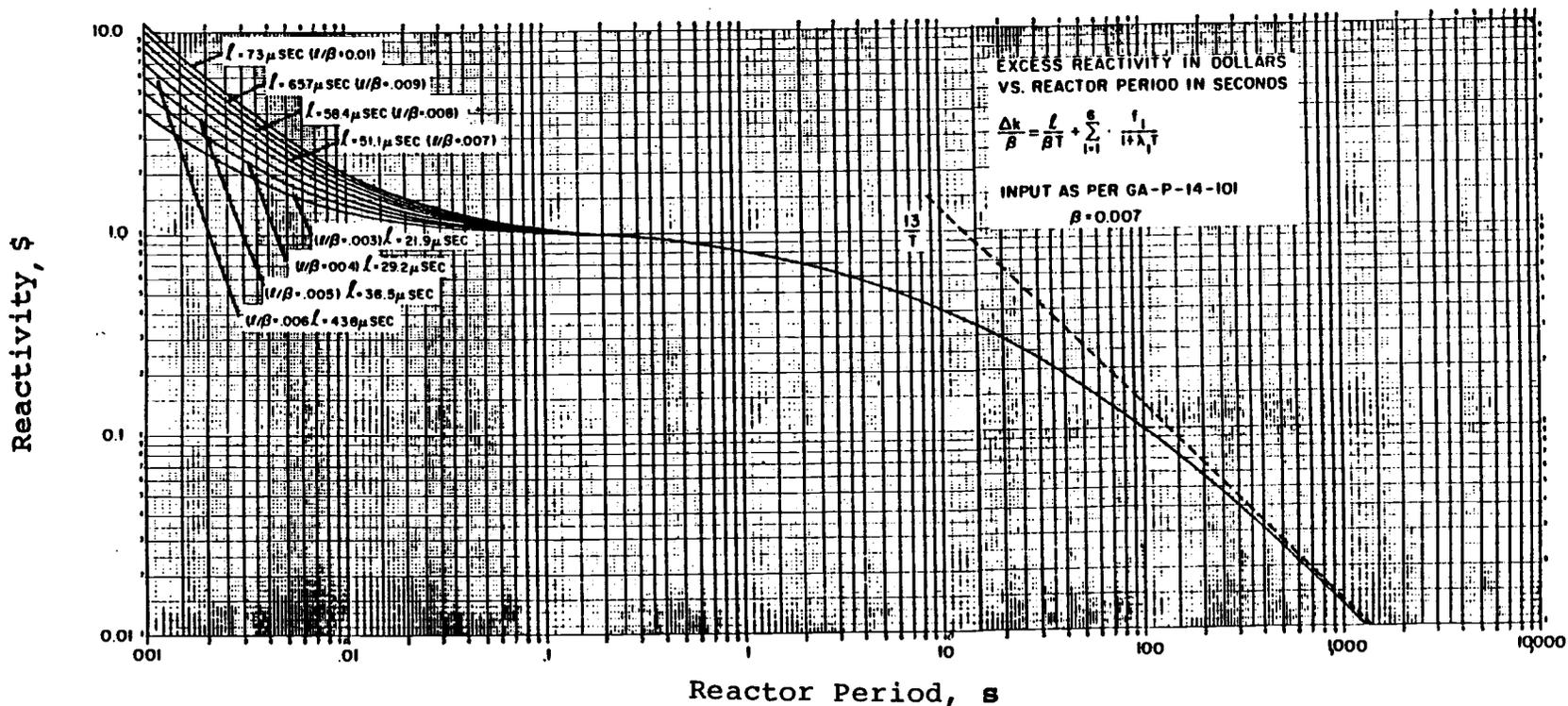


Figure V.7. Excess reactivity verses reactor period for OSTR.
 [Pg. 37 in (11)]

at the positions shown below:

<u>Transient</u>	<u>Safety</u>	<u>Shim</u>	<u>Regulating</u>
47 (in)	480	480	474

The transient rod was then moved to position 305 to add 70 cents of reactivity in the square wave mode. Power data were measured using a rate of 1 sample/s; a count of 600 samples was used to follow the power for 10 minutes (only the first 5 minutes of data is shown in Figures V.8 and V.9).

Figure V.8 shows the power trace and Figure V.9 shows the corresponding reactivity as a function of time. Table V.2 shows part of the output of the LOGGER program for this experiment.

An exponential least squares regression for this data was calculated to find an equation in the form

$$P = A \exp (Bt)$$

where

P = Reactor power, W

t = Time, s, and

A and B are constants to be determined.

B represents the reciprocal of the reactor period. The value of B was found to be 0.51768 s^{-1} , and the correlation coefficient for this curve was found to be 0.9964, so T, the reactor period, can be calculated as

$$T = \frac{1}{B} = \frac{1}{0.51768} = 1.93 \text{ s}$$

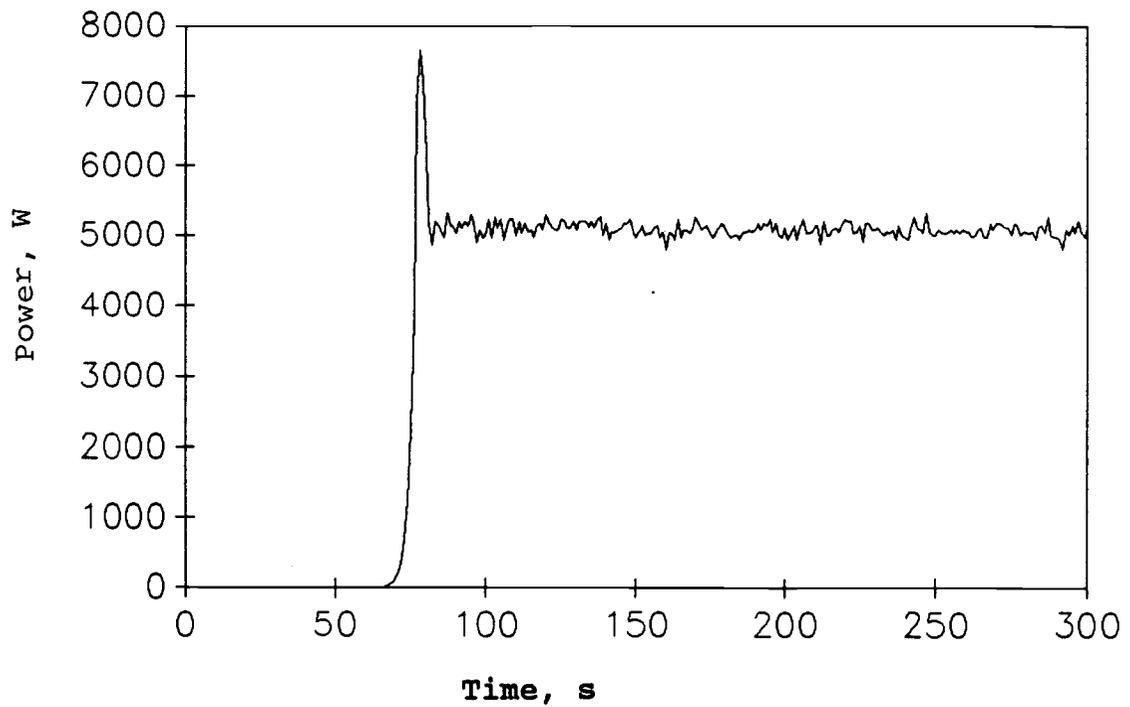


Figure V.8 Power trace following step reactivity change of \$0.70.

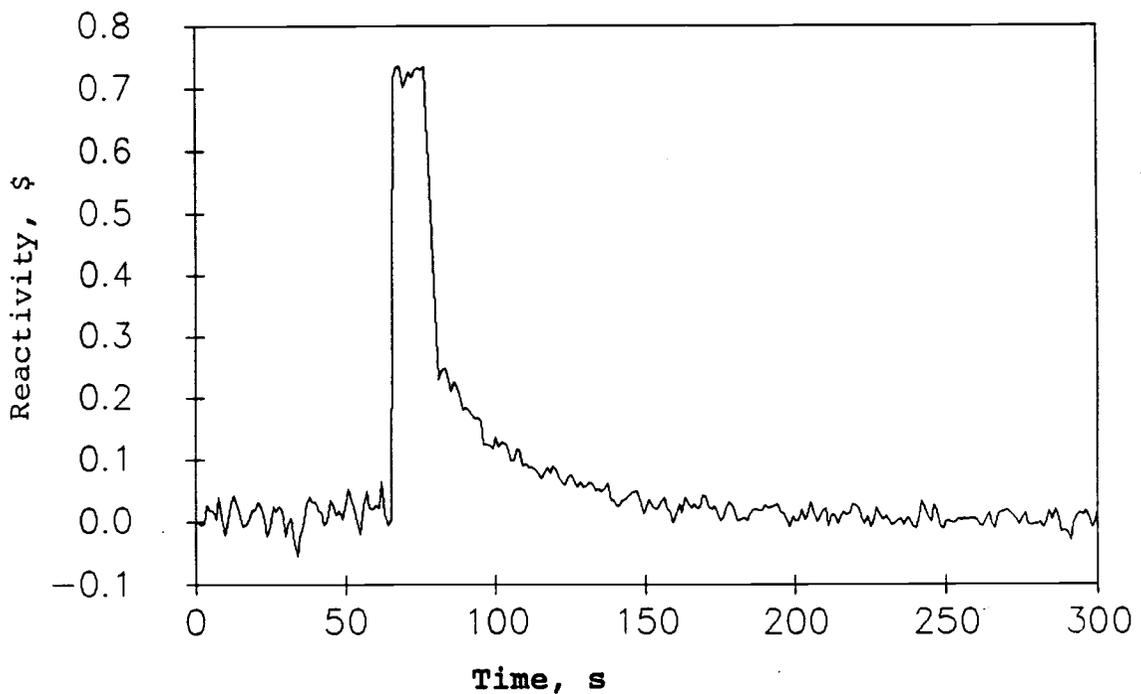


Figure V.9 Computed reactivity following step reactivity change of \$0.70.

From Figure V.7, this period gives a reactivity of \$0.7, which is consistent with the magnitude of the step change in reactivity for this experiment.

Table V.2

Part of the Output of LOGGER for + \$0.70 Pulse

<u>Time (s)</u>	<u>Power (W)</u>	<u>Reactivity (\$)</u>
69	92.65	0.70
70	137.79	0.71
71	236.9	0.72
72	425.32	0.71
73	701.81	0.72

V.4 Safety Rod Calibration Curve

A calibration curve for the safety control rod of the OSTR was generated from separate measurements, each consisting of reactivity determinations using the program LOGGER, and using the maximum reactivity measured for each segment of the rod.

The value of the sampling rate was 1 sample/s, and the power was measured for 150 samples. The power was maintained for the first 60 s at 5 W. Table V.3 shows the output worth for each segment of the safety rod as calculated by the LOGGER program.

The total rod worth at each position was determined by adding the worth of that segment to the sum of the reactivity worths of all the previous segments. Table V.4 shows the reactivity meter measurements and those from the reactivity curve for the safety rod which had been previously measured by the OSTR reactor operations staff using the period method with the inhour equation and times measured with stop watches. A comparison of these measurements shows a good agreement, especially for the lower position of the rod. Figure V.10 shows the two measurements. The reactivity meter gave a total rod worth that was about 5% less than that measured by the period method.

Table V.3

Reactivity Worth for Different Segments of the Safety Rod

<u>Start Position (units)</u>	<u>End Position (units)</u>	<u>Reactivity Worth (cents)</u>
100	235	34.16
235	317	35.91
317	386	35.25
386	450	34.72
450	525	36.72
525	615	36.25
615	828	36.36

Table V.4

Comparison Between Reactivity Meter Measurement and
 Period Method Measurement of the Reactivity
 Worth for the Safety Rod

<u>Rod Position (units)</u>	<u>Reactivity Meter (\$)</u>	<u>Period Method (\$)</u>
100 (in)	0.00	0.00
235	0.34	0.36
317	0.70	0.70
386	1.05	1.08
450	1.40	1.44
525	1.77	1.80
615	2.13	2.20
825 (out)	2.49	2.61

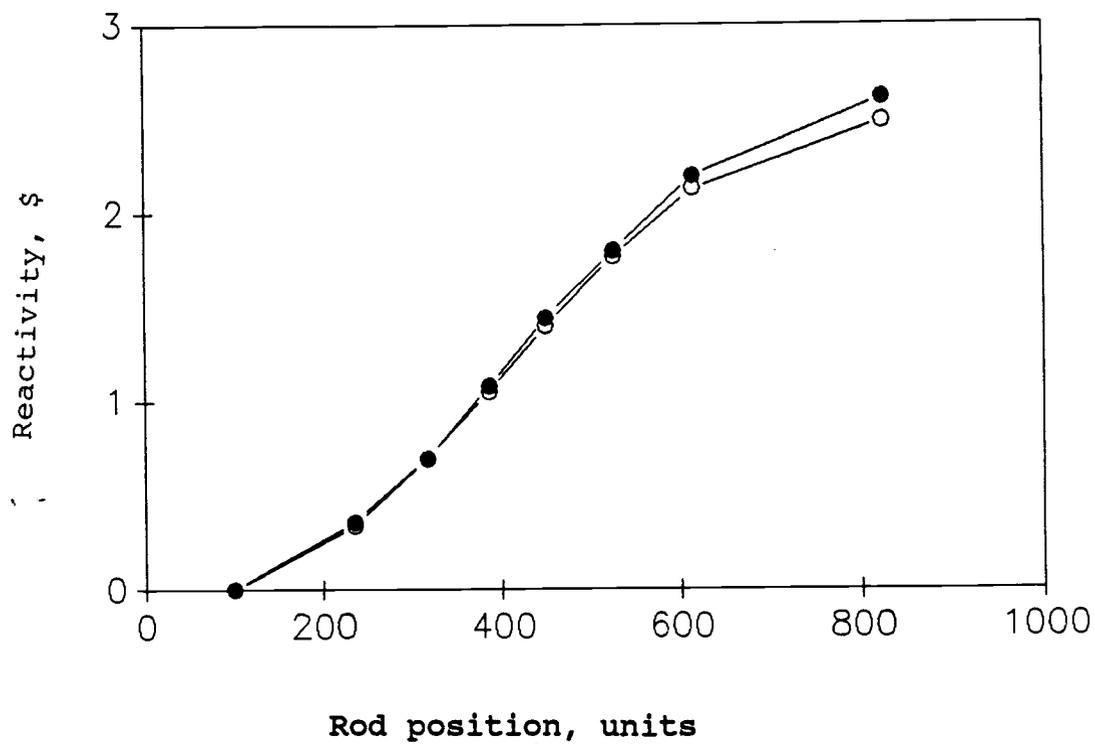


Figure V.10 Comparison of safety rod calibration curves

○ — ○ Reactivity Meter
● — ● Period Method

V.5 Positive Reactivity Prompt Jump

Another experiment was completed to show the performance of the reactivity program LOGGER during a step change in reactivity. In this experiment the reactor power was held constant at 11 W and the transient rod was moved rapidly to position 230 to add about 40 cents of reactivity (in the square wave mode). Power data were measured using a rate of 10 samples/s, and a count of 600 samples was used to follow the reactor power for 1 minute. The prompt power jump can be calculated using equation (2.20):

$$P_1 \approx P_0 \frac{\beta}{\beta - \rho}$$

where

- P_0 = power before the reactivity jump
- P_1 = power after the reactivity jump
- β = total delayed neutron fraction
- ρ = reactivity insertion

For this experiment,

$$P_1 = \frac{11}{(1-0.4)} = 18.3 \text{ W}$$

Following this jump, the power rises exponentially by the reactor period. Figures V.11 and V.12 show the reactor power trace for this experiment and the associated reactivity, respectively.

The average reactivity and standard deviation for $10 \text{ s} < t < 40 \text{ s}$ were calculated. These values were found

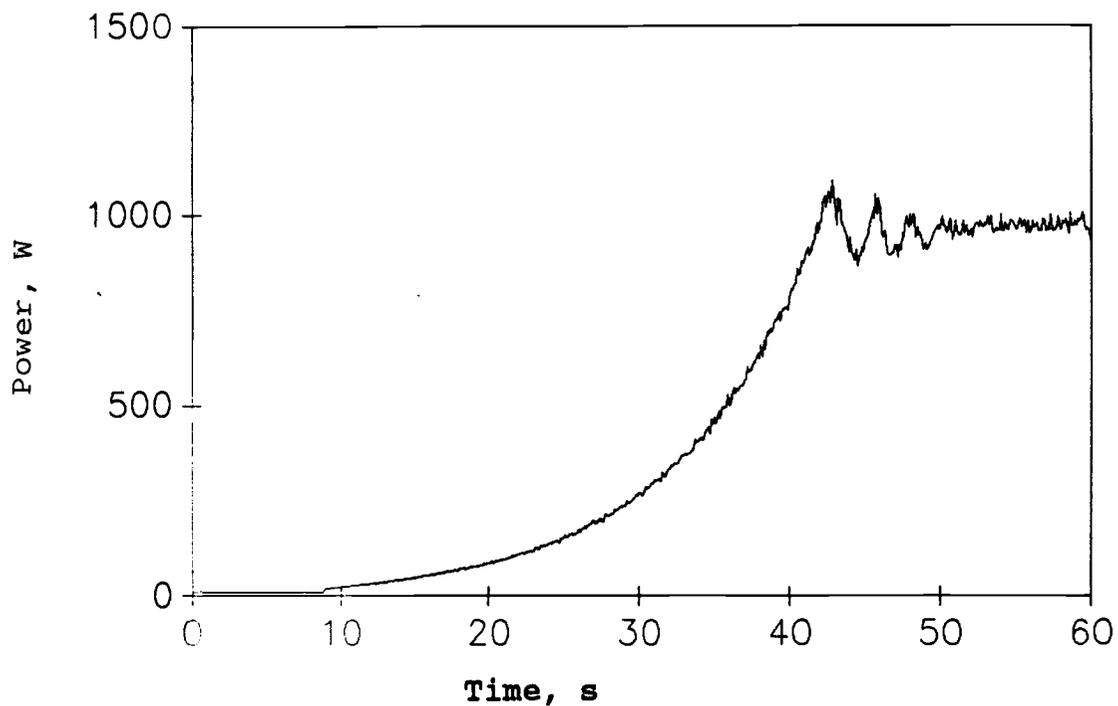


Figure V.11 Power trace following a + \$0.40 step reactivity insertion.

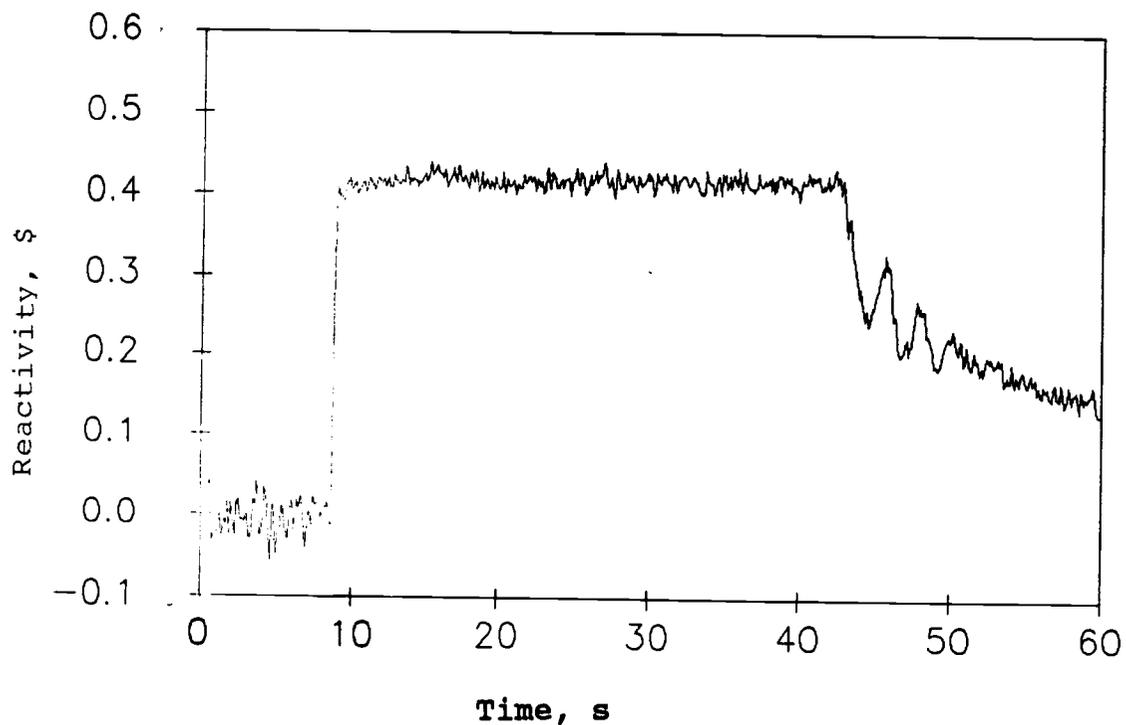


Figure V.12 Associated reactivity jump for a + \$0.40 step insertion.

to be $\beta = 0.42$ and $\beta = 0.008$, respectively. The reactivity value ($\beta = 0.42$) gives a difference of about 5% from that determined by the reactor staff by the period method ($\beta = 0.40$).

Figure V.13 shows the reactor power $P(t)$ as a function of time on an expanded scale after the change in the reactivity. It is noted that the power rises very suddenly at first, and then more slowly. This sudden rise is caused by the rapid decay of the exponential functions with the largest arguments in the reactivity equation [i.e., equation (2.10)]. This value agrees well with the calculated value (18.3 W).

Some oscillations appear on the power and reactivity curves after 40 s. These oscillations occur due to the underdamped response of the automatic control rod mechanism.

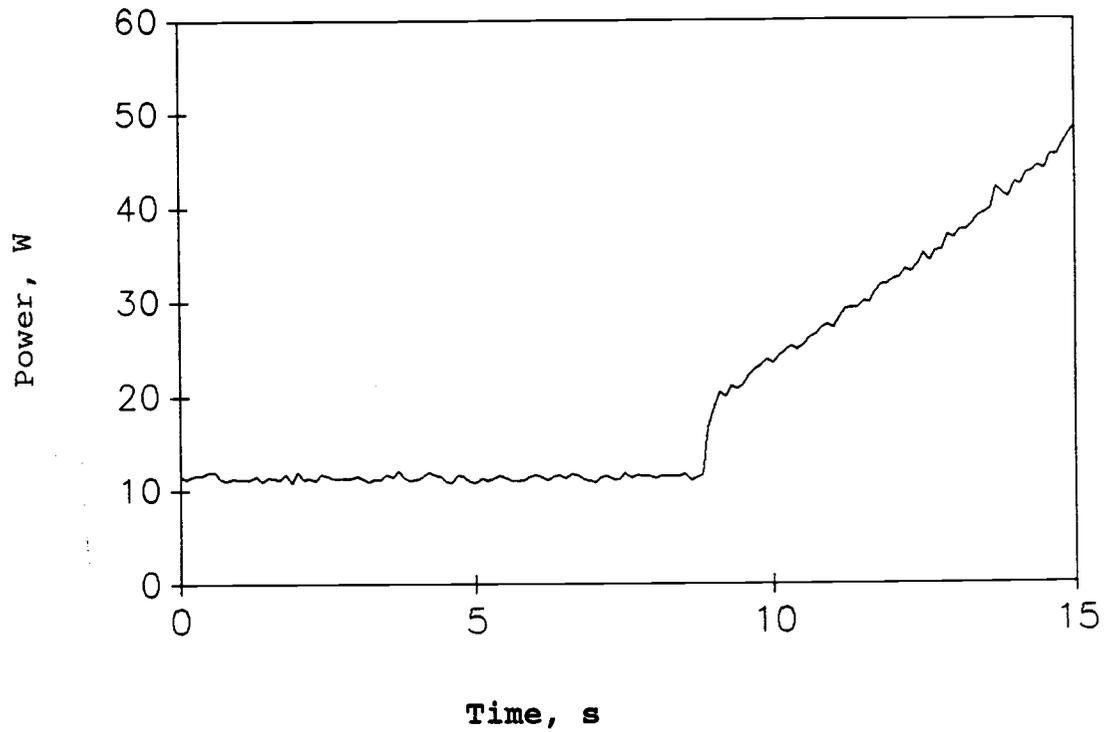


Figure V.13. Time behavior of reactor power following step reactivity insertion of + $\rho_{0.40}$ on an expanded scale.

V.6 Negative Reactivity Prompt Drop

At the conclusion of the previous experiment, the reactor power was 1 kW. A -40 cent reactivity was then inserted using the transient rod (the transient rod was scrambled). Figures V.14 and V.15 show, respectively, the power trace and the associated reactivity as a function of time. As for the previous experiment, this experiment used a rate of 10 samples/s and a count of 600 samples to follow the reactor power for 1 minute. Equation (2.20) was used to determine the prompt power drop,

$$\begin{aligned} P_1 &= 1000 \frac{1}{(1 + 0.4)} \\ &= 714 \text{ W} \end{aligned}$$

This value agrees well with power measured by LOGGER at the end of the prompt drop, as shown in Figure V.14.

The average reactivity and standard deviation for $10 \text{ s} < t < 60 \text{ s}$ were calculated; these values were found equal to - \$0.43 and \$0.0189, respectively. The reactivity value (- \$0.43) gives a difference of about 7% from that determined by the reactor staff by the period method (- \$0.40).

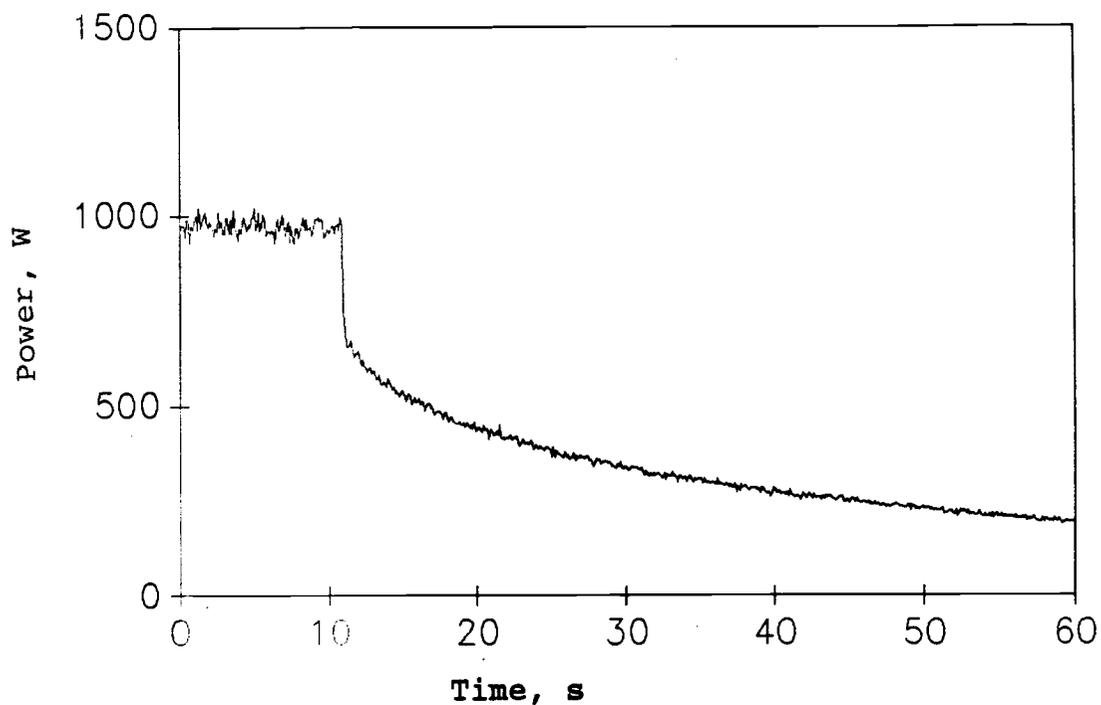


Figure V.14 Power trace following a - \$0.40 step reactivity insertion.

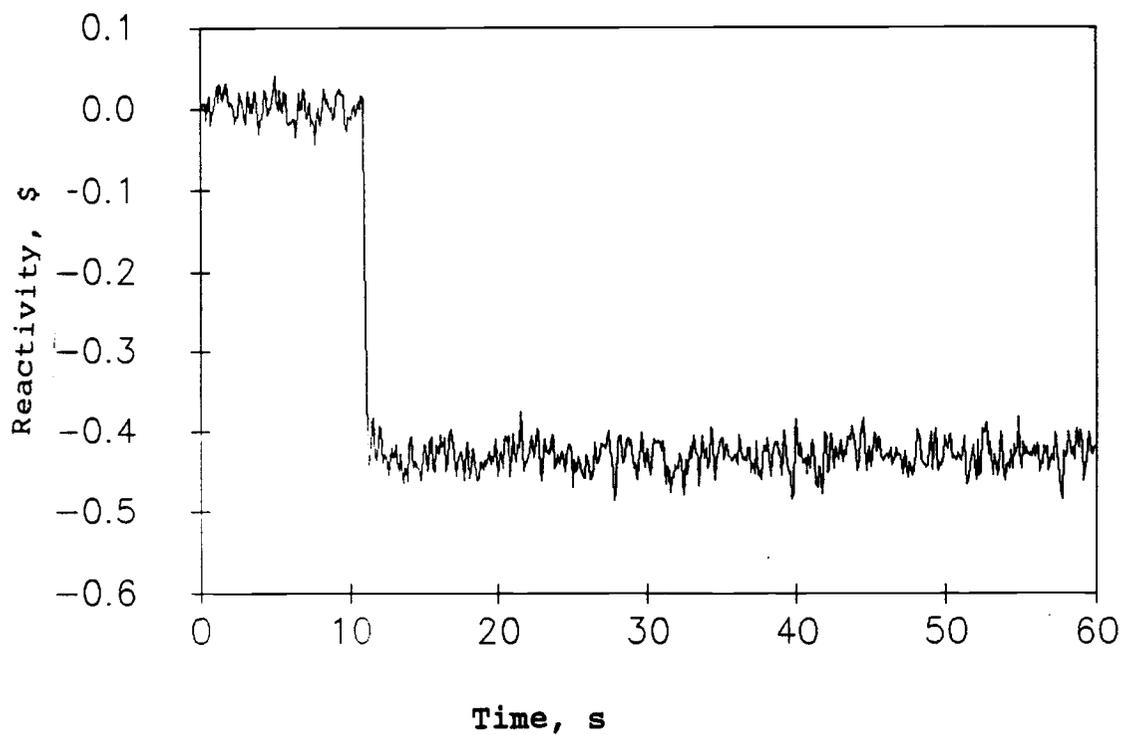


Figure V.15 Associated reactivity drop for a - \$0.40 step insertion.

CHAPTER VI

CONCLUSIONS AND RECOMMENDATIONS

A method based on the point reactor kinetics equations has been tested for the computation of the reactivity for the OSTR. An IBM Data Acquisition and Control Adapter inserted into a Leading Edge personal computer measured a voltage output signal of the wide range log-channel amplifier of the OSTR. This signal is proportional to the logarithm of the reactor power. The DACA produces a code that is linearly proportional to the input voltage. Ten volts corresponds to a code of 4095. A correlation between the ADC output code and the OSTR power was verified. The reactor power was sampled at different rates, and from the reactor power history the reactivity was determined and displayed on a digital read-out panel. The reactivity program was developed from a code originally written by R. T. Tuttle.⁵

Many measurements were performed using this technique which verified the performance of the equipment with especially good agreement for the control rod calibration curve.

Recommendations for Future Work

1. Since the reactivity calculation depends on the power measurement, the more exact the

power measurement, the more accurate is the calculation of the reactivity. Since the fission chamber output can drift somewhat, the power equations [i.e., equations (5.1), (5.2), and (5.3)] should be recalibrated from time to time.

2. The other input channels of the DACA can be used to measure other parameters of the OSTR, such as the fuel element temperature, bulk water temperature, and peak power in pulse mode. These measurements can be used to determine some of reactor specifications, such as the moderator temperature coefficient and the fuel temperature coefficient.

REFERENCES

1. James J. Duderstadt and Louis J. Hamilton, Nuclear Reactor Analysis, John Wiley & Sons, New York (1976).
2. John R. Lamarsh, Introduction to Nuclear Reactor Theory, second Edition, Addison Wesley, Reading, Mass. (1983).
3. Samuel Glasstone, and Alexander Sesonske, Nuclear Reactor Engineering, Third Edition, Van Nostrand, New York (1980).
4. Operator's Guide for the Model "D", Leading Edge, Leading Edge Hardware Products, Inc., Third Edition, Canton, MA, July (1986).
5. R. J. Tuttle, "Elimination of the Effect of a Constant Neutron Source in Reactivity Measurement", Atomics International, NAA-SR-MEMO-12290 REV., Canoga Park, California, (1967).
6. Arnold H. Vandoren, Data Acquisition Systems, Reston Publishing, Reston, Virginia, (1982).
7. IBM Personal Computer Data Acquisition and Control Adapter Technical Reference, IBM, March 18 (1986).
8. T. V. Anderson, ed., Oregon State TRIGA Reactor Manual, Third Edition, Vol. 2 (1986).
9. IBM Personal Computer Data Acquisition and Control Adapter Programming Support, IBM (1984).
10. John R. Lamarsh, Introduction to Nuclear Reactor Theory, Addison Wesley, Reading, Mass., (1972).
11. T. V. Anderson, B. Dodd, and J. C. Ringle, eds., Oregon State TRIGA Reactor Manual, Third Edition, Vol. 3., June (1986).
12. J. M. Harrer, R. E. Boyar, and Darwin Krucoff, "Measurement of CP-2 Reactor Transfer Function", Argonne National Lab., ANL-4373, Jan. 1 (1950), 26 p.

APPENDICES

APPENDIX A**FISSION CHAMBER DETECTOR AND AMPLIFIER**

reuter **RS** stokes

RS-C3-2510-114 Fission Counter/ Chamber

**For
Reactor Control
(Wide Range)**

The RS-C3-2510-114 has proven itself as the standard high-sensitivity fission counter/chamber for wide range reactor instrumentation.

It is designed for measurement of the neutron flux levels from shutdown to full power of nuclear reactor. The detector can be used to detect individual neutrons (counting mode) to 10^8 nv in the presence of an incident gamma flux of 10^6 R/hr.

It can also be used as a wide-range neutron sensor in conjunction with mean-square-voltage (MSV) type circuitry over a range of 10^4 to 10^{10} nv in the presence of an incident gamma flux of 10^6 R/hr.

Operation, as specified here, is greatly dependent on associated electronics. All data presented here is based on measurement using a wide band pre-amplifier such as the model PA-5 manufactured by General Atomic.

Concentric cylinders with uranium coatings provide the neutron sensitive area. Aluminum alloy is used in construction to minimize neutron absorption and residual activity. All seals are directly bonded ceramic to metal. Insulators are high-purity alumina ceramic and are designed to assure stable, long-term noise-free operation of the chambers even at elevated temperature.

This chamber meets the U.S. Specification RDT C15-1T "Fission Type Neutron Detector Assembly" which is part of LMFBR instrumentation development. It can be supplied to the RDT specification which includes integral cable detector housing and cable seals for minimum interference from external noise.

Electrical & Nuclear

Operating range nv (with wide range channels)	1 to 2×10^{10} nv
DC sensitivity	1.4×10^{13} amps/nv
Counting sensitivity (typical)	.5 cps/nv
Gamma sensitivity*	2×10^{-11} amps/R/hr
Alpha Current	3×10^{-9} amps
Operating Voltage Range	300 - 800 V
Operating Voltage Max	1000 V
Maximum operating temperature	125°C
Burn up life	10% for 10^{19} nvt
Expected Radiation Life	$> 10^{10}$ rads

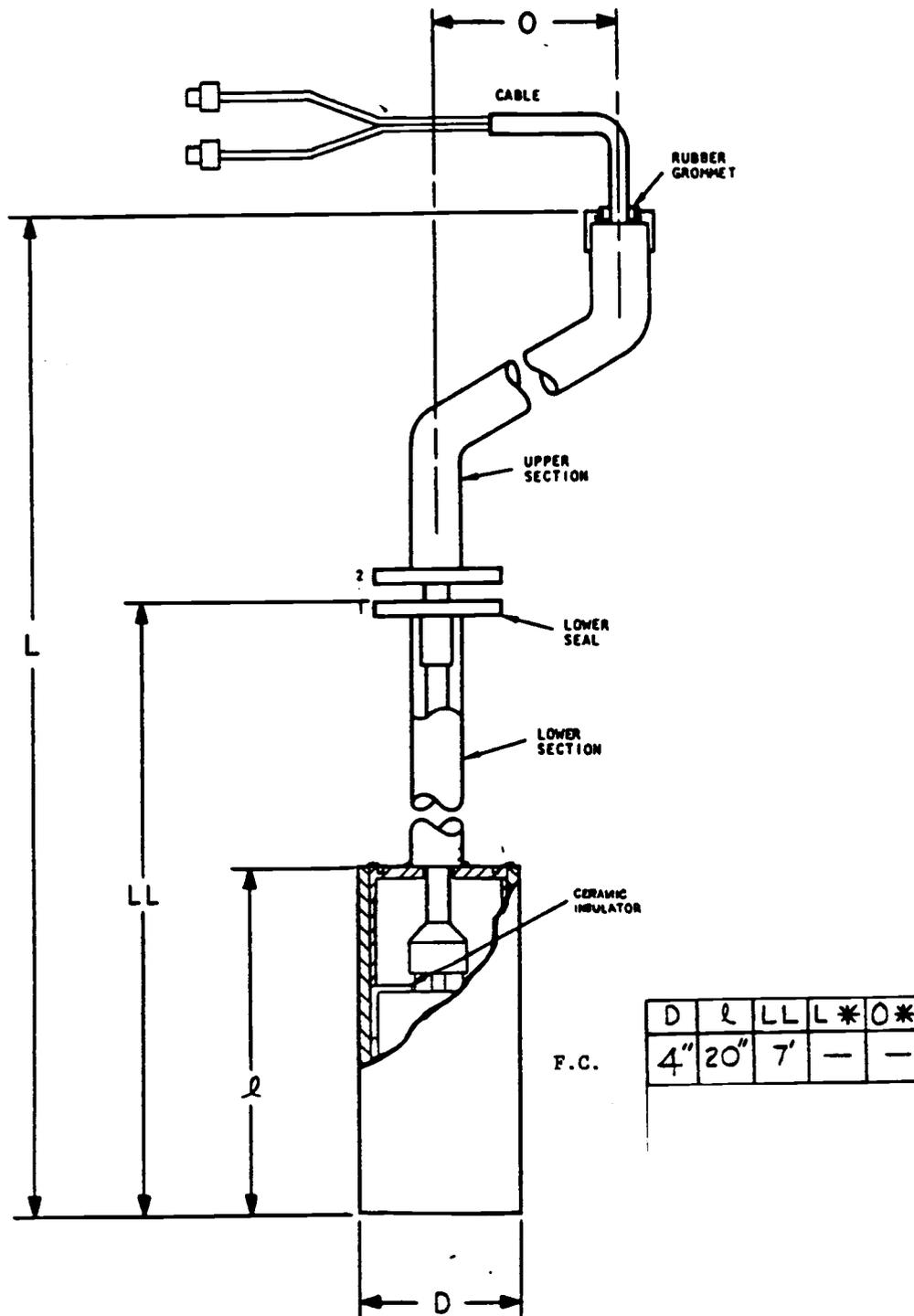
Cable Assembly

Expected Radiation Life (integral cable)	$> 10^{10}$ rads
Expected Radiation Life (non-integral cable)	$> 3 \times 10^9$ rads
Connector type	HN

NOTE 1: With other electrode grounded.

*d-c operation only; in mean square or counting applications gamma has negligible effect up to 10^6 R/hr for any neutron flux level.

Detector assembly



* TO BE DETERMINED BY CUSTOMER

GENERAL ATOMIC COMPANY

Wide Range Log Power Channel
Model NLW-2

- 10 decades with period
- in 10^6 R/hr γ background
- $\pm 1.5\%$ linearity
- all solid state circuitry

The wide range log power channel, Model NLW-2 measures 10 decades of neutron flux in a gamma background of 10^6 R/hr. It operates from a single fission counter, and the full 10-decade range is read out on a single meter or recorder. Using the ac signal from a fission chamber, the channel combines a pulse log-count rate technique for the lower 6 decades with a log ac (Campbell) technique for the upper 4 decades to produce a single output signal for the total range of 10 decades. Both techniques are affected very little by high gamma background, and the method used for combining them eliminates errors due to gamma and alpha background as well as the resolution counting loss error normally associated with high rate counting.

The circuitry is all solid state modular construction; and since no mechanical switching or combining techniques are used, the reliability is high and the response time is adequate for any power reactor transient. Test and calibration circuits are provided which feed

appropriate signals into the input of the preamplifier for checking six calibration levels. This method checks and calibrates all of the electronics (including the preamplifier), and a positive test for chamber and cable integrity based on alpha background is also provided.

Provision is also included for making an entire channel realignment in the field without any special test equipment or reactor operation required.

The wide range channel is normally provided as shown with a special low noise high voltage chamber supply, period or rate meter, and appropriate bistable trips mounted in a conventional nuclear drawer. Since the wide range log channel utilizes only the ac component of the fission chamber current, the d-c component, which is available at the back of the drawer, may be used with the Gulf Electronic Systems' linear power range channel to provide a linear power level indication and level trips within the normal range of the fission chamber. A conventional fission counter/chamber, such as the RSN 314, WL 7657, or GE NA04, is recommended for use with this channel. Up to 50 feet of triaxial cable or standard coaxial cable with an extra metal shield (e.g., aluminum conduit or aluminum pipe) may be used between the chamber and the preamplifier. Conventional coaxial cable of any required length may be used between the preamplifier and the wide range channel drawer located in the control room.

The all solid state preamplifier, PA-5, is sealed to prevent moisture penetration and is mounted in a rugged aluminum housing. It is double shielded so that the housing with mounting bracket can be bolted to the grounded reactor structure; the inner shield is isolated and connected to the instrument ground. The preamplifier is powered from the channel power supply through low voltage cable which also provides a d-c chamber current return lead. The preamplifier consists of a high gain feedback broad band amplifier followed by a low output impedance cable driver. The input impedance of the preamplifier is quite low so that leakage resistance, chamber capacitance, and cable capacitance have little effect.

High voltage for the detector is provided within the drawer by the HV-6 high voltage power supply, which is adjustable from 500V to 1,000V and includes a front panel meter* and loss of voltage annunciation. The HV-6 specifications are included in this section.

Buffer amplifier can be provided for remote output signals to insure safety action to the channel, even though the outputs are accidentally grounded or shorted to the 110-volt a-c line, or to the HV power supply (up to 1,000 volts).

*High voltage meter optional equipment. If is not required because adequate operating voltage is monitored automatically by the loss of voltage annunciate.

WIDE RANGE LOG CHANNEL
PERFORMANCE SPECIFICATIONS

RANGE: 10 decades log scale
2-6 nv* to 2×10^{10} nv in a gamma background of 10^6 R/hr. when used with a standard fission chamber

LINEARITY: $\pm 1.5\%$ of equivalent linear full scale over the temperature range 20°C to 30°C and $\pm 10\%$ line voltage variation

$\pm 3\%$ of equivalent linear full scale over the temperature range 10°C to 55°C and $\pm 10\%$ line voltage variation

APPROXIMATE AVERAGE RESPONSE TIME:

10^7 nv - 10^{10} nv	15 milliseconds
10^6 nv - 10^7 nv	20 milliseconds
10^4 nv - 10^6 nv	40 milliseconds
10^2 nv - 10^4 nv	500 milliseconds
10 nv - 10^2 nv	2 seconds
< 10 nv	20 seconds

OUTPUT: 0-10 volts dc full scale
0-1 ma (for remote meter)
0-100 mv (for recorder)

POWER: 115 V $\pm 10\%$ AT 1 AMP, 60 Hz or
 ± 15 vdc (regulated) at 1.3 amp
 $- 15$ v at 0.28 amp

*Depending upon gamma flux and electrical background noise.

APPENDIX B

**IBM PERSONAL COMPUTER
DATA ACQUISITION AND
CONTROL ADAPTER**

This appendix contains the specifications of the analog input and binary output devices. For other device specifications, refer to IBM Personal Computer Data Acquisition and Control Adapter Programming Support. Also, this appendix contains some information about the software and hardware used in this thesis.

FORTRAN FUNCTION LIST

This section contains information on the following functions:

AINM	Analog Input Multiple
AINS	Analog Input Simple
BITOUS	Binary Bit Output Simple
BOUS	Binary Output Simple
DELAY	Delay Execution

Analog Input Device

The analog input device has the following characteristics:

Resolution	12 bits
Input Channels	4 differential
Input Ranges	Switch-selectable ranges: 0 to +10 volts (unipolar), -5 to +5 volts (bipolar), and -10 to +10 volts (bipolar).
Input Resistance	100 megohms minimum
Input Capacitance	200 picofarads maximum; measured at the distribution panel connector
Input Leakage Current	± 300 nanoamperes maximum
Input Current	± 4 milliamperes at maximum input voltage
Digital Coding	Unipolar: binary. Bipolar: offset binary.
Safe Input Voltage	± 30 volts maximum (power On or Off)
Power Supply Rejection	$\pm 1/2$ LSB maximum change full scale calibration
Integral Linearity Error	± 1 LSB maximum

Differential Linearity Error	$\pm 1/2$ LSB maximum
Differential Linearity Stability	± 5 ppm/ $^{\circ}$ C maximum; guaranteed monotonic
Gain Error	$\pm 0.1\%$ maximum between ranges. Any range adjustable to zero.
Gain Stability	± 32 ppm/ $^{\circ}$ C of FSR maximum
Common-Mode Input Range	± 11 volts maximum
Common-Mode Rejection	72 dB minimum ratio (signal within common-mode range)
Unipolar Offset Error	Adjustable to zero
Unipolar Offset Stability	± 24 ppm/ $^{\circ}$ C of FSR maximum
Bipolar Offset Error	Adjustable to zero
Bipolar Offset Stability	± 24 ppm/ $^{\circ}$ C of FSR maximum

Settling Time	For channel acquisition: 20 microseconds maximum to $\pm 0.1\%$ of the input value
Conversion Time	35 microseconds maximum
Throughput to Memory	15,000 conversions per second, minimum
'A/D convert enable'	
Input Impedance	One LS TTL load plus 10-kilohm pull-up resistor
'A/D convert out'	
Fanout	10 LS TTL loads or 2 standard TTL loads

Binary Output (BO0 through BO15)

Fanout 28 LS TTL loads or 7 standard
TTL loads

Throughput from Memory 25,000 operations per second,
minimum

'BO Gate'

Input Impedance Two LS TTL loads plus one
10-kilohm pull-up resistor

BO CTS

Input Impedance One LS TTL load plus
10-kilohm pull-up resistor

'BO Strobe'

Fanout 10 LS TTL loads or 2 standard
TTL loads

Programming with FORTRAN

The FORTRAN bindings are supplied as an object module, DACF.OBJ and DACPF.OBJ. Include the module in the object modules list that the linker requires to make functions accessible to your FORTRAN program.

Editing, Compiling, and Linking

You can create source code for programs in compiled languages by using EDLIN or any other ASCII editor. Call functions just like any other external subroutine. You must observe the variable-declaration, parameter-passing, and array dimensioning conventions of the language.

After compiling the source code, link the resulting object modules with the proper object modules and libraries to form an executable (.EXE) file. Enter the correct one in response to the linker's prompt:

DACF.OBJ

for FORTRAN Version 2.00 and

DACPF.OBJ

for Professional FORTRAN.

Once the DAC.COM is loaded, the .EXE files execute in the normal way.

See the IBM Personal Computer *FORTRAN Compiler Version 2.00* or *Professional FORTRAN* for more information on compiling and linking your programs.

Arguments Are...

Every function requires at least one argument; most require several. The argument list determines:

- Which I/O device the function accesses
- On which adapter it is located
- Channel numbers
- The number of samples to read or write
- The variable or array that receives returning input data or sends output data
- The variable that receives the returning execution status.

Most arguments take the form of either integer variables or 2-byte unsigned integers. The language you are using may place other constraints on values or variables. These are explained in the "Remarks" section for each argument.

Types of Arguments

Arguments appear in an argument list following each function. Their purpose determines their place in the list. Not all arguments appear in the argument list of every function; however, the order of the arguments never changes. This order, divided into the following groups, is as follows:

- Adapter, device, and channel numbers. These tell the function which adapter to call. Further, they identify the specific device within that adapter, and the specific channel of that device, if applicable.
- Execution parameters. These supply additional information on execution and data storage.
- Count and rate. These tell iterative functions how many iterations to perform and how fast to perform them.
- Data variable. This is the variable to which an input function writes data, or from which an output function retrieves data.

Note: In all languages other than C, iterative I/O functions require the data variable to be the first element of a data array.

- Status variable. This is the variable to which the status of the function returns. It indicates the success or type of failure of the function.

When assigning values to arguments (integer arguments in particular), it is important to use the correct data type. It is also important to stay within the appropriate range. To assign a value greater than

32767, convert the unsigned integer value to the signed integer required by BASIC and FORTRAN. Lattice C programmers avoid this problem by using type *unsigned* or type *short integers* for these arguments.

Values greater than 32767, when assigned to integer variables, generate an overflow condition during execution. When returned to integer variables, they usually come out in two's complement form (as values in the range -32768 to -1.) This may affect the way your program tests and uses them.

One way to avoid this is to specify values in hexadecimal form, especially where the bitmasks AND and XOR are concerned. (For more on AND and XOR, see the function pages)

Reading the Argument Pages

The following pages contain detailed descriptions of each argument. In the examples, arbitrary alphabetic labels represent the arguments. You may change them in the code you write. Or, depending on the language you use, you can name them in more or less the same way. They are intended to clarify the purpose of each argument and to indicate its position in the argument list.

Arguments are position-specific. Be sure that arguments for adapter, device, and channel are consistent with the hardware you're accessing. Also make sure that commas (or other recognized delimiters) separate adjacent arguments.

Adapter Number

Label: adapt

Type: Integer value

Range: 0 to 3

Purpose: The adapter number indicates which of the adapters that function accesses.

Remarks: A single Personal Computer can accommodate up to four adapters. Switches on the card assign each an adapter number of 0 to 3. If a value for this argument lies outside this range (or is not assigned to an adapter currently installed in the system), an Unknown Adapter (128) error returns in the status variable.

Related Arguments:

Device Number

Bit Number

Label: bit

Type: Integer value

Range: 0 to 15

Purpose: This argument specifies a binary input or output bit to be tested, set, or cleared by the function.

Remarks: You must assign an integer of value 15 to 0 to this argument. Bit 15 is the most significant bit, and 0 is the least significant. Other values return an Unknown Bit Value (137) error in the status variable.

Channel Low

Label: chanlo

Type: Integer value

Range: 0 to 255

Purpose: This argument selects the channel that the input or output function accesses. In a scanning input function, it specifies the lowest numbered channel included in the scan.

Remarks: Functions accessing a single channel require only a single channel argument. By convention that argument is *chanlo*.

The wide range for this argument provides maximum room for the expansion bus interface. In practice, the accessed device determines the argument's effective range. The allowable values for the on-board devices are:

Device Name	Device #	Channel Range
Analog Input	9	0 to 3
Analog Output	9	0 to 1
Binary I/O	8	Not Applicable
Counter	10	0

For an expansion device, the range of values for this argument depends on the number of channels the device supports. If the value you use is outside the valid channel range for the device, the actual channel selected is determined by the value of *chanlo modulo* the number of channels supported by the device. If *chanlo* is less than 0 or greater than 255, an Invalid Channel Range (134) error returns in the status variable.

Related Arguments:

Channel High

Count

- Label:** count
- Type:** Long integer (or real) value
- Range:** 0 to 16 000 000
- Purpose:** This determines the number of times an iterative (multiple) function is performed. It also determines the time value of the DELAY function.
- Remarks:** The value for this argument must not exceed the amount of storage allocated for the target array of the function. It also must not exceed the amount of data in the source array. This is especially true when the function performs a scanning input. These involve count scans, each of which may generate several values.

In Compiled BASIC and Interpreted BASICA, this argument must be a real variable with an integer value in the specified range. Any fractional component is ignored. In C, this argument must be either a variable of type *long int*, or an expression that evaluates to type *long int*. In FORTRAN, this argument must be either a variable of type INTEGER*4 or an expression that evaluates to type INTEGER*4.

If *count* is 0, the function is called but not performed. If *count* is less than 0 or greater than 16 000 000, an Invalid Count Range (135) error returns in the status variable.

Related Arguments:

Data Variable.

Data Variable

Label: data

Type: Integer variable (integer array)

Range: -32768 to 32767

Purpose: This argument references a variable or array element to which a function will write data.

Remarks: If the function is simple (non-iterative), the variable must be an integer variable. If it is a multiple (iterative), or scanning function, the variable must be the first element of an integer array.

The on-board analog input and output devices have a resolution of 12 bits in the range 0 to 4095. Analog output data outside this range is interpreted *modulo* 4096.

The on-board binary and counter timer devices have a resolution of 16 bits; they return data in the range -32768 to 32767. The most significant bit is 15, and the least significant is 0.

Device Number

Label: device

Type: Integer value

Range: 0 to 255

Purpose: This argument determines which I/O device the function accesses. Every I/O device has a unique device number. Each adapter includes the following on-board devices:

Device #	Device
8	Binary I/O device
9	Analog I/O device
10	Counter device

Remarks: As noted above, values for this argument must fall in the range of 8 to 10. Values from 0 to 7 and from 12 to 255 access devices installed through the expansion bus interface. If a value outside this range appears in this argument, an Unknown Device (131) error returns in the status variable.

The device number chosen must correspond to either an adapter installed in the computer or an expansion device. Attempts to access a device that does not exist, or to access a device with an inappropriate function call, can return erroneous values or a Device Timeout (138) error.

Related Arguments:

Adapter Number

Mode

Label: mode

Type: Integer value

Range: 0 or 128

Purpose: This argument determines if system interrupts are enabled or disabled during the processing of multiple I/O functions.

Remarks: This argument applies only to the AINM, AOUM, BINM, BOUM, and CINM multiple I/O functions. Zero is the only allowed value for other functions.

If *mode* is 0, normal system interrupt processing continues during the processing of the multiple I/O functions. If *mode* is 128, the device driver inhibits system interrupts to increase I/O performance.

For values other than 0 and 128, an Unknown Mode (133) error returns in the status variable.

Related Arguments:

Rate.

Rate

- Label:** rate
- Type:** Long integer (or real) value
- Range:** 0 to 1 000 000
- Purpose:** This specifies the rate, in samples-per-second, at which the function executes.
- Remarks:** If this value is 0, an external clock signal on the IRQ line determines the sampling rate. Also, at 0, the function does not execute until the IRQ line goes from high to low.
- If you enter a value greater than either the function or the current device supports, then iterations occur at the maximum rate. A Timer Overrun (1) error or Excessive Timer Overrun (142) error returns in the status variable. ~-
 In Compiled BASIC and Interpreted BASICA programs, this argument must be a real variable containing an integer in the specified range. Decimal components are ignored. In C programs, this argument must be either a variable of type *long int* or an expression that evaluates to type *long int*. In FORTRAN programs, this argument must be either a variable of type INTEGER*4 or an expression that evaluates to type INTEGER*4.
- If *rate* is less than 0 or greater than 1 000 000, an Invalid Rate Range (139) error returns to the status variable.

Related Arguments:

Count, Mode

Status Variable

Label: stat

Type: integer

Range: -32768 to 32767

Purpose: This argument references the integer variable to which the function's status code returns.

Remarks: A non-zero return indicates a general execution failure of the function.

Storage Operation

Label: stor

Type: Integer value

Range: 0

Purpose: This argument is reserved.

Remarks: Zero is the only allowed value. For values other than zero, an Unknown Storage Operation (132) error returns in the status variable.

Analog Input Multiple AINM

Purpose: AINM samples analog values from the specified adapter, device, and channel.

Format: CALL AINM (adapt, device, chanlo, ctrl, mode, stor, count, rate, data, stat)

adapt	Adapter number accessed
device	Device number accessed
chanlo	Channel accessed
ctrl	Expansion device control
mode	Execution mode
stor	Must be 0
count	Number of times the function executes
rate	Execution rate, in samples-per-second
data	First element of the array that receives returning data
stat	Variable that receives the returning execution status.

Analog Input Simple

AINS

Purpose: AINS selects a single analog value from the adapter, device, and channel and stores it in the data variable.

Format: CALL AINS (adapt, device, chanlo, ctrl, data, stat)

adapt	Adapter number accessed
device	Device number accessed
chanlo	Channel accessed
ctrl	Expansion device control
data	Variable that receives the returning data
stat	Variable that receives the returning execution status.

Binary BIT Output Simple BITOUS

Purpose: BITOUS sets the state of a bit in the binary output word of the adapter and device. The bit takes the value of the data variable.

Format: CALL BITOUS (adapt, device, bit, data, stat)

adapt	Adapter number accessed
device	Device number accessed
bit	The bit number (15 to 0) for output
data	Variable from which the bit value is retrieved (must be 1 or 0)
stat	Variable that receives the returning execution status.

Remarks: When the function is finished, execution status returns to the status variable.

BITOUS neither reads nor affects handshaking lines on the binary input port. The function acts on only the bit specified. It numbers bits from 15 to 0, beginning with the most significant. The single exception to this rule occurs when BITOUS is the first binary output function executed after system initialization. In that case, BITOUS sets or clears the specified bit and zeroes all other bits.

Binary Output Simple BOUS

Purpose: BOUS outputs the contents of the data variable as a 16-bit binary word.

Format: CALL BOUS (adapt, device, hndshk, data, stat)

adapt Adapter number accessed

device Device number accessed

hndshk Handshake (must be 0)

data Variable from which data is retrieved

stat Variable that receives the returning
 execution status.

Remarks: BOUS operates through the adapter and device. When the function is finished, execution status returns to the status variable. The most significant bit is 15 and the least significant bit is 0.

A data value of 9 (binary word 0000 0000 0000 1001) sets bits 0 and 3 of the binary output word. This latched value remains in effect until changed by another binary output function.

Delay Execution

DELAY

Purpose: DELAY interrupts program execution.

Format: CALL DELAY (adapt, count, stat)

adapt Adapter number accessed

count Length of delay (milliseconds)

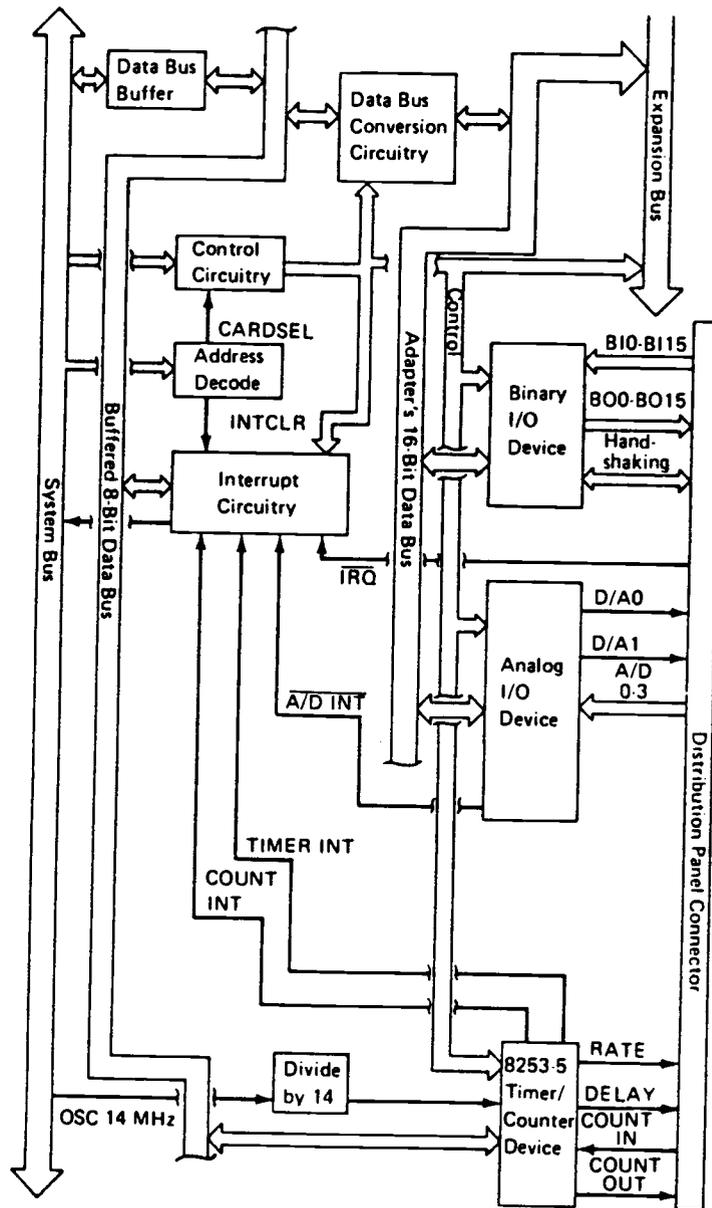
stat Variable that receives the returning
 execution status.

Remarks: DELAY allows you to perform timed sampling at intervals longer than one second allowed by iterative I/O functions. Software overhead must be taken into account. The time required to execute a DELAY function and a subsequent I/O function increases the length of delay by several to several hundred milliseconds.

When the function is finished, execution status returns to the status variable.

Major Components

Following is a block diagram of the Data Acquisition Adapter.



Analog I/O Device

The Data Acquisition Adapter's analog I/O device consists of two subsystems:

- Analog input: An analog-to-digital conversion subsystem.
- Analog output: A digital-to-analog conversion subsystem.

Analog Input Subsystem

On the following page is a block diagram of the analog input subsystem.

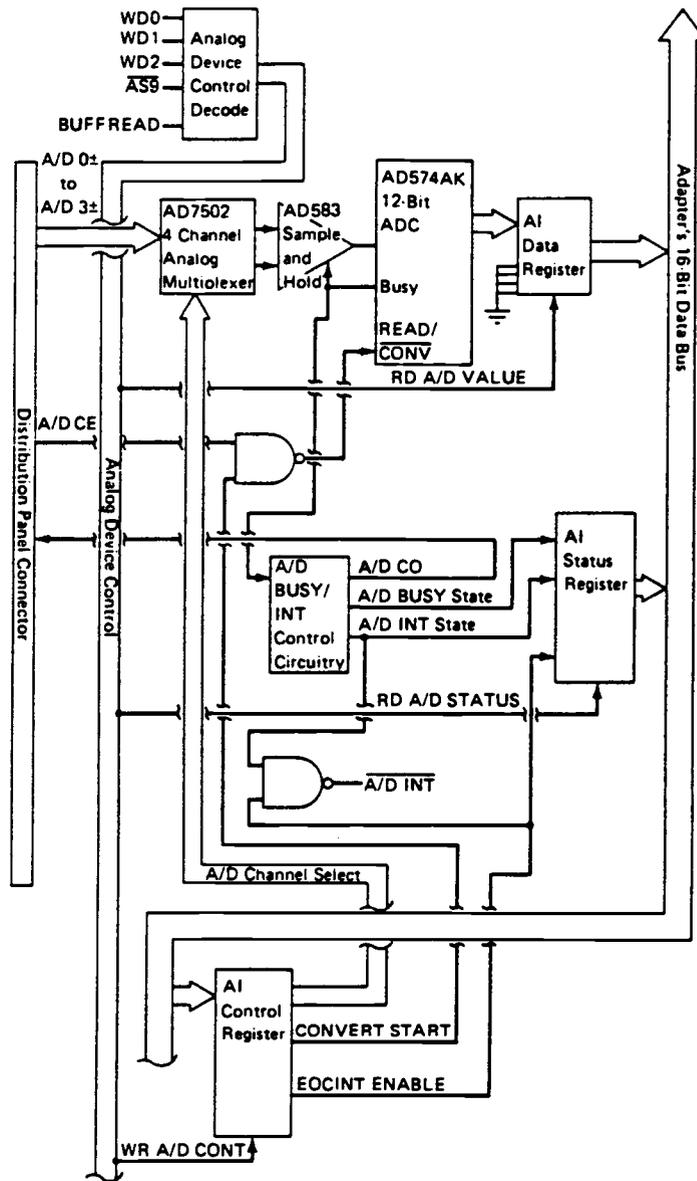
Analog-to-digital conversion is the process of converting analog signals (voltages) over a given range to digital values.

Unlike digital (binary) signals, which have only two voltage states, analog signals have infinite voltage levels over a particular range.

Analog-to-digital converters (ADCs) are categorized by the number of bits of resolution they allow. The greater the number of bits, the greater the number of discrete voltage levels that can be represented.

The Data Acquisition Adapter has an analog input device with the following features:

- Four, multiplexed, differential channels
- An ADC with 12-bit resolution
- Switch-selectable ranges
- Optional data-conversion control with 'A/D convert out' and 'A/D convert enable in' lines.



Analog input subsystem

The Data Acquisition Adapter's analog input device (device number 9) has four channels, which are multiplexed into a single ADC. This device converts analog signals in one of three ranges to digital values in the range of 0 to 4095.

The three switch-selectable ranges are:

- - 5 to +5 volts
- -10 to +10 volts
- 0 to +10 volts

The relationship of the analog input voltage to the returned digital value depends on the range for which the hardware is configured. The selected range setting for analog input is in effect for all analog input channels. For example, in the -5 to +5 volt configuration, an input of +4.997 volts generates a full-scale value of 4095; an input of 0 volts generates a value of 2048; and an input of -5 volts generates a value of 0.

Analog Input Device Control

The use of the $\overline{AS9}$ strobe causes the analog input device to be accessed as device number 9.

The control decode circuitry of the analog device decodes $WD0$ through $WD2$, $\overline{AS9}$, and $BUFFREAD$ to generate the following control signals:

WR A/D CONT	Write analog-to-digital control. Allows the AI control register to be written to.
RD A/D STATUS	Read analog-to-digital status. Allows reading of the AI status register.
RD A/D VALUE	Read analog-to-digital value. Allows reading of the AI data register.

Analog Input Device Registers

AI Control Register	The AI control register contains the analog-to-digital channel selection, analog-to-digital interrupt-enable information, and convert start bit information. The AI control register is cleared by $\overline{BUFFRES}$ during power-on-reset.
AI Status Register	The AI status register contains information about 'A/D busy,' the 'A/D interrupt status,' and the readback of the 'A/D interrupt enable.'
AI Data Register	The 16-bit AI data register contains the data from the ADC. Because the output of the ADC is a 12-bit digital value, the four highest bits of the register are grounded.

Starting an Analog-to-Digital Conversion

The convert start bit from the AI control register is logically ANDed with the external 'A/D convert enable' signal from the distribution panel connector. The result is inverted to generate an active low signal, which is brought to the $\overline{\text{READ/CONV}}$ pin of the AD574 ADC.

Reading an Analog-to-Digital Value

The $\overline{\text{READ/CONV}}$ pin must be taken high before the analog-to-digital value can be read. This is accomplished by writing a convert start bit equal to 0 to the AI control register.

Channel Selection

The differential analog to digital channel pair is selected by the AD7502 4-channel, analog multiplexer on the basis of the analog-to-digital channel-select bits of the AI control register.

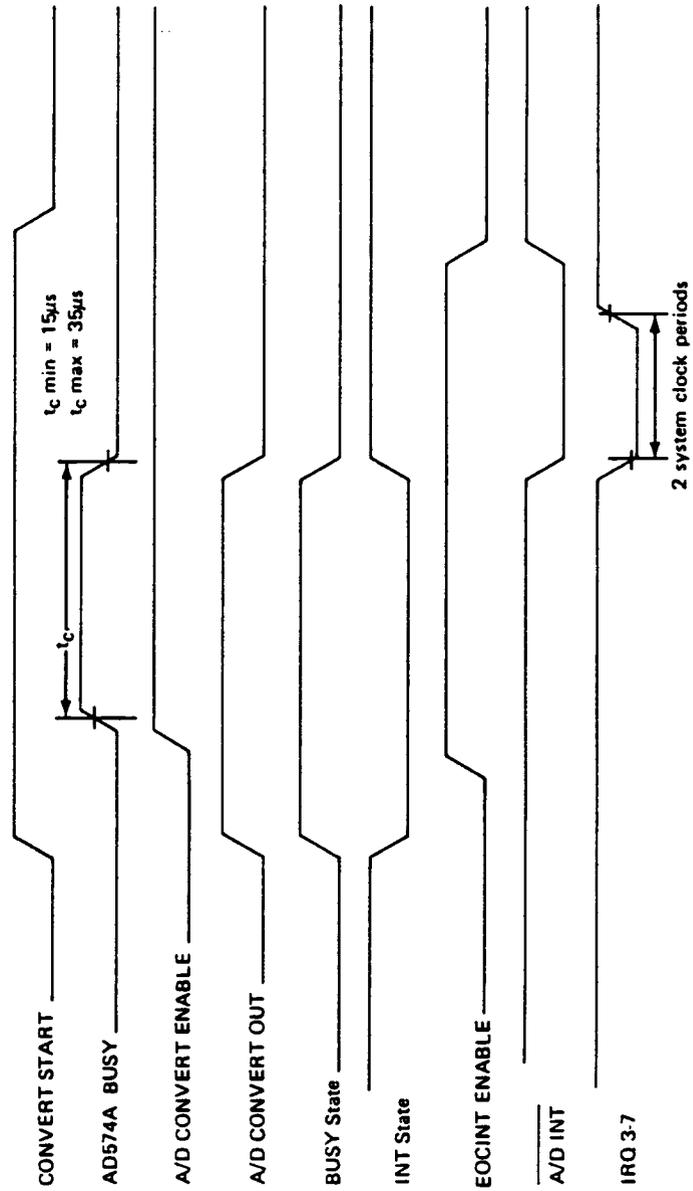
Sample and Hold

During a conversion, the 'busy' signal from the AD574A ADC causes the AD583 Sample and Hold to hold its present value when the 'busy' signal is high, and starts sampling again when it is low.

'A/D Busy' and Interrupt States

At the end of a conversion, the AD574 ADC's 'busy' signal goes low, and the AI status register shows that the analog input device is in the not-busy and interrupting state.

Following is a timing diagram of analog-to-digital conversion.



'A/D Interrupt'

The actual 'A/D interrupt' signal (A/D INT) is a result of the logical ANDing of the INT STATE status bit in the AI status register, and the EOCINT ENABLE bit from the AI control register. The inverted result generates $\overline{\text{A/D INT}}$ (an active low signal), which goes to the interrupt circuitry.

'A/D Convert Out'

The 'A/D convert out' (A/D CO) signal is brought out to the distribution panel connector on the Data Acquisition Adapter.

The 'A/D convert out' signal is set (TTL high) when a conversion has been commanded by programming the convert start bit. The signal remains high until the conversion is complete. If the analog signals received by the on-board analog input device are from an external device that can be made to send data on receipt of a TTL high pulse, you may use a synchronization scheme in which the program's request for an analog-to-digital conversion triggers (using 'A/D convert out') the output of analog data from the external device.

'A/D Convert Enable'

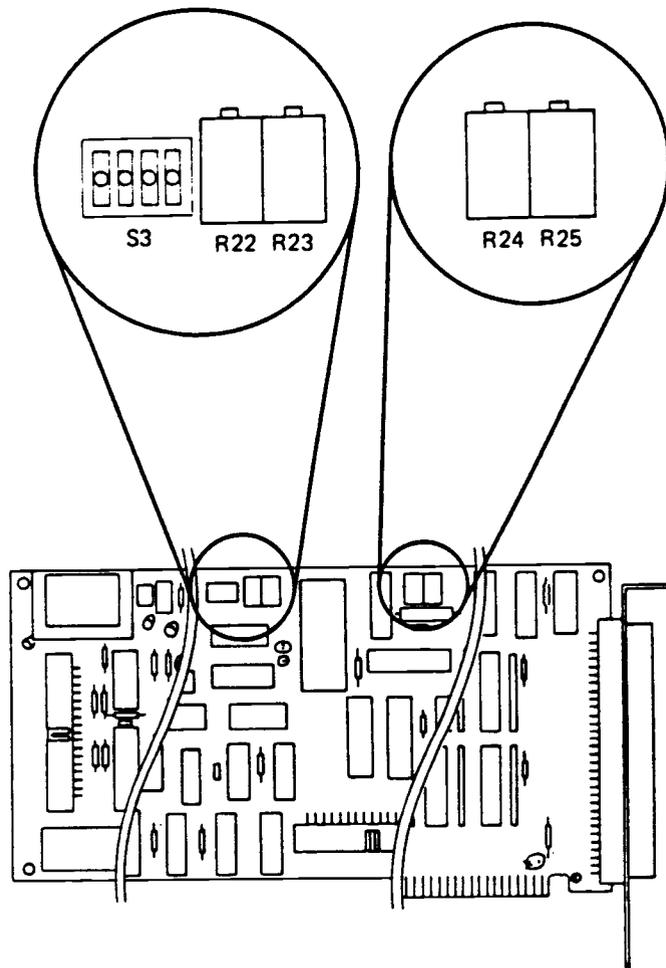
The 'A/D convert enable' (A/D CE) signal is brought out to the distribution panel connector on the Data Acquisition Adapter.

By holding the 'convert enable in' signal low (TTL), an external device can inhibit or delay all analog-to-digital conversions ordered by programming.

To be considered valid and allow an analog-to-digital conversion, the 'convert enable in' signal must remain high until the 'convert out' signal goes low again.

Analog Input Potentiometers

Four potentiometers (R22, R23, R24, and R25) on the Data Acquisition Adapter control bipolar offset, unipolar offset, gain, and common mode rejection for the analog input device. The following diagram shows the location of these potentiometers.



In the following “LSB” represents the weight of the least-significant bit of the 12-bit digital output code of the ADC.

The table shows the 1-LSB values for each analog input range.

Range	1 LSB
0 to +10 volts	2.44 mV
-5 to +5 volts	2.44 mV
-10 to +10 volts	4.88 mV

The ADC is intended to have a 1/2-LSB offset so the exact analog input for a given code will be in the middle of that code (halfway between the transitions to the codes above and below it). The information under “Bipolar Offset” and “Unipolar Offset” explains this 1/2-LSB offset.

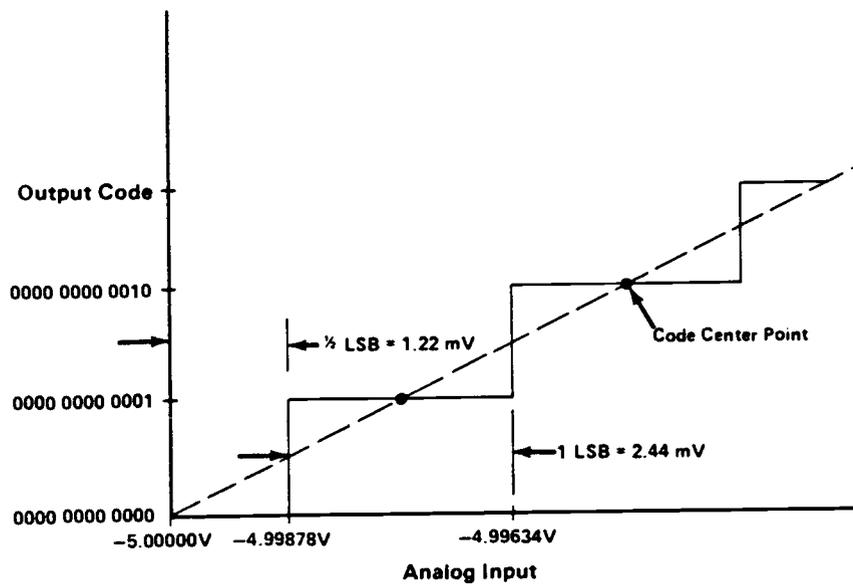
Bipolar Offset:

The value of R22 is set so the transition from the digital output code 0000 0000 0000 to 0000 0000 0001 occurs for an input voltage 1/2 LSB above negative full scale. R22 takes effect when a bipolar range (-5 to +5 volts or -10 to +10 volts) is selected.

The following shows the input voltages for the transition from the output code 0000 0000 0000 to 0000 0000 0001.

Range	Input Voltage for First Code Transition
-5 to +5 volts	-4.99878 volts
-10 to +10 volts	-9.99756 volts

The following shows the first few output-code transitions for the -5 to +5 volt range.



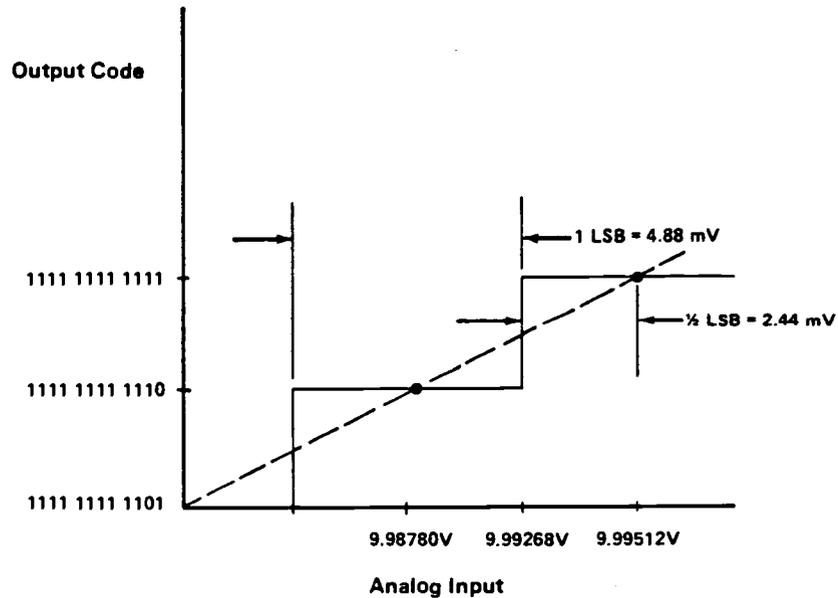
Gain:

The value of R23 is set so the last transition (1111 1111 1110 to 1111 1111 1111) occurs for an input voltage 1-1/2 LSB below full scale.

The following shows the input voltage for the transition from the output code 1111 1111 1110 to 1111 1111 1111.

Range	Input Voltage for Last Code Transition
0 to +10 volts	+9.99634 volts
-5 to +5 volts	+4.99634 volts
-10 to +10 volts	+9.99268 volts

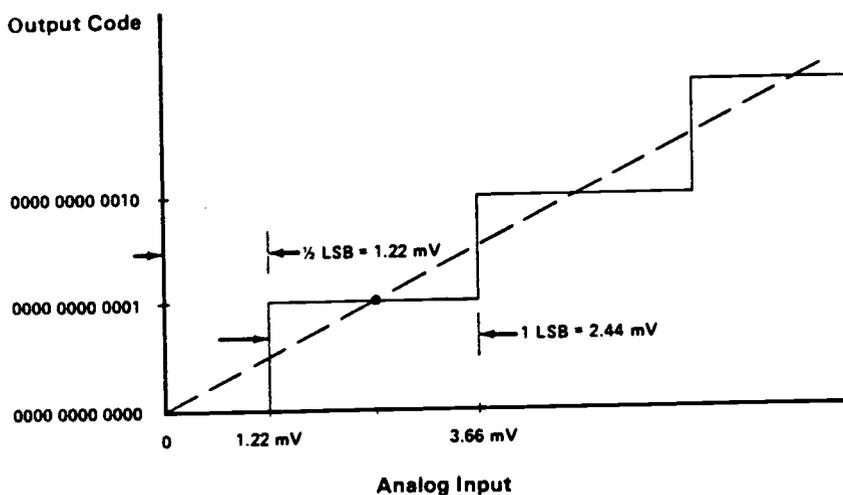
The following shows the last few output-code transitions for the -10 to +10 volt range.



Unipolar Offset:

The value of R24 is set so the first transition (0000 0000 0000 to 0000 0000 0001) occurs for an input voltage of $+1/2$ LSB. R24 takes effect when the unipolar range (0 to +10 volts) is selected.

The following shows the first few output-code transitions for the 0 to +10 volt range.

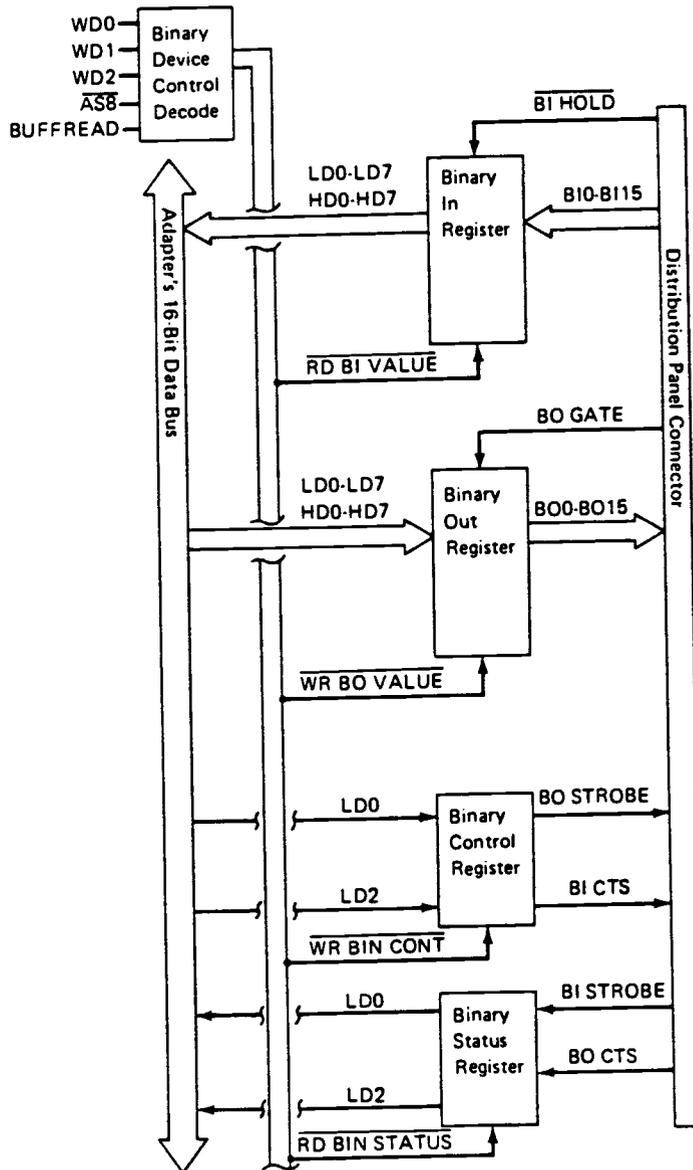


Common Mode Rejection:

R25 allows for the reduction and balancing of the error caused by common mode noise (voltage common to both sides of an analog input channel). The common-mode input range specification for the analog input device is ± 11 volts maximum. The value of R25 is set so on the most sensitive range (-5 to +5 volts), the effect of common mode voltage is balanced on each side of zero volts. For example, a common mode voltage of +11 volts produces the same output code as a common mode voltage of -11 volts.

Binary I/O Device

Following is a block diagram of the binary I/O device.



The Data Acquisition Adapter's binary I/O device has the following features:

- A 16-bit binary output port (BO0 through BO15)
- A 16-bit binary input port (BI0 through BI15)
- Input and output handshaking over the 'strobe' and 'clear-to-send' lines
- Direct control using BO GATE ('binary out gate') and BI HOLD ('binary in hold').

Digital signals have only two voltage states: On (high, +3 volts) and Off (low, +0.2 volts). Digital signals in this range are called *TTL signals*, because they are the proper levels to be interpreted by the transistor-to-transistor logic circuitry. These signals have many uses in data acquisition and control applications. Among these are sensing the state of two-state devices and controlling devices that require two-state control signals.

Binary I/O Device Control

The use of the $\overline{AS8}$ strobe causes the binary I/O device to be accessed as device number 8.

The $\overline{AS8}$ strobe as an enable, the WD0 through WD2 word bits, and the BUFFREAD signal are used to decode which binary decode operation is to occur.

Following are the four decode operations:

$\overline{WR\ BIN\ CONT}$	Write binary control: Controls the latching of the binary output strobe (BO STROBE) and the binary input clear-to-send (BI CTS) bits by the binary control register.
$\overline{RD\ BIN\ STATUS}$	Read binary status: Controls the reading of the binary input strobe (BI STROBE) and the binary output clear to send (BO CTS) bits by the binary status register.
$\overline{WR\ BO\ VALUE}$	Write binary value: Controls the writing of the binary output word (BO0 through BO15) to the binary output register.
$\overline{RD\ BI\ VALUE}$	Read binary value: Controls the reading of the binary input word (BI0 through BI15) from the binary input data register.

Binary I/O Device Registers

Following is a description of the binary I/O device registers.

Binary Control Register	Contains the BO STROBE bit and the BI CTS bit. These bits do not physically cause or prevent binary I/O events from occurring. They are programming control bits.
Binary Status Register	Allows the status of BO CTS and BI STROBE bits to be monitored. These bits do not physically cause or prevent binary I/O events from occurring. They are programming status bits.
Binary Input Register	When $\overline{\text{BI HOLD}}$ is brought high (or if no connection is made), the binary input register is not latched and allows the current state of the binary input lines to be monitored by reading the binary input register. Grounding $\overline{\text{BI HOLD}}$ causes the binary input register to latch the current state of all binary input lines. If the grounding of the $\overline{\text{BI HOLD}}$ line is maintained, any later read will obtain the value that was present when the line was initially grounded.
Binary Output Register	Contains the binary output word (BO0 through BO15). Grounding the BO GATE signal places the binary output port in the tri-state condition (all points floating). The binary outputs are gated out when the BO GATE signal is brought high (or if no connection is made).

Binary Output Subsystem

Following is a description of the binary output subsystem.

Binary Output Port (BO0 through BO15)

This subsystem uses high-power, tri-state, bus-driving devices. Changes in the binary output word are carried out on a per-bit basis. Only those bits affected by a change in the output word are actually changed. All others remain the same.

The output port of the binary I/O device supplies 16 high/low signals under program control. As with the input port, these signals can be used individually or considered as a 16-bit data word.

Binary Out Gate

You may place the output port in tri-state by pulling the binary out gate (BO GATE) lines low. These and all other data, handshaking, and control lines are pulled high by internal resistors to +5 volts. No connections to them are necessary unless your application requires handshaking or control.

Binary Output Handshaking

Because all communication lines are internally pulled up to their logical true state, you can use or not use binary output handshaking, depending on the requirements of your communication setup.

Binary output can be synchronized with the data input capabilities of the external device. The external device must be able to send a TTL signal to indicate it is ready for new data. It also must be able to accept parallel binary data when it receives a signal from the Data Acquisition Adapter's binary I/O device indicating the data is available.

Error Codes

Errors return in decimal to the status variable. Check this variable after an execution to see if an error has occurred. All errors return in the same way. There is, however, a difference between “hard” and “soft” errors, as explained below.

No Error

You see a No Error condition reported in the status variable when everything is working properly.

Error Code Description

0	No Error: The function has executed normally and control returns to the caller.
---	---

Soft Error

A soft error allows the function to execute, but affects the integrity of the data.

Error Code Description

1	Timer Overrun: The execution rate was faster at times than the device and function could manage. The function executed, but a small percentage of samples was taken at an irregular and slower rate than specified.
142	Excessive Timer Overrun: The execution rate was often faster than the device and function could manage. The function executed, but a large percentage of samples was taken at an irregular and slower rate than specified.

Error Code	Description
135	Invalid Count Range: An incorrect value is specified for the count argument.
136	Unknown Handshake Value: An incorrect value is specified for the handshake argument.
137	Unknown Bit Value: The value for the bit number argument is outside the range of valid bits.
138	Device Timeout: A device that does not exist was specified for the device argument or the external handshaking did not occur.
139	Invalid Rate Range: The value for the rate argument is outside the valid range.

Hard Errors

A hard error indicates a failure to execute. The function ends prematurely and control returns to the caller. No data is collected.

Error Code Description

- | | |
|-----|---|
| -2 | No Device Driver: (Compiled languages only). The device driver DAC.COM was not found. |
| 128 | Unknown Adapter: The requested adapter is either not in the system (that is, no adapter is addressed to that adapter number), or the adapter is not working. |
| 131 | Unknown Device: Either the requested device is not known, or a presence test on the device has failed. |
| 132 | Unknown Storage Operation: An incorrect value is specified for the storage argument. |
| 133 | Unknown Execution Mode: An incorrect value is specified for the execution mode argument. |
| 134 | Invalid Channel Range: Values for the channel low and channel high arguments do not set a valid channel range (that is, <i>chanhi</i> has a lower value than <i>chanlo</i> or is greater than 255). |

APPENDIX C

PROGRAMS

```

C *****POWER*****
C *
C * THIS PROGRAM MEASURES OUTPUT VOLTAGE FROM A WIDE RANGE *
C * LOG-CHANNEL AMPLIFIER WHICH IS RELATED TO THE POWER *
C * OF OREGON STATE UNIVERSITY TRIGA REACTOR (OSTR), *
C * 10 VOLTS CORRESPONDS TO 1 MEGA WATT. *
C * THE REACTOR POWER IS READ FROM THE LINEAR CHANNEL TO *
C * GET A MORE ACCURATE EQUATION BETWEEN REACTOR POWER AND *
C * THE ADC OUTPUT CODE (VOLTAGE). THE POWER IS MEASURED *
C * TEN TIMES FOR EACH CASE USING ANALOG INPUT SIMPLE *
C * FUNCTION (AINS), AND AN AVERAGE OF THESE MEASUREMENTS *
C * IS TAKEN. *
C *****
PROGRAM POWER
INTEGER*2 ADAPT, DEVICE, CHANLO, CTRL,
*RAWVAL, STAT, I
CHARACTER*14 ANS, DATAOUT
ADAPT=0
DEVICE=9
CHANLO=3
CTRL=0
STAT=0
I=0
WRITE(*,5)
5 FORMAT(' ENTER OUTPUT DATA FILE NAME '/')
READ(*,1)DATAOUT
1 FORMAT(A14)
OPEN(UNIT=81, FILE=DATAOUT, STATUS='NEW')
7 WRITE(*,10)
10 FORMAT(' ENTER R WHEN YOU ARE READY'/)
READ(*,1)ANS
IF(ANS.EQ.'R'.OR.ANS.EQ.'r') GOTO 15
GOTO 7
15 P1=0.0
DO 150 J=1,10
CALL AINS(ADAPT, DEVICE, CHANLO, CTRL,
*RAWVAL, STAT)
IF(STAT.NE.0) GOTO 300
P1=RAWVAL+P1
150 CONTINUE
P1=P1/10.0
WRITE(81,100)P1
WRITE(*,100)P1
100 FORMAT(1X, F10.0)
WRITE(*,20)
20 FORMAT(' ENTER Y WHEN YOU HAVE OTHER DATA OR ANY
*CHERACTOR IF NCT '/')
READ(*,1)ANS
IF(ANS.EQ.'Y'.OR.ANS.EQ.'y') GOTO 7
GOTO 400
300 WRITE(*,200)STAT
200 FORMAT(1X, 'EXECUTION ERROR', I6/)
400 CLOSE(81)
STOP
END

```

```

C *****CURVE*****
C *
C * THIS PROGRAM IS USED TO FIND AN EQUATION BETWEEN *
C * THE REACTOR POWER AND THE MEASURED ADC OUTPUT CODE. *
C * SINCE THE CODE IS PROPORTIONAL TO THE LOGARITHM OF *
C * REACTOR POWER , THE LOGARITHM OF THE POWER IS TAKEN, *
C * AND LINEAR REGRESSION IS USED TO FIND THE EQUATION. *
C * THE EQUATION IS IN THE FORM OF ( POWER = 10**(A+B*Cp)) *
C * WHERE Cp IS THE CODE ASSOCIATED WITH REACTOR POWER. *
C *****
DIMENSION X(100),Y(100),Z(100),R(100)
SUM0=0.0
SUM1=0.0
SUM2=0.0
SUM3=0.0
SUM4=0.0
PRINT*,' ENTER THE NUMBER OF DATA '
READ(*,1)I
1  FORMAT(I3)
   DO 10 J=1,I
   READ(*,2)X(J),Y(J)
2  FORMAT(2F10.0)
   Z(J)=ALOG10(Y(J))
10 CONTINUE
   WRITE(*,3)(X(J),Y(J),J=1,I)
3  FORMAT(1X,2F15.5)
   DO 20 JJ=1,I
   R(JJ)=Z(JJ)*X(JJ)
20 CONTINUE
   DO 30 II=1,I
   SUM0=SUM0+R(II)
   SUM1=SUM1+X(II)
   SUM2=SUM2+Z(II)
   SUM3=SUM3+X(II)*X(II)
   SUM4=SUM4+Z(II)*Z(II)
30 CONTINUE
   B=(SUM0-SUM1*SUM2/I)/(SUM3-SUM1*SUM1/I)
   A=(SUM2/I-B*SUM1/I)
   XR=(SUM0-SUM1*SUM2/I)**2/((SUM3-SUM1*SUM1/I)*(SUM4-SUM2*
*SUM2/I))
   WRITE(*,40)A,B,XR
40  FORMAT(1X,'A= ',F15.5,1X,'B= ',F15.5,1X,'R*2 = ',F15.10)
END

```

```

C *****LOGGER*****
C *
C * WRITTEN BY
C * ALLA J. MOHAMMAD BAKIR
C * COLLEGE OF ENGINEERING
C * NUCLEAR ENGINEERING DEPARTMENT
C * OREGON STATE UNIVERSITY
C * MAY 1 , 1988
C * THIS PROGRAM IS DERIVED FROM A PROGRAM WRITTEN BY:
C * ROBERT J. TUTTLE,ATOMICS INTERNATIONAL, MARCH 1,1967.
C * THIS PROGRAM MEASURES A VOLTAGE THAT IS RELATED TO
C * REACTOR POWER USING DATA ACQUISITION AND CONTROL
C * ADAPTER , AND FROM THE REACTOR POWER HISTORY THE
C * REACTIVITY CAN BE CALCULATED USING THE POINT KINETICS
C * EQUATIONS.
C * INPUT DATA CONSIST OF THE FRACTIONAL RELATIVE DELAYED
C * NEUTRON YIELDS(A(I),SUM OF A(I)=1.0),DELAYED NEUTRON
C * PRECURSOR DECAY CONSTANTS ( LAMBDA(I) IN RECIPROCAL
C * SECOND), THE PROMPT NEUTRON LIFETIME ( L IN SECONDS),
C * THE EFFECTIVE DELAYED NEUTRON FRACTION (BETA), THE
C * VALUE OF THE SOURCE CORRECTION TERM (SC IN CENTS/
C * SECOND/COUNT).
C * THIS PROGRAM USES A SPECIAL FUNCTION SUBROUTINE, FOR
C * MORE INFORMATION ABOUT THESE FUNCTIONS REFER TO THE
C * IBM PERSONAL COMPUTER DATA ACQUISITION AND CONTROL
C * ADAPTER PROGRAMMING SUPPORT.
C * REACTOR POWER SHOULD BE CONSTANT FOR ABOUT 1 MINTUE
C * BEFORE DATA COLLECTION BEGINS AND NO REACTIVITY
C * CHANGE SHOULD BE MADE DURING THIS PERIOD , THE POWER
C * DATA IS COLLECTED BY A RATE=RATE1, AND THE COLLECTION
C * TIME IS EQUAL TO COUNT1/RATE1 WHICH IS EQUAL TO
C * 1 MINUTE, THIS WAITING TIME PERMITS ESTABLISHMENT OF
C * THE EQUILIBRIUM PRECURSOR POPULATION WHICH IS
C * CALCULATED FROM THE AVERAGE POWER IN THE FIRST ONE
C * MINUTE.
C * REACTOR POWER FOR WHICH THE REACTIVITY TO BE FOUND IS
C * MEASURED BY A RATE=RATE2, AND COLLECTION TIME IS
C * EQUAL TO COUNT2/RATE2.
C * THIS PROGRAM HAS THREE SECTIONS, FIRST IS A POWER
C * MEASURING SECTION, IN WHICH A VOLTAGE RELATED TO THE
C * REACTOR POWER IS MEASURED BY THE DATA ACQUISITION AND
C * CONTROL ADAPTER, A SECOND SECTION INVOLVED WITH A
C * REACTIVITY CALCULATION , AND FINALLY DISPLAY OF
C * REACTIVITY ON A SEVEN SEGMENT DISPLAY.
C * *****
C PROGRAM LOGGER
C DIMENSION Y1(3000),Y2(3000),PIN(3000),XB(3000)
C INTEGER*2 RAWDTA(3000),POWDTA(3000)
C INTEGER*2 ADAPT,DEVICE,CHANLO,CTRL,MODE,
C *STOR,STAT
C INTEGER*4 COUNT1,RATE1,COUNT2,RATE2
C INTEGER*4 COUNT,RATE
C REAL A(6),LAMBDA(6),L,T(3000),R(3000)
C DOUBLE PRECISION
C 1B(6),BD(6),C(6),D(6),X1(6),K(3000),
C 2AT,DN,P0,DP,PBAR,SC,SUM
C CHARACTER*14 DATAOUT,FILENAME,RECIN
C ADAPT=0
C DEVICE=9
C CHANLO=3

```

```

CTRL=0
MODE=0
STOR=0
STAT=0
10 READ(*,10) (A(I), I=1,6)
   FORMAT(6F10.0)
   READ(*,10) (LAMBDA(I), I=1,6)
   READ(*,20) L, BETA, SC
20  FORMAT(3F10.0)
   WRITE(*,1)
1   FORMAT(' ENTER COUNT1, RATE1 2I4 FORMAT '/')
   READ(*,2) COUNT1, RATE1
2   FORMAT(2I4)
   WRITE(*,3)
3   FORMAT(' ENTER COUNT2, RATE2 2I4 FORMAT '/')
   READ(*,2) COUNT2, RATE2
   WRITE(*,4)
4   FORMAT(' ENTER OUTPUT DATA FILENAME '/')
   READ(*,5) DATAOUT
5   FORMAT(A14)
   WRITE(*,6)
6   FORMAT(' ENTER REC. FILENAME '/')
   READ(*,5) FILENAME
   WRITE(*,7)
7   FORMAT(' ENTER POWER FILENAME AND BE READY '/')
   READ(*,5) RECIN
   OPEN(UNIT=81, FILE=DATAOUT, STATUS='NEW')
   OPEN(UNIT=10, FILE=FILENAME, STATUS='NEW')
   OPEN(UNIT=90, FILE=RECIN, STATUS='NEW')
   COUNT=COUNT1
   RATE=RATE1
C
C   POWER MEASURING SECTION
C
   CALL AINM(ADAPT, DEVICE, CHANLO, CTRL,
*MODE, STOR, COUNT, RATE, RAWDTA(1), STAT)
   IF(STAT.NE.0) GOTO 300
   COUNT=COUNT2
   RATE=RATE2
   CALL AINM(ADAPT, DEVICE, CHANLO, CTRL,
*MODE, STOR, COUNT, RATE, POWDTA(1), STAT)
   IF(STAT.NE.0) GOTO 300
   WRITE(81,100) (RAWDTA(I), I=1, COUNT1)
   WRITE(81,100) (POWDTA(I), I=1, COUNT2)
100 FORMAT(1X,5I10)
   CLOSE(81)
C
C   REACTIVITY CALCULATION SECTION
C
   DT=1.0/RATE2
   NOP=COUNT2
   DO 200 I=1, COUNT1
   IF(RAWDTA(I).GE.0.AND.RAWDTA(I).LE.2488) THEN
   Y1(I)=10.**(-4.06015+0.00245*RAWDTA(I))
   ELSEIF(RAWDTA(I).GT.2488.AND.RAWDTA(I).LT.2680) THEN
   Y1(I)=10.**(-3.81625+0.00233*RAWDTA(I))
   ELSEIF(RAWDTA(I).GE.2680) THEN
   Y1(I)=10.**(-4.26878+0.0025*RAWDTA(I))
   ENDIF
200 CONTINUE

```

```

DO 350 I=1,COUNT2
IF(POWDTA(I).GE.0.AND.POWDTA(I).LE.2488)THEN
Y2(I)=10.**(-4.06015+0.00245*POWDTA(I))
ELSEIF(POWDTA(I).GT.2488.AND.POWDTA(I).LT.2680)THEN
Y2(I)=10.**(-3.81625+0.00233*POWDTA(I))
ELSEIF(POWDTA(I).GE.2680)THEN
Y2(I)=10.**(-4.26878+0.0025*POWDTA(I))
ENDIF
350 CONTINUE
AT=L/BETA/DT
SUM1=0.0
DO 400 I=1,COUNT1
SUM1=SUM1+Y1(I)
400 CONTINUE
C(1)=SUM1/COUNT1
SC=SC*DT/100.0
R(1)=-SC/C(1)
K(1)=1.0+BETA*(1.0+BETA*R(1))*R(1)
T(1)=DT
DO 1100 I=1,6
X1(I)=DBLE(LAMBDA(I)*DT)
B(I)=DEXP(-X1(I))
IF(X1(I).GT.0.1) GOTO 1000
BD(I)=X1(I)-0.5*X1(I)**2+0.162582*X1(I)**3
D(I)=0.5*X1(I)-0.166666*X1(I)**2+0.041847*X1(I)**3
GOTO 1100
1000 BD(I)=1.0D00-B(I)
D(I)=(X1(I)-BD(I))/X1(I)
1100 C(I)=K(1)*C(1)
N1=NOP-2
DO 1300 N=2,N1
K(N)=1.0+BETA*K(N-1)*R(N-1)
SUM=0.0
DP=0.5*(Y2(N+1)-Y2(N))
P0=Y2(N)-0.5*DP
DO 1200 I=1,6
C(I)=C(I)*B(I)+K(N)*(Y2(N)*BD(I)+2*DP*D(I))
1200 SUM=SUM+A(I)*C(I)
PBAR=(Y2(N+1)+Y2(N))/2.0
PIN(N)=DT/PBAR
DN=Y2(N+1)-Y2(N)
R(N)=1.0+(AT*DN-SUM-SC)/(PBAR*K(N))
K(N)=1.0+BETA*K(N)*R(N)
R(N)=1.0D00+(AT*DN-SUM-SC)/(PBAR*K(N))
WRITE(*,500)R(N)
500 FORMAT(1X,E16.6)
1300 T(N)=T(N-1)+DT
WRITE(10,19)(T(N),R(N),N=1,N1)
19 FORMAT(1X,F10.5,2X,E16.6)
WRITE(90,19)(T(N),Y2(N),N=1,N1)
CLOSE(10)
CLOSE(90)
C
C REACTIVITY DISPLAYING SECTION
C
COUNT=1000
BIT=0
HNDSHK=0
DEVICE=8
DO 205 I=1,N1

```

```
XB(I)=ABS(R(I))
NX=1000*XB(I)
NX1=NX/1000
NX2=NX/100-NX1*10
NX3=NX/10-10*NX2-100*NX1
NX4=NX-10*NX3-100*NX2-1000*NX1
RAWVAL=NX1*4096+NX2*256+NX3*16+NX4
CALL BOUS(ADAPT,DEVICE,HNDSHK,RAWVAL,
*STAT)
IF(STAT.NE.0) GOTO 300
IF(R(I).LT.0.0) THEN
RAW=1
GOTO 405
ELSE
RAW=0
ENDIF
405 CALL BITOUS(ADAPT,DEVICE,BIT,RAW,STAT)
IF(STAT.NE.0) GOTO 300
CALL DELAY(ADAPT,COUNT,STAT)
IF(STAT.NE.0) GOTO 300
205 CONTINUE
300 WRITE(*,150)STAT
WRITE(*,102)
102 FORMAT(1X,' EXECUTION COMPLETE. ',/)
GOTO 101
150 FORMAT(1X,' EXECUTION ERROR ',I6)
101 STOP
END
```