AD-A033 606 TRW SYSTEMS GROUP REDONDO BEACH CALIF NUCLEAR SURVIV--ETC F/G 9/2
NONELECTRICAL LANGUAGES SIMULATION MODULE (NELSIM). VOLUME II.(U)
SEP 76 M EPSTEIN, J R PISTACCHI F29601-73-C-0024
UNCLASSIFIED AFWL-TR-73-256-VOL-2 NL

1 OF 2
AD
A033606



AFWL-TR-73-256, Vol. II

AFWL-TR-73-256 Vol. II

ADA033606

NONELECTRICAL LANGUAGES SIMULATION MODULE (NELSIM)

Volume II

TRW Systems Group
Nuclear Survivability Dept.
Redondo Beach, CA 90278

DDC
RECEIVED
DEC 20 1976
C

September 1976

Final Report

Approved for public release; distribution unlimited.

This research was sponsored by the Defense Nuclear Agency under Subtask Z99QAXTC022, Work Unit 52, Work Unit Title: Interface Program for Circuit Analysis.

V1
A033592

Prepared for
Director
DEFENSE NUCLEAR AGENCY
Washington, DC 20305

AIR FORCE WEAPONS LABORATORY
Air Force Systems Command
Kirtland Air Force Base, NM 87117

7405

This final report was prepared by TRW Systems Group, Nuclear Survivability Department, Redondo Beach, California, under Contract F29601-73-C-0024, Job Order WDNE1302, with the Air Force Weapons Laboratory, Kirtland Air Force Base, New Mexico. Mr. White (ELP) was the Laboratory Project Officer-in-Charge.

When US Government drawings, specifications, or other data are used for any purpose other than a definitely related Government procurement operation, the Government thereby incurs no responsibility nor any obligation whatsoever, and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Information Office (OI) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

A. Brent White

A. BRENT WHITE
Project Officer

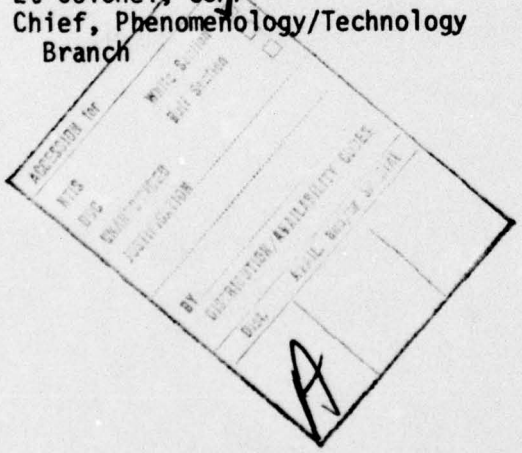
FOR THE COMMANDER

Larry W. Wood

LARRY W. WOOD
Lt Colonel, USAF
Chief, Phenomenology/Technology Branch

James L. Griggs Jr

JAMES L. GRIGGS
Colonel, USAF
Chief, Electronics Division



DO NOT RETURN THIS COPY. RETAIN OR DESTORY.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: AFWL-TR-73-256, Vol II	2. GOVT ACCESSION NO. / 3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): NONELECTRICAL LANGUAGES SIMULATION MODULE (NELSIM) Volume II	5. TYPE OF REPORT & PERIOD COVERED: Final Report
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): M. Epstein, J.R. Pistacchi	8. CONTRACT OR GRANT NUMBER(s): F29601-73-C-0024
9. PERFORMING ORGANIZATION NAME AND ADDRESS: TRW Systems Group, 1 Space Park, Redondo Beach, CA 90278	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 62704H, WDNE1302
11. CONTROLLING OFFICE NAME AND ADDRESS: Director, Defense Nuclear Agency, Washington, D.C. 20305	12. REPORT DATE: September 1976
	13. NUMBER OF PAGES: 106 (12) 100p.
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): Air Force Weapons laboratory (ELP), Kirtland Air Force Base, NM 87117	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

This research was sponsored by the Defense Nuclear Agency under Subtask Z99QAXTC022, Work Unit 52, Work Unit Title: Interface Program for Circuit Analysis.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Analysis
Computer Aided Analysis
CADA
Network Simulation
Non-electrical Languages for Simulation

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

NELSIM is a computer program developed to convert non-electrical system descriptions into electrical analogs or differential equations consistent with existing circuit analysis programs. The program provides an interface between system level problems and electrical simulation programs. The module permits engineers not familiar with disciplines generally associated with circuit simulation programs to solve system problems without manually deriving the electrical analogs and differential equations necessary to utilize such programs.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED 409 971

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

bpg

CONTENTS

SECTION		Page
I	INTRODUCTION	1
	1.1 NELSIM Input/Output Options	1
	1.2 NELSIM Data Organization	3
	1.3 NELSIM Usage Highlights	5
	1.4 Manual Organization	9
II	ANALOG SYSTEM INPUT LANGUAGE	10
	2.1 General Analog System Input Language	10
	2.2 Mechanical System Input Language	34
	2.3 Mechanical System State Variables	38
	2.4 Outputs	39
	2.5 Thermal System Input	41
	2.6 Electro-Mechanical Elements and Models	45
	2.7 Electro-Optical System Input	55
	2.8 Units and Scaling	57
III	TRANSFER FUNCTION INPUT LANGUAGE	65
	3.1 Flowgraph Input Preparation	66
	3.2 Graph Description	67
	3.3 Branches	67
	3.4 Input/Output Reference	68
IV	DIFFERENTIAL EQUATION INPUT LANGUAGE	71
V	COMBINED SYSTEM INPUTS	75
VI	NELSIM SAMPLE PROBLEMS	76
	6.1 Mechanical System Examples	76
	6.2 Thermal System Examples	80
	6.3 Electro-Mechanical Examples	85
	6.4 Electro-Optical System Examples	90

ILLUSTRATIONS

		Page
Figure 1.	Input/Output of NELSIM	2
Figure 2.	NELSIM Input Data Organization	4
Figure 3.	NELSIM Inputs	6
Figure 4.	Assumed Element Heat Flow Direction	12
Figure 5.	Temperature Source Polarity and Assumed Heat Flow	13
Figure 6.	Table Q1 Values as a Function of TC1	22
Figure 7.	Temperature Polarity and Heat Flow Direction	28
Figure 8.	Mutual Inductance Polarities	33
Figure 9.	Frictional Elements	37
Figure 10.	Mechanical Problem	39
Figure 11.	NELSIM Input for Mechanical System	40
Figure 12.	NELSIM Input for a Rotational Problem	41
Figure 13.	Heat Transfer Example	44
Figure 14.	Galvanometer Schematic Representation	45
Figure 15.	Lumped Element Galvanometer Equivalent	46
Figure 16.	Block Diagram of Rate Gyro	48
Figure 17.	NELSIM Equivalent Rate Gyro	50
Figure 18.	NELSIM Gyro Model	50
Figure 19.	Accelerometer Model	52
Figure 20.	NELSIM Accelerometer Equivalent	54
Figure 21.	NELSIM Accelerometer Model	55
Figure 22.	Equivalent Circuit of a Photodiode	56
Figure 23.	Graph Description Data Organization	66
Figure 24.	Prepared Flowgraph	66
Figure 25.	Unity Feedback System	68
Figure 26.	NELSIM Differential Equations Input	73
Figure 27.	Sample Mechanical Problem 1	76
Figure 28.	Sample Mechanical Problem 1 Representation to NELSIM	77
Figure 29.	Equivalent Electric Circuit for Mechanical System Problem 1	78
Figure 30.	Sample Mechanical Problem 2	78
Figure 31.	Thermal System Problem 1	80
Figure 32.	Thermal System Problem 2	83
Figure 33.	Electro-Mechanical Problem 1	85

ILLUSTRATIONS (Cont'd)

		Page
Figure 34.	Lumped Element Galvanometer Equivalent	85
Figure 35.	Galvanometer Electrical Analog	87
Figure 36.	Equivalent Circuit of a Photodiode	90

TABLES

		Page
Table I.	Standard Thermal Elements.	11
Table II.	Acceptable Constant Formats.	15
Table III.	Program Data Limits	34
Table IV.	Mechanical Standard Elements	35
Table V.	Mechanical to Electrical Analog Generation	36
Table VI.	Mechanical System State Variables.	38
Table VII.	Thermal Elements and State Variables	42
Table VIII.	System Analogies	62
Table IX.	Standard System Units.	62

SECTION I

INTRODUCTION

NELSIM (Nonelectrical Languages Simulation Module) is a digital computer program written in FORTRAN IV. The program translates non-electrical system descriptions into electrical analogs and differential equations acceptable for execution by the SCEPTRE (System for Circuit Evaluation and Prediction of Transient Radiation Effects) simulation program. The program thus provides an interface between system level problems and electrical simulation programs. The module allows engineers not familiar with disciplines generally associated with circuit simulation programs to solve system problems without manually deriving the electrical analogs and equations necessary to run such programs. The input language allows input of mechanical, thermal, electro-mechanical and electro-optical problems in terms of differential equations, transfer functions, or elements and units familiar to those particular disciplines.

The program features a free-field, user-oriented input language modeled after the SCEPTRE input language. The output is also in terms of the SCEPTRE language and allows direct execution by SCEPTRE. The output of NELSIM can be punched cards if requested by the user. Another option is scaling which allows the user to obtain an electrical analog with the units scaled to the actual response magnitudes expected.

1.1 NELSIM Input/Output Options

The program allows the three types of input to describe system problems illustrated in Figure 1.

The transfer function input allows users to enter the description of a system or part of a system in terms of signal flowgraphs or block

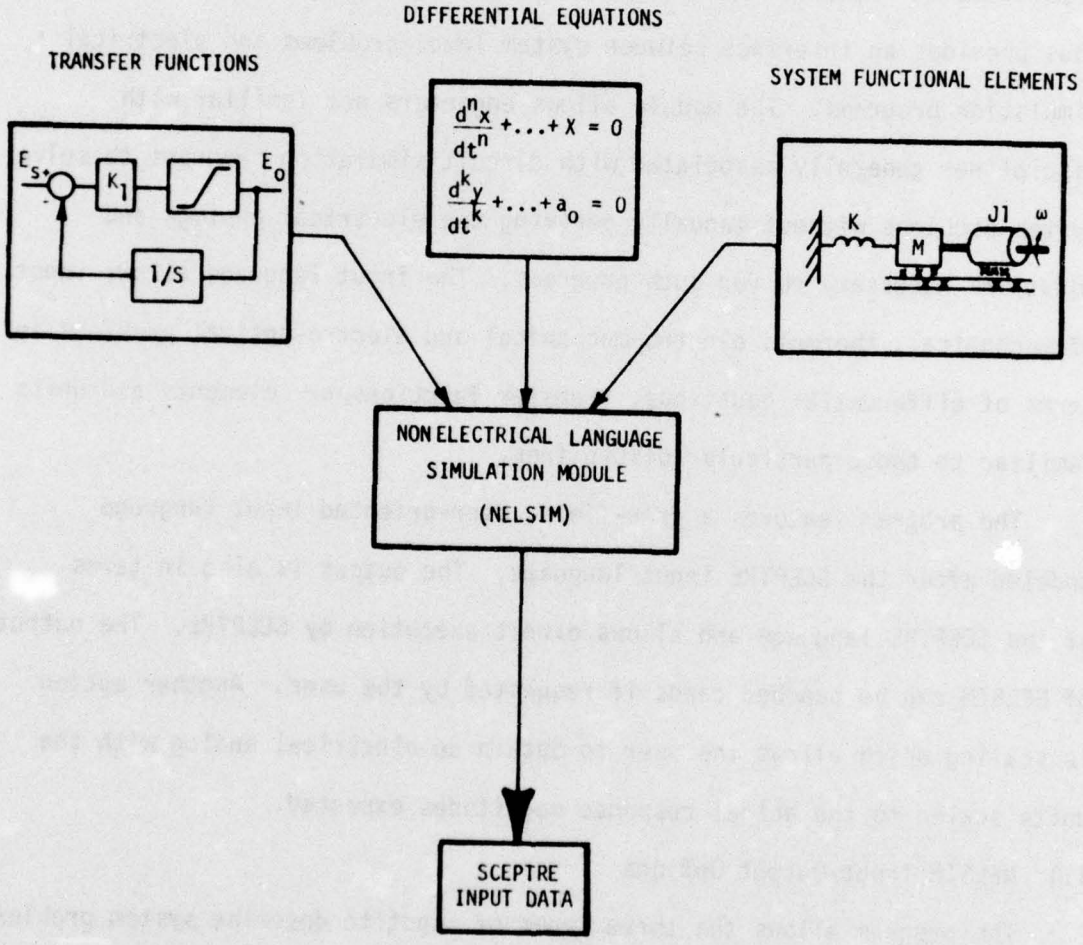


Figure 1. Input/Output of NELSIM

diagrams. The language for this option is nearly identical to the input language of the program SYNAP (Symbolic Network Analysis Program) developed for The Air Force Weapons Laboratory (AFWL) in 1972 and consists of descriptions of S-Domain transfer functions connected bwtween user derived node names.

The differential equations option allows the user to enter problem definitions in terms of sets of n^{th} order differential equations algebraically coupled together.

The user may also enter problems in terms of functional lumped elements and nodal connections. The program accepts elements of mechanical, thermal, electro-mechanical and electro-optical systems.

The output of the program consists of a set of first order differential equations for all differential equation entries and all transfer function entries. It also consists of electrical lumped element equivalent networks for all entires containing system elements. The user may use any option or combination of options to define the overall system to be analyzed. The user may have the output of NELSIM punched on cards and thus generate an input deck to the SCEPTRE program directly. The user may also call on the scaling algorithm built into NELSIM and obtain a listing of the analog generated with constant element values scaled to improve numerical accuracy upon execution.

1.2 NELSIM Input Data Organization

The data organization and input language of NELSIM have been kept identical to that of SCEPTRE wherever possible and consistent with it in all other cases. Figure 2 illustrates the data organization. As shown, the data are entered following heading and subheading cards. All problem descriptions entered in terms of lumped elements are entered under the

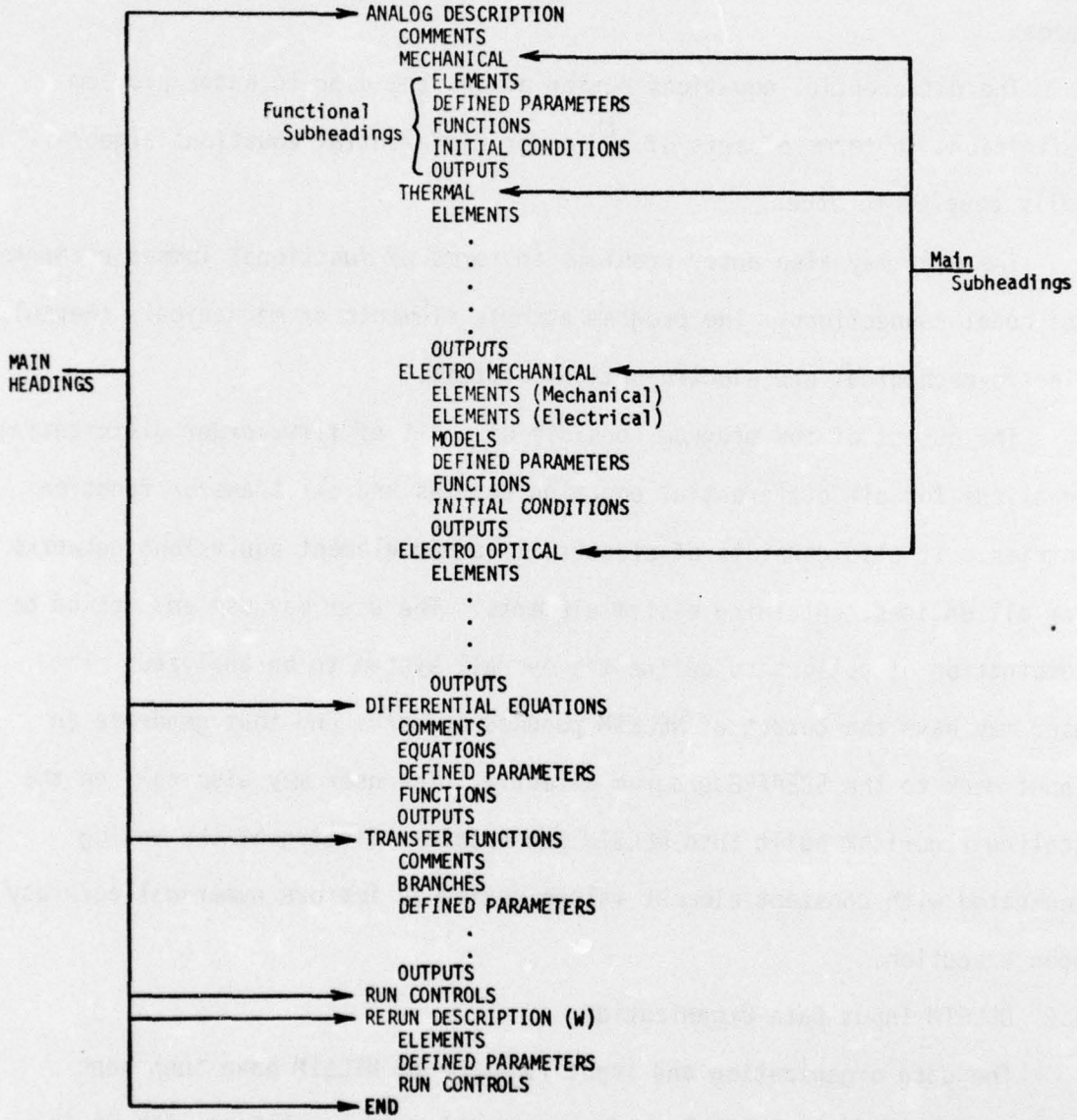


Figure 2. NELSIM Input Data Organization

ANALOG DESCRIPTION heading. All sets of differential equations are entered under the DIFFERENTIAL EQUATIONS heading and all transfer function descriptions are entered under the TRANSFER FUNCTIONS heading. The RUN CONTROLS, RERUN DESCRIPTION and END headings control the execution of the run, parameter value changes and program termination, respectively. The functional subheadings shown in Figure 2 identify the type of system being entered if lumped networks are entered. The functional elements serve to identify the remainder of variable values such as element values and description of equations used. The main headings must appear in the order shown in the figure. Main headings as well as subheadings however need not be included if not utilized. For example, if a problem is described strictly in terms of differential equations, the ANALOG DESCRIPTION headings, TRANSFER FUNCTION heading and all related subheadings may be deleted from the input. Sections II through IV elaborate the details of the input language as applicable to each input type.

1.3 NELSIM Usage Highlights

This paragraph provides the user a quick introduction to the input language by summarizing the input through three sample problem inputs. As discussed above three types of input are allowed in NELSIM. The input data organization for a differential equations problem, the input organization for a transfer function problem, and system element entry for a mechanical problem are shown in Figure 3.

The input data consists of descriptive statements constructed from user-derived component names, node names and value specifications. The information is placed on cards separated by special characters such as a comma, dash, parenthesis, or equal sign. In general, several complete statements can be punched on a card, separated only by a comma. The

Equations

$$K_1(V_2-V_1) + M_2\left(\frac{d^2V_2}{dt}\right) + B\left(\frac{dV_2}{dt} - \frac{dV_3}{dt}\right) + K_2(V_2-V_3) = 0$$

$$B\left(\frac{dV_3}{dt} - \frac{dV_2}{dt}\right) + K_2(V_3-V_2) + M_3\frac{d^2V_3}{dt} + K_3V_3 = 0$$

Parameter Values

$K_1 = 1$

$K_2 = 2$

$K_3 = 3$

$M_2 = .2$

$M_3 = .3$

$B = 1$

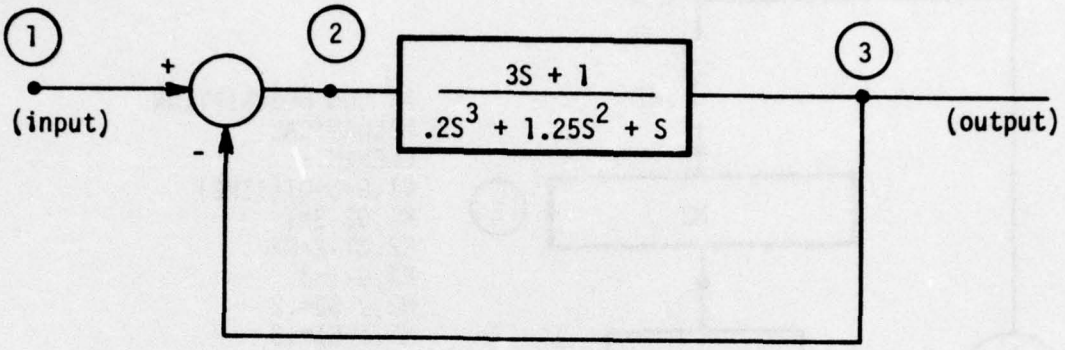
$V_1 = \cos 2t$

NELSIM Input

```
DIFFERENTIAL EQUATIONS
EQUATIONS
PK1*(PV2-PV1)+PM2*D2PV2+PB*(D1PV2-D1PV3)+PK2*(PV2-PV3)=0
PB*(D1PV3-D1PV2)+PK2*(PV3-PV2)+PM3*(D2PV3)+PK3*PV3=0
DEFINED PARAMETERS
PK1=1
PK2=2
PM2=.2
PM3=.3
PB=1
PV1=Q1(TIME)
FUNCTIONS
Q1(T)=(COS(2.*T))
OUTPUTS
PV2(V2),PV3(V3),PLOT
RUN CONTROLS
STOP TIME=100
INTEGRATION ROUTINE=IMPLICIT
END
```

a. Differential Equations

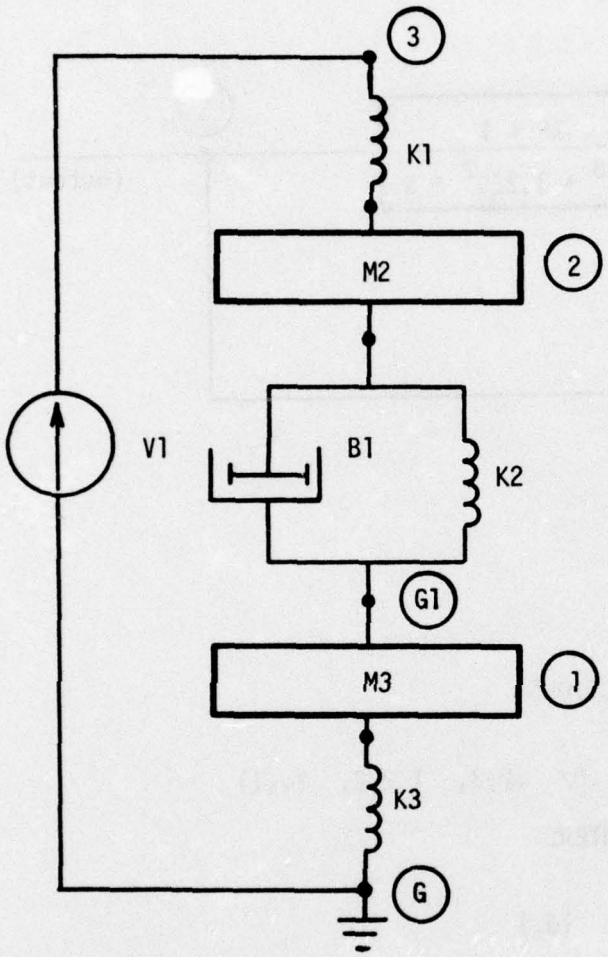
Figure 3. NELSIM Inputs




```
                TRANSFER FUNCTIONS
                BRANCHES
                1-2(1.,0)
                2-3(3.,1,  1.,0/  .2,3,  1.2,2,  1.,1)
                3-2(-1.,0)
                DEFINED PARAMTERS
Input at ---->  P1=T1
Node 1          FUNCTIONS
                T1=0,0,  0,1,  10,1
                OUTPUTS
                P1,P3,PLOT <---------- Output requests at
                RUN CONTROLS           Nodes 1 and 3
                STOP TIME=10
                INTEGRATION ROUTINE = IMPLICIT
                END
```

b. Transfer Function

Figure 3. NELSIM Inputs (Cont'd)




```
ANALOG DESCRIPTION
MECHANICAL
ELEMENTS
V1,G-3=Q1(TIME)
K1,G2-3=1
K2,G1-2=P2
K3,G-1=3
M2,2-G2=.2
M3,1-G1=.3
B1,G1-2=1
FUNCTIONS
Q1(T)=(COS(2.*T))
DEFINED PARAMETERS
P2=2
OUTPUTS
VM2(V2),VM3(V3),PLOT
RUN CONTROLS
INTEGRATION ROUTINE=IMPLICIT
STOP TIME=10
END
```

c. Mechanical Analog

Figure 3. NELSIM Inputs (Cont'd)

information is all placed under major heading and subheading such as TRANSFER FUNCTIONS and DEFINED PARAMETERS.

1.4 Manual Organization

This section has briefly described the capabilities of the NELSIM program and its general usage, and has given some examples of the input language. The remainder of the manual expands on the topics discussed above. Section II discusses the input language applicable to non-electrical system topologies. Sections III and IV describe the input language applicable to differential equations and transfer functions, respectively. The final section, Section V, contains several example problems varying in complexity and mode of analysis. These examples have been included to help the reader comprehend the wide usage range of NELSIM.

SECTION II

ANALOG SYSTEM INPUT LANGUAGE

As discussed in Section I, NELSIM allows the input of nonelectrical systems in terms of networks made up of lumped elements normally utilized within a particular discipline. The disciplines allowed are mechanical, thermal, electromechanical and electro-optical. The general input language for the analog description is discussed below followed by the documentation of each individual discipline.

2.1 GENERAL ANALOG SYSTEM INPUT LANGUAGE

All problems to be entered utilizing lumped elements must be entered using a nodal topology description. The general features of the network input language are explained using thermal quantities and elements. The entries covered below are applicable to all the systems disciplines allowed. Section 2.2 through 2.5 describe the input of mechanical, thermal, electromechanical and electro-optical systems, respectively.

2.1.1 Network Preparation

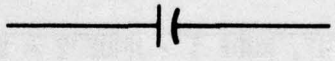
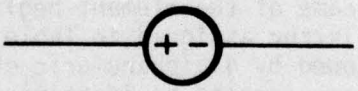
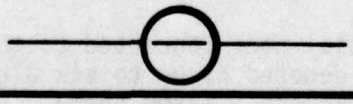
For a portion of a system described in terms of thermal elements, the network topology is accepted by NELSIM in terms of the standard elements shown in Table I. Each element may be linear or nonlinear. The six steps to be followed in preparing a network for a NELSIM run are summarized below:

- Draw an equivalent network comprised of thermal resistors, thermal capacitors, and temperature and heat flow sources
- Assign a name or number to all nodes in the circuit
- Give a name to each circuit element (beginning with R, C, T, Q)

- Assume arbitrary heat flow directions in each passive circuit element
- Indicate the direction of positive heat flow in each temperature and heat source
- Choose and record circuit values in a consistent set of parameter units.

A node is defined as the point of common potential (temperature) at the junction created by the connection of two or more network elements. Each node is given an arbitrary name consisting of six alphanumeric characters or less. Element names are chosen arbitrarily but must begin with the letter specified in Table I and must not contain more than 5 characters.

TABLE I. STANDARD THERMAL ELEMENTS

ELEMENT	SCHEMATIC REPRESENTATION	PROGRAM NOTATION
Resistor		R
Thermal Capacitor		C
Temperature Source		T
Heat Flow Source		Q

2.1.2 Element Input Description and Assumed Polarity

In preparing the circuit a heat flow direction and temperature difference are assumed for each element as shown in Figure 4.

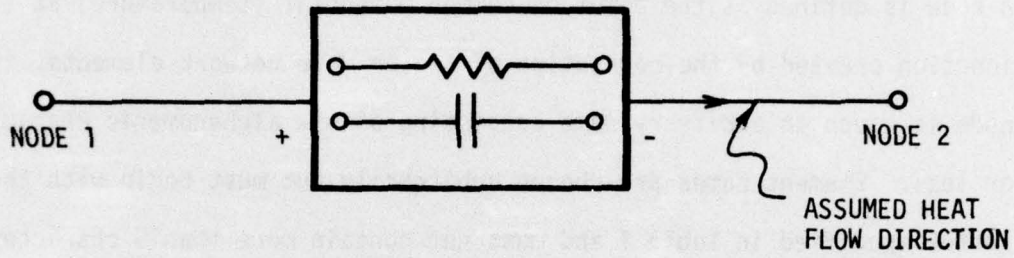


Figure 4. Assumed Element Heat Flow Direction

The elements are input into the program following the subheading card ELEMENTS using the format:

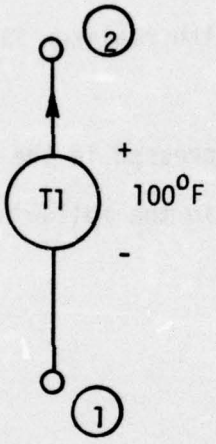
/ELEMENT, NODE 1 - NODE 2 = VALUE

where the heat flow direction is assumed to be from Node 1 to Node 2 and the parameters are defined as follows:

- ELEMENT - The name of the element beginning with the letter assigned in Table I and followed by 4 alphanumeric characters or less serving as identifiers. i.e., (R12, TIN).
- NODE 1 - Initial node connected to the element and denoted by up to six alphanumeric characters. (From node)
- NODE 2 - Final node connected to the element and denoted by up to six alphanumeric characters. (To node)

- VALUE - Value of the element. The form of VALUE may be expressed in six different formats to allow for linear and non-linear values of elements. The six formats available are discussed in Section 2.1.4.

The information may be placed anywhere on the card between columns 1 and 72 with the delimiters (comma, dash, equal sign) placed in the order shown. More than one element may be included on any card if separated by commas.



The heat in a source is assumed to flow from the negative to positive pole and VALUE is positive if the nodal connection is entered with the "FROM" node denoting the negative terminal. The source in Figure 5 for example can be entered in the two ways shown below.

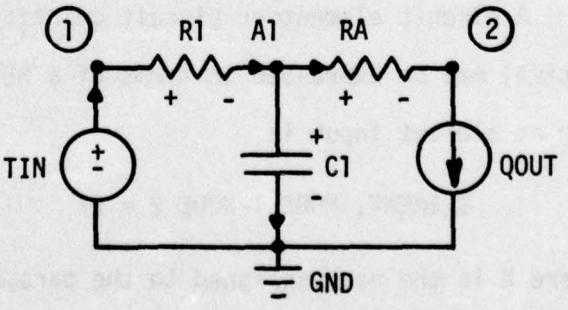
```
/ T1, 1-2 = 100
```

```
/ T1, 2-1 = -100
```

Figure 5. Temperature Source Polarity and Assumed Heat Flow

Example

The element entry into the program for the simple schematic shown is given below:




```
/ TIN, GND-1  = VALUE
/ R1, 1-A1    = VALUE
/ C1, A1-GND  = VALUE
/ RA, A1-2    = VALUE
/ QOUT, 2-GND = VALUE
```

An equally valid entry is:

```
/TIN, GND-1 =  VALUE, R1, 1-A1 =  VALUE
/C1, A1-GND =  VALUE
/RA, A1-2   =  VALUE, QOUT, 2-GND = VALUE
```

2.1.3 Element Temperatures, Heat Flows

The temperature or heat flow associated with any element is referred to by placing a T or an Q before the element name. TR1 and QR1 therefore refer to the temperature and heat flow associated with resistor R1.

2.1.4 Element and Circuit Value Formats

An element value or circuit quantity may be expressed in the six formats listed below. Each is explained in detail in the following paragraphs.

- Defined Parameter
- Constant
- Table
- Equation
- FORTRAN Function Subroutine
- Expressions

2.1.4.1 Quantities Expressed as Defined Parameters

A circuit element or circuit quantity (temperature, heat flow, derivative) may be expressed in terms of a DEFINED PARAMETER. The format for an element input is

```
ELEMENT, NODE 1-NODE 2 = PX
```

Where X is the name assigned to the parameter and may contain up to five alphanumeric characters. The prefix, P, must always proceed the name X as follows:

/R1, 1-2 = P12

/RSHNT, NOUT-GND = POUT

The actual values of defined parameters used are defined under the subheading DEFINED PARAMETERS. This input and its related formats are discussed in paragraph 2.1.5. A defined parameter may be expressed in terms of a constant, table, equation, function subroutine or another defined parameter.

2.1.4.2 Quantities Expressed as Constant

An element value, circuit quantity or defined parameter may be expressed as a numerical constant written in either integer or decimal form with or without an exponent. Up to 13 characters may be used to represent the constant. Table II summarizes the constant forms acceptable to the program. Sample element description cards using the constant notation are illustrated below.

/R1, 1-2 = 10

/R2, 3-4 = 1.E-3

/C6, GND-1 = E-12

Table II. Acceptable Constant Formats

Format*	Example	Value
±XX	10	10.
±XX.	-10.	-10.
±X.X	-10.2	-10.2
±X.XE±Y	+10.2E-6	10.2×10^{-6}
±XXE±Y	-10E-4	$-10.X10^{-4}$

2.1.4.3 Quantities Expressed as TABLES

A variable circuit quantity or defined parameter may be expressed in tabular form. The table format is specified as follows for an element:

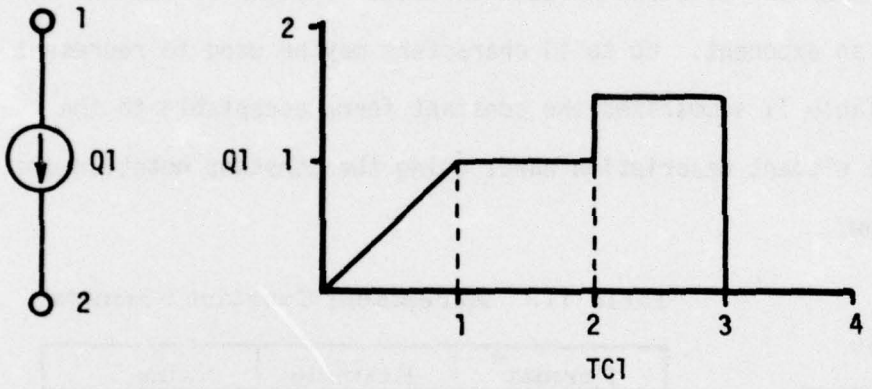
/ELEMENT, NODE 1 - NODE 2 = TABLE X (VARIABLE)

or

/ELEMENT, NODE 1 - NODE 2 = TX (VARIABLE)

where X is the table name containing up to five alphanumeric characters and the variable in parenthesis is the independent variable. The independent variable may be any circuit variable such as voltage, current, elements and defined parameters.

Source Q1 is a function of the temperature across capacitor C1 as shown in the table



Q1	TC1
0.	0
1.	1
1.	2
1.5	2
1.5	3
0.	3
0.	4

Q1 could be entered as

/Q1, 1-2 = TABLE 1(TC1)

or

/Q1, 1-2 = T1(TC1)

The tabular data shown in the table is entered as input pairs under the subheading FUNCTIONS and its format is discussed in Section 2.1.4.6. If the parentheses and variable are left off the table description, TIME is assumed to be the independent variable. Sample table entries follow.

/R12A, A1-A2 = TABLE T1(QR1)

/RA14, 8-12 = T3(TC3)

/TIN, GND-1 = TTIN ← Time assumed as independent variable

/Q1, 2-3 = T14(P3)

The occasion often arises that the same table data is utilized by more than one element or parameter. A common situation occurs when two current generators in the network are defined by the same tabular data, differing only in the independent variable. In this case the user may define both generators under ELEMENT as

/Q1, 1-2 = TABLE 1(TC1)

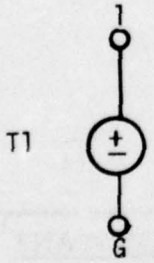
/Q2, 7-8 = TABLE 1(TC2)

2.1.4.4 *Quantities Expressed as Expressions*

A circuit quality, element or defined parameter may be expressed in terms of an expression. The element input format for this form is:

/ELEMENT, NODE1-NODE2 = XN (NUMERICAL EXPRESSION)

where N is the expression denotation containing up to five alphanumeric characters. An example is given below for source T1



/T1,G-1 = X1 (10.*COS(2.*3.14*PF)

T1 = 10.COS 2πf where PF = f

2.1.4.5 Quantities Expressed as EQUATIONS

A circuit quantity, element or defined parameter may be expressed in terms of an equation. The element input format for this form is:

/ELEMENT, NODE 1-NODE 2 = EQUATION X(a_1, a_2, . . . a_n)

or

/ELEMENT, NODE 1-NODE 2 = QX(a_1, a_2, . . . a_n)

where X is the name containing up to 5 alphanumeric characters given to the equation. The variable a_1 through a_n are described in the equation and may be any circuit quantity, number, defined parameter or table.

Example

Resistor R3 between nodes 1 and 2 is defined as:

R3 = f(QR3, TC1, P4)

where f(QR3, TC1, P4) = (QR3 - TC1)*P4.

The element input entry could be as follows:

/R3, 1-2 = EQUATION 1A(QR3, TC1, P4)

OR

/L3, 1-2 = Q1A(QR3, TC1, P4)

The actual mathematical expression f(QR3, TC1, P4) = (QR3 - TC1)*P4 must be defined under the subheading card FUNCTIONS. The details of this input are discussed in Section 2.9. More sample inputs are provided below.

/C2, 6-1 = EQUATIONS 2(.5, 120., TABLE 4(TC2))

/R17, 1-12 = Q3(TR3, TR2, 12., T1(QR2))

NOTE:	DECIMAL POINTS MUST BE SUPPLIED FOR ALL CONSTANT QUANTITIES ENTERED IN THE ARGUMENT LIST a_1 THROUGH a_n.

2.1.4.6 Quantities Expressed as FORTRAN FUNCTION SUBROUTINES

The user knowledgeable of FORTRAN may enter circuit elements, quantities and defined parameters as separate FORTRAN function subroutines. This feature is extremely useful when a quantity is too complex to be expressed as an equation. The element input format for this form is

/ELEMENT NAME, NODE 1-NODE 2 = FUNCTION X(a_1, a_2, . . . , a_n)

or simply

/ELEMENT NAME, NODE 1 - NODE 2 = FX(a_1, a_2, . . . , a_n)

where X or FX is the name of the subroutine and contains up to five alphanumeric characters and a_1 through a_n are the variables utilized in the function subroutine. These variables may be any circuit quantity, element or defined parameter. The subprogram name must be unique and begin with the letter F if the short form above is used.

Example

Capacitor C1 between nodes 1 and 2 is defined as follows:

C1 = f(TIN, TC3) for QC1 1E-3

C1 = f(TOUT, TC3) for QC1 1E-3

where $f(TIN, QC3) = \frac{TIN-TC3}{1E-6}$

and $f(TOUT, TC3) = \frac{TOUT-TC3}{1E-6}$

Using the subroutine input option the entry is as follows:

/C1, 1-2 = FUNCTION FCAP(TIN, TC3, TOUT, QC1)

or

/C1, 1-2 = FCAP(TIN, TC3, TOUT, QC1)

The function subroutine shown below must then be entered as part of the input deck. The variables A, B, C and D appearing in the parenthesis correspond to TIN, TC3, TOUT and QC1, respectively.

```
FUNCTION FCAP (A, B, C, D)
IF (D.GT.1.E-3) GO TO 10
FCAP = (A-B)/1.E-6
RETURN
10 FCAP = (C-B)/1E-6
RETURN
END
```

2.1.5 Defined Parameters Input Format

Every DEFINED PARAMETER utilized in the element input section must be defined. The parameter definition entries must be proceeded by the subheading card

/DEFINED PARAMETERS

The format utilized to define each defined parameter used is simply

/PX = VALUE

where X is up to five alphanumeric characters naming the parameter and VALUE can have the same formats used to define elements except for special values. Defined parameter values can therefore be written in the form of

- Constants
- Defined Parameters
- Tables
- Equations
- FORTRAN Function Subroutines
- Expressions

The formats for each have been explained in detail in paragraphs 2.1.4.1 through 2.1.4.7 and remain the same. It will, therefore, suffice here to

give some samples of possible entries under the DEFINED PARAMETERS card in the input deck.

4

```
/DEFINED PARAMETERS
/P1 = 3.E-12                          ← Constant Form
/POUT = TABLE A4(TR3)                 ← Tabular Form
/P3 = P4                              ← Defined Parameter Form
/P4 = Q3(TR2, QR1, 10.)               ← Equation Form
/P5 = FCAP(TC1, TC2, TIME)            ← FORTRAN Function
                                        Subroutine Form
/P6 = EQUATION 5(10., 20., P4, P5)    ← Equation Form
/P7 = X1 (P1 * P2)                    ← Expression Form
```

2.1.6 Functions

The FUNCTIONS portion of the input is used to define all TABLES and EQUATIONS called for under the ELEMENTS and DEFINED PARAMETERS subheadings. The input cards for tables and equations are preceded by the card

```
/FUNCTIONS
```

Their input formats are discussed in the next two paragraphs.

2.1.6.1 Table Input Format

Every TABLE referred to under ELEMENTS or DEFINED PARAMETERS must be explicitly defined under the FUNCTIONS subheading. The tabular data are entered as coordinate pairs separated by commas preceded by the table name. The independent variable must be the first coordinate and the dependent variable the second. If the data require continuation onto another card, the card must end with a dependent variable value. The next card then

begins with the next coordinate pair.

The data pair points must be supplied such that the independent variable is in increasing algebraic order. No specific limit to the number of point pairs that may constitute any table is defined. Only single value functions are allowed. It is permissible to supply two consecutive independent-variable values that are equal, but which have different dependent variable values. This may be done to produce step functions. The example below serves to illustrate various acceptable forms of table input.

Example

Referring to a previous example, the card

```
/Q1, 1-2 = TABLE 1(TC1)
```

was entered under ELEMENTS. The function is redrawn here and several valid inputs shown in Figure 6.

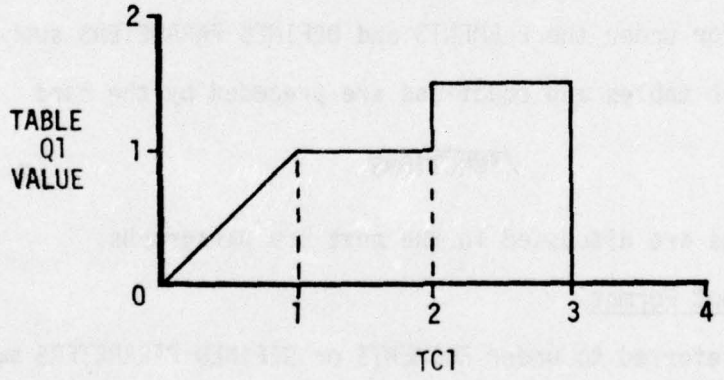


Figure 6. Table T1 Values as a Function of TC1

```
/TABLE T
```

/-1, 0

/0, 0, 1, 1, 2, 1, 2, 1.5, 3, 1.5

/3, 0, 4, 0

or

/T1 = -1, 0, 0, 0, 1, 1, 2, 1, 2, 1.5, 3, 1.5, 3, 0, 4, 0

or

/T1

/-1, 0, 0, 0, 1, 1, 2, 1

/2, 1.5, 3, 1.5, 3, 0, 4, 0

or

/TABLE 1

/-1, 0

/0, 0

/1, 1

/2, 1

/2, 1.5

/3, 1.5

/3, 0

/4, 0

When the table values are updated at each solution time step, linear interpolation is used between the points supplied. For independent variable values falling outside the range of values supplied in a table, linear

extrapolation is performed to determine the correct table value, and therefore, proper termination may be necessary. For example, in Figure 6, for any value of TIME in excess of 3 the dependent variable Q1 will be assigned the value zero because of the use of the point pair 4.0. In situations where the same table serves to define more than one quantity, the user need only explicitly define the table once. For example, if the two heat sources Q1 and Q2 were input under ELEMENTS as

/Q1, 1-2 = TABLE 1 (TC1)

/Q2, 7-8 = TABLE 1 (TC2)

only one TABLE I need be defined under FUNCTIONS.

2.1.6.2 Equation Input Format

Each unique equation referred to under ELEMENTS or DEFINED PARAMETERS must be explicitly defined under the FUNCTIONS subheading in terms of the mathematical expression it represents. The format utilized to enter the equation definition is as follows:

/EQUATION $X(b_1, b_2, \ldots b_n)$ = (MATHEMATICAL EXPRESSION)

/$QX(b_1, b_2, \ldots, b_n)$ = (MATHEMATICAL EXPRESSION)

X is the alphanumeric name given to the equation when referred to under ELEMENTS or DEFINED PARAMETERS. b_1 through b_n represents a dummy variable list each containing up to six alphanumeric characters and representing the variables called out in the argument list under ELEMENTS or DEFINED PARAMETERS. One restriction is that the DUMMY VARIABLE NAMES MUST NOT BEGIN WITH THE LETTERS I THROUGH N INCLUSIVE. The mathematical expression must appear on the card inside parentheses and be expressed in terms of the dummy variables b_1 through b_n.

The mathematical expression may be any combination of allowable operations, functions, or variables. The following mathematical operations and corresponding symbols are allowed.

OPERATION	SYMBOL
Exponentiation	**
Multiplication	*
Division	/
Addition	+
Subtraction	-

The order in which operations are performed is indicated by the order in which the operators are listed. The use of parentheses, to denote clearly the intended mathematical combination, is suggested to avoid ambiguity. For example, X + Y * Z should be written as (X + Y) * Z if X + (Y * Z) is not intended. All standard FORTRAN functions may be used, a few of the most commonly used are listed below.

FUNCTION	SYMBOL
Square root	SQRT (argument)
Sine Function	SIN (argument in radians)
Cosine Function	COS (argument in radians)
Exponential Function	EXP (argument)
Arctangent Function	ATNA (argument in radians)
Hyperbolic Tangent Function	TANH (argument in radians)
Natural Logarithm Function	ALOG (argument)
Common Logarithm Function	ALOG10 (argument)

The argument of any function may be any allowable mathematical expression. Mathematical expressions are not limited to 72 characters (one card) and may be continued on subsequent cards, using as many as necessary. In addition, the user may supply subprogram functions that he has written himself (see Section 2.1.4.7).

Example

Referring to a previous Example, the resistor R3 between nodes 1 and 2 was entered under ELEMENTS as:

```
/R3, 1-2 = Q1A(QR3, TC1, P4)
```

where f(QR3, TC1, P4) = (QR3 - TC1)*P4.

The definition of equation Q1A is entered under the subheading FUNCTIONS as:

```
/Q1A(A, B, C) = ((A-B) * C))
```

where A, B and C correspond to QR3, TC1 and P4 respectively.

Example

The defined parameter P1 = TC1 + QR2 - TABLE 2 (TC1) is entered under DEFINED PARAMETERS as

```
/P1 = Q2A(TC1, QR2, TABLE 2(TC1))
```

The equation Q2A is in turn defined under FUNCTIONS as

```
/Q2A(A, B, C) = (A + B - C)
```

At each solution step, the ordinate value of Table II would replace the dummy variable C and the computation (A + B - C) carried out.

When more than one element or defined parameter can be defined by the same equation, only one equation need be defined under FUNCTIONS. For example, if the two sources T1 and T2 are defined as:

```
T1 = (10*QC1 - TR2)/TIME

T2 = (10*QR4 - TC12)/TIME
```

The two sources could have been described under ELEMENTS as

```
/T1, 1-2 = Q1(10., QC1, TR2, TIME)
```

```
/T2, B-14 = Q1(10., QR4, TC12, TIME)
```

in which case the definition of Q1 need only appear once under FUNCTIONS as:

```
/Q1(X, Y, Z, T) = ((X * Y-Z)/T)
```

other equation definition inputs are shown below:

```
/Q12(XC1, P12, THETA) = (XC1 - P12 * COS (THETA))
```

```
/EQUATION 43(EIN, ZIN) = (EIN/ZIN)
```

```
/QT16(A, B, C) = (A ** B - C/12.E-4)
```

2.1.7 Initial Condition Input

Unless initial conditions for state variables are entered as input data, all voltage and current values are assumed to be zero at the first solution. Initial conditions other than zero often exist and must be input. For example, in AC analysis, initial conditions are needed by the program to linearize all non-linear functions about their operating points. Initial conditions follow the subheading card

```
/INITIAL CONDITIONS
```

and are entered in the forms

```
/TX = NUMBER
```

```
/QX = NUMBER
```

where TX and QX represent the temperature and the heat flow associated with element X. Number represents the value of the temperature or heat flow and must be a constant of the form shown in Table II.

If all initial conditions are zero, neither the subheading card nor the data are required. Care must be taken to establish the proper

polarities for initial conditions. Also, the initial capacitor temperatures are positive when they are consistent with the assumed temperature polarity for the capacitor. Proper statements are illustrated in Figure 7.

The assumed heat flow direction through capacitor C11 of Figure 7 is from node 4 to node 5. If the initial heat flow is in this direction, it is entered under the INITIAL CONDITIONS subheading as a positive quantity. If, however, the heat flows in the opposite direction, it is preceded by a negative sign. If the actual initial temperature polarity agrees with the arbitrarily chosen reference direction, that initial temperature is entered as positive; if not, it is entered as negative.

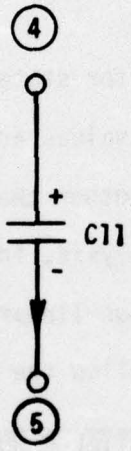


Figure 7. Temperature Polarity and Heat Flow Direction

2.1.8 Outputs Requests

Any variable used in the program may be requested as an output variable under the subheading

/OUTPUTS

The variables listed under OUTPUTS will appear in a printed listing as a function of time. The output variables may be any of the following:

- Temperature or heat flow associated with any element such as TC1, QC1
- Temperature or heat flow associated with any source such as T1, QE1, Q2, TJ2
- Any element value such as C17, R4
- Any DEFINED PARAMETER such as P13, POUT
- Any INTERNAL parameter. These are parameters within the program itself.

A typical output request is shown below:

/OUTPUTS

/TC1, QR1, R3, TJ4

/QX14, P16

The variables may appear anywhere on the card between column 1 and 72 separated by commas. As many variables as desired may appear on one card and as many cards as desired may be used. If more than one card is used, however, the last variable on any card must be followed by blanks.

The output is plotted if the following format is used:

/n_1, n_2, . . . n_n, PLOT

where n_1 through n_n are desired to be plotted against time. The output requests below would result in

/TRY, TC1, TSUP, PLOT

/QC8

a listing and plotting of TRY, TC1 and TSUP. The fourth quantity, QC8 would be output in printed form only. If the word PLOT is used, no other dependent variables may follow it on that card.

If a different independent variable is desired, the following format must be used:

/QC14, PLOT(TC14)

In this case the heat flow through capacitor C14 would be printed as well as plotted as a function of the temperature across it.

Additional flexibility is available to permit the user to attach a different label to any output quantity. Consider that elements C1 and R7 exist in a given network and that the temperature across both (TC1 and TR7) are of interest. If the user desires to rename them as TIN and TOUT, he can request the outputs as TC1(TIN), TR7(TOUT), PLOT. The plot, rename, and choice of independent variable options can be presented in terms of the general format as:

YQTY(YLABEL), ... , PLOT(XQTY(XLABEL))

If no rename is desired, the general format reduces to

/YQTY, ... , PLOT(XQTY)

If, in addition, only time is desired as the independent variable, this can be further reduced to

YQTY, ... , PLOT

If no plotted information is desired, the simplest form arises as

YQTY, ...

2.1.9 Run Control Input Data

The RUN CONTROLS subheading card is followed by instructions specifying the type of analysis to be performed and the general parameters to control the run. These parameters include such information as the type integration scheme to be used, time limits set on the run, etc. For all the Run Control input data available, the user is referred to a manual of

the SCEPTRE program. These cards are not processed by NELSIM as they apply strictly to the execution of the problem on SCEPTRE upon NELSIM output. A few of these commands are illustrated below.

2.1.9.1 Stop Time

All transient runs must have a problem duration in terms of time units calculated or assumed. This is the value indicating how far out in time the solutions are to be calculated. This entry is of the form:

/STOP TIME = NUMBER

where NUMBER is a constant in time units chosen.

2.1.9.2 Maximum Integration Passes

Another type of limit is available that operates on the number of passes made by the integration routine. In the absence of any instruction from the user, the program imposes a limit of 20,000 passes. The user may change this limit by entering

/MAX INTEGRATION PASSES = NUMBER

2.1.9.3 Start Time

Most transient analysis problems begin at time zero and for this case no entry is needed. If, however, the user desires to begin the run at other than time = 0, the following card is necessary

/START TIME = NUMBER

where NUMBER is a constant value in units of time assigned to the problem.

2.1.9.4 Minimum Step Size

This quantity is designed to terminate the run automatically whenever a time solution increment is required that is smaller than the minimum increment. If this quantity is not supplied, the program will automatically compute a minimum limit equal to 1×10^{-9} times the problem

duration (STOP TIME). If a specific minimum step size independent of the problem duration is desired, the format is

/MINIMUM STEP SIZE = NUMBER

For example if

STOP TIME = 5

is specified, the minimum size would automatically be taken to be 5×10^{-9}. If the user is not satisfied with this minimum step-size and desires to change it to 1×10^{-5} he may do so using the following entry

/MINIMUM STEP SIZE = 1E -5

2.1.9.5 Punched Output

Nominally, the output from NELSIM is in printed form and punched card output is suppressed. To obtain SCEPTRE input decks (a copy of the printed output), the punch option must be turned on by inserting a card in the run controls input data as,

/PUNCHED OUTPUT

2.1.10 Mutual Inductance Input Description

Although there is no thermal analogy for an inductor, inductors will appear in other simulations such as the representation of an electro-mechanical system. Additionally, the concept of mutual inductance can appear in a simulation such as for a transformer.

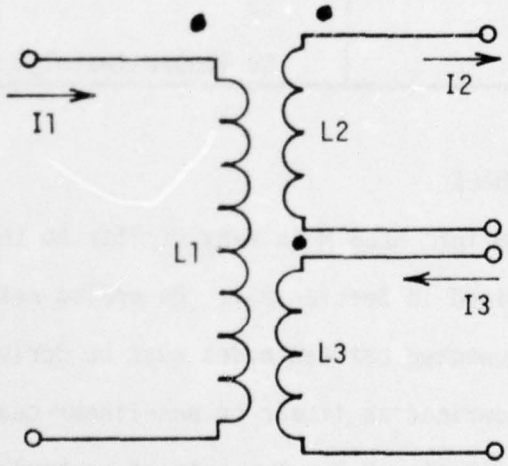
Mutual inductance is also entered under the ELEMENTS subheading but has a slightly different format. If coupling exists between inductors L1 and L2, the appropriate entry must include these elements in place of the node identification such as

/MX, L1-L2 = VALUE

In addition, a physical limitation of the principle of mutual inductance must be observed in order to have a physically realizable circuit. That is, since coefficient of coupling, k, is always less than unity and by definition

$$K = \frac{M}{\sqrt{L1\ L2}} < 1$$

the user should be certain that $M < \sqrt{L1\ L2}$. Stated in words, the mutual inductance between any two inductors must be less than the square root of the product of the self-inductances of the components between which the mutual inductance exists. The sign of M is positive if in a given winding the induced voltage of mutual inductance acts in the same direction as the induced voltage of self-inductance. If the induced voltage of mutual inductance opposes the induced voltage of self-inductance in a given winding, M is negative. The proper sign for M for the assumed current directions is illustrated in Figure 8.



MUTUAL INDUCTANCE	SIGN
M12	-
M13	+
M23	-

Figure 8. Mutual Inductance Polarities

2.1.11 NELSIM Program Limits

Table III gives the program data limits which have been established in the program. These limits are considered adequate for most normal circuit analysis work. Approximate limits are also given for the number of TABLE and EQUATION functions.

Table III. Program Data Limits

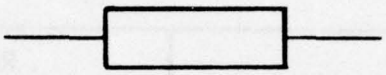
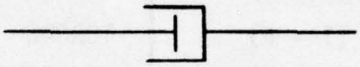
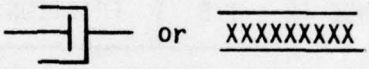
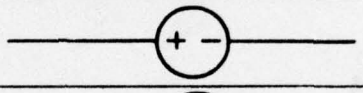
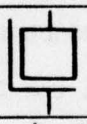
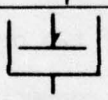
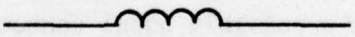
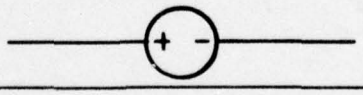
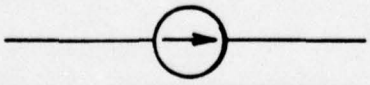
Description of Data	Maximum Number
Elements	300
Nodes	301
Defined Parameters	100
Defined Parameter Differential Equations	100
Mutual Inductances	50
Arguments in Equation Value Specification	50
Output Requests	100
Supplied Initial Conditions	300
Equation Functions (1 equation per card)	80 (approximately)
Cards per Equation Function	20
Table Functions	80 (approximately)

2.2 MECHANICAL SYSTEM INPUT LANGUAGE

Entry of a mechanical problem into NELSIM is very similar to the entry of an electrical network as explained in Section 2.1. An analog network made up of mechanical elements connected between nodes must be derived by the user. These elements may be defined as linear or non-linear quantities using the six formats outlined in Section 2.1. Two sets of mechanical elements are allowed as input for mechanical problems, one for translational systems, the other for rotational systems. The elements, their schematic

representation, as well as program notation, are shown in Table IV.

TABLE IV. MECHANICAL STANDARD ELEMENTS

	Element	Schematic Representation	Program Notation
Translational	Mass		M
	Dashpot		D
	Spring Stiffness		K
	Friction		B
Rotational	Force		F
	Velocity		V
	Moment of Inertia		J
	Friction		B
	Rotational Stiffness		K
	Torque		T
	Angular Velocity		W

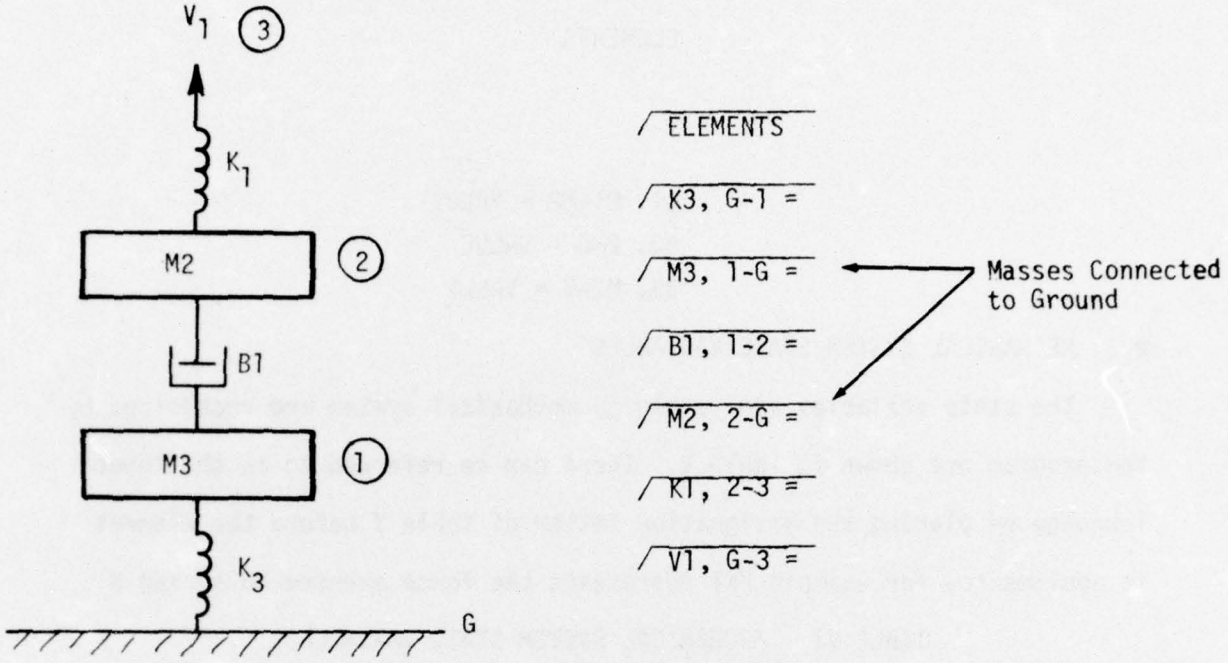
The elements must be entered in the format shown in Section 2.1 and begin with the letters indicated in Table IV. Table V shows the electrical analogs that are generated from each mechanical element and the numerical translation.

TABLE V. MECHANICAL TO ELECTRICAL ANALOG GENERATION

Translational			Rotational		
FORCE	=>	CURRENT (I = F)	TORQUE	=>	CURRENT (I = T)
VELOCITY	=>	VOLTAGE (V = V)	VELOCITY	=>	VOLTAGE (V = W)
MASS	=>	CAPACITANCE (C = M)	MOMENT OF I	=>	CAPACITANCE (C = J)
SPRING	=>	INDUCTOR (L = 1/K)	SPRING	=>	INDUCTANCE (L = 1/K)
FRICTION	=>	RESISTANCE (R = 1/B	FRICTION	=>	RESISTANCE (R = 1/B)

2.2.1 Mass Connections

The node names are user derived in general. For mass connections, however, a restriction is placed on the user. All masses must be connected to a ground node beginning with the letter G. Further, all masses must appear at node points only. If two or more masses are connected to different points, each must be connected to the common ground node. The following system shows a possible nodal connection for a 2 mass system.



2.2.2 Friction Connections

Friction can occur as the resistive force between two masses or be represented as a dashpot (Figure 9). Two types of nodal connections are, therefore allowed for a friction input, one between masses and one between nodes.

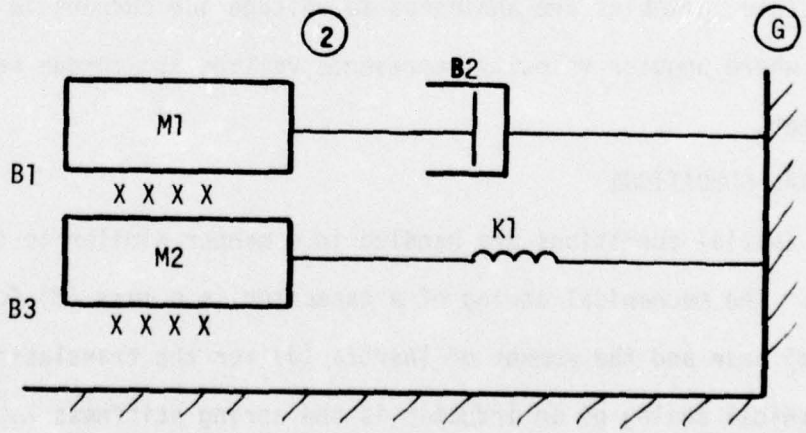


Figure 9. Frictional Elements

The frictional element inputs for Figure 9 are shown below:

```
ELEMENTS
.
.
.
B1, M1-M2 = VALUE
B2, 2-G = VALUE
B3, M2-G = VALUE
```

2.3 MECHANICAL SYSTEM STATE VARIABLES

The state variables applicable to mechanical system and recognized by the program are shown in Table V. These can be referred to in the input language by placing the designation letter of Table V before the element it applies to, for example FK1 represents the force exerted on spring K1.

TABLE VI. MECHANICAL SYSTEM STATE VARIABLES

Translational		Rotational	
State Variable	Program Notation	State Variable	Program Notation
Force	F	Torque	T
Velocity	V	Angular Velocity	W

The state variables are analogous to voltage and current in the electrical case where angular velocity represents voltage and torque represents current.

INITIAL CONDITIONS

Initial conditions are handled in a manner similar to the electrical case. The mechanical analog of a capacitor is a mass (M) for the translational case and the moment of inertia (J) for the translational case. The mechanical analog of an inductor is the spring stiffness K. Therefore, the initial conditions of a mechanical system can be entered as the force

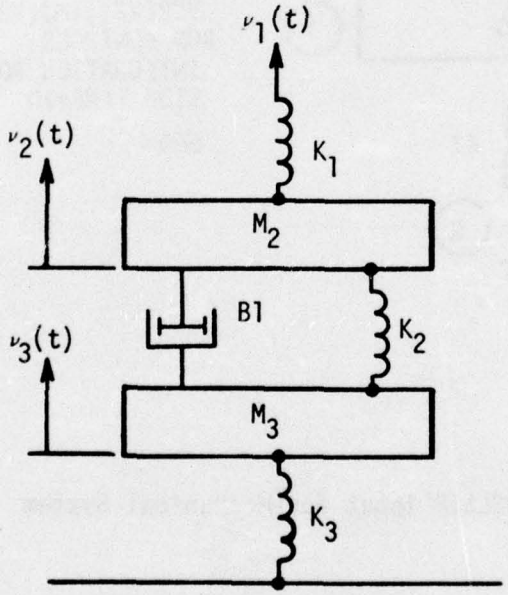
exerted on Ks and velocity on Ms and about Js. Some samples are given below.

INITIAL CONDITIONS
WJ5 = 10
WM10 = 15
VM2 = 5
FK12 = -20

2.4 OUTPUTS

The outputs are entered in the manner described in Section 2.1.10 with the electrical element names replaced by mechanical element names, voltage replaced by velocity (V) or angular velocity (W) and current replaced by force (F) or torque (T).

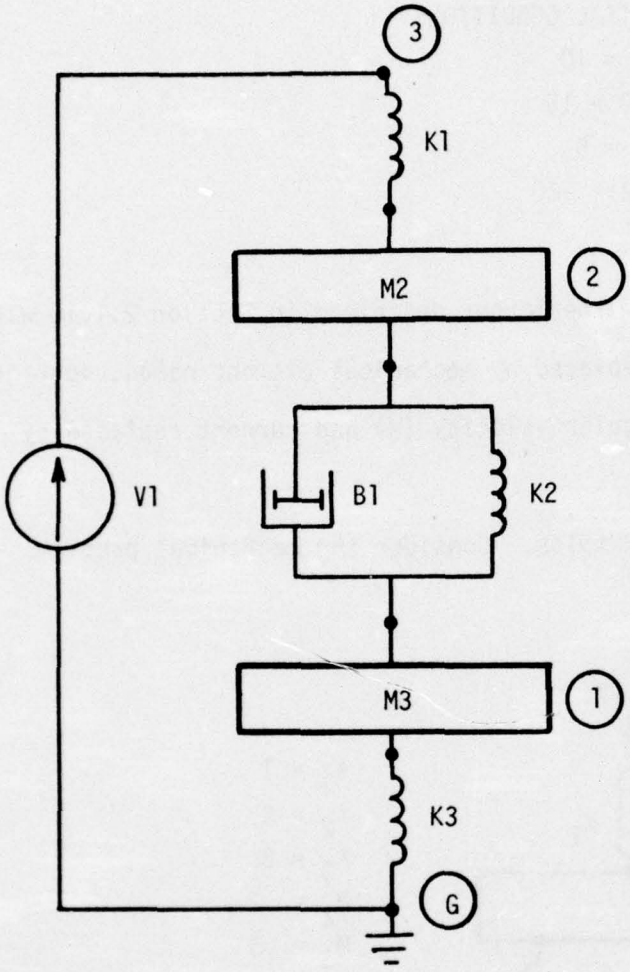
Mechanical System Input Examples. Consider the mechanical problem shown in Figure 10.



$K_1 = 1$
$K_2 = 2$
$K_3 = 3$
$M_2 = .2$
$M_3 = .3$
$B = 1$
$V_1 = \cos 2t$

Figure 10. Mechanical Problem

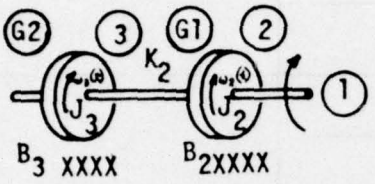
The noded mechanical system and input for NELSIM is shown in Figure 11.

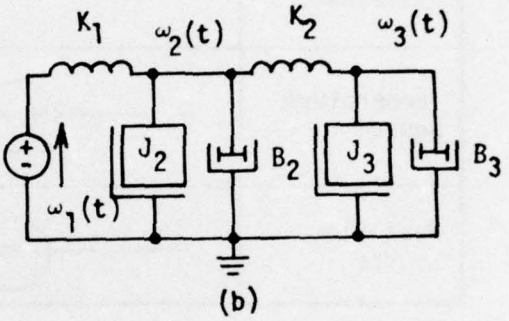



```
ANALOG DESCRIPTION
 MECHANICAL
  ELEMENTS
   V1,6-3=Q1(TIME)
   K1,62-3=1
   K2,61-2=P2
   K3,6-1=3
   M2,2-62=.2
   M3,1-61=.3
   B1,61-2=1
  FUNCTIONS
   Q1(T)= (COS(2.*T))
  DEFINED PARAMETERS
   P2=2
  OUTPUTS
   VM2(V2),VM3(V3),PLOT
  RUN CONTROLS
   INTEGRATION ROUTINE=IMPLICIT
   STOP TIME=10
   END
```

Figure 11. NELSIM Input for Mechanical System

Figure 12 represents a rotational system and its input for the normalized case where all values are set equal to 1.





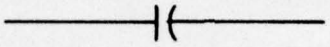
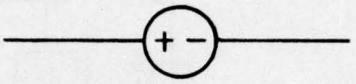
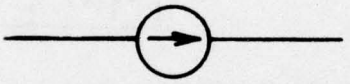

```
ANALOG DESCRIPTION
MECHANICAL ELEMENTS
W1, G-1 = 1
K1, 1-2 = 1
J2, 2-G1 = 1
K2, G1-3 = 1
J3, 3-G2 = 1
B3, J3-G = 1
B2, J2-G = 1
OUTPUTS
WJ2, WB2, PLØT
RUN CONTROLS
STOP TIME = 1
END
```

Figure 12. NELSIM Input for a Rotational Problem

2.5 THERMAL SYSTEM INPUT

The input to NELSIM for a thermal problem is similar to the input of a mechanical problem. The elements for a thermal system are shown in Table VII.

TABLE VII. THERMAL ELEMENTS AND STATE VARIABLES

Element	Schematic Representation	Program Notation
Thermal Capacitance		C
Thermal Resistance		R
Temperature Source		T
Heat Flow Source		Q

The user must derive a thermal analog network made up of the elements shown in Table VII. The value of these elements may be entered in any of the formats documented in Section 2.1.6.

Thermal State Variables

The state variables for thermal input are Temperature (T) and heat flow (Q) and are analogous to voltage (V) and current (I) respectively in the electrical network.

Initial Conditions

Initial conditions for a thermal system may be entered as the temperature across a thermal capacitance. This is analogous to a voltage across an electrical capacitance. Some examples follow.

```
INITIAL CONDITIONS
TC1 = 10
TC2 = 20
```

Outputs

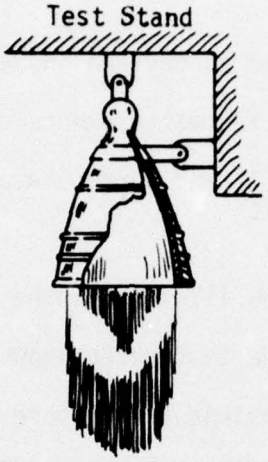
Outputs are requested as defined in Section 2.1.10 with the electrical elements replaced by the thermal elements of Table VII, voltage (V) replaced by Temperature (T) and current replaced by heat flow (Q).

Thermal Input Example

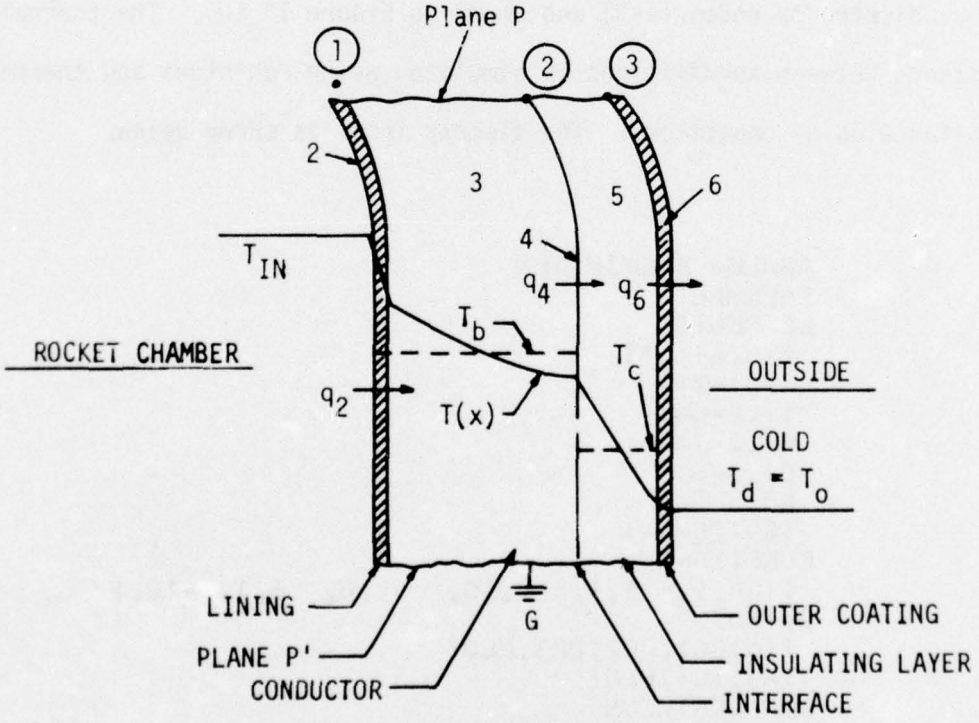
The following problem illustrates the input of a thermal system. Consider the rocket engine test setup shown in Figure 13 for which it is desired to study the transient temperature response within the engine wall. The input of the problem requires the various subdivisions within the wall to be indicated by nodes (1-3) and shown in Figure 13 (b). The thermal resistance between subdivisions is simulated using resistors and thermal capacitance using capacitors. The element input is shown below.

```
ANALOG DECRIPTION
THERMAL
ELEMENTS
 TIN,G-1=T1
 R1,1-2=1
 C1,2-G=1
 R2,2-3=1
 C2,3-G=1
 R3,3-4=1
 TOUT,G-4=1
FUNCTIONS
 T1=0,1,  1,1,  2,10,  5,10,  6,1,  10,1
OUTPUTS
 TIN,QC1,QC2,QR3,PLOT
 TR3,TC1,PLOT
RUN CONTROLS
 INTEGRATION ROUTINE=IMPLICIT
 STOP TIME=15
 RUN INITIAL CONDITIONS
END
```

The elements have all been entered as constant values for simplicity but in general will be non-linear functions of the materials involved and will be entered using a variety of value formats.



(a) Rocket-engine walls



(b) System Model Showing

Figure 13. Heat Transfer Example

2.6 ELECTRO-MECHANICAL ELEMENTS AND MODELS

The entry of electro-mechanical problems into NELSIM is possible in several ways. The differential equation and transfer function entries are defered until a general treatment of these topics in Sections III and IV, respectively. Two additional modes of entry, however, are available to the user, namely

1) Electro-mechanical element entry
2) Model entry

2.6.1 Electro-Mechanical Element Entry

NELSIM will allow for the input of an electro/mechanical system in terms of the lumped elements of both the electrical and mechanical portions of the system coupled together through common defined parameter names, algebraic or first order differential equations or dependent sources. The mechanical and electrical elements are entered following the subheadings ELEMENTS (MECHANICAL) and ELEMENTS (ELECTRICAL), respectively. The electrical elements allowed are discussed in Section 2.1. The input is better illustrated through an example. Consider the galvenometer sketch provided in Figure 14. A galvanometer ideally produces a deflection that is proportional to an electrical input.

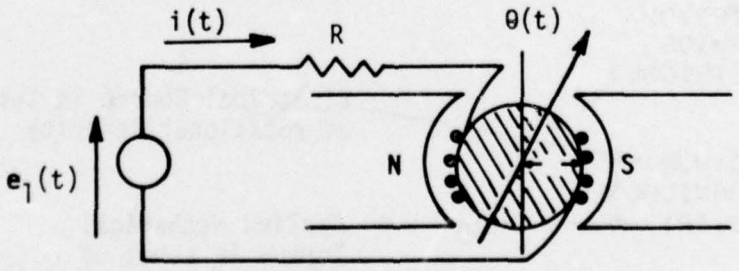


Figure 14. Galvanometer Schematic Representation

Figure 15 represents the equivalent lumped network in terms of its electrical and mechanical parts

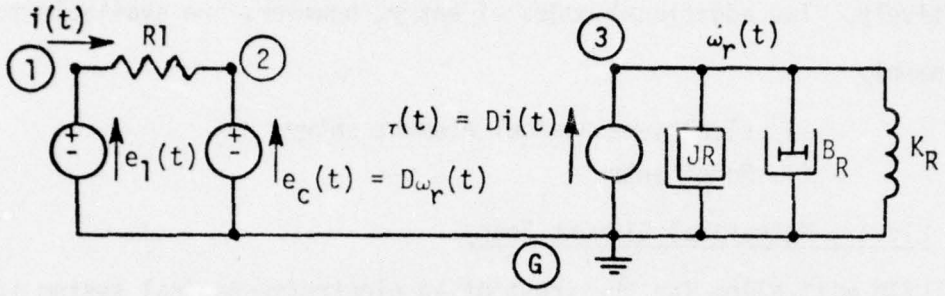


Figure 15. Lumped Element Galvanometer Equivalent

where J and B represent the moment of inertia of the rotor and the viscous damping that results from air friction. The element K_R represents the rotational stiffness produced by a spring attached to the rotor. The constant D is defined to be $\beta \ell a$ where β is the magnetic flux density, ℓ is the total length of the conductor and a is the radius of the rotor.

The input language where all the elements are normalized to one follows.

```
ANALOG DESCRIPTION
ELECTRO MECHANICAL
ELEMENTS(ELECTRICAL)
E1,G-1=1
R1,1-2=1
EC,G-2=DEC(PD,WJR)
ELEMENTS(MECHANICAL)
T1,G-3=DEC(PD,IR1)
JR,3-G=1
BR,3-G=1
KR,3-G=1
```

Electrical Source in terms of rotational Velocity

Applied Mechanical Torque in terms of current through resistor

The two dependent sources EC and T1 provide the interface between the two systems and are entered directly in terms of the elements involved.

Element NAME uniqueness

NELSIM converts all mechanical elements to electrical equivalent elements by replacing the first letter of the element. For example Mass M15 becomes capacitor C15 and spring constant K1 becomes inductor L1. Care must, therefore, be taken by the user to avoid duplication of names in the resulting network. This can be accomplished by checking that all electrical and mechanical elements have unique denotations following the first letter.

2.6.2 Built in Models

This option will allow the user to include models of generally used electro-mechanical devices in his system by simply calling for them. It is difficult to derive models general enough to suit all needs but it should be kept in mind that the capability to input system problems in terms of differential equations or transfer function blocks allows the user to build his own models. The two models presently implemented into NELSIM are that of a rate gyroscope and an accelerometer. The call to a model will be of the form

T1, N1-N2 = MODEL XXXX

where N1 represents the input to the model and N2 the output and XXX is the built in model name. The two models are discussed below. The model entries are submitted following the subheading MODELS.

2.6.2.1 Rate Gyro Model

Rate gyros are used widely in any application requiring tracking and stabilization. Aircraft and missiles utilizes gyros for stabilization when angular deviations are noticed. Rate gyros are also used extensively in conjunction with tracking antennas so that the angular velocity of the

tracked vehicle can be measured. A mathematical block diagram of a rate gyro is given in Figure 16.

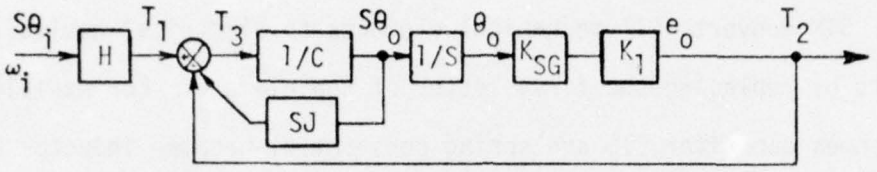


Figure 16. Block Diagram of Rate Gyro

In the figure ω_i represents the rate input to the gyro and e_o the proportional output voltage generated. The K's represent amplifiers within the system to attain the desired signal levels with K_{SG} representing the signal generator which generates an electrical signal proportional to the displacement angle θ_o. The transfer function between T2 and T1 is calculated as follows:

$$\frac{T2}{T1} = \frac{\frac{K/C}{S(1+\frac{J}{C}S)}}{1 + \frac{K/C}{S(1+\frac{J}{C}S)}}$$

when simplified the above equation becomes

$$\frac{T2}{T1} = \frac{1}{1 + \frac{C}{K}S + \frac{J}{K}S^2}$$

The above represents the closed loop transfer function of the rate gyro which can be recognized as having the form

$$F(S) = \frac{1}{1 + \frac{2\xi}{\omega_n} S + \frac{1}{\omega_n^2} S^2}$$

where ξ is the damping factor and ω_n the natural frequency. Like terms yield that

$$\omega_n = \frac{K}{J} \text{ and } \xi = \frac{C}{2\sqrt{KJ}}$$

which allows the user to adjust both of these quantities by adjusting the various gains.

The model equivalent for NELSIM input is shown in Figure 17 where PB = 1/C and PC = J. The feedback gain PF has been added for further generalization of the model and if set equal to one will result in the transfer function F(s). Use of the model requires the user to input the request under the ELECTRO-MECHANICAL heading and following the MODELS sub-heading card. Usage also requires the user to define all the gains needed by the model (PA, PB, PC, PD, PF). The letter P is attached to both the input node and the output node of a model as well as the name assigned by the user. If the model is referred to as M1 for example the input to the model becomes PINM1 and output PØUTM1. The model name is also attached to the internal parameters to maintain parameters name uniqueness (i.e., PIM1, PG2M1, PFM1). To illustrate consider Figure 18 for which H represents the gyro model.

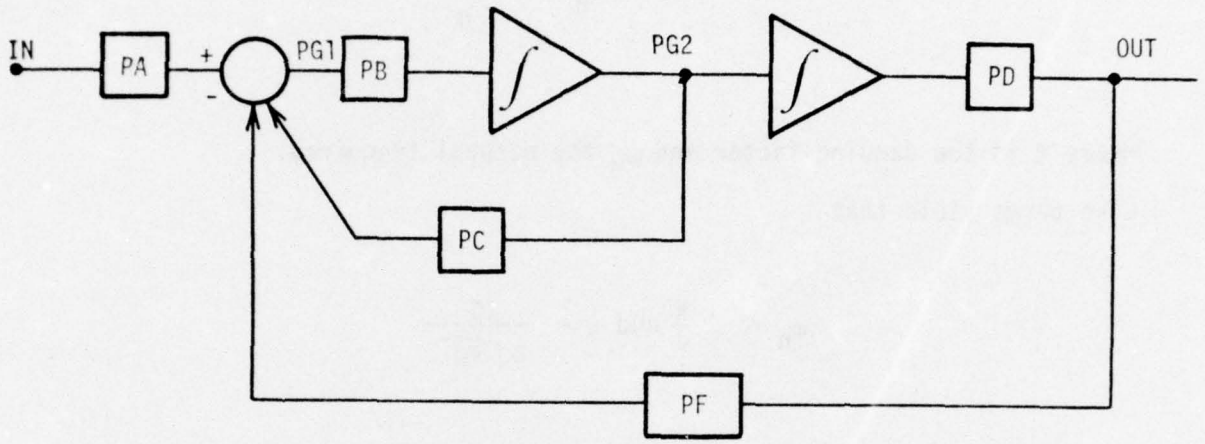


Figure 17. NELSIM Equivalent Rate Gyro

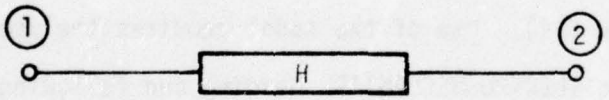


Figure 18. NELSIM Gyro Model

The input to the NELSIM program is shown below.

```
ANALOG DECRIPTION
ELECTRO MECHANICAL
MODELS
T1,1-2=GYRO
DEFINED PARAMETERS
PAT1=1
PBT1=1
PDT1=1
PCT1=1
PFT1=1
P1T1=T1
FUNCTIONS
T1=0,0,  1,0,  2,1,  10,1
OUTPUTS
P1T1(INPUT),P2T1(OUT),PG2T1,PLOT
RUN CONTROLS
STOP TIME=10
INTEGRATION ROUTINE=IMPLICIT
END
```

Model Name Added to Parameters

Input Step

Step Description

Response request at output node

2.6.2.2 Accelerometer Model

The accelerometer is the basic measuring element in an inertial navigation and guidance system. It is the function of the accelerometer to provide measurement and integration of linear acceleration to successfully accomplish the functions of position and velocity indication and generation of proper steering signals. Although numerous models are available the double integrating accelerometer shown in Figure 19 has been implemented into NELSIM.

In the model a_i represents the input acceleration, e_o the signal voltage generated and $\ddot{\theta}$ the angular acceleration of the motor shaft resulting from the application of e_o. The double integration of the angular velocity ($\ddot{\theta}$) yields the angular velocity$^{(\dot{\theta})}$ and relative angle

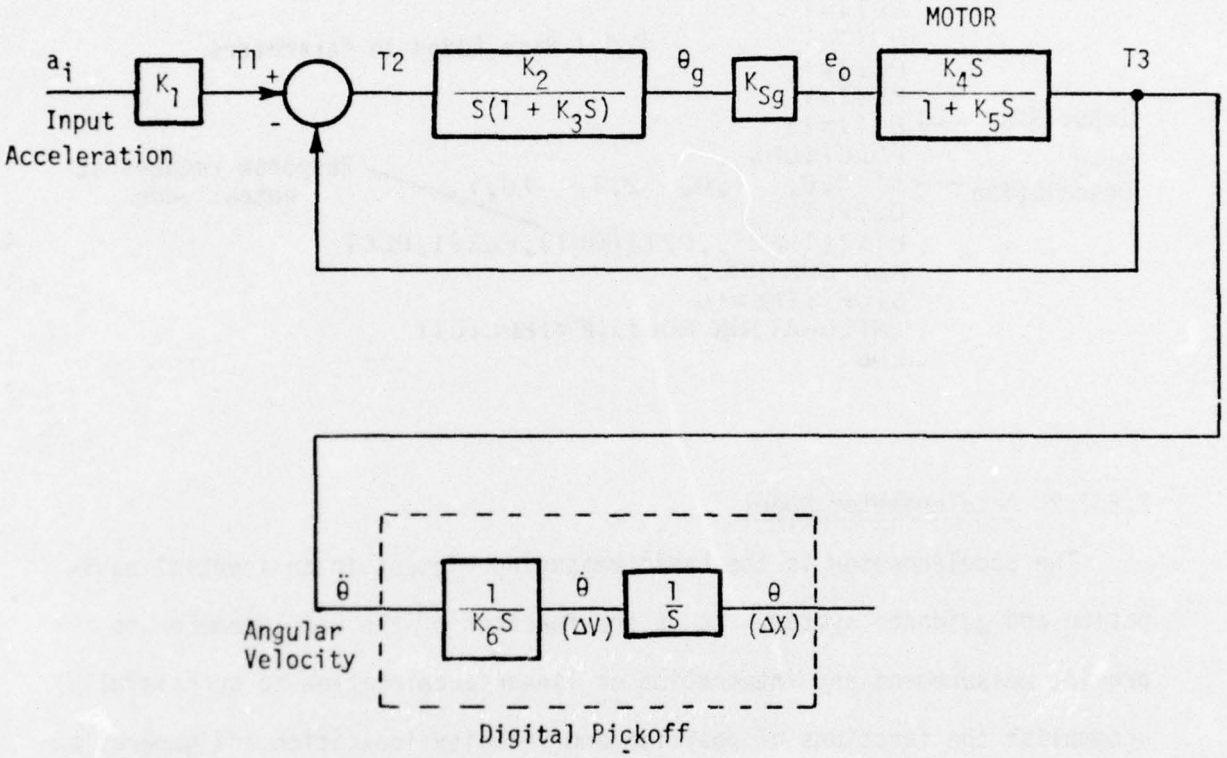


Figure 19. Acceleration Model

(θ) of the shaft. These two quantities are proportional to the craft velocity (V) and distance traveled (X). The mathematical formulation is given below.

$$\frac{T3}{T2} = \frac{(K_2K_{SG}K_4)S}{S(1 + \frac{K_3}{K_2}S)(1 + K_5S) + (K_2K_{SG}K_4)S}$$

$$= \frac{N}{1 + aS + bS^2 + N}$$

where: $N = (K_2K_{SG}K_4)S$

$$a = \frac{K_3}{K_2} + K_5$$

$$b = \frac{K_3K_5}{K_2}$$

inserting $T = K_1a_i$ and applying the final value theorem for an assured step input yields

$$T_3 = KK_1a_i$$

in the steady state which shows the angular acceleration of the shaft to be proportional to the input acceleration. Another relationship exist for T3, namely

$$T3 = K_6\theta S^2$$

equating the two yields

$$KK_1a_i = K_6\theta S^2$$

To determine velocity and distance changes as a function of a_i both sides of this equation are integrated.

$$KK_1 \Delta V = KK_1 \int_0^t a_i dt = K_6 \int_0^t \Delta\theta dt$$

$$K\Delta X = KK_1 \int_0^t \int_0^t a_i dt = K_6 \int_0^t \int_0^t \Delta\theta dt$$

$$\Delta V = \frac{K_6 \Delta\theta}{KK_1}$$

$$\Delta X = \frac{K_6 \Delta\theta}{KK_1}$$

The model programmed into NELSIM is illustrated in Figure 20.

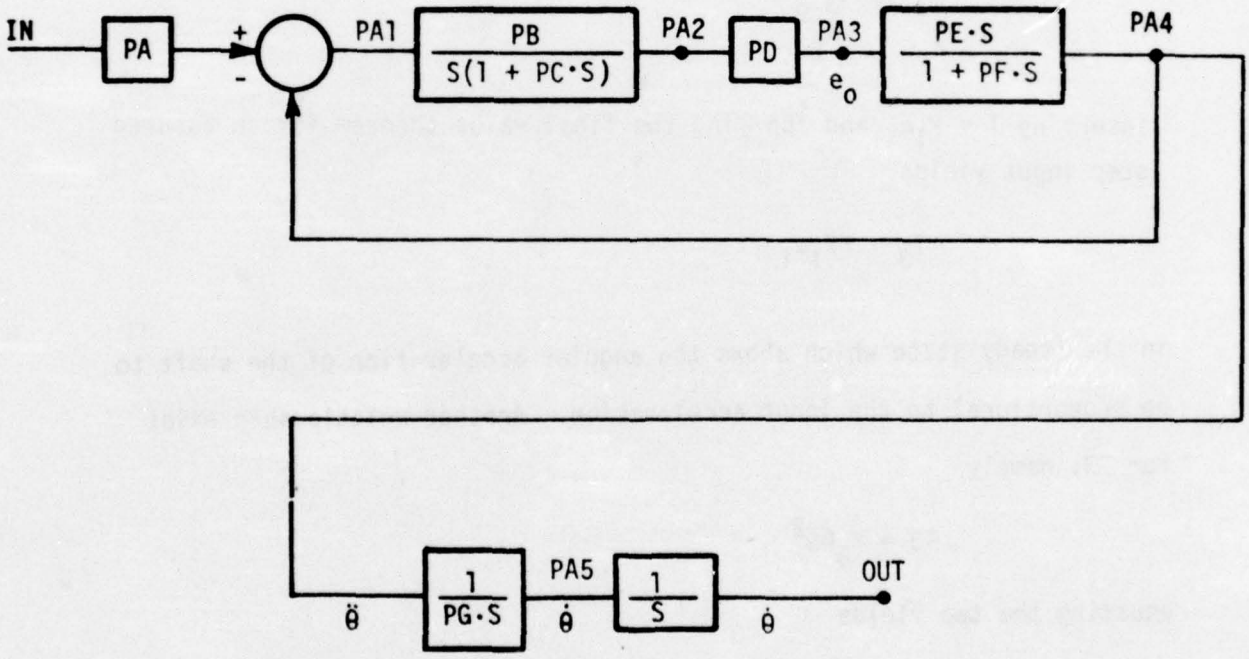


Figure 20. NELSIM Accelerometer Equivalent

The usage and response of the model is illustrated for the case where all gains have been normalized to 1 and the input to the model is a unit step. Consider Figure 21 where H is the accelerometer

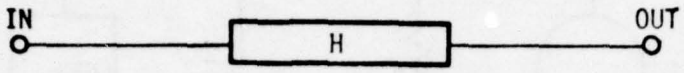


Figure 21. NELSIM Accelerometer Model

The input to NELSIM is given below:

```
ANLALOG DECRIPTION
ELECTRO MECHANICAL
MODELS
T1,IN-OU=ACCELEROMETER
DEFINED PARAMETERS
PAT1=1
PBT1=1
PCT1=1
PDT1=1
PET1=1
PFT1=1
PGT1=1
PINT1=T1
FUNCTIONS
T1=0,0,  0,1,  10,1
OUTPUTS
PINT1,PAT1,PA2T1,PA3T1,PA4T1,PLOT
PA5T1,POUT1,PLOT
RUN CONTROLS
INTEGRATION ROUTINE=IMPLICIT
STOP TIME=10
END
```

As was required for the gyro model, all gains within the model must be provided by the user with the model label (T1) added to each name.

2.7 ELECTRO-OPTICAL SYSTEM INPUT

Electro-optical systems are represented in NELSIM as equivalent electrical circuits capable of simulating the AC characteristics of photo diode devices.

The most commonly used circuit is illustrated in Figure 22 followed by the NELSIM input describing the circuitry.

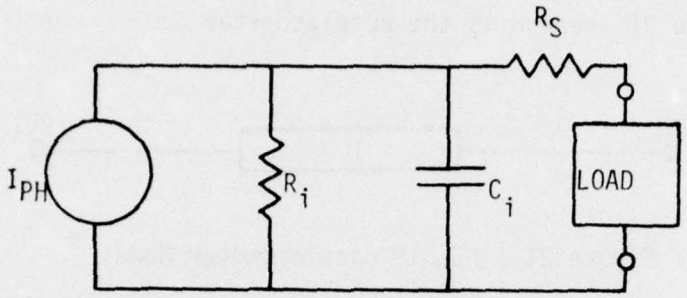


Figure 22. Equivalent Circuit of a Photo Diode

```
ANALOG DESCRIPTION
ELECTRO OPTICAL
ELEMENTS(ELECTRICAL)
JPD,G-1=OEO(PEFF,PCHG,PCON,POFR,TPL)
RI,2-G=6
CI,3-G=3
RS,3-4=12
RL,4-G=12
FUNCTIONS
TPL=0,0,.2,2,.4,6,.85,10,.6,12,0,15
OEO(A,B,C,D,E)=(A*B/(C*D)*E)
DEFINED PARAMETERS
PEFF=.25
PCHG=1E-7
PCON=123E-4
POFR=37
OUTPUTS
JPD,JRL,PLOT
RUN CONTROLS
INTEGRATION ROUTINE=IMPICIT
STOP TIME=15
END
```

In the above input, the photo current IPH is described by a perfect-square law formulation

$$I_{PH} = (n\ q\ P/hv)$$

where,

η = quantum efficiency

P/hv = average number of incident photons per unit time

I_{PH}/q = average number fps unit time of electrons emitted from the photocathode.

Since the input is as an equivalent circuit, the circuit elements are detailed as ELEMENTS (ELECTRICAL) and the element names are consistent with the network preparation rules of Section 2.1.2.

Since the device response is dependent upon the frequency of the energy source it is best to construct a circuit for frequency bands and sum the produced currents linearly, or to restrict the analysis to a certain frequency band.

2.8 UNITS AND SCALING

The following paragraphs describe the scaling option in NELSIM. In order to allow NELSIM to be completely compatible with SCEPTRE in the flexibility of its input language the scaling module is semiautomated and requires user interaction.

2.8.1 Units

SCEPTRE allows any set of parameter units as long as the element magnitudes with respect to these are consistent. To maintain the two input languages consistent, the same will be true of NELSIM. The user is allowed

to enter a problem in any set of units he chooses and is responsible for compatibility throughout his system.

2.8.2 Scaling Guidelines and Automation

One of the problems encountered in system analysis utilizing numerical integration is that of maintaining a consistent set of parameter units such that the magnitudes of all the solution variables are consistent with the specified error criteria. Numerical inaccuracies result and excessive solution time is spent when units are not carefully chosen due to the difficulties in meeting specified error criteria. Previous experience in the system analysis area and application of SCEPTRE and other codes to large-scale circuit and system problems has shown that is desirable to maintain the magnitude of the solution variables in the range between 10^{-3} and 10^{3}. For example, voltage and current are the two solution variables used by SCEPTRE and it is desirable to keep these variables within the limits shown below

$$10^{-3} \leq |V| \leq 10^{3}$$

$$10^{-3} \leq |I| \leq 10^{3}$$

The desired magnitude ranges can be accomplished through knowledge of the basic laws governing a particular system and a knowledge of the ranges expected. If the expected units of the solution variables and time are known, the system element units can be scaled to yield those units. The above is exemplified through a high speed transistorized circuit in which the response is known to be in terms of volts, milliamps and nanoseconds. Equations 1 through 3 illustrates the applicable relationships between state variables and electrical elements.

$$R = \frac{V}{I} \qquad (1)$$

$$C = I \frac{dt}{dV} \qquad (2)$$

$$L = V \frac{dt}{dI} \qquad (3)$$

Where: V = Voltage

I = Current

R = Resistance

C = Capacitance

L = Inductance

To obtain the response in terms of volts, milliamps and nanoseconds the elements are scaled as follows:

$$R = \frac{V}{I} = \frac{\text{Volts}}{\text{ma}} = \frac{1}{10^{-3}} = 10^{3} \text{ (kilohms)}$$

$$C = I \frac{dt}{dv} = \text{ma}, \frac{\text{nsec}}{\text{Volts}} = \frac{(10^{-3})(10^{-9})}{1} = 10^{-12} \text{(picofarads)}$$

$$L = V \frac{dt}{di} = \text{Volts}. \frac{\text{nsec}}{\text{ma}} = \frac{(1)(10^{-9})}{10^{-3}} = 10^{-6} \text{(microhenries)}$$

Definition of the network elements in terms of these units will bring the solution variables closer to their desired range.

The same procedure is applicable to other systems. Equations 4 through 8 present the relationships necessary for mechanical and thermal systems.

MECHANICAL SYSTEMS

$$B = \frac{F}{V} \qquad (4)$$

$$M = F \frac{dt}{dV} \qquad (5)$$

$$K = \frac{1}{V} \frac{dF}{dt} \qquad (6)$$

Where:

V = Velocity

F = Force

M = Mass

K = Spring Constant

B = Viscocity/Friction

THERMAL SYSTEM

$$C = Q \frac{dt}{dT} \qquad (7)$$

$$R = \frac{T}{Q} \qquad (8)$$

Where:

Q = Rate of heat flow

C = Thermal Capacitance

R = Thermal Resistance

T = Temperature

2.8.3 Automated Scaling

An algorithm has been implemented within the NELSIM program to achieve scaling of a network. The program generates an electrical analog from various types of system inputs. The scaling algorithm scales all the elements to magnitudes consistent with user provided guidelines. The option require the user to input the expected units of the solution variable and time in terms of electrical quantities. Table VIII shows the analogies

between the various systems involved. Both mechanical and thermal quantities are translated into the electrical quantities shown in the table, such that in the translated network the only solution variables are electrical. The user will enter a problem in terms of the standard units familiar to the field involved. Table IX illustrates the units that are expected if the scale option is utilized. The user indicates the scaling to be performed by entering the expected magnitudes of the solution variables and time. For example, consider the electronic network discussed earlier entered in terms of the standard elements shown in Table IX. To obtain the solution in terms of volts, milliamps and nanoseconds the network must be scaled. The user input would consist of the card shown below which states that the scaling option is desired

```
SCALE OPTION, VOLTAGE=1, CURRENT=10E-3, TIME=10E-9
```

and the voltage is to remain in terms of volts but current and time are to be scaled to milliamps and nanoseconds accordingly. Use of the scaling option for all system components will maintain a consistent set of variable values.

The output of the program consists of two listings. The first is a listing of the analog network generated with no scaling applied. The second consists of the analog network with all constant elements scaled and a list of the scale factors used on the elements. The user is then required to scale all non-constant element values to complete the scaling task.

To illustrate the use of the scaling option consider the mechanical problem presented earlier. It is desired to scale time by a factor of 10^{-3}, force by a factor of 10^{-3} and leave the velocity units the same.

TABLE VIII. SYSTEM ANALOGIES

SYSTEM FUNCTION	ELECTRICAL	MECHANICAL		THERMAL
		Translational	Rotational	
SOLUTION VARIABLES	Voltage	Velocity	Angular Velocity	Temperature
	Current	Force	Torque	Heat Flow
TIME	Time	Time	Time	Time
ELEMENTS	Resistance	Friction	Friction	Thermal Resistance
	Capacitance	Mass	Moment of Inertia	Thermal Capacitance
	Inductance	Spring Stiffness	Spring Stiffness	

TABLE IX. STANDARD SYSTEM UNITS

PHYSICAL MEDIUM	ELEMENTS	SYMBOL	STANDARD	UNITS MKS	ENGLISH	BTU
Electrical	Resistance	R	Ohms			
	Capacitance	C	Farads			
	Inductance	L	Henries			
Mechanical	Mass	M		Kilogram	Slug	
	Spring Stiffness	K		Newton/ meter	lb/ft	
	Friction	B		n-sec/m	lb/ft	
	Moment of Inertia	J		nm/ (rad/sec^2)	ft lb/ (rad/sec^2)	
	Angular Velocity	W		rad/sec	rads/sec	

TABLE IX. STANDARD SYSTEM UNITS (CONTINUED)

PHYSICAL MEDIUM	ELEMENTS	SYMBOL	STANDARD	MKS	ENGLISH	BTU
Thermal	Temperature	T				^{o}F
	Heat flow rate	Q				BTU/sec
	Capacitance	C				BTU/^{o}F
	Resistance	R				(BTU/^{o}F/ lb

As shown in Table VIII, the analog quantities for force and velocity are current and voltage, respectively. The scale request is

```
SCALE OPTION, TIME=1E-3, CURRENT=1E-3, VOLTAGE=1
```

The output of NELSIM will first list the network analog generated with no scaling applied, and the second lists, the value of analog elements with all constant elements.

SECTION III

TRANSFER FUNCTION INPUT LANGUAGE

The input language for a network topology was described in the preceding section. For a certain class of problems, a signal flowgraph representation of the system may already be at hand, for example, control systems.

An input language has been devised to allow the direct input of a flowgraph into NELSIM. The program interprets this input, and from it derives the set of first order differential equations necessary to execute the problem in SCEPTRE.

The input utilizes the same type of format available for circuit inputs in SCEPTRE and is nearly identical to SYNAP. The data are placed under headings and subheadings in a similar fashion and is format free. The statements consist of descriptive user-derived names for nodes, components, and value specifications separated by delimiters, such as commas, parentheses, and dashes. The input data may be punched anywhere on the card within the first 72 columns. To continue a statement from one card to another, the only requirement is that the discontinuation be made immediately after delimiters, with the delimiter being the last non-blank character on the card. The input data organization is shown in Figure 23. Each entry is described in the following paragraphs.

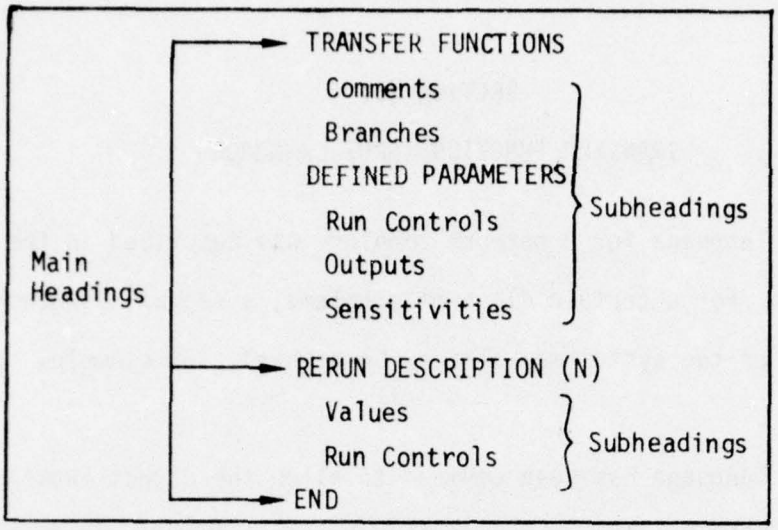


Figure 23. Graph Description Data Organization

3.1 FLOWGRAPH INPUT PREPARATION

The flowgraph description is defined in terms of branch transmittances connected between nodes. The steps listed below should be followed in preparing a signal flowgraph for input.

Assign a name or number to all nodes in the graph (four or less alphanumeric characters).

Assign a name or value to all transmittance parameters (four or less alphanumeric characters). All parameters must begin with the letter P and be defined under the DEFINED PARAMETERS subheading.

Label signal flow direction between node connections using arrowheads

Figure 24 illustrates a flowgraph ready for input generation.

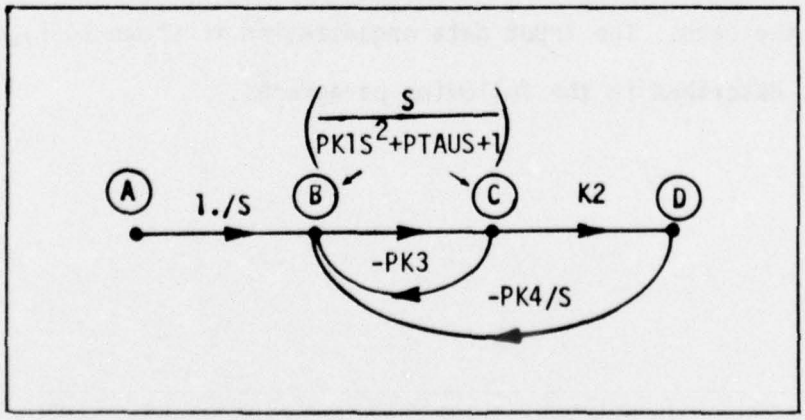


Figure 24. Prepared Flowgraph

3.2 GRAPH DESCRIPTION

For Network topologies the first card was ANALOG DESCRIPTION, in the same matter, the first card to describe a signal flowgraph is

/TRANSFER FUNCTIONS

This card indicates to the program that a flowgraph is being input and allows the program to enter the graph processor directly.

3.3 BRANCHES

The flowgraph is made up of branches containing some transmittance characteristics between one node and another. The format for any branch is shown below where:

/NODE1 - NODE2 (BRANCH TRANSMITTANCE)

- NODE1 denotes the initial node from which the signal originates (from node)
- NODE2 denotes the final node of the transmittance (to node)

The general form of the branch transmittance is a rational polynomial of the form

$$P(S) = \frac{N_0 * B_{01} * B_{02} \ldots B_{0n} + N_1 * B_{11} * B_{12} \ldots B_{1n} S + \ldots N_m * B_{m1} * B_{m1} * B_{m2} \ldots B_{mn} S^m}{M_0 * C_{01} * C_{02} \ldots C_{0n} + M_1 * C_{11} * C_{12} \ldots C_{1n} S + \ldots M_k * B_{k1} * B_{k2} \ldots B_{kn} S^k}$$

where

$N_0, N_1, N_2 \ldots N_m, M_0, M_1, \ldots M_k$

$B_{01}, B_{02}, \ldots B_{0n}, B_{11}, B_{12}, \ldots B_{1n},$

$C_{01}, C_{02}, \ldots C_{0n}, C_{11}, C_{12} \ldots C_{1n}$

are transmittance coefficient parameters

Polynomials of this form are recognized by the program by using the following input form:

/NODE1 - NODE2 (NUMERATOR/DENOMINATOR)

where the expanded form yields

/NODE - NODE (N_0*B_{01}*B_{02} ... B_{0n}, C_0, N_1*B_{11}*B_{12} ... B_{1n}, 1,

/N_2*B_{21}*B_{22} ... B_{2n}, 2, ..., N_m*B_{m1}*B_{m2} ... B_{mn}, m/

/M_0*C_{01}*C_{02} ... C_{0n}, 0, M_1*C_{11}*C_{12} ... C_{1n}, 1, M_2*C_{21}*C_{22} ... C_{2n}, 2,

/..., M_k*B_{k1}*B_{k2} ... B_{kn}, K)

3.4 INPUT/OUTPUT REFERENCE

Each node name entered in the branch description is translated into a defined parameter by NELSIM by adding the letter P to each node. This is done to satisfy the SCEPTRE input as well as interfaces requirements that will be discussed later. The user desiring to define or output a node must refer to it as PNODE in the input. An example will illustrate the procedure. The branches of the feedback system shown in Figure 25 are described as indicated below.

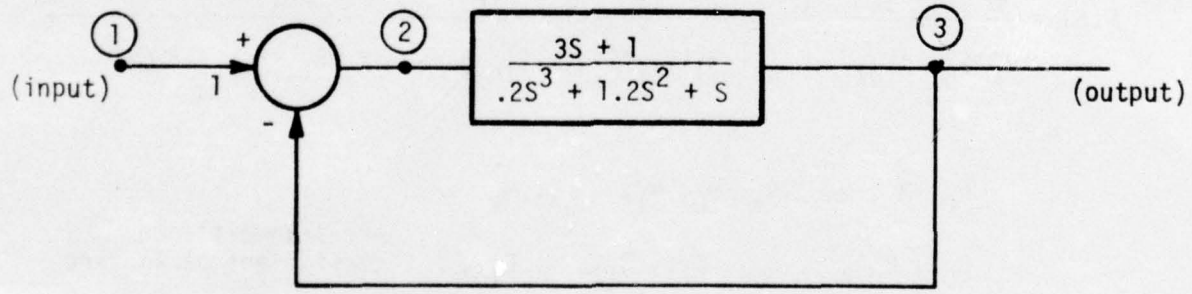


Figure 25. Unity Feedback System

```
TRANSFER FUNCTIONS
BRANCHES
1-2(1.,0)
2-3(3.,1,  1.,0/ .2,3.  1.2,2,  1.,1)
3-2(-1.,0)
```

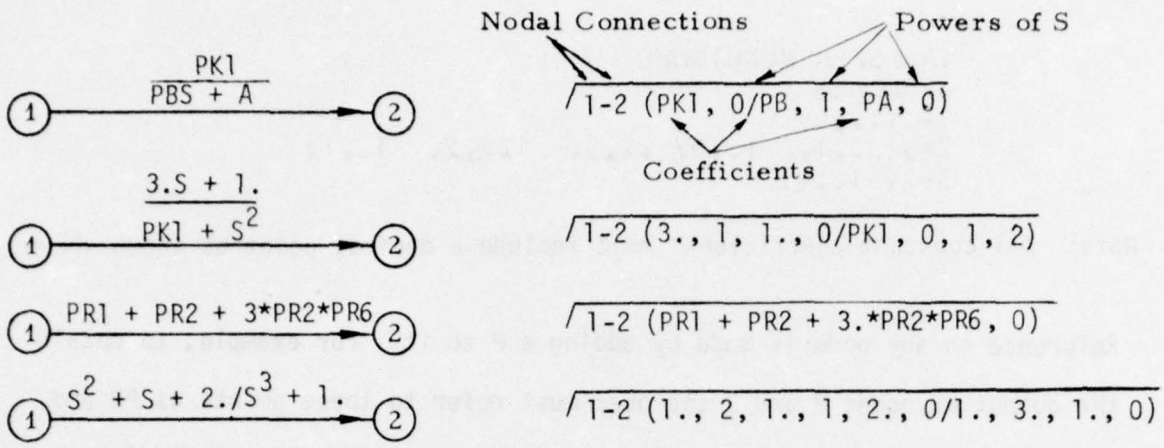
Note: All constant coefficients must include a decimal point as shown above.

Reference to any node is made by adding a P to it. For example, to obtain the output at nodes 2 and 3 the user must refer to these points as P2 and P3 under outputs. A more complete input description of Figure 22 will clarify. As shown below, the input to the system is defined as P1 and described as Table T1.

```
TRANSFER FUNCTIONS
BRANCHES
1-2(1.,0)
2-3(3.,1,  1.,0/ .2,3,  1.2,2,  1.,1)
3-2(-1.,0)
DEFINED PARAMETRS
P1=T1                          <-- Input at Node 1
FUNCTIONS
T1=0,0,  0,1,  10,1
OUTPUTS
P1,P3,PLOT                     <-- Output requests at Nodes 1 and 3
RUN CONTROLS
STOP TIME=10
INTEGRATION ROUTINE=IMPLICIT
END
```

The integer numbers between commas denote the power of S associated with the individual coefficient.

The coefficients can be made up of literal and numerical values. Several sample branch transmittances will better illustrate the required input:



The branch descriptions for the flowgraph shown in Figure 23 are given below:

```
GRAPH DESCRIPTION
  BRANCHES
    A-B  (1.,0/1,1)
    B-C  (1.,1/PK1,2,PTAU,1,1,1)
    C-D  (PK2,0)
    C-B  (-PK3,0)
    D-B  (-PK4,0/1,1)
```

SECTION IV

DIFFERENTIAL EQUATION INPUT LANGUAGE

Differential equations are a very general form of system definition and all the systems seen so far can be described in terms of these. NELSIM accepts as input sets of n^{th} order differential linear or nonlinear differential equations of the form given below.

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ \vdots & & & \vdots \\ a_{m1} & \cdots & \cdots & a_{mm} \end{bmatrix} \bullet \begin{bmatrix} \left(\frac{d^n x_1}{dt^n}\right)^i \\ \left(\frac{d^n x_2}{dt^n}\right)^j \\ \vdots \\ \left(\frac{d^n x_m}{dt^n}\right)^k \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1m} \\ \vdots & & & \vdots \\ b_{m1} & \cdots & \cdots & b_{mm} \end{bmatrix} \bullet \begin{bmatrix} \left(\frac{d^{n-1} x_1}{dt^{n-1}}\right)^i \\ \left(\frac{d^{n-1} x_2}{dt^{n-1}}\right)^j \\ \vdots \\ \left(\frac{d^{n-1} x_m}{dt^{n-1}}\right)^k \end{bmatrix}$$

$$+ \cdots + \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1m} \\ \vdots & & & \\ c_{m1} & \cdots & \cdots & c_{mm} \end{bmatrix} \bullet \begin{bmatrix} (x_1)^i \\ (x_2)^j \\ \vdots \\ (x_m)^k \end{bmatrix} + \begin{bmatrix} d_1 \\ \vdots \\ d_m \end{bmatrix} = \begin{bmatrix} F_1 \\ F_2 \\ \vdots \\ F_m \end{bmatrix} \qquad (1)$$

The above equations are a general set of m nonlinear differential equations in m unknowns, x_1 through x_m and with forcing functions F_1 through F_m. For convenience, the equations are represented in matrix form, but will be

input into NELSIM individually. The term $\left(\frac{d^n x_1}{dt^n}\right)^i$ is the nth derivative of the variable x_1 raised to the i^{th} power where i can vary from 1 to i and is not necessarily the same for all derivative orders. For the systems considered in Section II, the highest order of derivative n generally encountered is two. The coefficients a_{11}, b_{11}, etc. may be time varying or nonlinear. It will be possible to enter the coefficients into NELSIM as constants, defined parameters, or any other type documented in section.

NELSIM automatically takes equations and separates them into first order differential equations in a format directly compatible with SCEPTRE. It should be realized that to determine the transient response of the system, it will be necessary for SCEPTRE to integrate a number of equations equal to the total number of derivatives - up to n derivatives for each unknown for a maximum of mn equations to integrate. The maximum number of first order differential equations currently allowed in SCEPTRE is one hundred, and NELSIM will warn the user if over one hundred equations are generated. If the SCEPTRE limit is increased, NELSIM will be easily modified to accommodate the increase.

VARIABLE PARAMETER NAMES

All names entered in the descriptions must be entered as defined parameter names. For example velocity V1 in an equations should be expressed as PV1. The value of these variables are then defined under the DEFINED PARAMETERS subheading.

SPECIFICATION OF DERIVATIVE ORDER

The order of a derivative is specified by a D followed by an integer number indicating the derivative order. For example, $\frac{d^2V2}{dt^2}$ is defined as

D2PV2 where PV2 is the defined parameter name give to V2 and D2 denotes the second order derivative of V2.

As an example of the differential equation input in NELSIM, consider the equations below which represent the two second order differential equations

$$K1\ (V2-V1) + M2\ \ddot{V}2 + B\ (\dot{V}2-\dot{V}3) + K2\ (V2-V3) = 0$$

$$B\ (\dot{V}3-\dot{V}2) + K2\ (V3-V2) + M3\ \ddot{V}3 + K3\ V3 = 0$$

resulting from the mechanical system shown in Figure 10.

The NELSIM input for differential equations (3) and (4) is shown in Figure 26.

```
DIFFERENTIAL EQUATIONS
UNITS=ENGLISH (MECHANICAL)
K1 * (V2-V1) + M2 * D2V2 + B * (D1V2-D1V3) + K2 * (V2-V3) = 0
B * (D1V3-D1V2) + K2 * (V3-V2) + M3 * (D2V3) + K3 * V3 = 0
DEFINED PARAMETERS
K1=1.
K2=2.
M2=.2
M3=.3
B=1.0
V1=X1(COS(2*TIME))
OUTPUTS
V2,V3,PLOT
RUN CONTROLS
PUNCH OUTPUT          <-- REQUEST TO OBTAIN DECK OF INPUT CARDS READY
STOP TIME=100             FOR SCEPTRE INPUT
END
```

Figure 26. NELSIM Differential Equations Input

The symbols D1 and D2 preceding a variable refer to the first or second time derivatives of that variable and the defined parameters express the variables involved.

INPUT RESTRICTIONS

To accommodate the SCEPTRE input language, several restrictions are placed on the input of differential equations and are listed below:

1. The highest order term must be
   a) Uncoupled
   b) Not exceed an order of 9
2. Derivative variables cannot exceed 4 characters (6 including the derivative definition), e.g.,
   D2PV2
   D1PX14
3. The highest order derivative must appear only once within the equation, and be of one of the following forms only:

   ± (expression) $\dot{x}$ derivative

   ± expression $\dot{x}$ derivative

SECTION V

COMBINED SYSTEM INPUTS

A complete system is usually made up of a number of subsystems not necessarily each being of the same type. NELSIM accepts as input a combination of systems as well as simple systems as seen in the previous sections. A valid combined system entry of the problem into NELSIM is

```
ANALOG DESCRIPTION
      THERMAL
         o
         o
         o
         o
         o
      MECHANICAL
         o
         o
         o
         o
         o
      ELECTRO OPTICAL
         o
         o
         o
         o
         o
      THERMAL
         o
         o
         o
         o
         o
      MECHANICAL
```

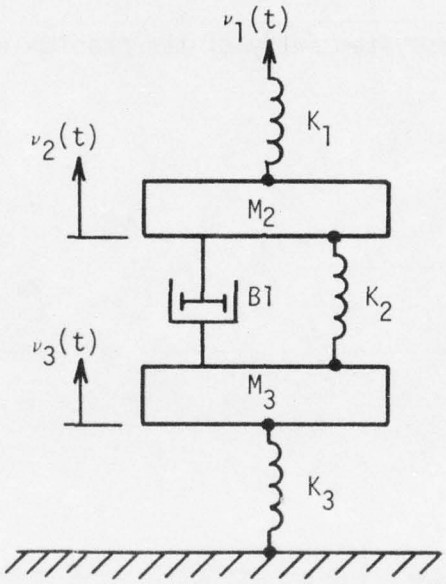
SECTION VI

NELSIM SAMPLE PROBLEMS

The following sample problems have been formulated to illustrate the use of NELSIM with each of the four disciplines considered - mechanical, thermal, electro-mechanical, and electro-optical.

6.1 MECHANICAL SYSTEM EXAMPLES

Problem 1 - The problem depicted in Figure 27



$K_1 = 1$
$K_2 = 2$
$K_3 = 3$
$M_2 = .2$
$M_3 = .3$
$B = 1$
$V_1 = \cos 2t$

Figure 27. Sample Mechanical Problem 1

can be represented as shown in Figure 28 which carries a NELSIM input as shown below.

```
ANALOG DESCRIPTION
MECHANICAL
ELEMENTS
V1,G-3=Q1(TIME)
K3,G-1=3
M3,1-G=.3
K2,1-2=2
B1,1-2=1
M2,2-G=.2
K1,2-3=1
FUNCTIONS
Q1(T)=(COS(2.*T))
OUTPUTS
VM2(V2),VM3(V3),PLOT
RUN CONTROLS
INTEGRATION ROUTINE=IMPLICIT
STOP TIME=10
END
```

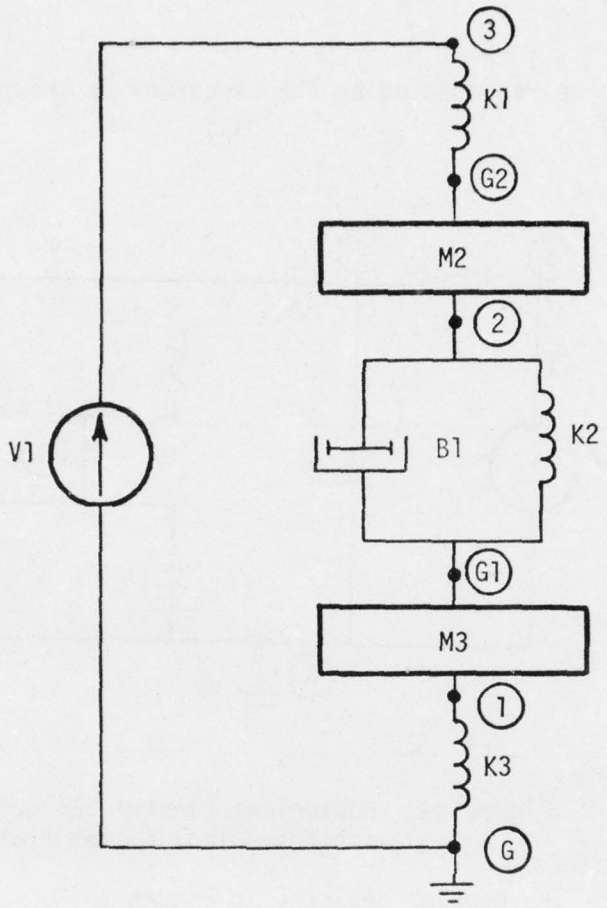


Figure 28. Sample Mechanical Problem 1 Representation to NELSIM

The output from NELSIM will be:

```
CIRCUIT DESCRIPTION
FUNCTIONS
QDIV(A)=(1./A)
Q1(T)=(COS(2.*T))
ELEMENTS
E1          ,G          -3          =Q1(TIME)
L3          ,G          -1          =   3.33333E-01
C3          ,1          -G          =   3.00000E-01
L2          ,1          -2          =   5.00000E-01
R1          ,1          -2          =   1.00000E+00
C2          ,2          -G          =   2.00000E-01
L1          ,2          -3          =   1.00000E+00
OUTPUTS
VC2(V2),PLOT
VC3(V3),PLOT
RUN CONTROLS
STOP TIME          =    1.000E+01
INTEGRATION ROUTINE=IMPLICIT
END
```

This output can be represented by the circuitry as shown in Figure 29.

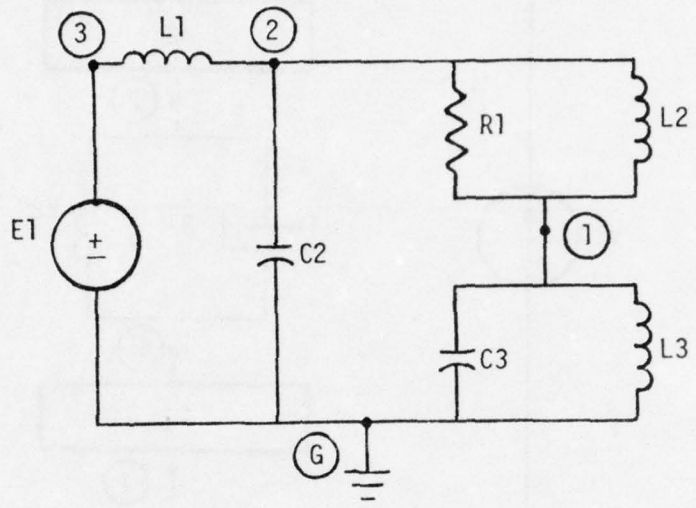


Figure 29. Equivalent Electric Circuit for Mechanical System Problem 1

Problem 2 - The problem depicted in Figure 30.

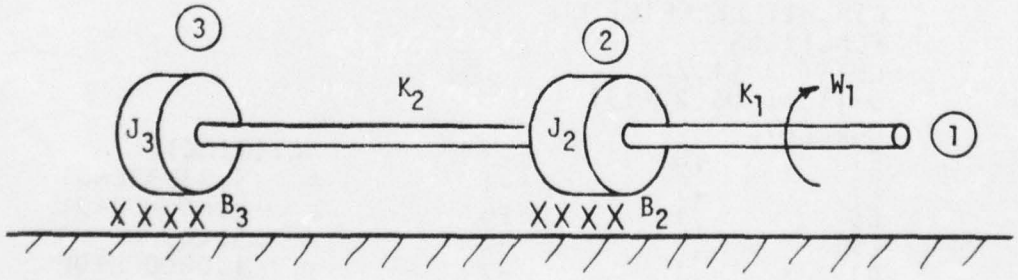


Figure 30. Sample Mechanical Problem 2

Can be represented to NELSIM as:

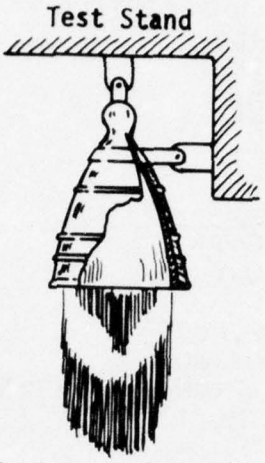
```
ANALOG DESCRIPTION
MECHANICAL
ELEMENTS
W1,G-1=1
K1,1-2=1
J2,2-G=1
K2,2-3=1
J3,3-G=1
B3,J3-G=1
B2,J2-G=1
OUTPUTS
WJ2,WB2,PLOT
RUN CONTROLS
INTEGRATION ROUTINE=IMPLICIT
STOP TIME=1
END
```

The output from NELSIM will be

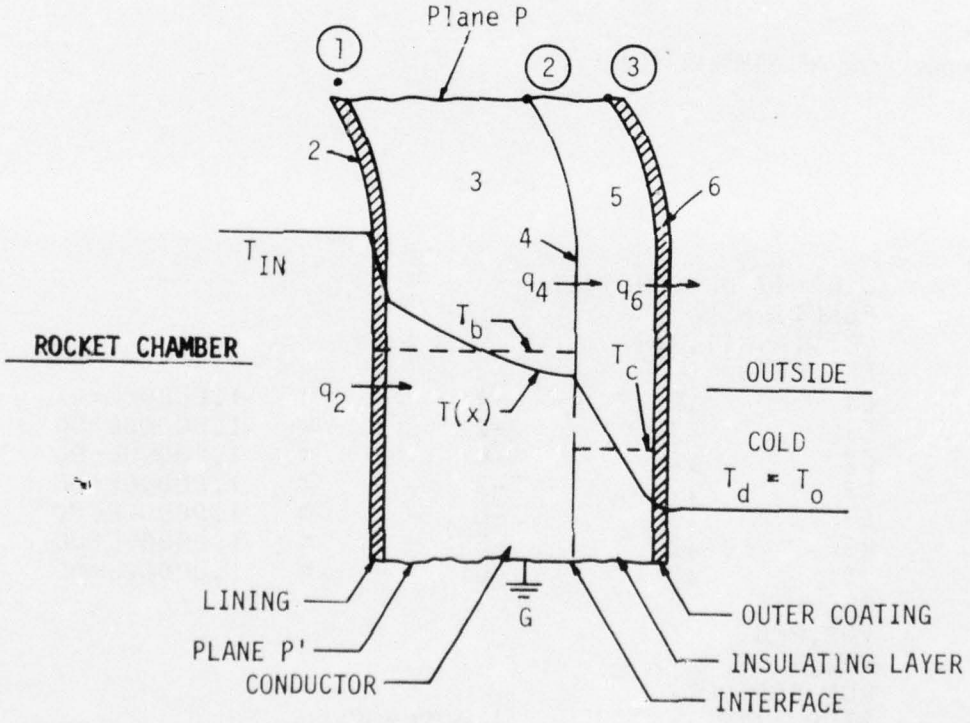
```
CIRCUIT DESCRIPTION
FUNCTIONS
QDIV(A)=(1./A)
ELEMENTS
E1          ,G          -1          =    1.00000E+00
L1          ,1          -2          =    1.00000E+00
C2          ,2          -G          =    1.00000E+00
L2          ,2          -3          =    1.00000E+00
C3          ,3          -G          =    1.00000E+00
R3          ,3          -G          =    1.00000E+00
R2          ,2          -G          =    1.00000E+00
OUTPUTS
VC2,PLOT
VR2,PLOT
RUN CONTROLS
STOP TIME          =    1.000E+00
INTEGRATIONROUTINE=IMPLICIT
END
```

6.2 THERMAL SYSTEM EXAMPLES

Problem 1



(a) Rocket-engine walls



(b) System Model Showing

Figure 31. Thermal System Problem 1

The input of the problem requires the various subdivisions within the wall to be indicated by nodes (1-3). The thermal resistance between subdivisions is simulated using resistors and thermal capacitance using capacitors. The input is shown below.

```
ANALOG DESCRIPTION
THERMAL
ELEMENTS
TIN,G-1=T1
R1,1-2=1
C1,2-G=1
R2,2-3=1
C2,3-G=1
R3,3-4=1
TOUT,G-4=1
FUNCTIONS
T1=0,1,1,1,2,10,5,10,6,1,10,1
OUTPUTS
TIN,QC1,QC2,QR3,PLOT
TR3,TC1,PLOT
RUN CONTROLS
INTEGRATION ROUTINE=IMPLICIT
STOP TIME =15
RUN INITIAL CONDITIONS
END
```

The elements have all been entered as constant values for simplicity but in general will be nonlinear functions of the materials involved and depend on assumptions made by the user. All the nonlinear entry capabilities allowed in SCEPTRE apply to the NELSIM modules. The output of NELSIM is shown below with the elements and output requests altered to electrical analogies.

```
CIRCUIT DESCRIPTION
FUNCTIONS
QDIV(A)=(1./A)
T1=0,1,1,1,2,10,5,10,6,1,10,1
ELEMENTS
EIN         ,G          -1          =T1
R1          ,1          -2          =    1.00000E+00
C1          ,2          -G          =    1.00000E+00
R2          ,2          -3          =    1.00000E+00
C2          ,3          -G          =    1.00000E+00
R3          ,3          -4          =    1.00000E+00
EOUT        ,G          -4          =    1.00000E+00
OUTPUTS
EIN,PLOT
IC1,PLOT
IC2,PLOT
IR3,PLOT
VR3,PLOT
VC1,PLOT
RUN CONTROLS
STOP TIME          =    1.500E+01
INTEGRATIONROUTINE=IMPLICIT
RUN INITIALCONDITIONS
END
```

Problem 2 - For the rocket motor shown in Figure 32, heat is transferred by convection and radiation to the chamber wall, then through the wall by conduction, and finally to the coolant by convection

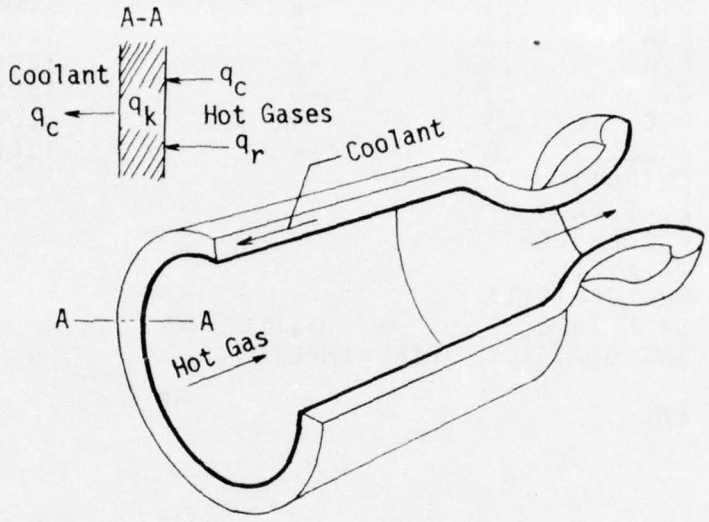


Figure 32. Thermal System Problem 2

This problem is described to NELSIM as follows:

```
ANALOG DESCRIPTION
THERMAL
ELEMENTS
TGAS,G-1=T1
R1R,1-2=1
R1C,1-2=1
R2K,2-3=1
C2,3-G=1
R3C,3-4=1
TCOOL,G-4=1
FUNCTIONS
T1=0,1,1,1,2,10,5,10,6,1,10,1
OUTPUT
TGAS,TCOOL,PLOT
TR2K,PLOT
RUN CONTROLS
INTEGRATION ROUTINE=IMPLICIT
STOP TIME=10
END
```

The produced NELSIM output for this case will be:

```
CIRCUIT DESCRIPTION
FUNCTIONS
QDIV(A)=(1./A)
T1=0,1,1,1,2,10,5,10,6,1,10,1
ELEMENTS
EGAS        ,G          -1          =T1
R1K         ,1          -2          =    1.00000E+00
R1C         ,1          -2          =    1.00000E+00
R2K         ,2          -3          =    1.00000E+00
C2          ,3          -G          =    1.00000E+00
R3C         ,3          -4          =    1.00000E+00
ECOOL       ,G          -4          =    1.00000E+00
OUTPUTS
EGAS,PLOT
ECOOL,PLOT
VR2K,PLOT
RUN CONTROLS
STOP TIME          =    1.000E+01
INTEGRATION.ROUTINE=IMPLICIT

END
```

6.3 ELECTRO-MECHANICAL EXAMPLES

Problem 1 - For the galvanometer shown in Figure 33.

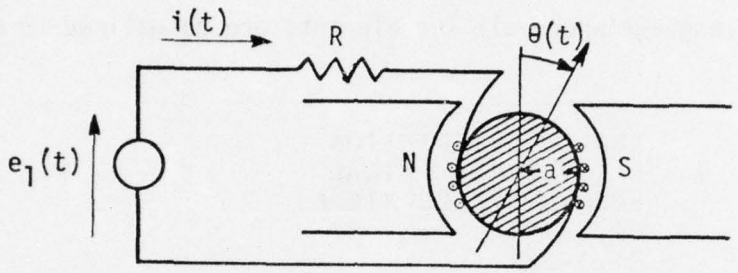


Figure 33. Electro-Mechanical Problem 1

Figure 34 represents the equivalent lumped network in terms of its electrical and mechanical parts.

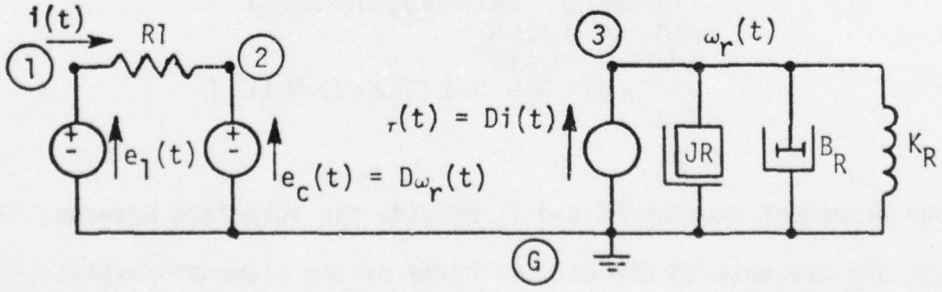


Figure 34. Lumped Element Galvanometer Equivalent

where J and B represent the moment of inertia of the rotor and the viscous damping that results from air friction. The element K_R represents the rotational stiffness produced by a spring attached to the rotor. The constant D is defined to be $B\ell a$ where B is the magnetic flux density, ℓ is the total length of the conductor and a is the radius of the rotor.

The input language where all the elements are normalized to one is shown below.

```
ANALOG DESCRIPTION
ELECTRO MECHANICAL
ELEMENTS(ELECTRICAL)
E1,G-1=1
R1,1-2=1
EC,G-2=QEC(PD,WJR)
ELEMENTS(MECHANICAL)
T1,G-3=QEC(PD,IR1)
JR,3-G=1
BR,3-G=1
KR,3-G=1
DEFINED PARAMETERS
PD=QPD(PA,PL,PB)
PA=1
PB=1
PC=1
PL=1
FUNCTIONS
QEC(A,B)=(A*B)
QPD(A,B,C)=(A*B*C)
OUTPUTS
WKR(WKR),TKR(TKR),VR1,PLOT
RUN CONTROLS
STOP TIME=10
INTEGRATION ROUTINE=IMPLICIT
END
```

The two dependent sources EC and T1 provide the interface between the two systems and are entered directly in terms of the elements involved. The output of NELSIM is shown below with the electrical analog represented shown in Figure 35. As can be seen, the network is entirely in terms of electrical quantities.

```
CIRCUIT DESCRIPTION
FUNCTIONS
QDIV(A)=(1./A)
QEC(A,B)=(A*B)
QPD(A,B,C)=(A*B*C)
ELEMENTS
J1          ,G          -3          =QEC(PD,IR1)
CR          ,3          -G          =    1.00000E+00
RR          ,3          -G          =    1.00000E+00
LR          ,3          -G          =    1.00000E+00
E1          ,G          -1          =    1.00000E+00
R1          ,1          -2          =    1.00000E+00
EC          ,G          -2          =QEC(PD,VCR)
DEFINED PARAMETERS
PD          =QPD(PA,PL,PB)
PA          =    1.00000E+00
PB          =    1.00000E+00
PC          =    1.00000E+00
PL          =    1.00000E+00
OUTPUTS
VLR(WKR),PLOT
ILR(TKR),PLOT
VR1,PLOT
RUN CONTROLS
STOP TIME          =    1.000E+01
INTEGRATION.ROUTINE=IMPLICIT
END
```

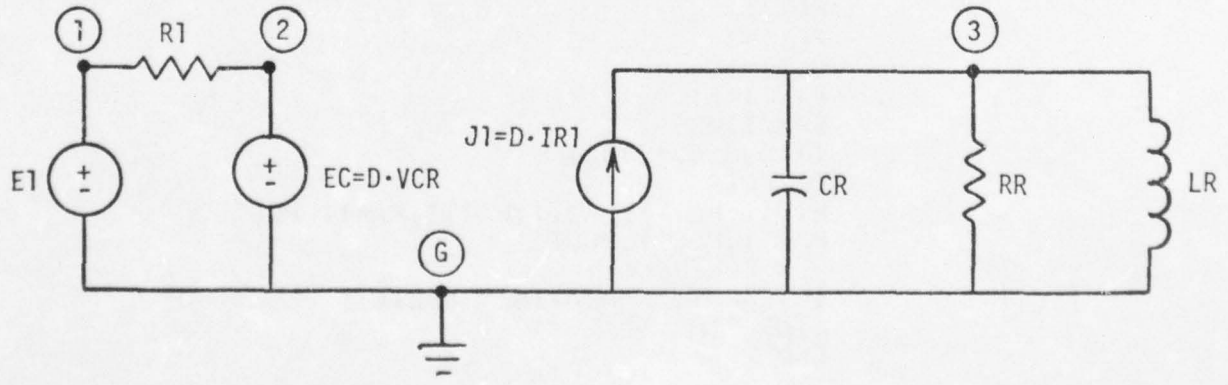
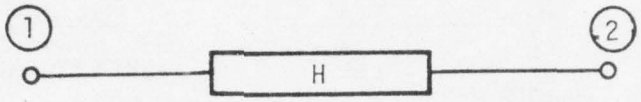


Figure 35. Galvanometer Electrical Analog

Problem 2 - Implementation of the accelerometer model built-in to the electro-mechanical system is accomplished below. The usage and response of the model is illustrated for the case where all gains have been normalized to 1 and the input to the model is a unit step. Consider Figure below where H is the accelerometer.



The input to NELSIM is given below:

```
ANALOG DESCRIPTION
ELECTRO MECHANICAL
MODELS
T1,IN-CU=ACCELEROMETER
DEFINED PARAMETERS
PAT1=1
PBT1=1
PCT1=1
PDT1=1
PET1=1
PFT1=1
PGT1=1
PINT1=T1
FUNCTIONS
T1=0,0,0,1,10,1
OUTPUTS
PINT1,PA1T1,PA2T1,PA3T1,PA4T1,PLOT
PA5T1,POUT1,PLOT
RUN CONTROLS
INTEGRATION ROUTINE=IMPLICIT
STOP TIME=10
END
```

All gains within the model are provided by the user with the model label (T1) added to each name.

AD-A033 606 TRW SYSTEMS GROUP REDONDO BEACH CALIF NUCLEAR SURVIV--ETC F/G 9/2
NONELECTRICAL LANGUAGES SIMULATION MODULE (NELSIM). VOLUME II.(U)
SEP 76 M EPSTEIN, J R PISTACCHI F29601-73-C-0024

UNCLASSIFIED AFWL-TR-73-256-VOL-2 NL

2 OF 2
AD
A033606



END
DATE
FILMED
2-77

The output generated by the program is shown below.

```
CIRCUIT DESCRIPTION
FUNCTIONS
ODIV(A)=(1./A)
T1=0,0,0,1,10,1
QMA1(A,B,C)=(A*B-C)
QMA2(A,B,C,D)=((A*B-C)/D)
QMA3(A,B)=(A*B)
QMA4(A,B,C)=((A-B)/C)
QMA5(A,B,C,D)=(A*((B-C)/D))
QMA6(A,B)=(A/B)
DEFINED PARAMETERS
PAT1        =    1.00000E+00
PBT1        =    1.00000E+00
PCT1        =    1.00000E+00
PDT1        =    1.00000E+00
PET1        =    1.00000E+00
PFT1        =    1.00000E+00
PGT1        =    1.00000E+00
PINT1       =T1
PA1T1=QMA1(PINT1,PAT1,PA4T1)
DPA2T1=PA7T1
DPA7T1=QMA2(PA1T1,PBT1,PA7T1,PCT1)
PA3T1=QMA3(PA2T1,PDT1)
DPA6T1=QMA4(PA3T1,PA6T1,PFT1)
PA4T1=QMA5(PET1,PA3T1,PA6T1,PFT1)
DPA5T1=QMA6(PA4T1,PGT1)
DPOUT1=PA5T1
PA2T1=0
PA5T1=0
PA6T1=0
PA7T1=0
POUT1=0
OUTPUTS
PINT1,PLOT
PA1T1,PLOT
PA2T1,PLOT
PA3T1,PLOT
PA4T1,PLOT
PA5T1,PLOT
POUT1,PLOT
RUN CONTROLS
STOP TIME           =    1.000E+01
INTEGRATION.ROUTINE=IMPLICIT
END
```

6.4 ELECTRO-OPTICAL SYSTEM EXAMPLES

Figure 36 details the equivalent circuit selected to simulate the AC characteristics of photo diodes.

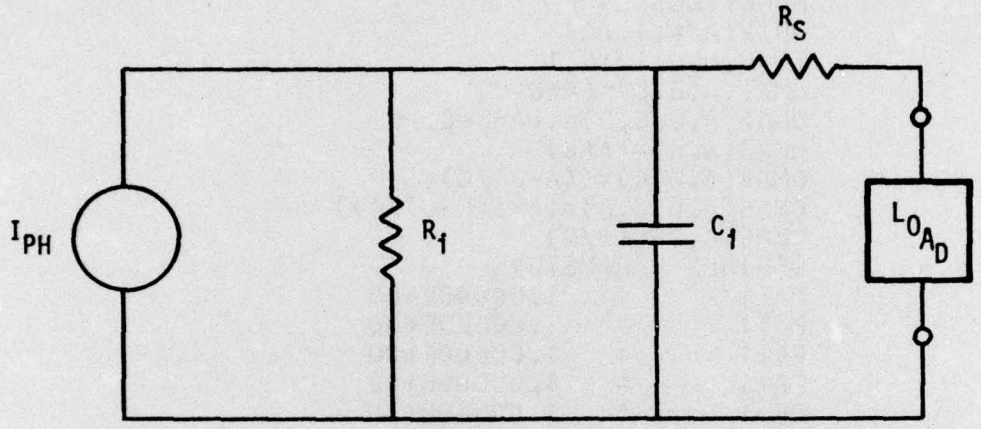


Figure 36. Equivalent Circuit of a Photo Diode

As detailed in Volume I, the photocurrent, I_{PH} is expressed as

$$I_{PH} = (\eta q \, P/hV)$$

Problem 1

The normalized power input at 5 m to an $I_N S_b$ device is described as

POWER	TIME (SEC)
0	0
.2	2
.4	6
.85	10
.6	1
0	15

Using

$$q = 25\%$$

$$R_S = 18\,\Omega$$

$$C_j = 7.1\ \text{pf}$$

$$R_j = 6\,\Omega$$

$$R_{LOAD} = 12\,\Omega$$

Plot the current time history at the load and at the source.

The NELSIM input is

```
ANALOG DESCRIPTION
ELECTRO OPTICAL
ELEMENTS(ELECTRICAL)
JPD,G-1=CEO(PEFF,PCHG,PCON,POFR,TPL)
RI,2-G=6
CI,3-G=3
RS,3-4=12
RL,4-G=12
FUNCTIONS
TPL=0,0,.2,2,.4,6,.85,10,.6,12,0,15
CEO(A,B,C,D,E)=(A*B/(C*D)*E)
DEFINED PARAMETERS
PEFF=.25
PCHG=1E-7
PCON=123E-4
POFR=37
OUTPUTS
JPD,JRL,PLOT
RUN CONTROLS
INTEGRATION ROUTINE=IMPLICIT
STOP TIME=15
END
```

For which the NELSIM output is

```
CIRCUIT DESCRIPTION
FUNCTIONS
QDIV(A)=(1./A)
TPL=0,0,.2,2,.4,6,.85,10,.6,12,0,15
QEO(A,B,C,D,E)=(A*B/(C*D)*E)
ELEMENTS
JPD         ,G          -1          =QEO(PEFF,PCHG,PCON,POFR,TPL)
RI          ,2          -G          =   6.00000E+00
CI          ,3          -G          =   3.00000E+00
RS          ,3          -4          =   1.20000E+01
RL          ,4          -G          =   1.20000E+01
DEFINED PARAMETERS
PEFF        =   2.50000E-01
PCHG        =   1.00000E-07
PCON        =   1.23000E-02
POFR        =   3.70000E+01
OUTPUTS
JPD,PLOT
JRL,PLOT
RUN CONTROLS
STOP TIME          =   1.500E+01
INTEGRATION ROUTINE=IMPLICIT
END
```

Problem 2

Rerun Problem 1 using an $I_N A_S$ device whose characteristics are

$$q = 25\%$$

$$R_S = 12\,\Omega$$

$$C_i = 3\text{ pf}$$

$$R_i = 6\,\Omega$$

For the same load and compare results.

The NELSIM input and output are

```
ANALOG DESCRIPTION
ELECTRO OPTICAL
ELEMENTS(ELECTRICAL)
JPD,G-1=QEO(PEFF,PCHG,PCON,POFR,TPL)
RI,2-G=6
CI,3-G=7.1
RS,3-4=18
RL,4-G=12
FUNCTIONS
TPL=0,0,.2,2,.4,6,.85,10,.6,12,0,15
QEO(A,B,C,D,E)=(A*B/(C*D)*E)
DEFINED PARAMETERS
PEFF=.25
PCHG=1E-7
PCON=123E-4
POFR=37
OUTPUTS
JPD,JRL,PLOT
RUN CONTROLS
INTEGRATION ROUTINE=IMPLICIT
STOP TIME=15
END
```

```
CIRCUIT DESCRIPTION
FUNCTIONS
QDIV(A)=(1./A)
TPL=0,0,.2,2,.4,6,.85,10,.6,12,0,15
QEO(A,B,C,D,E)=(A*B/(C*D)*E)
ELEMENTS
JPD       ,G          -1          =QEO(PEFF,PCHG,PCON,POFR,TPL)
RI        ,2          -G          =   6.00000E+00
CI        ,3          -G          =   7.10000E+00
RS        ,3          -4          =   1.80000E+01
RL        ,4          -G          =   1.20000E+01
DEFINED PARAMETERS
PEFF      =   2.50000E-01
PCHG      =   1.00000E-07
PCON      =   1.23000E-02
POFR      =   3.70000E+01
OUTPUTS
JPD,PLOT
JRL,PLOT
RUN CONTROLS
STOP TIME       =   1.500E+01
INTEGRATIONROUTINE=IMPLICIT
END
```

DISTRIBUTION LIST

DEPARTMENT OF DEFENSE

Director
Armed Forces Radiobiology Research Institute
Defense Nuclear Agency
ATTN: Robert E. Carter
ATTN: Tech. Lib.

Director
Defense Communications Agency
ATTN: Code 540.5
ATTN: Code 930, Monte I. Burgett, Jr.

Defense Documentation Center
12 cy ATTN: TC

Director
Defense Nuclear Agency
ATTN: DDST
ATTN: STSI
3 cy ATTN: STTL, Tech. Lib.

Headquarters
European Command
J-5
ATTN: ECJ6-PF

AUL
ATTN: LDE

Commander
Field Command
Defense Nuclear Agency
ATTN: FCPR

Chief
Livermore Division, Field Command, DNA
ATTN: FCPRL

OJCS/J-3
ATTN: J-3, RDTA Br., WWMCCS, Plans Div.

DEPARTMENT OF THE ARMY

Commander
Frankford Arsenal
ATTN: SARFA-FCD, Marvin Elnick

Commander
Harry Diamond Laboratories
ATTN: AMXDO-EM, R. Bostak
ATTN: AMXDO-RB, Joseph R. Miletta
ATTN: AMXDO-RBI, John A. Rosado
ATTN: AMXDO-RCC, John E. Thompkins
ATTN: AMXDO-EM, Robert F. Gray
ATTN: AMXDO-EM, Raphael Wong
ATTN: AMXDO-RC, Robert B. Oswald, Jr.
ATTN: AMXDO-EM, J. W. Beilfuss

Commander
Picatinny Arsenal
ATTN: SMUPA-FR-S-P, Lester W. Doremus
ATTN: SARPA-ND-N
ATTN: SARPA-ND-C-E, Amina Nordio

Commander
TRASANA
ATTN: ATAA-EAC, Francis N. Winans

DEPARTMENT OF THE ARMY (Continued)

Director
U.S. Army Ballistic Research Labs.
ATTN: AMXBR-X, Julius J. Meszaros
ATTN: AMXBR-VL, John W. Kinch
ATTN: AMXRD-BVL, David L. Rigotti

Commander
U.S. Army Electronics Command
ATTN: AMSEL-GG-TD, W. R. Werk
ATTN: AMSEL-TL-MD, Gerhart K. Gaule
ATTN: AMSEL-TL-IR, Edwin T. Hunter

Commanding Officer
U.S. Army Electronics Command
ATTN: CPT Allan S. Parker

Commander
U.S. Army Electronics Proving Ground
ATTN: STEEP-MT-M, Gerald W. Durbin

Commandant
U.S. Army Field Artillery School
ATTN: ATSFA-CTD-ME, Harley Moberg

Commander
U.S. Army Mat. & Mechanics Rsch. Ctr.
ATTN: AMXMR-HH, John F. Dignam

Commander
U.S. Army Materiel Dev. & Readiness Cmd.
ATTN: AMCRD-WN-RE, John F. Corrigan

Commander
U.S. Army Missile Command
ATTN: AMSI-RGP, Victor W. Ruwe
ATTN: AMSMI-RGP, Hugh Green
ATTN: AMSMI-RRR, Faison P. Gibson

Commander
U.S. Army Nuclear Agency
ATTN: ATCN-W, LTC Leonard A. Sluga

Project Manager
U.S. Army Tactical Data Systems, AMC
ATTN: Dwaine B. Huewe

Commander
White Sands Missile Range
ATTN: STEWS-TE-NT, Marvin P. Squires

DEPARTMENT OF THE NAVY

Chief of Naval Research
Navy Department
ATTN: Code 427

Commanding Officer
Naval Ammunition Depot
ATTN: Code 7024, James Ramsey

Commander
Naval Electronic Systems Command
ATTN: ELEX 05323, Cleveland F. Watkins
ATTN: PME 117-21
ATTN: Code 50451

DEPARTMENT OF THE NAVY (Continued)

Commander
Naval Electronics Laboratory Center
ATTN: H. F. Wong

Director
Naval Research Laboratory
ATTN: Code 2627, Doris R. Folen

Commander
Naval Surface Weapons Center
ATTN: Code 431, Edwin B. Dean
ATTN: Code WX21, Tech. Lib.

Commander
Naval Surface Weapons Center
ATTN: Code FUR, Robert A. Amadori

Commanding Officer
Naval Weapons Evaluation Facility
ATTN: Code ATG, Mr. Stanley

Director
Strategic Systems Project Office
ATTN: SP-2701, John W. Pitsenberger

DEPARTMENT OF THE AIR FORCE

Commander
Aeronautical Systems Division, AFSC
ATTN: ASD-YH-EX, Lt Col Robert Leverette

AF Aero-Propulsion Laboratory, AFSC
ATTN: POD, P. E. Stover

AF Weapons Laboratory, AFSC
ATTN: ELA
ATTN: ELC
ATTN: ELP, Carl E. Baum
ATTN: HO
2 cy ATTN: SUL

AFTAC
ATTN: TAF

Air Force Avionics Laboratory, AFSC
ATTN: AFAL, TEA, Hans J. Hennecke
ATTN: AFAL, AAA

Commander
Foreign Technology Division, AFSC
ATTN: FTD, PDJC

Commander
Ogden Air Logistics Center
ATTN: MMEWM, Robert Joffs

SAMSO/DY
ATTN: DYS, Maj Larry A. Darda

SAMSO/YD
ATTN: YDD, Maj Marion P. Schneider

Commander in Chief
Strategic Air Command
ATTN: XPFS, Maj Brian G. Stephan

ENERGY RESEARCH & DEVELOPMENT ADMINISTRATION

Los Alamos Scientific Laboratory
ATTN: Doc. Con. for Bruce W. Noel
ATTN: Doc. Con. for J. Arthur Freed

ENERGY RESEARCH & DEVELOPMENT ADMINISTRATION (Continued)

University of California
Lawrence Livermore Laboratory
ATTN: Donald J. Meeker, L-153
ATTN: E. K. Miller, L-156
ATTN: Lawrence Cleland, L-156
ATTN: Frederick R. Kovar, L-94
ATTN: William J. Hogan, L-531

DEPARTMENT OF DEFENSE CONTRACTORS

Aerojet Electro-Systems Co. Div.
Aerojet-General Corporation
ATTN: Thomas D. Hanscome

Aeronutronic Ford Corporation
Aerospace & Communications Ops.
ATTN: E. R. Poncelet, Jr.
ATTN: Ken C. Attinger
ATTN: Tech. Info. Section

Aeronutronic Ford Corporation
Western Development Laboratories Div.
ATTN: Samuel R. Crawford, M.S. 531

Avco Research & Systems Group
ATTN: Research Library, A-830, Rm. 7201

The BDM Corporation
ATTN: T. H. Neighbors
ATTN: William Druen

Bell Aerospace Company
Division of Textron, Inc.
ATTN: Carl B. Schoch, Wpns. Effects Grp.

The Bendix Corporation
Communication Division
ATTN: Doc. Con.

The Bendix Corporation
Research Laboratories Division
ATTN: Donald J. Niehaus, Mgr. Prgm.

The Boeing Company
ATTN: Robert S. Caldwell, M.S. 2R-00
ATTN: Donald W. Egelkrout, M.S. 2R-00
ATTN: David L. Dye, M.S. 87-75
ATTN: Aerospace Library
ATTN: Howard W. Wicklein, M.S. 17-11

Booz-Allen & Hamilton, Inc.
ATTN: Raymond J. Chrisner

Brown Engineering Company, Inc.
ATTN: David L. Lambert, M.S. 18

Charles Stark Draper Laboratory, Inc.
ATTN: Kenneth Fertig
ATTN: Paul R. Kelly

Computer Sciences Corporation
ATTN: Richard H. Dickhaut

Cutler-Hammer, Inc.
AIL Division
ATTN: Anne Anthony, Central Tech. Files

DEPARTMENT OF DEFENSE CONTRACTORS (Continued)

University of Denver
Colorado Seminary
ATTN: Sec. Officer for Fred P. Venditti

The Dikewood Corporation
ATTN: L. Wayne Davis

E-Systems, Inc.
Greenville Division
ATTN: Library, 8-50100

Exp. & Math. Physics Consultants
ATTN: Thomas M. Jordan

Fairchild Industries, Inc.
ATTN: Mgr., Config. Data & Standards

The Franklin Institute
ATTN: Ramie H. Thompson

Garrett Corporation
ATTN: Robert E. Weir, Dept. 93-9

General Electric Company
Space Division
ATTN: Larry I. Chasen
ATTN: John R. Greenbaum
ATTN: Joseph C. Peden, CCF 8301
ATTN: John L. Andrews
ATTN: James P. Spratt

General Electric Company
Re-Entry & Environmental Systems Div.
ATTN: Robert V. Benedict

General Electric Company
TEMPO-Center for Advanced Studies
ATTN: DASIAC
ATTN: William McNamera
ATTN: Royden R. Rutherford
ATTN: M. Espig

General Electric Company
ATTN: CSP 0-7, L. H. Dee

General Electric Company
Aerospace Electronics Systems
ATTN: W. J. Patterson, Drop 233
ATTN: George Francis, Drop 233

General Electric Company-TEMPO
ATTN: DASIAC for William Alfonte

General Research Corporation
ATTN: Robert D. Hill

General Research Corporation
Washington Operations
ATTN: David K. Osias

GTE Sylvania, Inc.
Electronics Systems Grp.-Eastern Div.
ATTN: Leonard L. Blaisdell
ATTN: James A. Waldon

GTE Sylvania, Inc.
ATTN: Charles H. Ramsbottom
ATTN: Herbert A. Ullman
ATTN: David P. Flood

DEPARTMENT OF DEFENSE CONTRACTORS (Continued)

Gulton Industries, Inc.
Engineered Magnetics Division
ATTN: Eng. Magnetics Div.

Harris Corporation
Harris Semiconductor Division
ATTN: T. L. Clark, M.S. 4040
ATTN: Wayne E. Abare, M.S. 16-111
ATTN: Carl F. Davis, M.S. 17-220

Hazeltine Corporation
ATTN: M. Waite, Tech. Info. Ctr.

Honeywell, Incorporated
Government & Aeronautical Products Division
ATTN: Ronald R. Johnson, A-1622

Honeywell, Incorporated
Aerospace Division
ATTN: Stacey H. Graff, M.S. 725-J
ATTN: James D. Allen, M.S. 775-D
ATTN: Harrison H. Noble, M.S. 725-5A

Honeywell, Incorporated
ATTN: Tech. Lib.

Hughes Aircraft Company
ATTN: Billy W. Campbell, M.S. 6-E-110
ATTN: Kenneth R. Walker, M.S. D-157

Hughes Aircraft Company
Space Systems Division
ATTN: William W. Scott, M.S. A-1080
ATTN: Edward C. Smith, M.S. A-620

IBM Corporation
ATTN: Frank Frankovsky
ATTN: Harry W. Mathers, Dept. M-41

IIT Research Institute
ATTN: Irving N. Mindel

Intelcom Rad Tech
ATTN: R. L. Mertz
ATTN: Leo D. Cotter
ATTN: Eric P. Wenaas
ATTN: MDC

Kaman Sciences Corporation
ATTN: John R. Hoffman
ATTN: Albert P. Bridges
ATTN: Donald H. Bryce
ATTN: W. Foster Rich
ATTN: Walter E. Ware

Litton Systems, Inc.
Guidance & Control Systems Division
ATTN: R. W. Maughmer
ATTN: Val J. Ashby, M.S. 67

Lockheed Missiles & Space Co., Inc.
ATTN: George F. Heath, Dept. 81-14
ATTN: Benjamin T. Kimura, Dept. 81-14
ATTN: Hans L. Schneemann, Dept. 81-64

LTV Aerospace Corporation
ATTN: Technical Data Ctr.

DEPARTMENT OF DEFENSE CONTRACTORS (Continued)

Martin Marietta Aerospace
Orlando Division
ATTN: Jack M. Ashford, MP-537
ATTN: Mona C. Griffith, Library, MP-30
ATTN: William W. Mras, MP-413

Martin Marietta Corporation
Denver Division
ATTN: J. E. Goodwin, Mail 0452
ATTN: Paul G. Kase, Mail 8203

McDonnell Douglas Corporation
ATTN: Tom Ender
ATTN: Tech. Lib.

McDonnell Douglas Corporation
ATTN: Stanley Schneider
ATTN: Raymond J. DeBattista

McDonnell Douglas Corporation
ATTN: Tech. Lib., C1-290/36-84

Mission Research Corporation
ATTN: William C. Hart

Mission Research Corporation
ATTN: J. Roger Hill
ATTN: David E. Merewether

The Mitre Corporation
ATTN: M. E. Fitzgerald

National Academy of Sciences
ATTN: National Materials Advisory Board, for R. S. Shane, Nat. Materials Advsy.

Northrop Corporation
Electronic Division
ATTN: Boyce T. Ahlport
ATTN: George H. Towner
ATTN: Vincent R. DeMartino

Northrop Corporation
ATTN: Orlie L. Curtis, Jr.
ATTN: James P. Raymond
ATTN: David N. Pocock

Northrop Corporation
Electronic Division
ATTN: Joseph D. Russo

Physics International Company
ATTN: Doc. Con. for John H. Huntington

R & D Associates
ATTN: S. Clay Rogers
ATTN: Leonard Schlessinger

The Rand Corporation
ATTN: Cullen Crain

Raytheon Company
ATTN: Gajanan H. Joshi, Radar Sys. Lab.

Raytheon Company
ATTN: James R. Weckback
ATTN: Harold L. Flescher

DEPARTMENT OF DEFENSE CONTRACTORS (Continued)

RCA Corporation
Government & Commercial Systems
ATTN: George J. Brucker

RCA Corporation
ATTN: E. Van Keuren, 13-5-2

Rockwell International Corporation
ATTN: James E. Bell, HA-10
ATTN: George C. Messenger, FB-61

Rockwell International Corporation
Electronics Operations
ATTN: Mildred A. Blair
ATTN: Alan A. Langenfeld
ATTN: Dennis Sutherland

Sanders Associates, Inc.
ATTN: Moe L. Aitel, NCA, 1-3236

Science Applications, Inc.
ATTN: Frederick M. Tesche

Science Applications, Inc.
ATTN: William L. Chadsey

Science Applications, Inc.
ATTN: J. Robert Beyster
ATTN: Larry Scott

Science Applications, Inc.
Huntsville Division
ATTN: Noel R. Byrn

Simulation Physics, Inc.
ATTN: John R. Uglum

The Singer Company
ATTN: Irwin Goldman, Eng. Management

Sperry Flight Systems Division
Sperry Rand Corporation
ATTN: D. Andrew Schow

Sperry Rand Corporation
Sperry Division
ATTN: Paul Marraffino

Stanford Research Institute
ATTN: Philip J. Dolan
ATTN: Arthur Lee Whitson

Sundstrand Corporation
ATTN: Curtis B. White

Systron-Donner Corporation
ATTN: Gordon B. Dean

Texas Instruments, Inc.
ATTN: Gary F. Hanson
ATTN: Donald J. Manus, M.S. 72

Texas Tech. University
ATTN: Travis L. Simpson

DEPARTMENT OF DEFENSE CONTRACTORS (Continued)

TRW Systems Group
ATTN: A. M. Liebschutz, R1-1162
ATTN: Richard H. Kingsland, R1-2154
ATTN: A. A. Witteles, R1-1120
ATTN: Aaron H. Narevsky, R1-2144
ATTN: Benjamin Sussholtz
ATTN: Lillian D. Singletary, R1-1070
ATTN: Jerry I. Lubell
ATTN: M. Epstein
ATTN: J. R. Pistacchi

TRW Systems Group
San Bernardino Operations
ATTN: John E. Dahnke
ATTN: H. S. Jensen

DEPARTMENT OF DEFENSE CONTRACTORS (Continued)

United Technologies Corporation
Hamilton Standard Division
ATTN: Raymond G. Giguere

Victor A. J. Van Lint, Consultant
Mission Research Corporation
ATTN: V. A. J. Van Lint

Westinghouse Electric Corporation
ATTN: Henry P. Kalapaca, M.S. 3525

Official Record Copy (P.O.)