NAVORD REPORT 4412

AN ADES ENCODER FOR THE 650 CALCULATOR

19 DECEMBER 1956

U. S. NAVAL ORDNANCE LABORATORY
WHITE OAK, MARYLAND

NAVORD Report 4412

Aeroballistic Research Report 360

AN ADES ENCODER FOR THE 650 CALCULATOR

Prepared by:

E. K. Blum and S. Stern

ABSTRACT: An experimental model of an Encoder has been constructed to translate the mathematical language of ADES into programs for the 650 Magnetic Drum Calculator. This model was used primarily to demonstrate that the logical design was sound, and that ADES could actually be put into operation. Several problems were programmed and then computed successfully. Although this particular model is not recommended for production, it can be studied profitably as a source of ideas for machine techniques which can be used in a production model. The experience gained with this model indicates that ADES is a feasible automatic programming system for computers with magnetic tape storage.

U. S. NAVAL ORDNANCE LABORATORY
WHITE OAK, MARYLAND

19 December 1956

This report describes an experimental model of a device for translating mathematical formulas into programs for digital computers. This device is part of a general system of automatic programming called ADES. The model proves the feasibility of such an automatic system. The experimental model is designed for use with the 650 Magnetic Drum Calculator, but embodies the logical design for a general device to be used with any computer.

The data and conclusions presented here are the opinion of the author and do not necessarily represent the final judgment of the Laboratory.

This work was performed under Task No. FR-30-1-57.

W. W. WILBOURNE
Captain, USN
Commander

H. H. KURZWEG
By direction

PREFACE

It is assumed that the reader is familiar with NAVORD Reports 4209 and 4411, which describe the ADES language and the logical design of the ADES Encoder. They shall be referred to frequently in the ensuing pages. As described in those reports, ADES was still only a system on paper. It was decided to construct an actual Encoder to prove to interested, but somewhat skeptical, persons that such a device would really operate as claimed. The job was undertaken with the foreknowledge that the computer to be used to synthesize the Encoder was not entirely adequate. Nevertheless, since the 650 Calculator (without tapes) was the only computer directly accessible, it was chosen, with the understanding that the resulting device would be regarded only as an experimental model of the Encoder.

The experimental model was completed by two persons in about three months. It consists of approximately three thousand 650 instructions, which are read on to the drum in phases. This model does not include all the features of the ADES II Encoder described in report 4411. It is estimated that to do this would require an additional two thousand instructions at most. Had magnetic tapes been available, the complete ADES II Encoder could have been built and operated successfully on the 650 calculator. The results obtained with the experimental model amply justify a future undertaking of this kind. It is hoped that Encoders will be built for the larger and faster machines now in use.

We wish to acknowledge the fine work of Mrs. M. Zawatzky who assisted in preparing this report for publication.

CONTENTS

CONTENTS (continued)

AN ADES ENCODER FOR THE IBM 650 CALCULATOR

I. INTRODUCTION

In NAVORD Reports 4209 and 4411, a system for the automatic programming of mathematical problems for a digital computer was described. This system, called ADES, consists of a language similar to ordinary mathematical notation, a translating device called the Encoder and a digital computer. ADES is a general system designed for use with any modern electronic computer. In this report, we describe its application to the 650 magnetic drum calculator.

The 650 calculator at NOL is used with an interpretive speed-code system which converts the machine to a single-address computer. Interpretive codes are provided for the usual arithmetical (floating-point) and logical operations. There is also a means of modifying instructions by using index registers. (See NAVORD 4169.) An Encoder which translates from the ADES mathematical notation into the speed-code language has been constructed, that is, the Encoder will accept the formulation of a problem in ADES language as input and produce a program of speed-code instructions as output. This program can later be loaded into the speed-code computer together with data and the problem can be run.

We shall refer to this particular Encoder as Model I. It was constructed primarily for experimental purposes and has certain limitations which were imposed by insufficient storage. For example, it will accept only those formulations which can be written with variables of degree one at most. No auxiliary f-equations (see NAVORD 4209) are permitted. Only a special form of minimization is allowed, as will be explained more fully below. However, aside from these restrictions, this Encoder contains most of the essential design features required of a device for translating ADES language into programs for any digital computer. Therefore, it should be of interest even to those computer groups which do not have a 650 calculator.

The general logical design is given in NAVORD Report 4411. In the present report, we give the details of the construction of the Model I Encoder, listing the 650 instructions used. This should serve as an illustration of how an actual Encoder can be synthesized with a digital computer. Further, many of the techniques should be adaptable to similar machines.

II. General Description of Model I

To avoid confusion with the speed-code computer, we shall refer to the translating subroutines of the Encoder as "schemata". These schemata are

composed of basic 650 instructions, punched on IBM cards, 6 instructions per card. There are approximately 3500 instructions in all. Since the capacity of the 650 drum is only 2000 words, and tapes were not available, the schemata cards were arranged to be loaded in phases.

The ADES formulation of a problem is likewise punched on IBM cards. Each alphabetical symbol in the formulation consists of a letter and a two-digit subscript, and occupies a three-column field; e.g., a_{32} is punched in the card as a32. There are 10 fields per card. Numerical symbols are also permitted, but these are limited to 3-digit positive integers. Symbols are punched in the cards in the order in which they are written in the formulation. The formulation cards must therefore be in order. Model I will accept formulations containing up to 600 symbols.

The cards containing the schemata are divided into three decks. In the translation process deck 1 is read first and is followed by the formulation deck. The instructions in deck 1 do a preliminary translation. The main translation is done by decks 2 and 3, which are read alternately into the 650 in phases. This phasing is necessitated by the limited storage. Each equation in the formulation requires one pass of decks 2 and 3. These decks are recycled until the last equation is translated.

As the translation proceeds, the speed-code program is punched out on cards, five instructions per card. This program deck, together with data cards for the problem, can be loaded into the computer at some later time. The problem will be run in the speed-code system. In problems which require library subroutines, special cards are punched out by the Encoder. Before computation, these special cards are used in conjunction with a compiling system to obtain a library program which is added to the main program deck.

The allocation of storage for the program and data in the computer is performed by the Encoder and is as follows: All data is loaded into storage in a block starting with storage number 0600. Immediately following the data, the program of speed instructions is stored. Starting from storage 1999 and working backward, storage is allocated for intermediate results. If there is an overlap, the Encoder will stop and give an alarm. Addresses in the computer will be denoted by Greek letters; e.g., $\alpha\ a_{01}$ is the initial address assigned to the data for a_{01}.

The allocation of storage in the Encoder requires some explanation. The formulation is read into storages 1600-1899. Each symbol occupies a half-storage, which we shall call a "cell". The location of a cell is called a "cell number". The cell number, 8XXX, designates the left half of storage 1XXX. The cell number, 9XXX, designates the right half of storage 1XXX - 1; e.g., 8604 designates the left half of 1604 and 9605 designates the right half of 1604. Certain tables compiled by the Encoder are also stored in cells. The number of the cell containing a symbol is denoted by "#" followed by the symbol; e.g., $\#r_{01}$ denotes the number of the cell which contains r_{01}.

One of the most frequent operations performed in the Encoder is scanning of the formulation and the tables. This is accomplished by two subschema, the "Scan Right" and "Scan Left" subschema. Scan Right works with two standard registers, S and Sc. S occupies all of storage 0140 and Sc occupies storage 0109. Sc contains a cell number, while S contains the contents of that cell. The operation, Scan Right, causes the cell number in Sc to advance by 1 and the contents of this new cell are placed in S. Thus, if (Sc) = 8601, then after a Scan Right, (Sc) = 9602 and the contents of 9602 (the right half of storage 1601) are placed in S. If (Sc) = 9602, then after a Scan Right, (Sc) = 8602. The following exception should be noted. A cell which contains zero is a "spacer" cell. Spacer cells are automatically skipped over by the Scan Right operation. Scan Left operates similarly.

III. Summary Description of Schemata

Let us assume that deck 1 and the formulation deck have been read on to the drum (see section IV). Deck 1 contains the schema known as the Initial Addressor (I.A.).

The Initial Addressor comprises two subschema, a scan of the Computer Table and four scans of the equations in a formulation.

Starting at the first cell of the Computer Table (8600 = left half of 1600), I.A. begins to scan right. For each independent variable, a_{mn}, in the Computer Table, it computes an initial address, αa_{mn}. A table of entries of the form mn αa_{mn}0000 is compiled in storages 0350–0389. At the same time the degree of a_{mn} is determined. The degree and the address of a_{mn} are recorded in storage 1150 + mn as follows: 0000 αa_{mn}9 if the degree is zero, 10000αa_{mn}6 if the degree is one. The cell to the right of a_{mn} initially contains the integer which specifies the number of data for a_{mn}. This integer is replaced by αa_{mn}9. In all of the above, "αa_{mn}" stands for a four-digit address.

When the colon which marks the end of the Computer Table is reached, the table of mnα0000 entries is punched out one per card. These entries are to be loaded into the computer which will use them to compose instructions to read in the data for computation.

I.A. continues scanning right from the colon to the comma, which marks the end of the Computer Table. For each q_{mn} that it encounters, an address, μq_{mn}, is assigned as an index register in the computer. q_{mn} is replaced

by $\mu q_{mn}7$ and 00000 $\mu q7$ is recorded in 1200 + mn. For an integer-valued a_{mn}, 00000 $\mu q_i 4$ is recorded in 1150 + mn, where q_i is the index associated with a_{mn} in the Computer Table.

When the comma after the Computer Table is reached, its cell number is recorded in P_r(=1500). Control is then sent to the First Scan Subschema (F.S.). F.S. scans the formulation for equal signs. For each equal sign, it scans left to determine whether an r or a b is being defined. If r_{mn} is being defined, it records 00000 $r_{mn}3$ in 1250 + mn. If b_{mn} is being defined, it records 00000 $b_{mn}2$ in 1300 + mn.

The Second Scan Subschema then scans the formulation for a's, b's, f's, r's and numerical constants on the right side of each equation. b_{mn} is replaced by the contents of 1300 + mn, f_{mn} is replaced by the absolute value of the contents of 1400 + mn. (In 1400 + mn is a constant which will be used later to distinguish the f's.) r_{mn} is replaced by the contents of 1250 + mn. Numerical constants are changed to floating-point numbers, are assigned computer addresses, and are then punched out on cards to be loaded into the computer later as data.

The Third Scan Subschema scans for q's in the equations and treats them as explained in the processing of the Computer Table.

The Fourth Scan Subschema scans the Computer Table for r's and replaces each r_{mn} that it finds by the contents of 1250 + mn.

After the fourth scan has been completed, constants initializing decks 2 and 3 are read in. From this point on, decks 2 and 3 will be read in alternately. The schemata contained in decks 2 and 3 are sometimes referred to as phases 2a and 2b respectively.

Let us assume that deck 2 has been read on to the drum. Deck 2 consists of the b-schema, the q-schema, the Addressor, the r-schema, and the Recursion schema. The operation of these schemata can best be understood by reading Report 4411. However, in Model I, certain features had to be eliminated to meet storage limitations. For example, special library f's are not permitted. Vector equations are excluded, as are vector recursions in which all the b's are the same. Likewise, double recursions are excluded. In Model I, any of these formulations will cause the b-schema to give an error indication. However, the coding is such that if sufficient storage were available, the instructions for translating these special formulations could easily be appended. It should be pointed out that certain branching situations are not permitted since the branch schema has been oversimplified. This is taken into account in Report 4411.

Minimization in Model I is limited to the definition of an r-symbol used as the upper bound in a quantification. Thus, the table look-up aspect is not provided for.

Variables of degree 2 (i.e. matrices) are not permitted. However, many of the instructions for translating such variables are included in Model I. Again, if more storage were available, the necessary instructions could easily be appended.

When phase 2a has finished processing the right side of an equation, it will call for a read-in of deck 3. Deck 3 contains the instructions for the Interpreter.

Once again, for a description of the over-all design of the Interpreter, we refer to Report 4411. However, some of the techniques in Model I are worth describing in some detail.

For example, let us assume that the Encoder is programming the equation,

$$b_{01} = f_{02}\ a_{01}\ a_{02}\ q_{02}\ ,$$

that is,

$$b_{01} = a_{01} + a_{02}(q_{02}).$$

When the Interpreter starts to operate, the a's will have been replaced by addresses (say 1987 for a_{01} and 1986 for a_{02}). An index register will have been set up for q_{02}, and q_{02} will have been replaced by the address of this index register, say 1989. The function, f_{02}, will have been replaced by 14021, where 1402 is the location of the library schema for f_{02}. Thus, the equation will look as follows to the Interpreter:

02018 00000 00000 05008 14021 19879 19866 19899 07018.

Starting with the comma (07018), the Interpreter scans left for f's. (See instructions in 0300-0310 on page 1 I. of list.) An f is identified by a low-order '1'. When 14021 is found, the degree of f_{02} is obtained by referring to register 1402. In this case, the degree is 2. Therefore, the Interpreter scans right for two operands. Operands are identified by either a '9' or a '6' in the low-order position. In this case, the first operand is 19879. The low-order '9' indicates that this operand does not require an index register (see page 4I). Hence, 00 1987 0000 is stored in register O_1 and zero is stored in M_1. The second operand is 19866. The low-order '6' indicates that an index register must be used. Hence, 00 1986 0000 is placed in register O_2 and the Interpreter scans right to obtain 19899. It places 00 1989 0000 in M_2.

The Interpreter also scans left from f_{02} to determine whether the result of this operation will have to be stored temporarily (see pages 21, 31). In this case, no temporary storing is required.

The Interpreter is now ready to compose instructions. It goes to $Li(f_{02})$, which is 1402, and obtains the cell number of the library subschema (see p.71) for f_{02}. This subschema places the operation code for "add" in register OP. (The operation code is 65 - - - - 0225 in the speed-code system.) It then tests M_1. Since $(M_1)=0$, it is ignored. It then composes the instruction "Insert 1987", which is coded as 60 1987 0180. Then M_2 is tested. Since $(M_2)=0$, the next instruction composed is "Modify 1989", which is coded as 65 1989 0093. Then the instruction, "Add 1986", is composed by combining the contents of 0_2 and OP to form 65 1986 0225. This completes the computational part of the program.

The Interpreter then goes into an end-procedure which composes instructions for storing the result (see p.91 ff.), for punching the result, for closing loops etc.

IV. Input-Output

To start the Model I Encoder, the decks are placed in the hopper as described in section II, 70 0007 0088 + is set on the 8000 switches, and the Computer Reset and Program Start buttons are pressed. This causes the first load card to be read into storages 1-8. The next instruction is taken from storage 7. This instruction starts a "clear" routine that stores zero in storages 0010 through 1999. Control then goes to 8000.

Six more load cards are read, placing the 'ADES Read-in' instructions on the drum. Control goes to this routine which reads in Deck 1, containing six ADES instructions per card, until another load card transfers control to the "Formulation Read-in".

The formulation is then read in, ten symbols per card, until a load card transfers control to the first instruction of deck 1. After deck 1 completes its preliminary translation, control is sent again to the 'ADES Read-in' instructions and deck 2 is read into the storages previously occupied by deck 1. A load card then starts the operation of phase 2a. When phase 2a is completed, it places the proper initial instruction for phase 2b in storage 0024 and then calls for a read. The instructions for phase 2b are read in with Deck 3 until a load card is reached. This starts phase 2b at the instruction in 0024. When phase 2b is completed, it, in turn, sets up the proper instruction for phase 2a in storage 0024 and calls for a read. Deck 2 is read in again and once more a load card sends control

to 0024, whence operation of phase 2a is resumed. In this way, decks 2 and 3 are used alternately for each equation until the last equation is translated.

ADES Instruction Cards.

The board wiring used for the read-in of the ADES instruction decks is as follows:

(1) Wire from First Reading column 4 to Digit Selector common and then from the '8' exit hub to Load Entry A hub. Thus, we must put an '8' punch in column 4 of every ADES instruction card to identify it.

(2) Wire from Read Card A hubs 17-20 to Word 2 positions 4-1 of storage entry A. In an ADES instruction card, columns 17-20 contain the location in which the first instruction of that card is to be stored (see list of ADES instructions). The remaining five instructions are stored in sequence. Wire from word size Emitter 4 to Word 2 of A.

(3) Wire from Read Card A hubs 7-15 to Word 5 positions 10-2. Split wire from Read Card A hub 16 to Word 5, position 1 and to the sign position. Card columns 7-16 contain the first instruction word on that card.

(4) Wire from Read Card A hubs 22-31 to Word 6.

Wire from Read card A hub 21 to the sign position of Word 6. Card columns 21-31 contain the second instruction word.

(5) Wire from Read Card A hubs 34-43 to Word 7 and from hub 33 to the sign position of Word 7. Card columns 33-43 contain the third instruction word.

(6) Similarly, the 4th, 5th, and 6th instruction cards, which are in card columns 45-55, 57-67, and 69-79, respectively, are wired to Words 8, 9, and 10.

(7) Wire from Word Size Emitter 10 to Words 5-10 of A. Wire from Word Size Emitter 0 to Words 1, 3, and 4 of A.

Formulation Cards

A formulation card contains ten 3-column fields. Each field contains either (1) an alphabetic symbol, consisting of a letter and the two digits of its subscript or (2) a numerical constant, consisting of three digits. The letter in an alphabetic symbol is punched as an alphabetic character. Hence, this column will contain a digit punch and a "high" punch. A formulation card is identified by a '9' punch in column 4. The wiring for reading in formulation cards is as follows:

(1) Wire from Read Card C, 1–3, to Storage Entry C, 10–8 of Word 1. Card columns 1–3 contain the problem number.

(2) Wire from Read Card C, 24–26 to positions 4–2 of Word 1 of Storage Entry C. Split wire column 24 to Read Column Split Common, and from the 12–x exit hub to a Pilot Selector IPU. Wire from Read Impulse 8 to Transfer Hub of Pilot Selector. Wire from Read Impulse 9 to Normal Hub of Pilot Selector. Wire from Common Hub of Pilot Selector to position 1 of Word 1 of Entry C.
Card columns 24–26 contain the first symbol, of the form $\overset{(x)}{n_o}n_1n_2$ where $\overset{12}{n_o}$ or $(\overset{x}{n_o})$ is a letter, or simply $n_on_1n_2$ when the first symbol is a number. Thus, an 8 is emitted into position 1 if the symbol is alphabetic, a 9 if it is numeric.

(3) Split wire column 26 to the sign position of word 1. The entry, $0\overset{x}{0}0$, means that the field is ignored.

(4) Wire Read Impulse Zero to remaining positions of Word 1. Wire from Word Size Emitter 10 to W.1. of C.

(5) Wire Words 2–10 similarly. Columns 30–32 are wired to Word 2, columns 36–38 to Word 3, columns 42–44 to Word 4, columns 48–50 to Word 5, columns 54–56 to Word 6, columns 60–62 to Word 7, columns 66–68 to Word 8, columns 72–74 to Word 9, and columns 78–80 to Word 10. These columns contain the remaining nine symbols on the card.

Output.

The output of the Encoder consists of 650 instructions, program constants, any data which was contained in the formulation, and cards for use with a subroutine compiling system.

The punching of instructions is controlled by the "Store–Punch Subroutine" in Model I (see list of instructions below). Instructions are composed as the Encoder translates. When five instructions have been recorded in storages 0027–0031, they are punched simultaneously into one card. At the same time, the address which specifies the assigned location of the first instruction is punched into the card. This address will be used later by the 650 Speed–Code Computer to store the five instructions in the proper storages of the computer.

The punching of constants is done by the "Data-Punch" subroutine in Model I. Constants are punched one per card. The address assigned to each constant is also punched in the card. When the problem is to be run on the computer, this address will be used by a read routine in the computer to store the constant in the proper storage.

The cards for the compiler will not be described here. Suffice it to say that they contain information for compiling subroutines such as "cosine", "arcsine", etc. into the main program.

The wiring of the board for punching is as follows:

The punching of instructions and program constants by Model I is done from punch level B. The card punched is a Speed-Code type '5' card, (see NAVORD 4169) which contains six 10-digit 650 words.

(1) Position 10 of the control information hub is wired to the punch B hub. Thus, an 8 in position 10 of word 10 will select punch level B.

(2) The instructions produced by Model I are punched, 5 instructions per card as follows: Word 1 is wired to card columns 22-31 and the sign to card column 21.

Word 2 is wired to c.c. 34-43, the sign to c.c. 33
Word 3 is wired to c.c. 46-55, the sign to c.c. 45
Word 4 is wired to c.c. 58-67, the sign to c.c. 57
Word 5 is wired to c.c. 70-79, the sign to c.c. 69

(3) Word 6, which may be used for identification, is wired to c.c.7-16.

(4) Positions 10,9, and 8 of Word 7 contain the problem number and are wired to c.c. 1,2, and 3. Positions 7,6, and 5 of Word 7 contain zeros and are wired to c.c. 32,44,56,68, and 80. The remaining positions of Word 7 are not punched.

(5) The 650 address which specifies the location of the first word punched in the card is located in positions 8-5 of word 8. These positions are wired to c.c. 17-20.

(6) Words 9 and 10 and the remaining positions of Word 8 are not wired.

(7) Digit Selector Punch 5 is wired to c.c. 4, Digit Selector Punch 0 to c.c. 5, and Digit Selector Punch 1 to c.c. 6.

V. Operating Experience with Model I

Although Model I has not been put into regular operation, several problems have been programmed by it for experimental purposes. These

problems are of a comparatively simple nature, since each was chosen to test a specific part of the Encoder. Three of them are presented here to illustrate the actual physical operation of Model I rather than as examples of what ADES can do. Each problem was formulated, punched on cards, and run through the Encoder. The Encoder produced a deck of program cards which were then loaded into the computer together with data cards. The computation was carried out on the computer, result cards were punched and the results printed and checked.

Example 1. This problem illustrates quantification. We wish to compute $b_1(q_1) = a_1(q_1) + a_2(q_1)$ for all q_1 from 0 to 4 inclusive, store the results in sequence, and punch them out one per card in the first card field. The ADES formulation is as follows:

ADES FORMULATION

$$a_{01}\ 5\ a_{02}\ 5\ P_{02}\ r_{01}\ 5\ P_{01}$$

$$b_{00}\ e_{00}\ f_{00}\ b_{01}\ P_{01}$$

$$P_{03}\ 0\ q_{01}\ 4\ r_{01}\ d_{11}\ b_{01}\ e_{00}\ f_{02}\ a_{01}\ q_{01}\ a_{02}\ q_{01}\ P_{01}$$

$$r_{01}\ e_{00}\ f_{01}\ q_{01}\ P_{01}$$

Remarks. P_{01} denotes a comma. P_{02} denotes a colon. P_{03} denotes the quantifier, "for all". e_{00} denotes an ordinary equal sign. The first line constitutes the Computer Table. It says that a_{01} will have five data, a_{02} will have five data, and that r_{01} will have five values to control storing of b_{01}. The Second line is the Master Phase Equation, which indicates that b_{01} is to be programmed first. The third line is the equation for b_{01}. Here, the left side of the equation contains the quantification, "$P_{03}\ 0\ q_{01}\ 4$" meaning "for all $0 \leq q_{01} \leq 4$". It also contains d_{11}, which indicates that the results are to be punched in the first field of a card, one per card. Finally, the left side contains r_{01}, which means that successive values of b_{01} are to be stored in successive storages. r_{01} is defined by the fourth line as being equal to q_{01}. The right side of the equation for b_{01} contains the formula for b_{01}, with "f_{02}" denoting "+".

The formulation is punched on four cards as follows:

	Card Field									
	1	2	3	4	5	6	7	8	9	10
Card 1:	a01	005	a02	003	p02	r01	005	p01	b00	e00
Card 2:	f00	b01	p01	p03	000	q01	004	r01	d11	b01
Card 3:	e00	f02	a01	q01	a02	q01	p01	r01	e00	f01
Card 4:	q01	p01	000	000	000	000	000	000	000	000

The fields containing "000" with the "high" x punch are ignored by the read routine. The formulation is read into storages 1600–1618 and looks as follows after read-in:

Storage	Contents
1600	01018 00059
1601	01028 00059
1602	07028 09018
1603	00059 07018
1604	02008 00000
1605	00000 05008
1606	06008 02018
1607	07018 07038
1608	00009 08018
1609	00049 09018
1610	04118 02018
1611	00000 00000
1612	05008 06028
1613	01018 08018
1614	01028 08018
1615	07018 09018
1616	00000 00000
1617	05008 06018
1618	08018 07018

Example 2. This problem illustrates a simple scalar recursion. We wish to compute the sum of all $a_{01}(q_{01})$, where $0 \quad q_{01} \quad 5$, and print the final result. Denoting the sum by b_{01}, and observing that $b_{01}(q_{01}+1)=b_{01}(q_{01}) + a_0(q_{01})$, we write the ADES formulation as follows:

ADES FORMULATION

$$a_{01}\ 6\ P_{02}\ q_{01}\ 6\ P_{01}$$

$$d_{11}\ P_{03}\ 0\ q_{01}\ 5\ b_{01}\ e_{11}\ f_{02}\ b_{01}\ q_{01}\ a_{01}\ q_{01}\ P_{01}\ f_{01}\ 0\ P_{01}$$

$$b_{00}\ e_{00}\ f_{00}\ b_{01}\ P_{01}$$

The formulation is punched on three cards as follows:

	Card Field									
	1	2	3	4	5	6	7	8	9	10
Card 1:	a01	006	p02	q01	006	p01	d11	p03	000	q01
Card 2:	005	b01	e11	f02	b01	q01	a01	q01	p01	f01
Card 3:	000	p01	b00	e00	f00	b01	p01	000	000	000

The 650 speed-code program produced by the Encoder is as follows (Operations are listed by name for explanatory purposes.)

Location of Instruction	Operation	Address	Remarks
0606	Insert Positive	1985	Compute
0607	Store	1983	$b_1(0)=0$
0608	Insert Positive	1985	Lower bound
0609	Store	1989	of q_{01}
0610	Insert Positive	1989	
0611	Add		$(q_{01}+1) \bmod 2$
0612	Modulo	0002	
0613	Store	1982	
0614	Modify	1981	
0615	Insert Positive	1983	
0616	Modify	1989	$a_1(q_1)+b_1(q_1)$
0617	Add	0600	
0618	Modify	1982	
0619	Store	1983	
0620	Increment Modifier	1989	
0621	Modify	1982	
0622	Insert Positive	1983	Punch out
0623	Store	0327	final value
0624	Punch	0015	
0625	Store	1980	
0626	No Operation	0000	
0627	No Operation	0000	
0628	Stop	1111	

Remarks: The numerical value of the index, q_{01}, is stored in 1989. The instruction, "Modify 1989", causes the next instruction, "Add 0600", to be executed as if the contents of 1989 had been added to the address, 0600; i.e. it makes 1989 an index register. The instruction, "Modulo 0002" reduces the number in the result register to the residue modulo 2. The instruction "Increment Modifier" adds 1 to the contents of 1989 and tests to see if q_{01} has reached its upper bound. If not, control is transferred back to 0614, where the loop begins. The Encoder has assigned computer storages

as follows: q_{01} in 1989; $b_{01}(q_{01})$ in 1983; the constant, zero, in 1985; a_{01} in 0600–0605.

Example 3. This problem illustrates vector recursion. We wish to integrate an ordinary first-order differential equation of the form,

$$y' = \phi(x,y),$$

with initial values x_o, y_o. We shall use a "predictor-corrector" method similar to Adams' method to step ahead and a Runge-Kutta formula to get started. A numerical statement might look as follows:

For $1 \leq n \leq 20$, $x_{n+1} = x_o + (n+1)h$, $x_o = a_1$, $x_1 = a_1 + h$

$$y'_{n+1} = \phi(x_{n+1}, \bar{y}_{n+1}),\ y'_o = \phi(x_o, y_o),\ y'_1 = \phi(x_1, b_6),$$

$$v_{n+1} = 2(y'_{n+1} + 2y'_n)h,\ v_o = 0,\ v_1 = 2(y'_1 + 2y'_o),$$

$$y_{n+1} = y_n + (1/5)(v_{n+1} - u_{n+1}),\ y_o = a_3, y_1 = b_6,$$

$$r_{n+1} = y_{n+1} - \bar{y}_{n+1},\ r_o = 0,\ r_1 = 0$$

$$u_{n+1} = y_n - y_{n-1},$$

$$\bar{y}_{n+1} = y_n + 2(y'_n - y'_{n-1})\,h + v_n - 5u_{n+1},$$

$$b_6 = y_o + (k_1 + 2k_2 + 2k_3 + k_4)/6,$$

$$k_1 = h\,\phi(x_o, y_o),$$

$$k_2 = h\,\phi(x_o + h/2,\ y_o + k_1/2),$$

$$k_3 = h\,\phi(x_o + h/2,\ y_o + k_2/2),$$

$$k_4 = h\,\phi(x_o + h,\ y_o + k_3),\ .$$

The last five equations will be recognized as the Runge-Kutta method. $\bar{y}_{n+1}$ denotes the predicted value of y_{n+1}. v_{n+1} and u_{n+1} are intermediate results.

To obtain the corresponding ADES formulation, we shall use the following notation:

$x_o = a_1$, $x = b_1$, $\bar{y} = b_2$, $y' = b_3$, $y = b_4$, $r = b_5$, $h = a_2$, $y_o = a_3$, $n = q_1$, $k_1 = b_7$, $k_2 = b_8$, $k_3 = b_9$, $k_4 = b_{10}$, $x+h/2 = b_{11}$, $y+k_1/2 = b_{12}$, $y+k_2/2=b_{13}$, $y+k_3 = b_{14}$, $v = b_{15}$, $u = b_{16}$, $\phi = f_{51}$. To make it somewhat easier to read, we shall write the formulation using the conventional symbols for punctuation, equal signs, and arithmetic operations. Thus, we shall write, "," and punch "p_{01}" on the card. We shall write "=" and punch "e_{oo}" ; write "+" and punch "f_{02}" ; write "−" and punch "f_{03}" ; write "/" and punch "f_{05}" ; write "." and punch "f_{04}".

ADES FORMULATION

$$a_{01}\ 1\ a_{02}\ 1\ a_{03}\ 1 : q_{01}\ 21 ,$$

$$r_{01} = +\ 1\ q_{01} ,$$

$$r_{02} = -\ q_{01}\ 1 ,$$

$$b_{00} = f_{00} b_{01},$$

$$p_{03}\ 1\ q_{01}\ 20\ b_{01}\ d_{10} = 31 + b_{01}\ q_{01}\ a_{02},\ f_{01}\ a_{01},\ +\ a_{01}\ a_{02},$$

$$b_{03} = 31\ f_{51}\ b_{01}\ r_{01}\ b_{02},\ f_{51}\ a_{01}\ a_{03},\ f_{51}\ b_{01}\ 1\ b_{06},$$

$$b_{15} = 31\ \cdot\cdot\ 2 + b_{03}\ r_{01}\ \cdot\ 2\ b_{03}\ q_{01}\ a_{02},\ f_{01}\ 1,\ \cdot\cdot 2 + b_{03}\ 1\ \cdot\ 2\ b_{03} a_{02},$$

$$b_{04}\ d_{20} = 31 + b_{04}\ q_{01} - b_{15}\ r_{01}\ b_{16}\ 5,\ f_{01}\ a_{03},\ f_{01}\ b_{06},$$

$$b_{05}\ d_{31} = 31 - b_{02}\ b_{04}\ r_{01},\ f_{01}\ 0,\ f_{01}\ 0\ ,\ )$$

$$b_{16} = -\ b_{04}\ q_{01}\ b_{04}\ r_{02} ,$$

$$b_{02} = +\ \cdot - b_{03}\ q_{01}\ b_{03}\ r_{02} - 2\ a_{02} + b_{04}\ q_{01} - b_{15}\ q_{01}\ \cdot\ 5\ b_{16} ,$$

$$b_{06} = +\ a_{03} / +\ b_{07} + \cdot\ 2 + b_{08}\ b_{09}\ b_{10}\ 6 ,$$

$$b_{07} = \cdot b_{03}\ 0\ a_{02} ,$$

$$b_{08} = \cdot\ f_{51}\ b_{11}\ b_{12}\ a_{02},$$

$$b_{11} = +\ a_1 / a_{02}\ 2 ,$$

$$b_{12} = +\ a_{03} / b_{07}\ 2,$$

$$b_{09} = \cdot\ f_{51}\ b_{11}\ b_{13}\ a_{02},\qquad b_{10} = \cdot\ f_{51}\ b_{01}\ 1\ b_{14}\ a_{02} ,$$

$$b_{13} = +\ a_{03} / b_{08}\ 2 ,\qquad b_{14} = +\ a_{03}\ b_{09} ,$$

The above formulation is for an arbitrary function, f_{51}. f_{51} would have to be defined by an f-equation. Since Model I does not allow for f-equations, the formulation was tried for the specific function, $\phi(x,y) = x \cdot y$; i.e. we integrated $y' = xy$. The Encoder programmed the formulation correctly, producing 182 instructions.

A. PHASE 1

Title - Initial Load Cards

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	Word 1	00	0000	0000	
	2	45	0005	8000	8000 = (70 0007 0088)
	3	21	0010	0004	
	4	16	0006	0002	
	5	15	0008	8002	Card 1 Read into
	6	21	1999	0004	Storages 0001-0008
	7	65	0003	0004	This card clears storages
	8	21	2000	0004	0010-1999
	Word 7	70	1007	9999	Card 2
	Word 1	00	0000	0000	Card 3
	2	67	0001	1003	Reads into storages 1001-1008
	3	15	1004	1005	Sets up read-in routine
	4	24	0000	1007	For ADES constants 1/card
	5	69	0002	8002	under control of C.C. 17-20
	6	00	0000	0000	
	7	70	0007	1002	
	8	00	0000	0000	

Title - Phase 0: The punched load cards

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	Word 1	66	0801	0000	These cards load in the
	2	10	0094	0095	ADES Read-In routine
	3	24	0093	0358	
	4	69	0004	8002	Card 1 Reads into 0001-
	5	24	0094	0356	0008
	6	11	0099	0097	
	7	70	0357	9000	
	8	24	0095	0352	
	Word 1	66	0801	0001	
	2	69	0353	0354	
	3	69	0010	8002	
	4	24	0099	0355	Card 2
	5	70	0007	9001	Reads into 0351-0358
	6	69	0006	0008	
	7	69	0002	0003	
	8	69	0004	0005	
	Word 1	66	0801	0002	
	2	44	0098	0087	
	3	24	0097	0358	
	4	10	0096	8003	Card 3 Reads into 0001-0008
	5	24	0098	0356	
	6	69	0011	8002	
	7	69	0002	0003	
	8	24	0096	0355	

Title - Phase 0: The punched load cards

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	Word 1	66	0501	0003	
	2	15	0091	0095	
	3	24	0047	0358	
	4	00	0001	0000	Card 4
	5	24	0091	0356	Reads into 0001-0008
	6	70	0007	0088	
	7	69	0002	0003	
	8	24	0087	0355	
	Word 1	66	0501	0004	
	2	67	0002	0089	
	3	24	0088	0358	
	4	35	0004	0090	Card 5
	5	24	0089	0356	Reads into 0001-0008
	6	69	0049	0092	
	7	69	0002	0003	
	8	24	0090	0355	
	Word 1	66	0501	0005	
	2	24	0000	0047	
	3	24	0049	0358	Card 6
	4	22	0049	0190	Reads into 0001-0008
	5	24	0092	0356	
	6	65	8001	0093	
	7	69	0002	0003	
	8	24	0190	0087	I: Go to ADES Read In

Title - Read-in routine

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
					This routine reads in the ADES instructions 6/card under control of c.c. 17-20. It remains on the drum for all phases.
	0087	70	0007	0088	
	0088	67	0002	0089	
	0089	35	0004	0090	
	0090	69	0049	0092	
	0049	24	0000	0047	
	0092	22	0049	0190	
	0190	65	8001	0093	
	0093	10	0094	0095	
	0094	69	0004	8002	
	0095	11	0099	0097	
	0099	69	0010	8002	
	0097	44	0098	0087	
	0098	10	0096	8003	
	0096	69	0011	8002	
	8003	69	00XX	8002	
	8002	24	XXXX	0047	
	0047	15	0091	0095	
	0091	00	0001	0000	

Title - Formulation Read-In Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0350	70	0007	0351	
	0351	69	0352	0353	Initialize Storage Callout
	0352	= (60	0001	0360)	
	0353	24	0354	0355	(0354 = storage for callout instr.)
	0355	65	0001	0356	
	0356	20	0083	0357	MAR XX XXXXX → 0033 for punch
	0357	20	0033	0358	out of Program cards, 0083 for data cards
	0358	60	8002	0360	initial word → WA
	0359	46	0366	0395	
	0360	46	0377	0361	If negative, skip to next word
	0361	35	0005	0362	Truncate '9' in posit.6.
	0362	30	0008	0363	$0\ldots0_d \mid n_1n_2800$
	0363	11	0364	0365	
	0364	(00	0000	0005)	Test for =
	0365	44	0366	0359	
	0366	10	0364	0367	
	0367	35	0003	0368	
	0368	69	0369	0370	
	0369	= (00	0000	0008)	9 initially; 9 ⇒ store in left half
				0009)	8 ⇒ store in right half
	0370	91	0385	0371	
	0371	35	0005	0372	
	0372	21	0373	0374	Store left half temporarily
	0373	= (temp.store for left half)			
	0374	69	0375	0376	
	0375	= (00	0000	0008)	
	0376	24	0369	0377	Prepare for next rt.cell
	0377	65	0354	0378	
	0378	16	0380	0381	Prepare for next word
	0380	= (60	0010	0360)	
	0381	45	0382	0350	

Title - Formulation Read-In

Came from	Loc. of Inst.		Op. Code	Data Address	Instr. Address	Notes
	0382		15	0383	0384	
	0383	=	(60	0011	0360)	Prepare for next word
	0384		20	0354	8001	
	8001	=	(60	00XX	0360)	call in next word
	0385		69	0386	0387	
	0386	=	(00	0000	0009)	Prepare for next left cell
	0387		24	0369	0388	
	0388		10	0373	0399	add left half to rt. half and store
	0399		(21	XXXX	0390)	initially 21 1600 0390 finally
	0390		65	0399	0391	21 (L(F) 0390 where L(F) = location of end of formulation
	0391		15	0392	0393	
	0392	=	(00	0001	0000	Prepare for next 'store'
	0393		20	0399	0394	
	0394		21	0373	0377	Store zero in 0373
0408	0409		20	0373	0385	" " " "
0365	0395		10	0364	0396	
	0396		35	0003	0397	
	0397		21	0398	0389	
	0398		(temp. store)			Store = sign temporarily
	0389		65	0399	0400	
	0400		15	0401	0402	
	0401	=	(00	0000	0013)	
	0402		10	0373	8002	
	8002	=	(21	XXXX	0403)	Store word with 1 or 2 blanks
	0403		16	0401	0404	
	0404		15	0392	0405	Prepare for next store: note
	0405		20	0399	0406	that machine will stop automatically if formulation is too long.
	0406		44	0408	0407	
	0408		60	0398	0409	" = " → U.A.
	0407		60	0398	0371	

Title - Load card to follow ADES formulation deck : completes read-in of formulation and sends control to phase 1.

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	Word 1	66	0501	0330	Reads into 0001-0008
	2	24	0390	0003	
	3	60	0373	0004	Test for last half-word
	4	45	0902	0904	
	5	00	0000	0000	
	6	00	0000	0000	
	7	69	0008	0002	
	8	(69	0250	0251)	First instruction for initial addressor to be loaded into 0390
	0843	65	0399	0844	If last symbol was stored in art. cell decrease add. part of 0399 by 1 to get L(F). Go to 0390. I.A.
	0844	16	0392	0845	
	0845	20	0399	0390	
	0902	10	0903	0399	
	0903	(00	0000	5008)	
	0904	65	0903	0905	
	0905	10	0399	0906	
	0906	10	0907	8003	Store $-_{00}$ sign at end of
	0907	(00	0000	0600)	formulation
	8003	21	XXXX	0990	
	0990	00	0000	0843	

Title - Initial Addressor

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
last load card	0390	69	0250	0251	
	0250	(00	9600	0000)	#-1 of first cell in
	0251	24	0109	0252	formulation → Sc.
	0252	69	0253	0100	Scan rt.
	0253	90	0255	0254	Test alphabetic
	0254	01	0254	0000	Error
	0255	91	0265	0256	Test for A
	0256	97	0258	0257	Test for P
	0257	01	0257	0000	Error
	0258	16	0259	0260	
	0259	(00	0000	7028)	Test for Po2. If Po2, → Punch
	0260	45	0257	0325	out ∝ (a). I: Go to 5 I.A.
	0265	69	0266	0100	Scan rt.
	0266	90	0267	0269	Test for numeric
	0267	01	0267	0000	Error
	0269	30	0001	0270	Truncate '9'
	0270	45	0272	0271	zero test
	0271	01	0271	0000	Error
	0272	16	0273	0274	
	0273	(00	0000	0001)	
	0274	45	0275	0276	
	0276	69	0302	0306	(deg=o) I: Go to 4 I.A.
	0275	69	0277	0306	(deg >o) I: Go to 4 I.A.
	0277	69	0278	0100	Scan rt
	0278	90	0280	0254	Test alphabetic

Title - Initial Addressor

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0280	97	0281	0287	Test for P
	0281	20	0282	0283	
	0282	(temp.store Po2)			
	0283	60	0284	0285	
	0284	(10	0000	0006)	deg 1 tag
	0285	69	0286	0303	I: Go to 3 I.A.
	0286	65	0282	0258	
	0287	91	0288	0290	Test for A
	0288	60	0284	0289	
	0289	69	0265	0303	I: Go to 3 I.A.
	0290	99	0294	0257	
	0292	60	0293	0301	
	0293	(00	0000	0005)	deg 2 tag
	0294	69	0295	0150	Scan left to α
	0295	65	0109	0297	(Sc) = # (α)
	0297	30	0003	0299	
	0299	20	0298	0300	Replace α by # (α)
	0300	69	0292	0100	Scan rt to r
	0301	69	0252	0303	I: Go to 3 I.A.
0276	0302	60	0446	0301	
	0446	(00	0000	0009)	deg zero tag

Title - Initial Addressor: Degree Subschema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0303	24	0296	0304	00 000 #∝(A)5 for deg 2
	0304	10	0298	0670	Form: 00 000∝(A)9 for deg 0
	0305	65	0263	0279	1 0000 ∝ (A)6 for deg 1
	0279	15	0264	8002	
	0264	(10	1050	0672)	
	8002	(10	1150+mn	0672)	
	0670	21	0671	0305	
	0671	(temp.store)			
	0672	44	0673	0674	
	0673	01	0673	0000	Error
	0674	15	0675	0676	
	0675	(10	9999	9624)	
	0676	10	0671	8002	
	8002	(21	1150+mn	0296)	

Title - IA: $\sum$ Subschema and Compile $\propto$(a) Table

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0306	24	0296	0307	
	0296	(exit)			
	0307	15	0273	0444	
	0273	(00	0000	0001)	N(ai) → L.A.
	0308	15	0309	0310	
	0309	(00	0000	6000)	Initially. See 0311
	0310	10	8001	0311	
	0311	20	0309	0312	00000 $\sum_{1}^{k}$ N(ai) 0 → 0309
	0312	21	0298	0313	00000 $\propto$ (a_k) 0 → 0298
	0298	(temp store)			
	0313	69	0314	0150	01 mn8 → L. A.
	0314	30	0001	0261	
	0315	35	0001	0316	
	0316	15	0298	0317	
	0317	35	0003	0318	
	0318	20	(0350)	0319	mn($\propto$) 0000 → 0350-0389
	0319	65	0318	0320	
	0320	15	0434	0321	increment 0318
	0321	20	0318	0322	
	0322	69	0443	0100	Scan rt. to orig. pos.
	0443	65	0298	0445	$\propto$ → L.A.
	0445	15	0446	0447	
	0446	(00	0000	0009)	
	0447	10	0109	0448	
	0448	69	0296	0200	Replace N by $\propto$9
	0444	35	0001	0308	
	0261	35	0004	0262	
	0262	20	0263	0315	for use in deg. subschema
	0263	(STORE: 01mn 0000)			

Title - I.A. Punch Out Instructions for Data Read Routine

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0325	65	0309	0326	(0309) = γ_1 + 1. i.e. add.of
	0326	35	0003	0333	lst storage after data
0249	0327	10	0328	0291	(See 4 IA) I:GO to 6 I.A.
	0328	(00	0343	0000)	
0291	0329	69	0330	0050	Punch: Initilize-Sequencer Card
	0330	65	0309	0332	
	0331	20	0391	0335	
	0332	15	0334	0331	
	0333	20	0309	0249	I: GO to 6 I.A.
	0334	(66	0000	0049)	
	0335	10	0336	0346	
	0336	(00	0566	0000)	
	0346	69	0347	0050	Punch: 66(γ_1 +1)0049 →0566
	0347	65	0391	0348	
	0348	16	0434	0349	
	0349	10	0410	0411	
	0410	(00	0554	0000)	
	0411	69	0412	0050	Punch: 66(γ_1)0049 → 0554
	0412	66	0318	0413	
	0413	15	0414	0415	
	0414	(20	0350	0319)	
	0415	15	0309	0416	(γ_1 +1)- no. of entries = γ_2
	0416	20	0391	0417	
	0417	15	0334	0418	
	0418	10	0419	0420	
	0419	(00	0592	0000)	
	0420	69	0421	0050	Punch: 66 (γ_2) 0049 →0592 GO to 6 IA

Title - I.A. Punchout: α(a) Table

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0421	65	(0350)	0422	
	0422	10	0391	0423	= γ_2, initially
	0423	69	0424	0050	Punch: mn α0000 $\rightarrow$ γ_2
	0424	65	0391	0428	00 γ_2 0000
	0425	16	0309	0426	00 γ_1 +1 0000
	0426	45	0427	0435	I: 60 to 7 IA
	0427	15	0309	0429	
	0428	15	0434	0425	γ_2 +1 $\rightarrow$ 0391
	0429	20	0391	0430	
	0430	65	0421	0431	
	0431	15	0434	0432	Increment 0421
	0432	20	0421	0421	
0327	0291	15	0433	0329	I: 60 to 5 I.A.
	0433	(10	0000	0182)	Sequencer instruction
	0434	(60	0001	0000)	
0333	0249	20	1504	0327	Initialize instruction ctr. 60 to 5 IA
	1506	00	1990	0000	Initialize Int. Stor. Ctr. by read-in of phase 1
	1505	00	1989	0000	Initialize β ctr. by read-in of phase 1

Title - I.A. Process Computer Table from : to)

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0435	69	0436	0100	Scan rt
	0436	90	0437	0254	Test alph. Numeric ⇒error
	0437	91	0667	0438	Test for a
	0438	98	0677	0439	Test for q
	0439	99	0678	0440	Test for r
	0440	97	0600	0257	Test for P, i.e. end of table
	0441	69	0442	0100	Scan rt.
	0442	98	0450	0449	Test for q
	0449	01	0449	0000	error
	0450	69	0841	0470	I: GO to
	0841	65	0488	0842	subscript of q mn
	0842	15	0323	8002	
	0323	(60	1200	0324)	
	8002	(60	1200+mn	0324)	
	0324	11	0454	0665	00000 $(M_q)4 \rightarrow$U.A
	0454	(00	0000	0003)	
	0665	20	0298	0666	Zero →0298
	0666	69	0435	0303	60 to 3IA. $M_q 4 \rightarrow$1150 + i.j.
	0677	69	0678	0470	I:GO to 8 IA
	0678	69	0679	0100	Scan rt.
	0667	30	0001	0668	
	0668	35	0004	0669	
	0669	20	0263	0441	
	0679	90	0267	0435	
	0600	69	0109	0601	#( ) after computer table)
	0601	24	1500	0456	Pr=1500. GO to 1 F.S.

Title - I.A. q - Subschema: Replace q_{mn} by M_q7. $M_q7 \rightarrow 1200 + mn$

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0470	24	0296	0481	
	0494	16	0495	0496	
	0495	(00	0050	0000)	
	0496	46	0497	0498	
	0498	01	0498	0000	Error
	0497	15	0482	8002	
	0482	(65	1250	0483)	
	8002	(65	1200+mn	0483)	
	0483	45	0484	0471	
	0484	10	0109	0485	
	0485	69	0296	0200	$M_q7 \rightarrow \#(q)$
	0471	65	1505	0472	
	0472	15	0473	0474	
	0473	(00	0000	7000)	
	0474	30	0003	0475	
	0475	10	0488	0476	
	0476	10	0477	8003	8003=20(1200+mn) 0478
	0477	(20	1200	0478)	
	0478	69	0479	0040	Decrement φ ctr.
	0479	69	0480	0040	Decrement φ ctr.
	0480	65	0140	0481	qmn $\rightarrow$ L.A.
	0481	69	0494	0486	Extract subscript of qmn see 2 F.S.

Title - First Scan

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0451	69	1500	0452	# (, after computer table)
	0452	24	0109	0453	→ Sc.
	0453	69	0455	0100	Scan rt.
	0455	95	0499	0467	Test for =. . D Go to 3 FS
0601,IA	0456	65	0399	0457	(0399) = 21 1XXX 0390
	0457	16	0458	0459	1XXX is loc. of last symbol (s)
	0458	(20	2000	0390)	of formulation.
	0459	20	0109	0460	009XXX → Sc
	0460	69	0461	0100	Scan rt.
	0461	97	0462	0464	Test for comma
	0462	69	0109	0463	
	0463	24	1501	0451	# (last comma) → P_s = 1501
	0464	69	0465	0100	Scan rt.
	0465	97	0462	0466	Test for comma
	0466	01	0466	0000	Error
	0467	65	0109	0468	
	0468	16	1501	0469	Test Sc = P_s
	0469	45	0453	0854	I: Go to IS.S.

Title - Extract subscript of symbol in L.A.

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0487	30	0001	0492	
	0492	35	0008	0489	
	0489	11	8003	0490	
	0490	30	0004	0491	00 00mm 0000 → L.A.
Start	0486	24	0493	0487	
	0493	(exit		)	
	0491	20	0488	0493	
	0488	= (00	00mn	0000)	

Title - First Scan

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0499	20	0500	0501	$= n_1 n_2 \longrightarrow$ 0500
	0500	(temp store)			
	0501	69	0502	0150	Scan left
	0502	96	0630	0503	Test for f. D: Go to 9 F.S.
	0503	99	0571	0504	Test for π. D: Go to 6 F.S.
	0504	92	0506	0908	Test for b
	0505	01	0505	0000	Error
	0506	69	0507	0486	Extract subscript of bmn
	0507	15	0508	8002	
	0508	(10	1300	0509)	
	8002	(10	13mn	0509)	
	0509	44	0519	0510	Zero test (1300 + mn)
	0510	15	0511	0512	
	0511	(10	0000	0009)	
	0512	20	0513	0514	
	0513	(20	13xx	0518)	00000 # (bmn) 2 $\rightarrow$ 13 mn
	0514	65	0109	0515	
	0515	15	0516	0517	
	0908	94	0501	0505	Test for d

Title - First Scan: Vector b

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0516	= (00	0000	2000)	
	0517	30	0003	0513	
0513	0518	65	0140	0551	bmn → L.A. Go to 5 F.S.
0509	0519	65	0500	0520	= n_1n_2 → L.A.
	0520	69	0521	0486	
	0521	16	0522	0523	
	0522	(00	0009	0000)	
	0523	45	0524	0533	Test for vector
	0524	16	0525	0526	
	0525	(00	0022	0000)	
	0526	45	0527	0553	Test for vec. rec. (prec. line)
	0527	16	0528	0529	
	0528	(00	0010	0000)	
	0529	45	0530	0553	Test for vec. rec. (C.O.V.)
	0530	16	0528	0531	
	0531	45	0532	0553	Test for vec. rec. (one line)
	0532	01	0532	0000	Error
	0533	69	0534	0100	Scan rt.
	0534	95	0543	0535	test for =
	0535	97	0536	0533	test for P
	0536	16	0537	0538	
	0537	(00	0000	7048)	test for Po4
	0538	45	0539	0453	I: Go to IF.S.
	0539	65	0109	0540	
	0540	16	1501	0541	Ps
	0541	45	0533	0542	Test for end of formulation
	0542	01	0542	0000	Error
	0543	69	0544	0150	
	0544	69	0545	0486	Extract subscript of bmn
	0545	16	0547	0548	Compare with preceding subscript

Title - First Scan: Vector b

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0546	20	0547	0552	
	0547	(Store subscript of b)			
	0548	45	0549	0550	
	0549	01	0549	0000	Error
	0550	69	0533	0100	
	0551	69	0546	0486	Extract subscript of bmn
	0552	69	0909	0100	Scan rt.
0531	0553	69	0554	0150	Scan left for = of first
	0554	95	0555	0553	eq. in vect. rec.
	0555	69	0556	0150	Scan left for P
	0556	97	0557	0555	
	0557	16	0558	0559	
	0558	(00	0000	7038)	Test for P03
	0559	45	0560	0561	
	0560	01	0560	0000	Error
	0561	69	0562	0150	
	0562	97	0563	0561	
	0563	16	0564	0565	
	0564	(00	0000	7018)	Test for P01
	0565	45	0566	0567	
	0566	01	0566	0000	Error
	0567	69	0568	0100	Scan rt. for first =
	0568	95	0569	0567	in vect. rec.
	0569	69	0570	0100	Scan rt. for second =
	0570	95	0543	0569	in vect. rec. D: Go to 4 F.S.
	0909	94	0552	0453	Test for d. I: Go to 1 F.S.

Title - First scan: Double Recursion

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
0503	0571	69	0572	0150	Scan left
	0572	92	0573	0595	Test for b. I: Go to 7 F.S.
	0573	69	0574	0486	Extract subscript of b
	0574	15	0575	8002	
	0575	(10	1300	0576)	
	8002	(10	13mn	0576)	
	0576	44	0577	0578	
	0577	01	0577	0000	Error
	0578	15	0579	0580	
	0579	(10	0000	0008)	
	0580	20	0513	0581	
	0581	65	0109	0582	# (bmn) = (Sc)
	0582	15	0516	0583	
	0583	30	0003	0513	
	0513	(20	13mn	0584)	00000 # (bmn) 2 $\rightarrow$ 13mn
	0584	69	0552	0100	Scan rt. to r. Go to 5 F.S.

Title - First Scan: $\mathcal{A}$-eq

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
0572	0595	69	0596	0100	Scan rt to r
	0596	69	0597	0100	Scan rt. to =
	0597	69	0598	0100	Scan rt to f
	0598	96	0602	0599	Test for f
	0599	01	0599	0000	Error
	0602	69	0603	0486	Extract subscript of F
	0603	16	0649	0604	
	0604	45	0605	0648	Test for Fo2 $\equiv$ +. I: Go to 8 F.S.
	0605	16	0434	0606	
	0606	45	0607	0585	Test for Fo3 $\equiv$ -. I: Go to 8 F.S.
	0607	69	0608	0609	
	0608	(65	0109	0623)	set up 0622
	0609	24	0622	0610	
	0610	69	0611	0150	Scan left to =
	0611	69	0612	0150	Scan left to rmn
	0612	69	0613	0486	Extract sub. mn of r
	0613	16	0495	0614	
	0614	46	0615	0498	Test mn $\geq$ 50. I: see 8 I.A.
	0615	15	0616	8002	
	0616	(10	1300	0617)	
	8002	10	1250+mn	0617	
	0617	44	0618	0619	
	0618	01	0618	0000	Error
	0619	15	0620	0621	
	0620	(09	9999	9935)	
	0621	20	0513	0622	
	0622	65	0109	0623	
	0623	15	0624	0517	(0517) = 30 0003 0513
	0624	(00	0000	3000)	
	0513	(20	1250+mn	0552)	00000 # (Rmn)3 $\rightarrow$1250 + mn

Title - First Scan: simple r-eq.

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0585	20	0500	0654	Zero → 0500
	0586	90	0587	0656	Test for numeric
	0587	98	0625	0588	Test for q
	0588	69	0607	0150	Scan left to f. Go to 7 FS
	0589	69	0590	0100	Scan rt
	0590	98	0592	0591	Test for q
	0591	69	0588	0150	Scan left to numeric or q
	0592	69	0593	0100	Scan rt
	0593	97	0650	0594	Test for P
	0594	01	0594	0000	Error
	0625	69	0626	0100	Scan rt from q
	0626	90	0591	0659	Test numeric
	0627	69	0628	0609	See 7 FS
	0628	(66	0109	0629)	For simple r, negative
	0629	16	0624	0663	→1250 + mn
	0648	69	0652	0653	
	0650	69	0651	0150	Scan left to q (or numeric)
	0651	69	0662	0150	Scan left to q (or numeric)
	0652	(10	0000	0000)	
	0653	24	0500	0655	
	0654	69	0587	0100	Scan rt from fo3
	0655	69	0586	0100	Scan rt from fo2
	0656	35	0005	0657	
	0657	15	0500	0658	
	0658	20	0500	0589	(ō) xxx9 ooooo →0500
	0659	35	0005	0660	
	0660	15	0500	0661	
	0661	20	0500	0592	
	0662	69	0627	0150	
	0663	30	0003	0664	Form - (ō) xxx9 # (r) 3
	0664	16	0500	0513	

Title - First Scan: f-equation

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0630	69	0631	0486	Extract subscript of f mn
	0631	20	0547	0632	Store mn
	0632	15	0633	8002	
	0633	(65	1400	0634)	
	8002	(65	14mn	0634)	
	0634	45	0635	0642	
	0635	46	0636	0637	
	0636	01	0636	0000	Error
	0637	69	0638	0100	Scan rt to " = m_1m_2"
	0638	69	0639	0486	Extract subscript m_1m_2
	0639	16	0522	0640	
	0640	45	0641	0453	Test m_1m_2=09
	0641	01	0641	0000	Error
	0642	65	0109	0643	# f mn = (Sc)
	0643	30	0003	0644	00000 # (f mn) 1→L.A.
	0644	15	0273	0645	
	0645	10	0547	0646	
	0646	10	0647	8003	
	0647	(20	1400	0552)	
	8003	(20	14mn	0552)	
	0649	(00	0002	0000)	

Title - Second Scan

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0680	69	0681	0100	Scan rt (init. from , after table)
	0681	92	0688	0682	Test for b
	0682	96	0698	0683	Test for f
	0683	99	0755	0684	Test for r
	0684	91	0760	0762	Test for a. I: Go to 4 S.S.
0762	0685	65	0109	0686	
	0686	16	1501	0687	Test for end of formulation
	0687	45	0680	0805	Go to T.S.
	0688	69	0689	0100	Scan rt
	0689	95	0680	0690	Test for =
	0690	99	0691	0910	Test for r
	0691	69	0692	0100	Scan rt
	0692	95	0757	0693	Test for =.
	0693	69	0694	0150	Scan left to r
	0694	69	0695	0150	Scan left to b
	0695	10	0696	0697	
	0696	(00	1300	0000)	Location of b-table
	0697	69	0680	0700	Go to 2 S.S.
	0698	69	0699	0100	Scan rt from f
	0699	95	0680	0752	Test for =
	0752	69	0753	0150	Scan left to f
	0753	10	0754	0697	
	0754	(00	1400	0000)	Location of f-table
	0755	69	0756	0100	Scan rt from r
	0756	95	0680	0757	Test for =
	0757	69	0758	0150	Scan left to r
	0758	10	0759	0697	
	0759	(00	1250	0000)	Location of r-table
	0760	10	0761	0697	
	0761	(00	1150	0000)	Location of a-table
	0910	94	0688	0694	

Title - Replace - alphabetic-symbol subschema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0700	24	0500	0701	
	0500	(exit		)	
	0701	21	0671	0702	
	0671	(temp store loc. table)			
	0702	69	0703	0486	Extract subscript
	0703	15	0671	0704	
	0704	15	0705	8002	
	0705	(65	0000	0706)	
	8002	(65	XXXX+mn	0706)	XXXX=1150, 1250, 1300 or 1400
	0706	45	0708	0707	
	0707	01	0707	0000	Error
	0708	46	0714	0709	
	0709	35	0001	0710	
	0710	44	0717	0711	Test for a of deg. 1
	0711	30	0001	0712	
	0712	10	0109	0713	Replace symbol
	0713	69	0500	0200	By table entry
	0714	35	0005	0715	Truncate high order 5 digits
	0715	67	8002	0716	Change - to + (simple r)
	0716	30	0005	0712	
	0717	20	0671	0718	Store: + 0000$\propto$(a)60
	0718	69	0719	0100	Scan rt from A mn
	0719	99	0722	0720	Test for r
	0720	69	0721	0150	Scan left
	0721	65	0671	0711	
	0722	69	0723	0486	Extract subscript of r_{ij}
	0723	15	0724	8002	
	0724	(65	1250	0725)	
	8002	(65	1250+ij	0725)	

Title - S.S. Replace-alphabetic-symbol subschema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0725	46	0726	0720	
	0726	35	0001	0727	-(0) ¦ XXX9(#r eq)30
	0727	44	0728	0729	0=fo3 , 1=fo2
	0728	67	8002	0729	
	0729	35	0003	0730	
	0730	21	0282	0731	
	0282	(temp.store	±0 XXX)		
	0731	69	0109	0732	
	0732	24	1512	0733	# rij →Sc_1
	0733	67	8002	0734	9(#r eq) 3 0000
	0734	30	0001	0735	
	0735	69	8003	0736	0 → distributor
	0736	22	0109	0737	#(r eq) → Sc
	0737	69	0738	0100	Scan rt
	0738	98	0739	0737	Test for q
	0739	10	1512	0741	#(rij) ⟶ U.A. q in L.A.
	0740	69	0742	0200	Store q in #(rij)
	0741	21	0109	0740	Sc_1 → Sc.
	0742	69	0743	0150	Scan left to a
	0743	65	0282	0744	± XXX (numeric)
	0744	35	0002	0745	
	0745	15	0671	0746	0000 (∝ ± XXX)60
	0746	46	0747	0748	
	0747	01	0747	0000	Error
	0748	30	0001	0749	
	0749	10	0109	0750	# (Amn)
	0750	69	0751	0200	(∝ ±N)6 →#(Amn)
	0751	69	0500	0100	Scan rt to q (replaced r)

Title - S.S. Replace numerical constants

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
0684	0762	90	0685	0911	Test numeric
	0911	69	0912	0913	
	0912	(65	1099	0773)	
	0913	24	0764	0766	
	0766	65	0764	0767	Init. (0764)=65 1099 0773
	0767	16	0768	0769	
	0768	(65	1150	0773)	
	0769	45	0770	0804	
	0770	15	0771	0772	
	0771	(65	1151	0773)	
	0772	20	0764	8002	
	8002	(65	11XX	0773)	
	0773	45	0774	0780	
	0774	35	0005	0775	+00000 0XXX9 ¦ (β) 9 00000
	0775	11	0140	0776	
	0776	44	0766	0777	
	0777	30	0005	0778	0---0¦ 00000 (β)9
	0778	10	0109	0779	
	0779	69	0680	0200	(β)9 → #(N) Go to 1 S.S.
	0780	60	0140	0781	0---0 $X_1X_2X_3$9 → U.A.
	0781	11	0446	0782	(0446) = 0---09
	0782	45	0783	0790	
	0783	36	0000	0786	
	0784	10	0785	0789	
	0785	(58	0000	0000)	
	0786	10	8002	0787	Convert $X_1X_2X_3$ to
	0787	30	0002	0788	floating decimal
	0788	11	8002	0784	
	0789	67	8003	0790	

Title - S.S. replace numerical constants

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0790	10	1505	0792	βctr → U.A.
	0791	69	0793	0050	Punch: N → β
	0792	21	0282	0791	
	0282	(Store β temporarily)			
	0793	65	0282	0794	
	0794	30	0003	0795	
	0795	15	0446	0796	
	0796	20	0282	0797	00000 (β)9 → 0282
	0797	65	0140	0798	00000 $0X_1X_2X_3$9
	0798	35	0005	0799	
	0799	15	0282	0800	
	0800	10	0764	0801	
	0801	11	0802	8003	
	0802	(44	9999	9970)	
	8003	(20	11XX	0803)	$0X_1X_2X_3$9(β)9 → 11XX
	0803	65	0282	0778	
	0804	01	0804	0000	Error
	0854	69	1500	0855	
	0855	24	0109	0680	

Title - Third Scan: replace q's after computer table

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
1 S.S.	0805	69	1500	0806	Pr → Sc
	0806	24	0109	0807	
	0807	69	0808	0100	Scan rt
	0808	98	0812	0809	Test for q
	0809	65	0109	0810	
	0810	16	1501	0811	Test end of formulation
	0811	45	0807	0813	I: Go to Fourth Scan
	0812	69	0807	0470	q-replacement subschema see 8 I.A.

Title - Fourth Scan: replace r's in computer table

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
T.S.	0813	69	0250	0814	
	0250	(00	9600	0000)	
	0814	24	0109	0815	
	0815	69	0816	0100	Scan rt.
	0816	99	0820	0817	Test for r
	0817	65	0109	0818	
	0818	16	1501	0819	Test end of computer table
	0819	45	0815	0822	I: Go to End Phase 1
	0820	10	0759	0821	
	0759	(00	1250	0000)	Location of r-table
	0821	69	0815	0700	Replace-alphabetic subschema see 2 S.S.

Title – Scan Right

Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
0100	24	0103	0056	
0106	65	0109	0113	0109 Scan counter
0113	98	0117	0118	9take left half, 8 take right half
0118	16	0121	0126	0109: $00(\overset{9}{8})X_1X_2X_3$ 0000
0121	00	1000	0000	Increment Sc
0126	10	8002	0135	
0135	10	0138	0143	form 65 1XXX 0145 in U.A.
0138	64	3000	0145	
0143	20	0109	8003	
8003	65	1XXX	0145	
0145	35	0005	0108	
0108	65	8003	0115	L.A. contains contents of
0115	45	0119	0106	cell in positions 1–5
0119	30	0001	0125	Isolate d_1, i.e. low order
0125	35	0001	0132	digit of symbol
0132	16	8001	0137	
0137	24	0140	0144	Store symbol
0144	35	0004	0101	Shift d_1 to posit. 5
0101	16	0104	8002	
0104	60	0164	0182	Note: for phase 2 this routine
8002	60	$0164+d_1$	0182	is altered as shown in p.1
0164	89	9999	9999	of 1 End
0165	99	9999	9998	
0166	99	9999	9989	
0167	99	9999	9899	
0168	99	9999	8999	
0169	99	9998	9999	
0170	99	9989	9999	
0171	99	9899	9999	
0172	89	9999	9999	

Title – Scan Right

Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
0173	99	9999	9999	
0182	90	0136	0187	Br. on letter (8 in pos.8)
0187	15	0140	0195	
0195	11	8003	0103	
0136	65	0140	0183	
0183	30	0003	0141	
0141	35	0004	0201	
0201	16	0154	0159	
0154	30	0010	0130	
0159	11	0112	0217	
0112	10	0000	0000	
0217	16	8002	8001	
8001	30	$000x_c$	0130	
0130	10	0133	$018'_1$	
0133	99	8999	9999	
0117	15	0221	0175	
0221	00	1001	0000	
0175	10	8002	0134	
0134	10	0139	0146	
0139	64	1999	0225	
0146	20	0109	8003	
8003	65	1XXX	0225	
0225	35	0005	0131	
0131	11	8003	0145	clear upper
0103	(	exit	)	
0056	69	0060	0064	
0060	(45	0119	0106)	Initialize Scan rt.
0064	24	0115	0106	for blanks

Title – Scan left: initialize scan register

Loc. of Instr.		Op. Code	Data Address	Instr. Address	Notes
0150		24	0103	0065	
0156		65	0109	0163	
0163		98	0066	0218	
0218		16	0221	0126	
0066		15	0121	0175	
		Initialize Scan Counter			
0209		24	0162	0116	
0116		30	0001	0123	
0123		35	0004	0142	
0142		20	0109	0162	
0162	=	(	exit	)	
0065		69	0069	0072	
0069		(45	0119	0156)	
0072		24	0115	0156	

Title – Replace U.A. cell Loc. by Low order five positions of L.A.

Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
0200	24	0103	0206	
0206	20	0111	0114	
0114	30	0005	0127	
0127	44	0184	0232	
0184	35	0005	0161	
0232	35	0004	0193	
0193	60	8003	0151	
0151	35	0004	0161	
0161	69	8003	0268	
0268	98	0122	0223	9 rep.right cell of previous register
0223	10	0226	0181	8 rep.left cell of same reg.
(0226	19	1999	0103)	
0181	21	0186	0189	
0189	10	0192	8003	
(0192	45	0000	0026)	
8003	65	$\alpha - 1$	0129	
0129	30	0005	0147	
0147	35	0005	0208	
0208	15	0111	0186	
0186	20	1XXX	0103	
0122	10	0075	0179	
(0075	19	3000	0103)	
0179	21	0186	0239	
0239	10	0242	8003	
(0242	45	0000	0085)	
8003	65	α	0188	
0188	35	0005	0149	
0149	11	8003	0158	
0158	10	0111	0120	

0120	30	0005	0186
0186	20	1XXX	0103

Title – Store–Punch subroutine (store from L.A.) for Instructions

Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
0000	24	0103	0043	
0043	20	00^{28}_{31}	0038	
0043	71	0027	0023	
(0043	20	0027	0038)	initially
0038	65	0043	0048	
0048	16	0102	0107	
0107	45	0110	0011	
0110	15	0013	0017	
(0102	20	0031	0038)	
(0013	20	0032	0038)	
0017	20	0043	0051	
0011	65	0014	0017	
0023	20	0027	0061	
0061	65	0034	0018	
0018	15	0022	0037	
(0022	00	0005	0000)	
(0014	71	0027	0023)	
0037	20	0034	0046	
0046	60	1504	0052	1504 = I.C.
0052	11	1505	0105	1505 = β ctr.
0105	46	0058	9999	I exceeded storage
0058	65	0062	0017	
(0062	20	0028	0038)	
(0036	80	0005	0110)	
0033	MAR	– –	–	
0051	65	1504	0059	
0059	15	0063	0067	
(0063	00	0001	0000)	
0067	20	1504	0103	

Title – Punch subschema for Data (punch from L.A. into Permanent storage specified by position 8–5 of U.A.)

Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
0050	24	0103	0057	
0057	20	0077	0016	
0016	21	0084	0020	Neg.Zero wired into all
0020	71	0077	0053	other fields.Punch 1 field
0053	67	8003	0015	
0015	16	1505	0019	If (U.A.)=(βctr), then
0019	45	0103	0012	decrement β ctr. before exiting
0083	(MAR	–	–)	Placed in 0083 by Phase 1
0086	80	0005	0110	Control word for punching
		Decrement β Ctr by 1		
0040	24	0103	0012	
0026	15	1505	0073	1505 = β ctr.
0012	66	0068	0026	
0068 =	(00	0001	0000)	
0073	20	1505	0103	
		Decrement β Ctr. by contents of L.A. positions 8–5		
0025	24	0103	0026	(i.e. L.A. should be negative)

Title – Phase 1: Constants for f-table; library f's

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
f01	1401	-00	0001	4011	Identity function
f02	1402	-00	0001	4021	+
f03	1403	-00	0001	4031	–
f04	1404	-00	0001	4041	•
f05	1405	-00	0001	4051	/
f06	1406	-00	0001	4061	Neg. Identity
f07	1407	-00	0001	4071	Absolute value
f08	1408	-00	0001	4081	Minimization ftn
f09	1409	-00	0001	4091	Branch ftn =
f10	1410	-00	0001	4101	Branch ftn –
f11	1411	-00	0001	4111	Branch ftn $\geq$
f12	1412	-00	0001	4121	Square root
f13	1413	-00	0001	4131	Neg. absolute value
f14	1414	-00	0001	4141	
f00	1400	-00	0001	4001	
f16	1416	-00	0001	4161	log base e
f17	1417	-00	0001	4171	sine
f18	1418	-00	0001	4181	cos
f19	1419	-00	0001	4191	tan
f20	1420	-00	0001	4201	arctan
f21	1421	-00	0001	4211	arcsin
f22	1422	-00	0001	4221	arc cos
f23	1423	-00	0001	4231	sinh
f24	1424	-00	0001	4241	cosh
f25	1425	-00	0001	4251	tanh
f26	1426	-00	0001	4261	arc cosh
f27	1427	-00	0001	4271	arc tanh
f28	1428	-00	0001	4281	arc sinh
f15	1415	-00	0001	4151	exponential

Title - End Phase 1

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0822	69	0823	0824	
	0823	(99	8999	9999)	
	0824	24	0172	0825	
	0825	69	0826	0827	
	0826	(98	9999	9999)	Modify scan right sub-
	0827	24	0173	0828	schema for Phase 2
	0828	69	0829	0830	
	0829	(98	0136	0187)	
	0830	24	0182	0831	
	0831	69	0832	0833	
	0832	(99	8999	9999)	
	0833	24	0133	0834	
	0834	65	1300	0835	00000 #(boo eq)2
	0835	45	0836	0839	
	0836	30	0001	0837	Truncate 2
	0837	35	0004	0838	
	0838	20	1510	0846	# (boo eq) → E
	0839	01	0839	0000	Error
	0840	70	0007	0088	Read the first card of phase 2a
	0846	65	0847	0848	Initialize table of q's
	0847	(24	1220	0848)	for phase 2
	0900	69	1504	0901	Initialize punch out
	0901	24	0034	0840	17-20 for instructions
	0848	16	0849	0850	
	0849	(24	1230	0848)	
	0850	45	0851	0900	
	0851	15	0852	0853	8002 = 24(1221+j) 0848
	0852	(24	1231	0848)	
	0853	69	1532	8002	1532 = 00001 00001

B. INITIALIZE PHASES 2a and b

Title - Init. Phase 2 : Scan table of set up q's. $M_q7 \rightarrow$ L.A.

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	1200	24	1203	1204	
	1206	69	1209	1212	Initialize Sc
	1209	(00	9222	0000)	with first cell in table
	1212	24	0109	1213	
	1213	20	1217	1220	ooooo$M_q7 \rightarrow$1217
	1220	69	1231	0100	Scan rt.
	1231	91	1279	0236	Test end of table. Non-zero $\longrightarrow$ Acc.
	1203	(exit		)	
	0236	16	1217	1232	Test q
	1232	45	1220	1214	Zero in LA $\Rightarrow$ q setup
	1204	69	0109	1205	
	1205	24	1211	1206	Store Sc
	1214	69	1211	1215	Restore Sc to its initial value
	1215	24	0109	1203	
	1279	69	0109	1280	
	1280	24	1281	1214	
	1281	(store		)	
	1221	00	0010	0001	
	.	.	.	.	Initialize table of q's by read in
	.	.	.	.	
	.	.	.	.	
	1230	00	0010	0001	

Title - Init. Phase 2: Scan TAI for symbol in L.A.

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	1202	24	1203	1207	
	1207	69	1210	1216	
	1210	(00	9250	0000)	# (TAI)-1 → Sc
	1216	24	0109	1244	
	1244	20	1217	1245	
	1245	69	0227	0100	Scan rt
	0227	91	1203	0228	Non-Zero → L.A. if symbol
	0228	16	1217	0229	not in TAI
	0229	45	1245	1203	Zero → L.A. if symbol in TAI
	1282	24	1203	1244	Resume scan with this instruction
			Remove 9 Tag and shift		
	1201	24	1203	1208	
	1208	30	0001	1218	
	1218	35	0004	1203	
	1203	= (exit		)	

Title - Init. Phase 2: $B^i \rightarrow B^{i+1}$ (also A^i, R^i)

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	1198	24	1203	1576	Entry for $B^i \rightarrow B^{i+1}$
	1576	65	1579	1583	
	1579	(00	8574	0000)	= $\#B^{48}$
	1583	69	1586	1589	
	1586	(69	1590	0150)	Set up scan left
	1589	24	1592	0202	
	1595	20	0109	1577	Initialize Sc
	1577	20	1582	1592	Initialize 1582 with Sc
	1582	(temp.storage)			
	1592	69	1590	0150	Scan left; $B^i \rightarrow$ L.A.
	1590	98	0220	1594	Test for end of table
	1594	10	1582	1587	$\#(B^{i+1}) \rightarrow$ U.A.
	1587	69	1591	0200	Store B^i in B^{i+1}
	1591	69	0109	1578	$\#B^{i+1} \rightarrow 1582$
	1578	24	1582	1592	
		Entry for $A^i \rightarrow A^{i+1}$			
	1246	24	1203	1596	
	1596	65	1599	1583	
	1599	(00	8549	0000)	
		Entry for $R^i \rightarrow R^{i+1}$			
	1248	24	1203	1593	
	1593	65	1598	1583	
	1598	(00	8540	0000)	
	0202	69	0109	0212	
	0212	24	0213	1595	
	0220	69	0213	0224	
	0224	24	0109	1203	

Title - Init. Phase 2: $B^{i+1} \longrightarrow B^{i}$ (also A^{i+1}, R^{i+1})

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	1199	24	1203	1584	
	1584	65	1588	1597	Entry for $B^{i+1} \longrightarrow B^{i}$
	1588	(00	9551	0000)	= #B°
	1597	69	1219	1589	
	1219	(69	1590	0100)	Set up scan rt.
			Entry for $A^{i+1} \longrightarrow A^{i}$		
	1247	24	1203	1580	
	1580	65	1585	1597	
	1585	(00	9542	0000)	# A°
			Entry for $R^{i+1} \longrightarrow R^{i}$		
	1249	24	1203	1581	
	1581	65	1523	1597	
	1523	(00	9532	0000)	# R°

Title - Initialize Phase 2. B^i, A^i, R^i tables, TAI

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	1530	00	0080	0008	
	1531	00	0000	0001	R° ; $\#R^\circ = 9532$
	1532	00	0010	0001	
	.	.	.	.	
	.	.	.	.	
	.	.	.	.	
	.	.	.	.	
	.				
	1539	00	0010	0001	
	1540	00	0010	0008	R^{17} ; $\#R^{17} = 8540$
	1541	00	0000	0001	A° ; $\#A^\circ = 9542$
	1542	00	0010	0001	
	.	.	.	.	
	.	.	.	.	
	.	.	.	.	
	.	.	.	.	
	1548	00	0010	0001	
	1549	00	0010	0008	A^{15} ; $\#A^{15} = 8549$
	1550	00	0000	0004	B° ; $\#B^\circ = 9551$
	1551	00	0010	0001	
	.	.	.	.	
	.	.	.	.	
	.	.	.	.	
	1573	00	0010	0001	
	1574	00	0010	0008	B^{48} ; $\#B^{48} = 8574$
	1250	00	0010	0001	
	.	.	.	.	
	.	.	.	.	
	.	.	.	.	TAI
	1281	00	0010	0001	

C. PHASE 2a

Title - b-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0300	69	1510	0301	E $\longrightarrow$ Sc
	0301	24	0109	0302	
	0302	69	0391	0100	Scan right Go to 4B
	0303	16	0304	0305	
	0304	(00	0000	5098)	Test for Eo9. i.e. Vector eq.
	0305	45	0307	0359	I: Go to 11B
0332 0327	0306	01	0306	0000	Error
	0307	15	0308	0309	
	0308	(00	0000	0090)	Test for n_1n_2=oo in En_1n_2.
	0309	45	0789	0310	D: Go to 1 Rec
	0310	69	0311	0150	
	0311	16	1497	0312	
	1497	(00	0000	7018)	Scan left for P01 i.e. comma
	0312	45	0310	0313	
	0313	69	0314	0100	Scan right
	0314	93	0315	0317	Test for r (storage r)
	0315	69	1177	0316	
	1177	(00	0313	0000)	e5 $\longrightarrow$ Ts
	0316	24	1519	0757	I: Go to 6R
	0317	98	0319	0318	Test alphabetic: c,d,e,p,b
	0318	01	0318	0000	Error
	0319	94	0313	0320	Test for d
	0320	92	0324	0389	Test for b I: Go to 4B
0389	0321	16	0322	0323	
	0322	(00	0000	7038)	Test for $\forall \Xi$ P03
	0323	45	0390	1300	I: Go to 1Q. D: Go to 4B
	0324	69	0325	0100	Scan right
	0325	98	0338	0324	Test for alphabetic
	0326	69	0327	0100	Scan right
	0327	91	0328	0306	Test for F
	0328	95	0326	0324	Test for =

Title - b-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0328	16	0329	0330	
	0329	(00	0001	4901)	
	0330	46	0333	0331	
	0331	15	0329	0332	Test for Special Library F
	0332	30	0001	0306	I: Go to 1B
	8002	(00	0000	149X)	
	0333	15	0334	0335	
	0334	(00	0000	0900)	Test for Foo, i.e. phase eq.
	0335	45	0340	0336	
	0336	65	1550	1080	B° I: Go to 12B
0241	0337	15	1069	0339	
	1069	(00	0000	0004)	Tag for phase eq. → B°
	0339	20	1550	0340	
	0340	69	0341	0100	Scan right
	0341	92	0353	0342	Test for b
	0342	16	1497	0343	
	0343	45	0340	0344	Test for comma
	0344	69	0109	0345	# (,) → C
	0345	24	1513	0346	
	0346	69	0347	0150	Scan left from ,
	0347	98	0960	0348	Test for = sign. D: Go to 12B
	0348	93	0552	0346	Test for R D: Go to 10B
	0349	69	0368	0350	Enter subroutine to test
	0350	24	1203	1981	bmn = B^1
	0351	20	0152	0671	Store b (or r) I: See 3R
	0152	(temp.store		)	
0341	0353	10	0354	0349	
	0354	(00	8550	0000)	Initialize Sc with #B°-1
	0355	21	0109	0356	
	0356	69	0367	0100	Scan rt to B^i (R^i)
	1981	16	1033	0351	Subtract o----02

Title - b-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr Address	
	0357	69	0109	0358	
	0358	24	0176	0360	Store Sc, i.e. position in B^i table
	0176	(temp.	store)		
	0360	69	0361	0209	#bi in $B^i \rightarrow$ Sc
	0361	69	0362	0150	Scan left
	0362	69	0363	0100	Scan right bi→L.A.
	0363	16	0152	0364	Test b = bi
	0364	45	0365	0386	I: Go to 4B
	0365	69	0176	0366	
	0366	24	0109	0356	
0356	0367	16	1069	1999	Test end of B^i (R^i) table. See
0349	0368	65	0152	0369	Initialize Sc with # (beq)
	0369	69	0370	0209	
	0370	69	0371	0100	Scan right from b
	0371	94	0370	0372	Test for d
	0372	99	0376	0387	Test for $\alpha \leq 9$ or =. See 4B
0377	0373	69	0374	0375	Set exit of Br Subschema to
	0374	(00	0000	0513)	be 0513. See 8B
	0375	24	0457	0422	(I: Go to 5B
0372	0376	16	1023	0377	
	1023	(00	0000	0003)	Put deg.1 (i.e. '6') tag
	0377	20	1508	0373	on α. $\alpha_6 \rightarrow Q_0$
0401	0378	69	0379	1198	$B^i \rightarrow B^{i+1}$
	0379	65	1512	0381	00000 #(b)0 $\longrightarrow B^0$
	0380	20	1550	0382	
	0381	30	0003	0380	
	0382	65	0152	0383	#(b eq) 2
	0383	69	0384	1201	Remove '2' tag
	0384	20	1510	0402	00 # (b eq) 0000 → E. See 4B
0402	0385	98	0394	0393	Go to 4B

Title - b-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
0364	0386	20	0176	1203	I: Go to 0368, 3B. See 0350, 2B 0 → 176 if B in B^i, #0 → 0176 if no b in B^1
0372	0387	65	0176	0388	
	0388	45	0400	0421	I: Go to 5B
0320	0389	97	0321	0318	Test for P. Go to 1B
0323	0390	46	0318	0313	Test for P=Pol. Go to 1B
0302	0391	98	0392	0302	Test for =. Go to 1B
	0392	95	0303	0302	
0388	0400	69	0109	0401	#(=) → 0176
	0401	24	0176	0378	Go to 3B
0384	0402	60	0176	0385	Go to 3B
0385	0393	11	0221	0395	Form #(=)-1
0385	0394	10	0121	0395	
	0395	21	0176	0396	
	0396	65	1504	0397	α_π =I.C.
	0397	30	0003	1331	Go to 12B
1331	0398	10	0176	0399	
	0399	69	0403	0200	α_π 9 → blank cell left of =.
	0403	65	1514	0404	Min. register
	0404	45	0405	0300	I: Go to 1B
	0405	65	1550	0406	B° = ooooo #(b)0
	0406	15	1023	0407	Add M in Tag = '3' tag
	0407	20	1550	0408	# (b)3 → B°
	0408	21	1514	0409	Store 0 in Min.
	0409	21	1521	0410	Store 0 in T
	0410	65	1598	0411	(1598) = $\#R^{17}$
	0411	69	0412	0414	Go to store blanks subschema (R^i)
	0412	65	1599	0413	(1599) = $\#A^{15}$
	0413	69	0999	0414	Store blanks in A^i D: Go to 6Q

Title - b-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0414	24	1203	0415	
	0415	20	0109	0416	#R^{17}(or #A^{15})→Sc
Store blanks subschema	0416	65	1082	0417	(1082) = 00000 00001
	0417	10	0109	0418	
	0418	69	0419	0200	Store 00001 in R^i (or A^i)
	0419	69	0420	0150	Scan left
	0420	98	1203	0416	Test for beginning R° (orA°)
0388	0421	01	0421	0000	Error
0375	0422	69	0423	0100	Scan rt. from α s9
	0423	69	0424	0100	Scan rt. from $\alpha_\pi 0$
	0424	98	0560	0425	Test for = or λ; if =, in
	0425	65	0109	0426	midst of recursion. D: Go to 10B
	0426	20	0472	0433	Sc → 0472
	0433	98	0470	0432	
	0432	15	0434	8002	8002 = 65(left half of K (bmn))0435
	0434	(64	2000	0435)	
	0435	20	0176	0427	Store (K) = $\delta_0 \delta_1$ ----δ_q. Go to 6B
	0470	15	0471	8002	8002 = 65 (rt half of K(bmn) 1989
	0471	64	3000	1989	Go to 5a B
	0472	(temp.store		)	

Title - b-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	1989	35	0005	1988	
	1988	20	0176	1987	
	1987	65	0109	1986	
	1986	15	1985	8002	
	1985	(64	3001	1984)	
	1984	30	0005	1983	
	1983	15	0176	0435	Go to 5B

Title - b-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0427	69	0428	0150	Scan left
	0428	93	0431	0429	Test for storage r
	0429	16	1497	0430	1497 = ,
	0430	45	0427	0431	Test for comma
	0431	20	1509	0436	0 or # (r) 3 → S+.
	0436	60	1511	0437	1511 = B I
	0437	11	0176	0440	0176 = K(temp store) see 0435, 5B
	0438	36	0000	0439	
	0439	20	1203	0441	
	0440	45	0438	0457	
	0441	67	8003	0442	
	0442	15	0438	0443	
	0443	21	1582	0444	
	0444	65	1203	0445	Compare K.I. with B.I and
	0445	16	1582	0446	obtain first disparate
	0446	35	0004	0447	digit, Si, if any.
	0447	15	0448	0449	
	0448	(35	0000	0453)	B.I. = $\gamma_0 \gamma_1 - - - \gamma_q$
	0449	20	1203	0450	K.I. = $\delta_0 \delta_1 - - - \delta_q$
	0450	15	1033	0451	
	0451	20	1582	0452	1582 = 35 [000(i-1)] 0455
	0452	65	0176	1203	
	0453	65	8002	0454	
	0454	35	0001	0468	I: Go to 7B
	0456	44	0458	0457	Test δ_i = 0
	0457	00	0000	0513	See 0375, 3B. Go to 8B or 4R
	0458	60	1511	1582	B.I. 1582=35 [000(i-1)] 0455
	0459	(10	0000	0000)	
	0460	46	0469	0461	Test γ_i = 0, (i.e.) < 1
1582	0455	11	0459	0460	Go to 7B

Title - b-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0461	11	0459	0462	$\delta_i \neq 0, \gamma_i \neq 0 \Rightarrow \gamma_i = 2, \delta_i = 1$
	0462	45	0469	0463	Test all $\gamma_j = 0$, $j > 1$
	0463	65	1203	0464	δ_{i+1}, δ_{i+2}, - - -
	0464	45	0469	0465	
	0465	65	0176	0466	K
	0466	16	1511	0467	- B.I.
	0467	15	0176	0473	Replace δ_i by o
0454	0468	20	1203	0456	
0460	0469	65	1511	0473	
	0473	35	0005	0474	Store B.I. or new K.I. (low order 5)
	0474	20	1203	0475	
	0475	65	8003	0476	B.I. or new K.I.(high order 5) →L.A.
	0476	10	0472	0071	# (=sign). See 5B I: Go to 7aB
	0477	69	0479	0200	Store B.I.(or K) in K(bmn)
0070	0478	21	0109	0477	#(=sign)→Sc
	0479	65	0109	0035	Scan rt. to #(=)+1 Go to 7a B
0045	0480	65	1203	0481	
	0481	30	0005	0482	Store low order 5 digits in K (bmn)
	0482	10	0109	0483	
	0483	69	0484	0200	
	0484	65	1504	0485	π = (I.C.) = 1504
	0485	15	0486	0487	
	0486	(00	0002	0129)	Form: "Jump to π +2"
	0487	10	1505	0489	βctr. = 1505
	0488	69	0490	0050	Punch: "Jump to π +2" → β
	0489	24	0021	0488	Store β in 0021(temp)
	0490	30	0004	0491	00 0000 (β-1)→L.A.
	0491	15	0021	0492	00 β (β-1)

Title - b-schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
0479	0035	98	0039	0041	
	0039	15	0221	0045	
	0041	16	0121	0045	
	0045	20	0109	0480	I: Go to 7B
0476	0071	98	0055	0070	
	0055	10	0221	0478	
	0070	11	0121	0478	I: Go to 7B
0496	0074	20	0109	0085	
	0085	69	0082	0150	
	0082	69	0497	0100	Go to 8B

Title - b-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0492	15	0493	0512	
	0493	(69	0000	0000)	
0512	0494	69	0495	0000	Punch: 69 $\beta(\beta-1) \rightarrow \pi$
	0495	65	0109	0496	Scan left
	0496	16	0063	0074	Scan left to λ. Go to 7a B
0082	0497	35	0003	0498	00 (λ) 0000
	0498	15	0499	0500	
	0499	(24	0000	0340)	
	0500	10	1505	0501	Punch: 24 λ 0340 $\rightarrow$ B-1
	0501	69	0502	0050	
	0502	69	0503	0150	Scan left to $\alpha\pi 9$
	0503	69	0504	1201	00 ($\alpha\pi$) 0000 $\rightarrow$ L.A.
	0504	15	0486	0505	1470 = 00 0002 0000
	0505	16	1470	0506	00 $\alpha\pi$ 0129 $\rightarrow$ L.A.
	0506	69	0507	0000	Punch: 00 $\alpha\pi$ 0129 $\rightarrow \pi+1$
	0507	65	0508	0509	
	0508	(00	0086	0000)	= loc. of S.C.C. "No OP"
	0509	69	0021	0510	69 β β-1 $\rightarrow$ dist.
	0510	22	0021	1381	Go to 12B
1381	0511	69	0457	0000	Punch "69 α (No OP) β-1" $\rightarrow \pi+2$ Go to 6B
0492	0512	20	0021	0494	
0457	0513	69	0354	0514	#B°-1 $\rightarrow$ Sc.
	0514	24	0109	0515	See 2B
	0515	69	0516	0100	Scan rt. to B°
	0516	94	0517	0521	Test B° for phase mark: '4'
	0517	65	1508	0523	(Qo) = (α)6 I: Go to 9B
	0518	10	1512	0519	Sc_1
	0519	24	0109	0520	$Sc_1 \rightarrow$ Sc i.e. #b mn

Title - b-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0520	69	0340	0200	$\alpha_q \rightarrow$ #b mn Go to 2B
	0521	65	1509	0522	Zero test S+
	0522	45	0524	0517	I: Go to 8B
0517	0523	15	1023	0518	1023=0---03 i.e. $\alpha_q \rightarrow$L.A.
	0524	69	0525	0209	#(req) $\rightarrow$ Sc
	0525	69	0526	0100	Scan rt.
	0526	97	0527	0525	Test for q;
	0527	69	0529	0100	Scan rt.
	0529	98	0531	0675	Test for q
0675	0530	20	1503	0532	Store non-zero in Ic
	0531	21	1503	0532	Store zero in Ic
	0532	69	1512	0533	# b mn $\rightarrow$ Sc
	0533	24	0109	0534	
	0534	69	0535	0100	
	0535	99	0538	0536	Test for address
	0536	97	0538	0537	Test for q
	0537	93	0538	0528	Test for r
	0528	01	0528	0000	Error
	0538	20	1502	0539	Row index $\rightarrow$ I_R
	0539	65	1503	0540	Zero test Ic
	0540	45	0541	0546	
	0541	69	0542	0100	
	0542	99	0545	0543	Test for const.
	0543	97	0545	0544	Test for r
	0544	93	0545	0528	Test for q
	0545	20	1503	0546	Col. index $\rightarrow$Ic
	0546	65	1512	0547	S_1
	0547	30	0003	0548	
0529	0675	97	0530	0531	

Title - b-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0548	20	1541	0549	# (b mn)→A*
	0549	69	0550	0551	
	0550	(00	0340	0000)	0340 = e2
	0551	24	1521	1100	e2 →T. Go to 3A
0348	0552	69	0553	0150	
	0553	96	0558	0554	Test for a of deg. 1
	0554	95	0558	0555	Test for a of deg. 2
	0555	69	0556	0150	
	0556	95	0558	0557	
	0557	01	0557	0000	Error
	0558	69	0550	0559	e2 →T
	0559	24	1521	1003	Go to 3A
0424	0560	16	0304	0561	0304 = Eo9 (See 1B)
	0561	45	0562	0359	0359 = Vector Schema. Go to 11B
	0562	15	0308	0563	00 0000 $0m_1m_20$ L.A. (See 1B)
	0563	35	0008	0564	
	0564	21	0176	0565	Store m_1 temp.
	0565	65	8002	0566	Clear U.A.
	0566	16	0459	0567	0459 = 10 --- 0
	0567	45	0598	0568	Test m_2 = 1. if $\neq$ 1 Go to 12B
	0568	65	1033	0569	
	0569	16	0176	0570	Test $m_1 \leq 2$. Branch for $m_1>2$
	0570	46	0571	0571	has been omitted on model 1.*
	0571	69	1512	0572	Sc_1=#(b mn)→Sc
	0572	24	0109	0573	
	0573	69	0574	0100	Scan rt.
	0574	99	0575	1975	Test for address. I: Go to 13B

*Vector recursions for which all b's are same has been omitted in Model 1.

Title - b-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0575	65	1508	0576	Qo = α (b mn) 6
	0576	10	1512	0577	# b mn
	0577	69	0340	0200	α 6 → # b mn Go to 2B'
1972	0578	21	1503	0579	Zero I.C.
	0579	65	0176	0580	m_1
	0580	16	1033	0581	test m_1=2.
	0581	45	0582	0532	I: Go to 9B
	0582	16	1033	0583	test m_1=4
	0583	45	0584	0532	I: Go to 9B
	0584	46	0591	0585	Test m_1=5
	0585	65	1508	0586	Qo = α_6
	0586	15	1023	0587	1023=0---03 i.e. α9 → L.A.
	0587	10	1512	0588	
	0588	69	0589	0200	α9 → # b mn
	0589	65	8003	0590	0 → L.A. and U.A.
	0590	10	0109	0577	# index. erase index
0584	0591	65	0152	0592	# (b eq) 2 → Sc
	0592	69	0593	0209	
	0593	69	0594	0150	Scan left
	0594	16	0322	0595	Test for $\forall$, i.e. P03
	0595	45	0593	0596	
	0596	69	0597	0100	Scan rt to μ_q
	0597	20	1515	0532	0---0 μ_{q7} → Qind. Go to 9B.
0561 0305	0359	01	0359	0000	Error

Title - b-Schema

Came from		Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
0567		0598	16	0459	0599	0459 = 1 0---0
		0599	45	0600	0600	
		0600	01	0600	0000	Error
0347		0960	69	0961	0962	
		0961	(69	0301	0302)	Read in Phase 2B and
		0962	24	0024	0963	go to 1 Int.
		0963	70	0007	0000	
0625	IR	0964	69	0965	0962	
		0965	(69	0417	0418)	
		0966	01	0966	0000	Error
0336		1080	30	0001	0241	
		0241	35	0001	0337	I: Go to 2B
0398		1331	15	0980	0398	(0980) ='9' tag
		1470	(00	0002	0000)	See 0505, 8B
0510		1381	65	8001	0511	Go to 8B
		1999	45	1997	1998	Check for master phase
		1998	15	1069	0386	Eq, B oo

Title - b-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	1997	15	1069	1980	
	1980	16	0888	1979	Test for end rec. mark see 4R
	1979	45	1978	0356	I: Go to 2B
	1978	16	0063	1977	
	1977	45	1976	0356	I: Go to 2B
	1976	15	0902	0357	I: Go to 3B
0574	1975	97	1974	1973	Test for μ_q 7
	1974	69	1033	1972	1033 = 0 ---02
	1973	69	8003	1972	Zero
	1972	24	0238	0578	0238 = Σ , Go to 11B

Title - r-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0601	91	0602	0306	Test f. I: Go to 1B
	0602	21	1092	1093	Test for special
	0603	46	0604	0331	library fmn. i.e. mn $\geq$ 90
	0604	15	0329	0605	
	0605	16	1342	0606	Test for f min
	0606	45	0611	0607	
	0607	65	1521	0608	No TLU Min in Model 1
	0608	45	2002	0609	This branch will not be taken
	0609	65	1598	0610	Store blanks in R^i (See 5B)
	0610	69	1070	0414	Go to 1A
	0611	69	0981	0100	Scan Rt.
0981	0612	97	0613	0622	Test for q
	0613	69	0614	1200	Test setup of q
	0614	45	0619	0615	
	0615	15	1217	0616	$\mu_{q7} \rightarrow$ L.A.
	0616	15	1033	0617	Change tag to '9'
	0617	10	0109	0618	#q $\rightarrow$ U.A.
	0618	69	0611	0200	$\mu_{q9} \rightarrow$ #q
	0619	65	1519	0620	Ts
	0620	45	0621	0609	Zero test Ts
	0621	01	0621	0000	Error
0612 0981	0622	16	1497	0623	Test for comma
	0623	45	0626	0624	
	0624	69	0109	0625	#, $\rightarrow$C and exit to
	0625	24	1513	0964	interpreter. (see 12B)
	0626	69	0627	0150	Scan left
	0627	69	0628	0100	Scan rt.
	0628	94	0629	0630	Test for μ_{q4}
	0629	15	1023	0613	$\mu_{q7} \rightarrow L.A_1$
	0630	96	0631	0652	Test for a^1. Go to 2R
0611	0981	98	0622	0612	

Title - r-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0631	69	0632	0100	Scan rt.
	0632	99	0611	0633	Test for address D: Go to 1R
	0633	97	0634	0637	Test for q
	0634	65	1521	0635	Zero test T
	0635	45	0636	0637	
	0636	01	0636	0000	Error
	0637	65	1519	0638	Zero test Ts
	0638	45	0648	0639	
	0639	69	0640	0150	Scan left to a
	0640	69	0109	0641	$Sc \longrightarrow Sc_1$
	0641	24	1512	0642	
	0642	69	0643	1246	$A^i \longrightarrow A^{i+1}$
	0643	65	1512	0644	
	0644	30	0003	0645	$(\#a)\ o \longrightarrow A^\circ$
	0645	20	1541	0646	
	0646	65	1598	0647	Store blanks in R^i and
	0647	69	1038	0414	go to 1A.
	0648	16	1356	0649	Test Ts=e6.1 See 4Q
	0649	45	0650	0651	
	0650	01	0650	0000	Error
	0651	24	1517	0552	#0→Tq Go to 10B
0630	0652	95	0656	0653	Test for a of deg. 2
	0653	93	0668	0654	Test for r. D: Go to 3R
	0654	99	0611	0655	Test for address. D: Go to 1R
	0655	01	0655	0000	Error
	0656	65	1521	0657	Zero test T
	0657	45	0659	0658	
	0658	69	0637	0100	Scan rt
	0659	69	0660	0100	Scan rt
	0660	97	0661	0663	Test for q

Title - r-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0661	69	0662	1200	Test setup of q
	0662	45	0636	0663	D: Go to 2R
	0663	69	0664	0100	Scan rt.
	0664	97	0665	0637	Test for q
	0665	69	0666	1200	Test setup of q
	0666	45	0636	0667	
	0667	69	0637	0150	Scan left. Go to 2R
0653	0668	69	0109	0669	$Sc \longrightarrow Sc_1$
	0669	24	1512	0676	
0351	0671	69	0109	0672	See 2B
	0672	24	1512	0355	$Sc \rightarrow Sc_1$
	0676	69	0677	0209	$\#(r\ eq) \longrightarrow Sc$
	0677	69	0678	0100	Scan rt.
	0678	99	0698	0683	Test for $\alpha \leq 9$ D: Go to 4R
1930	0679	69	0680	1248	$R^i \longrightarrow R^{i+1}$
	0680	65	1512	0681	$\#(rmn)o \longrightarrow R$
	0681	30	0003	0682	
	0682	20	1531	0692	Go to 4R
0678	0683	60	0109	0684	$\#(=$ of r eq.$) \rightarrow$ U.A.
	0684	98	0685	0686	
	0685	10	0121	0687	Form # (=) - 1
	0686	11	0221	0687	
	0687	21	0176	0688	
	0688	65	1504	0689	$\alpha_\pi = (I.C.)$
	0689	30	0003	0690	
	0690	10	0176	0691	

Title - r-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0691	69	1931	0200	$\alpha\pi \rightarrow \#(=)-1$ Go to 7R
1930 0682	0692	69	0176	0693	$\#(=)-1 \rightarrow$ Sc
	0693	24	0109	0694	
	0694	69	0695	0100	Scan rt to =
	0695	69	0601	0100	Scan rt to f. Go to 1R
0699	0696	69	0697	0375	Set exit of Br Subschema
	0697	(00	0000	0700)	to be 0700. Go to 3B
0678	0698	10	1512	1113	$Sc_1 = \#rmn$ Go to 7R
1113	0699	69	0696	0200	$\alpha s_9 \rightarrow \#rmn$
	0700	65	1092	1936	$\#rmn \rightarrow$ Sc See 7R
1932	0701	24	0109	0611	Go to 1R
7A 1178	0702	69	0703	0704	
	0703	(00	9233	0000)	$\#Wo-1 \rightarrow$ 0710 (temp)
	0704	24	0710	0705	
	0705	65	1509	0706	S+. $\rightarrow$ L.A. i.e. #(req)3
	0706	69	0707	0209	$\#(req) \rightarrow$ Sc
	0707	69	0709	0100	Scan rt.
0717 0718	0708	21	0710	0711	
	0709	10	0710	0716	
	0710	(temp store		)	
	0711	69	0712	0200	$S \rightarrow W^i$
	0712	65	0140	0713	
	0713	16	1497	0714	Test for comma
	0714	45	0707	0715	
	0715	69	0703	0719	
0709	0716	98	0717	0718	
	0717	10	0221	0708	
	0718	11	0121	0708	
	0719	24	0109	0720	

Title - r-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0720	69	0721	0100	Scan rt. to W^i
	0721	93	0723	0722	Test for C
	0722	97	0727	0720	Test for q
	0723	16	0724	0725	
	0724	(00	0000	3018)	Test for Col
	0725	45	0726	0728	
	0726	01	0726	0000	Error
	0727	20	0710	0728	$q \rightarrow R^q$
	0728	65	1502	0729	Ir = $\nearrow_{q9} \rightarrow$ L.A.
	0729	10	0109	0730	
	0730	69	0731	0200	$\nearrow_{q9} \rightarrow W^1$
	0731	69	1391	0100	Scan rt.
	0732	97	0739	0733	
	0733	16	0734	0735	
	0734	(00	0000	3028)	Test for Co2
	0735	45	0736	0739	
	0736	65	1519	0737	
	0737	16	1119	0738	Test Ts=el7
	0738	45	1094	0726	D: Go to 7R
	0739	65	1503	0740	
	0740	10	0109	0741	Ic $\rightarrow W^2$
	0741	69	0742	0200	
	0742	69	1383	0100	D: Go to 7R
	0743	97	0744	0749	Test for q
	0744	16	0710	0745	Compare: Sand R^q
	0745	45	0726	0746	
	0746	65	1502	0747	
	0747	10	0109	0748	
	0748	69	0742	0200	
	0749	16	0724	0750	
0731	1391	98	0733	0732	Test for alphabetic

Title - r-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0750	45	0751	0746	
	0751	15	0724	0752	
	0752	16	1497	0753	(1497)=comma
	0753	45	0742	0754	
	0754	69	0703	0755	#W° -1 → Sc
	0755	24	0109	0756	
	0756	69	1382	0100	Scan rt to =
	1382	69	0601	0100	Scan rt to F. Go to 1R
0316 1B	0757	20	1509	0758	00000 #(req)3 → S+.
	0758	65	0109	0759	
	0759	30	0003	0760	#r → R°
	0760	20	1531	0761	
	0761	65	1509	0762	#(req) → Sc
	0762	69	0763	0209	
	0763	69	0764	0100	Scan rt to =
	0764	69	0765	0100	Scan rt to f
	0765	91	0766	0306	Test for f I: Go to 1B
	0766	69	0767	0100	Scan rt
	0767	97	0768	0726	Test for q
	0768	69	0769	1200	Test setup of q
	0769	45	0770	0771	
	0770	01	0770	0000	Error
	0771	65	1217	0772	
	0772	15	1033	0773	$\mu_{q7+2}=\mu_{q9}$ → IR
	0773	20	1502	0774	
	0774	69	1390	0100	Scan rt
	0775	97	0776	0780	Test for q D: go to 7R
	0776	69	0777	1200	
	0777	45	0770	0778	Test setup of q
	0778	65	1217	0779	μ_{q7}
0774	1390	98	0780	0775	

Title - r-Schema

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0779	15	1033	0780	$\mu_{q9} \rightarrow$ I.C.
	0780	20	1503	0781	
	0781	69	0782	0783	
	0782	(20	1502	0784)	Set exit of 0148, 7A
	0783	24	0148	1160	and go to 6A
0148	0784	69	0785	0786	
	0785	(20	1502	1127)	Reset exit of 0148
	0786	24	0148	0787	
	0787	65	1531	0788	R° → Sc
	0788	69	1379	0209	
	1379	69	1082	1380	
	1380	24	1531	0124	Store blank in R°
	0124	21	1519	0313	0 → Ts Go to 1B
0742	1383	98	0749	0743	Go to 5R
0602	1093	16	0329	0603	Go to 1R
0698	1113	20	1114	0699	Store α_{s9} Go to 4R
0700	1936	45	1935	1932	
	1935	65	1114	1934	α_{s9}
	1934	10	1933	0266	Go to 5A
	1933	(00	9503	0000)	
	1932	69	1512	0701	Go to 4R
0691	1931	65	1092	1930	
	1930	45	0692	0679	D: Go to 4R I: Go to 3R
0738	1094	69	0742	0150	Go to 5R

Title - Addressor

Came from	#	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0	1000	69	1500	1001	Pr → Sc
	1	1001	24	0109	1002	
	2	1002	69	1017	0100	Scan right
	3	1003	96	1035	1004	Test S for a of deg.1
	4	1004	95	1035	1005	Test S for a of deg.2
	5	1005	94	1026	1006	Test S for a, integer
	6	1006	65	0109	1007	
	7	1007	16	1501	1008	Test (Sc) =(P_s)
	8	1008	45	1002	1009	
	9	1009	60	1550	1010	oooooB° → U.A.
	10	1010	35	0009	1011	
	11	1011	11	1012	1013	Test for M in.Tag in B°
		1012	= (90	0000	0000)	Min. tag is a 9 in position
	12	1013	44	1014	1016	1 of B°.
	13	1014	69	1507	1015	$\overline{Sc}$ →Sc
	14	1015	24	0109	0313	Go to e5 1,b
	15	1016	65	1550	1018	B° → LA
1002	16	1017	98	1006	1003	Test for alphabetic.
	17	1018	69	1060	0209	# b_1= (B°)→Sc . Go to 4,A
1060	18	1019	69	1020	0100	Scan right b_1→L.A.
	19	1020	69	1021	0209	Init. Sc with #(= of b_1 eq.)
	20	1021	20	1510	0301	#(=)→E. Go to el.o, b.
	21	1022	15	1023	1024	Change μq4 to μ q7
	22	1023	= (00	0000	0003)	
	23	1024	69	1025	1200	Test setup of q
	24	1025	45	1028	1030	
1005	25	1026	69	0109	1027	Sc →Sc_1
	26	1027	24	1512	1022	
	27	1028	69	1512	1029	Sc_1→Sc
	28	1029	24	0109	1002	

Title - Addressor

Came from	#	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	29	1030	60	1512	1031	#(a int)→U.A. 8-5
	30	1031	15	1217	1032	M_q 7 →L.A. 5-1
	31	1032	15	1033	1034	Change M_q7 to M_q9
	32	1033	= (00	0000	0002)	
	33	1034	69	1028	0200	Replace A int. by Mq9
1003 1004	34	1035	65	0109	1036	
	35	1036	30	0003	1037	Sc = #(a)o→A°
	36	1037	20	1541	1038	
	37	1038	65	1541	1039	
	38	1039	35	0003	1040	A° → Sc
	39	1040	20	0109	1041	
	40	1041	69	1042	0100	Scan Right
	41	1042	20	1502	1043	Row-index →IR
	42	1043	69	1044	0150	Scan left
	43	1044	20	1508	1388	Store a in Q.
1388	44	1045	96	1046	1047	Test for a of deg 1
	45	1046	21	1503	1100	Zero →Ic. Go to 3R
	46	1047	69	1048	0100	Scan right
	47	1048	69	1049	0100	Scan right
	48	1049	20	1503	1050	Col index →I.C. 5-1
	49	1050	65	1508	1051	A^2, i.e. a of deg 2, →LA
	50	1051	69	1052	0209	#($\alpha(a^2)$) in computer table →Sc
	51	1052	69	1053	0100	Scan right
	52	1053	20	1509	1054	Storage r →ST
	53	1054	69	1055	0150	Scan left $\alpha(a^2)9$ →L.A.
	54	1055	16	1056	1057	
	55	1056	= (00	0000	0003)	
	56	1057	20	1508	1100	$\alpha(a^2)6$ →Q. Go to 3R
1126 1109	57	1058	69	0263	1061	Left half of I_R (I_C) →Sc
	58	0263	(Temp.Store		)	

Title - Addressor: Initialize for IR

Came from	Loc.of Inst.	Op. Code	Data Address	Instr. Address	Notes
	1100	69	1101	1102	
	1101	=(00	8502	0000)	#of left half of I R
	1102	24	0263	1103	
	1103	65	1104	1105	
	1104	=(00	1098	0000)	e13 = 1098
	1105	69	0264	1106	Set exit of 0264,
	1106	22	0264	1110	4A to be e 13
	1107	30	0004	1108	
	1108	69	0265	1109	Set exit of 0265
	1109	23	0265	1058	to be e13
					Go to 2A
	1110	69	0266	1111	Set exit of 0266
	1111	22	0266	1107	to be e13
1065	1112	10	0109	1066	Go to 4A

Title - Addressor

Came from	#	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
1018	17.1	1060	69	1019	0150	Scan left Go to 1A
	59	1061	24	0109	1062	
	60	1062	69	0264	0100	Scan rt to IR (Ic)
	61	0264	99	9999	1064	Test numeric. D: go to el3 or el7
	62	1064	97	1065	1089	Test for q I: Go to 5R
	63	1065	15	1033	1112	Change μq 7 to μq 9
1112	64	1066	69	1067	0200	μq 9 → IR (Ic)
	65	1067	65	0140	1068	μq 7 → L.A.
	66	1068	69	0265	1200	Test setup of q. If yes,
	67	0265	45	1070	9999	go to el3 or el7
	68	1070	65	1521	1071	Zero test T
	69	1071	45	1072	1073	
	70	1072	01	1072	0000	Error
	71	1073	69	1074	1075	
	72	1074	= (00	8541	0000)	#A° - 1 → Sc
	73	1075	24	0109	1076	
	74	1076	69	1077	0100	Scan rt and test for first
	75	1077	91	1078	1076	blank, A^k
	76	1078	69	1079	0150	Scan left to A^{k-1}
	77	1079	20	0267	1081	Store #(a) = (A^{k-1})
	78	0267	(temp.store		)	
	79	1081	65	1082	1083	
	80	1082	= (00	0000	0001)	
	81	1083	10	0109	1084	00001 (i.e. blank) → A^{k-1}
	82	1084	69	1085	0200	
	83	1085	69	1086	0150	Scan left
	84	1086	98	1087	1081	Test for beginning of table
	85	1087	65	0267	1088	# (a) → Sc
	86	1088	69	1002	0209	Go to 1A

Title - Addressor

Came from	#	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
1064	87	1089	93	1091	1090	Test for r
	88	1090	01	1090	0000	Error
	89	109L	20	1092	1929	$\neq 0 \rightarrow$ 1092 Go to 9A
7R 1934	92	0266	69	9999	0200	$\alpha(r)9 \rightarrow$ IR (Ic) Go to el3 or el7
	93	1095	69	0601	0100	Scan rt. to $\neq$ Go to 1R
	96	1098	65	1503	1099	Zero test Ic 1098=el3
	97	1099	45	1115	1127	
	98	1115	69	1116	1117	
	99	1116	(00	8503	0000)	# left half of Ic
	100	1117	24	0263	1118	
	101	1118	65	1119	1120	
	102	1119	= (00	1160	0000)	= el7
	103	1120	69	0264	1121	
	104	1121	22	0264	1122	Replace IR by Ic and change exits 0264,0265,0266
	105	1122	69	0266	1123	
	106	1123	22	0266	1124	
	107	1124	30	0004	1125	
	108	1125	69	0265	1126	
	109	1126	23	0265	1058	I.: Go to 2A
	110	1127	65	1515	1128	Test Qind; if $\neq 0$
	111	1128	45	1182	1129	go to 7A
	112	1129	60	1541	1130	$(A^\circ)=\#(a) \rightarrow$ U.A.
	113	1130	35	0003	1133	
	114	1131	15	1508	1132	$(Q^\circ)=\alpha(a)6$
	115	1132	69	1134	0200	$\alpha(a)6 \rightarrow \#(a)$
	116	1133	21	0109	1131	$\#(a) \rightarrow$ Sc

Title - Addressor

Came from	#	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	117	1134	69	1135	0100	Scan rt. from #(a)
	118	1135	65	1502	1136	IR → L.A.
	119	1136	10	0109	1137	9 replaces row index
	120	1137	69	1138	0200	
	121	1138	65	1503	1139	Zero test Ic
	122	1139	45	1140	1149	
	123	1140	69	1141	0100	
	124	1141	60	0109	1142	Store Zero in place of Col. index
	125	1142	69	1149	0200	
1150	126	1143	69	1151	1247	$A^{i+1} \longrightarrow A^{i}$
	127	1144	65	1541	1145	
	128	1145	16	1082	1146	Test A° for blank
	129	1146	45	1038	1147	D: Go to 2A
	130	1147	65	1521	1148	Zero test T
	131	1148	45	1153	1002	I: Go to 1A
	132	1149	69	0109	1150	Sc → Sc_1
	133	1150	24	1512	1143	
	134	1151	69	1512	1152	Sc_1 → Sc
	135	1152	24	0109	1144	
	136	1153	21	1521	1154	Zero→T
	137	1154	65	1517	1155	Zero test Tq
	138	1155	45	1156	0340	I: Go to 2B
	139	1156	21	1517	1157	Zero → Tq
	140	1157	69	1518	1158	Scq → Sc
	141	1158	24	0109	1159	
	142	1159	69	1304	0150	Go to IQ
	143	1160	67	1509	1161	Scan TAI for (ST)=r
	144	1161	69	1162	1202	
	145	1162	45	1173	1163	Non-zero ⟹ no r in TAI
	146	1163	69	1164	0100	Scan rt.

Title - Addressor

Came from	#	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	147	1164	16	1502	1165	Test S=IR
	148	1165	45	1166	1168	
	149	1166	67	1509	1167	Resume scan for S$\mathbf{\mathfrak{T}}$
	150	1167	69	1162	1282	
	151	1168	69	1169	0100	
	152	1169	16	1503	1170	Test S=Ic
	153	1170	45	1166	0243	I: Go to 10A
	154	1171	69	0148	0100	$\mu_R \longrightarrow$ IR
	155	0148	20	1502	1127	Go to 5A (7 R if set by r-schema)
1162	156	1173	65	1509	1174	Test ST minus
	157	1174	46	0966	1175	D: Go to 12B
	158	1175	65	1519	1176	
	159	1176	16	1177	1178	Test Ts = e5
	160	1177	= (00	0313	0000)	
	161	1178	45	1179	0702	I: go to 4R
	162	1179	69	1180	1181	
	163	1180	= (00	1160	0000)	e17 $\longrightarrow$ Ts
	164	1181	24	1519	0702	Go to 4R
1096	165	1182	46	1183	0975	Test Qind ≤ 0 I: Go to 8A
1128	166	1183	65	1502	1970	Scan TAI for IR = row index
	167	1184	69	1185	1202	of rec. b with $n_2=2$, $1 \leq n_1 \leq 8$
	168	1185	45	1194	1186	D: Go to 8A
	169	1186	69	1187	0100	
	170	1187	16	0322	1188	Scan rt. and test for P03
	171	1188	45	1189	1191	(see 1B)
	172	1189	65	1217	1190	Resume scan for μ_q 7
	173	1190	69	1185	1282	
	174	1191	69	1192	0100	Scan rt.
	175	1192	20	1502	1193	$\mu \rightarrow$IR $\mu :=$(IR)mod(n_1+1)
	176	1193	21	1515	1129	
1183		1970	16	0238	1184	

Title - Addressor

Came from	#	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
1185	177	1194	65	1502	1195	IR
	178	1195	69	1196	1201	
	179	1196	15	1431	1197	Form and Punch: "Ins (IR)"
	180	1197	69	0998	0000	
	181	0998	67	1515	0997	Qind
	182	0997	69	0996	1201	
	183	0996	15	1285	0995	Form and Punch:
	184	0995	69	0994	0000	"Modulo (Qind)"
	185	0994	65	1505	0993	βctr = $\bar{\mu}$
	186	0993	15	1442	0992	Punch: "Store $\bar{\mu}$ "
	187	0992	69	0991	0000	
	188	0991	65	1502	0990	
	189	0990	10	0109	0989	IR → TAI
	190	0989	69	0988	0200	
	191	0988	69	0987	0100	
	192	0987	65	0322	0986	P03 → TAI
	193	0986	10	0109	0985	
	194	0985	69	0977	0200	
	195	0984	65	1505	0983	
	196	0983	30	0003	0982	
	195	0982	15	0980	0979	$\bar{\mu}$ → TAI
	196	0980	= (00	0000	0009)	
	197	0979	10	0109	0978	
	198	0978	69	0976	0200	
0985	199	0977	69	0984	0100	
	200	0976	69	1183	0040	Decrement βctr Go to 7A
1182	201	0975	69	0974	1202	Scan TAI for μq = (Qind)
	202	0974	45	0973	0972	
	203	0973	01	0973	0000	Error
	204	0972	69	0971	0100	Scan rt

Title - Addressor

Came from	#	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	205	0971	16	1497	0970	
		1497	(00	0000	7018)	Test for P01
	206	0970	45	0969	0967	
	207	0969	65	1515	0968	Qind
	208	0968	69	0974	1282	Scan TAI for q Go to 8A
	209	0967	69	1097	0100	Scan rt
	210	1097	68	8002	1096	- (m+2)9 → Qind
	211	1096	20	1515	1182	Go to 7A
1091		1929	65	1541	1928	# bmn (or a mn) → Sc
		1928	69	1927	0209	
		1927	69	1926	0100	
		1926	69	0109	1925	
		1925	24	1512	1924	
		1924	65	1092	0676	Go to 3R
1044		1388	69	1389	0150	Scan left
		1389	69	1045	0100	Scan rt Go to 2a

Title - Phase 2a Addressor

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
			Compare KI with BI		
1170	0243	65	0109	0244	
	0244	98	0199	0245	
	0245	15	0246	8002	
	0246	= (64	2000	0247)	
	0247	16	1511	0248	
	0248	45	1166	0196	D: Go to 7A
	0196	65	0109	0197	
	0197	15	0063	0198	
	0198	20	0109	1171	I: Go to 7A
	0199	15	0194	8002	
	0194	= (64	3000	1996)	
	1996	35	0005	1995	
	1995	20	0140	1994	
	1994	65	0109	1993	
	1993	15	1992	8002	
	1992	(64	3001	1991)	
	1991	30	0005	1990	
	1990	15	0140	0247	

Title — Q-Schema

Came from	Loc. of Instr.		Op. Code	Data Address	Instr. Address	Notes
e 6	1300		21	0257	1302	St. Zero in $\overline{Q}_L$
	0257	=	(	$\overline{Q}_L$	)	
	1302		21	0258	1412	St. zero in $\overline{Q}u$
	0258	=	(	$\overline{Q}u$	)	
e 8	1412		65	1413	1414	
	1413	=	(00	0259	0000)	
	1414		69	0251	1415	Set Qu to QL in 6.1
	1415		22	0251	1416	
	1416		69	0252	1417	Set Qu to QL in 6.3
	1417		22	0252	1418	
	1418		69	0253	1419	Set Qu to QL in 6.4
	1419		22	0253	1420	
	1420		69	0254	1421	Set Qu to QL in 6.5
	1421		22	0254	1422	
	1422		69	1423	1424	
	1423		(20	0257	1370)	set $\overline{Q}u$ to $\overline{Q}L$ in 6.2
	1424		24	0255	1425	
	1425		69	1426	1427	
	1426	=	(69	1386	0200)	set exit of box 7 to be e7.1
	1427		24	0256	1304	
e 6.4	1304		69	1305	0100	scan right
	1305		99	0253	1308	Test S for Address
	0253		20	0259	1384	S → QL(QU). Go to page 4,Q
	0259	=	(	QL	)	
	1308		96	0251	1318	Test S for a of deg.1. I: go
	0251		20	0259	1312	to page 2,Q.00 000 α(a)6→QL(Qu)
	1312		69	1313	0100	Scan right
	1313		99	0255	1315	test S for address
6.2	0255		20	9999	1370	S → $\overline{Q}L$ ($\overline{Q}u$).Go to 2,Q
	1315		93	1373	1317	Test S for r. D: Go to 2,Q
	1317		01	1317	0000	error indication

Title – Q-Schema

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
0255	1370	60	0109	1371	Zero→L.A. Sc→U.A.
	1371	69	1372	0200	Zero→index of ℓ.b.
	1372	69	1384	0150	scan left. Go to 4,Q
1315	1373	69	1310	0150	scan left
	1310	69	0109	1311	sc→sc q
	1311	24	1518	1316	
	1316	20	1517	0558	store≠0 in Tq. Go to 5,b
1308	1318	97	1319	1327	test S for q
	1319	69	0109	1320	sc→Sc_1
	1320	24	1512	0252	i.e.# (9)→sc_1
6.3	0252	20	0259	1321	00 000 μq7→QL(QU)
	1321	69	1322	1200	test q setup
	1322	45	1326	1324	zero implies q is setup
	1324	69	1512	1325	sc_1→sc
	1325	24	0109	1384	Go to 4,Q
	1326	01	1326	0000	error indication
1318 e6.1	1327	93	1328	1329	test S for r
	1329	95	1310	1326	test S for a of deg.2
	1328	69	0109	1333	sc→sc_1
	1333	24	1512	1334	
	1334	69	1337	0209	Initialize Sc with#r eq.
	1337	69	1335	0100	scan right
	1335	99	0254	1340	test for α(9)
	0254	20	0259	1324	α(9)→QL
	1340	69	1338	0100	scan right
	1338	91	1341	1330	Test S for f
	1330	01	1330	0000	error
	1341	16	1342	1343	1408=# of register for fμ
	1342	(00	0001	4081)	test for fμ
	1343	45	1355	1344	D: Go to 4,Q
	1344	69	1374	0100	scan right

Title – Q-Schema

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
1344	1374	16	0261	1376	
	0261 =	(	QM	)	
	1376	45	1377	1345	
	1377	01	1377	0000	error in q of minimization
	1345	69	1347	0100	scan right
	1347	99	1351	1348	test S for constant
	1348	96	1346	1349	Test S for a of deg 1
	1349	95	1366	1350	Test S for a of dg. 2
	1350	01	1350	0000	error
	1346	20	0260	1352	$S \rightarrow QU$
	0260 =	(	Qu	)	
1347	1351	20	0260	1358	u.b. $\rightarrow$ QU
	1352	69	1353	0100	scan right
	1353	99	1354	1365	test S for address
	1354	20	0258	1358	$\alpha \rightarrow \overline{QU}$
	1358	69	1359	0100	scan right
	1359	92	1360	1361	test S for b
	1361	01	1361	0000	error
	1360	10	1512	1362	$Sc_1 = \#rmn \rightarrow U.A.$
	1362	69	1363	0200	Replace rmn by b in $\forall$ eq.rmn
	1363	69	1512	1364	$Sc_1 \rightarrow Sc$
	1364	24	0109	1428	Go to 5,Q
	1365	93	1368	1350	Test S for r
1349	1366	69	1512	1367	$Sc_1 \rightarrow Sc$ q
	1367	24	1518	1316	Go to 2,Q
1365	1368	69	1366	0150	

Title – Q-Schema

Came from	Loc. of Instr.		Op. Code	Data Address	Instr. Address	Notes
1343	1355		69	1356	1357	
	1356	=	(00	1327	0000)	e 6.1 → TS
	1357		24	1519	1369	
	1369		69	1512	1378	Sc_1 = #r → sc q
	1378		24	1518	0611	Go to 1, r
e 7	1384		60	0109	0256	Store zero in (Sc)
	0256		69	1386	0200	
e 7.1	1386		69	1387	0100	scan right to μq7
	1387		20	0261	1392	00000 μq7 → QM
	1392		69	1393	1200	test setup of q
	1393		45	1473	1394	
	1394		01	1394	0000	error
	1473		60	1281	1474	Indicate q setup
	1474		15	0261	1475	00000 μq7
	1475		69	1395	0200	μq7 → Table of q's
	1395		65	1396	1397	
	1396	=	(00	0260	0000)	
	1397		69	0251	1398	change QL to QU in 6.1
	1398		22	0251	1399	
	1399		69	0252	1400	change QL to QU in 6.3
	1400		22	0252	1401	
	1401		69	0253	1402	change QL to QU in 6.4
	1402		22	0253	1403	
	1403		69	0254	1404	change QL to QU in 6.5
	1404		22	0254	1405	
	1405		69	1406	1407	
	1406	=	(20	0258	1370)	Replace $\overline{QL}$ by $\overline{QU}$ in 6.2
	1407		24	0255	1408	

Title – Q-Schema

Came from	Loc. of Instr.		Op. Code	Data Address	Instr. Address	Notes
	1408		69	1409	1410	
	1409		(69	1428	0200)	change exit of 7 to e 8
	1410		24	0256	1304	
1364	1428		65	0257	1429	zero test $\overline{Q}$L
	1429		45	1430	1435	
	1430		69	1432	1201	Remove 9 tag
	1431	=	(60	0000	0180)	"Insert + – " see 1437
	1432		15	1433	1434	
	1433	=	(60	0000	0229)	"Modify – "
	1434		69	1435	0000	Punch "Modify ($\overline{Q}$L)"
	1435		65	0259	1436	
	1436		69	1437	1201	Remove 9 tag
	1437		15	1431	1438	
	1438		69	1439	0000	Punch "Insert (QL)"
	1439		65	0261	1440	QM
	1440		69	1441	1201	Remove 9 tag
	1441		15	1442	1443	
	1442	=	(46	0000	0155)	"Store"
	1443		69	1444	0000	Punch "Store (QM)"
	1444		65	0258	1445	Zero test $\overline{Q}$U
	1445		45	1446	1467	I: Go to 6,Q
	1446		69	1447	1201	Remove 9 tag
	1447		15	1433	1448	
	1448		69	1449	0000	Punch "Modify ($\overline{Q}$U)"
	1449		65	0260	1450	
	1450		69	1451	1201	Remove 9 tag
	1451		15	1431	1452	
	1452		69	1453	0000	Punch "Insert (QU)"
	1453		65	1505	1454	βctr L.A.

Title — Q-Schema

Came from	Loc. of Instr.		Op. Code	Data Address	Instr. Address	Notes
	1454		15	1442	1455	
	1455		69	1456	0000	Punch "Store β"
	1456		65	1504	1457	00 (I.C.)0000 →L.A.
	1457		30	0004	1458	
	1458		15	1505	1459	00(β)(I.C.)
	1459		20	0262	1332	
	0262		00	(0000	0000)	Int.Storage
	1332		69	1461	0040	Decrement βctr by 1
	1461		65	0261	1462	μq 9
	1462		69	1466	1201	Remove 9 tag
	1463		60	8002	1464	
	1464		15	0262	1465	
	1465		69	1476	0050	Data punch:00(β)(I.C.)→μq-1
	1466		16	0068	1463	(0068)=00 0001 0000
1445	1467		65	0260	1468	QU
	1468		69	1469	1201	Remove 9 tag
	1469		35	0004	1471	
	1471		15	1504	1472	
	1472		30	0004	1411	00(QU)(I.C.) in L.A.
	1411		20	0262	1461	
1465	1476		69	0109	1477	Sc → $\overline{Sc}$
	1477		24	1507	1478	
	1478		60	1522	1479	Test sub = 0. $05n_1n_2$8
	1479		45	1480	0999	
	1480		35	0008	1481	
	1481		11	1482	1483	test n_2=1
	1482	=	(18	0000	0000)	
	1483		45	0966	1288	0966=error stop.12,b
1479	0999		21	1519	1000	zero→TS. Go to e9,A

Title – Q-Schema

Came from	Loc. of Instr.		Op. Code	Data Address	Instr. Address	Notes
	1288		65	0261	1289	$\mu_q 9 \rightarrow$ L.A.
	1289		69	1290	1201	Remove 9 tag
	1290		15	1431	1291	
	1291		69	1292	0000	Punch "Insert μ_q"
	1292		65	1293	1294	
	1293	=	(00	0018	0000)	0018 = address in S.C.C.which
	1294		15	1295	1296	contains 1 x 10^0
	1295	=	(65	0000	0225)	"Add – "
	1296		69	1484	0000	Punch "Add 1"
	1484		65	1522	1485	
	1485		30	0002	1486	
	1486		16	1487	1488	
	1487	=	(00	0000	0052)	test n_1 = 2,4
	1488		45	1489	1059	I: Go to 8,Q
	1489		16	1487	1490	
	1490		45	1491	1059	I: Go to 8,Q
	1491		65	0261	1492	00000 $\mu_q 7 \rightarrow$ L.A.
	1492		69	1493	1202	scan TAI for q
	1493		45	1494	1495	
	1494		01	1494	0000	error
	1495		69	1496	0100	scan right
	1496		16	1497	1498	
	1497	=	(00	0000	7018)	test (S) = Pol
	1498		45	1499	1385	
	1499		65	0261	1375	
	1375		69	1493	1282	resume scan
	1282		24	1203	1244	
1498	1385		69	1283	0100	scan right

Title – Q-Schema

Came from	Loc. of Instr.		Op. Code	Data Address	Instr. Address	Notes
	1283		69	1284	1201	
	1284		15	1285	1965	
	1285	=	(00	0000	0235)	"Modulo – "
	1965		20	1964	1286	Store "Modulo m+2"
	1286		69	1287	0000	Punch "Modulo m+2"
	1287		65	1505	1297	βctr → L.A.
	1297		15	1442	1298	form "Store β" β=$\bar{\mu}_q$
	1298		69	1299	0000	Punch "Store β"
	1299		69	1336	0100	Scan right in TAI from m+2
	1336		91	1339	1299	until end of table
	1339		60	0109	1301	
	1301		15	0261	1303	store $\mu_q 7$ in TAI
	1303		69	1306	0200	
	1306		69	1307	0100	scan right
	1307		60	0109	1314	
	1314		15	1309	1323	
	1309	=	(00	0000	7028)	store P02 in TAI
	1323		69	1063	0200	
	1063		69	1238	0100	scan right
	1238		65	1505	1239	
	1239		30	0003	1969	store $\bar{\mu}_q 9$ in TAI
	1240		10	0109	1241	
	1969		15	0980	1240	9 tag
	1241		69	1242	0200	
	1242		69	1243	0040	Decrement βctr by 1
	1243		21	1522	1968	Store zero in sub, Go to 9,Q
1488	1059		69	1460	1202	Scan to end of TAI
	1460		69	1963	0150	scan left. Go to 9,Q

Title – Q-Schema

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	1963	21	1964	1287	store zero in $\overline{M}$
	1964	(	$\overline{M}$	)	Go to 8,Q
1243	1968	65	1964	1962	Zero test $\overline{M}$
	1962	45	1961	0999	I: Go to 6,Q
	1961	65	0261	1960	$\mu_q 7 \rightarrow$ L.A.
	1960	69	1959	1201	Remove tag
	1959	15	1431	1958	
	1958	69	1957	0000	Punch "Insert μq"
	1957	65	1964	1956	(1964)="Modulo m+2"
	1956	69	1955	0000	Punch "Modulo m+2"
	1955	65	1505	1954	
	1954	15	1442	1953	Punch "Store $\overline{\overline{\mu}}_q$"
	1953	69	1952	0000	
	1952	69	1951	0100	
	1951	91	1950	1952	scan to end of TAI
	1950	60	0109	1949	
	1949	15	0261	1948	Record $\mu_q 7$ in TAI
	1948	69	1947	0200	
	1947	69	1946	0100	scan right
	1946	60	0109	1945	
	1945	15	0322	1944	P03 $\rightarrow$ TAI
	1944	69	1943	0200	
	1943	69	1942	0100	
	1942	65	1505	1941	
	1941	30	0003	1940	
	1940	15	0980	1939	$\overline{\overline{\mu}}_q \rightarrow$ TAI
	1939	10	0109	1938	
	1938	69	1937	0200	
	1937	69	0999	0040	decrement βctr. Go to 6,Q

Title - Recursion

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0789	21	0790	0791	Store Zero in P
	0790	(temp store		)	
	0791	35	0008	0792	U.A. L.A. 0---0 n_1 | n_2 0---0
	0792	21	0793	0794	
	0793	(Temp. store		)	0---0 n_1 $\rightarrow$ 0793
	0794	65	8002	0797	
	0795	16	0459	0796	Test n_2 = 1
	0796	45	0959	0799	D: Go to 6 Rec.
	0797	45	0795	0798	
	0798	01	0798	0000	Error
	0799	65	0793	0800	n_1
	0800	16	1033	0801	(1033)=0---02
	0801	45	0802	0803	
	0802	16	1033	0803	
	0803	20	0804	0805	Store 0 if n_1=2,4
	0804	(temp.store		)	Store - if n_1=5, + if n_1=1,3
	0805	65	1033	0806	
	0806	16	0793	0807	test $n_1 > 2$
	0807	46	0816	0808	
	0808	69	0809	0100	Scan right
	0809	98	0810	0808	Test for =
	0810	95	0811	0808	
	0811	69	0812	0150	Scan left
	0812	16	1497	0813	Test for ,
	0813	45	0811	0814	
	0814	69	0109	0815	# (,) $\rightarrow$ P
	0815	24	0790	0816	
	0816	69	0817	0150	
	0817	16	0322	0818	Scan left to $\forall$ = Po3
	0818	45	0816	0822	

Title - Recursion

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0819	69	0820	0100	
	0820	98	0821	0819	Scan rt for = of first eq.
	0821	95	0824	0819	
0818	0822	65	0804	0823	Test $\overline{N}<0$, i.e. n_1= 1,3
	0823	46	0819	0833	
	0824	21	0804	0825	Store 0 in $\overline{M}$
	0825	69	0826	0100	Scan rt
	0826	98	0827	0828	Test for =
	0827	95	0857	0828	D: Go to 3 Rec
	0828	16	1497	0829	test for ,
	0829	45	0825	0830	
	0830	65	0804	0831	
	0831	15	0063	0832	Add 00 0001 0000 to $\overline{M}$
	0832	20	0804	0825	
0823	0833	69	0834	0100	Scan right
	0834	96	0836	0835	Test for a of degree 1
	0835	95	0837	0838	Test for a of degree 2
	0836	69	0838	0100	
	0837	69	0836	0100	
	0838	69	0839	0100	Scan right to q mn, rec. index
	0839	20	0710	0840	q mn $\longrightarrow$ temp store
	0840	69	0109	0841	Sc $\longrightarrow$ Sc_1
	0841	24	1512	0842	
	0842	69	1500	0843	# (,) after computer table
	0843	24	0109	0844	$\longrightarrow$ 0109
	0844	69	0845	0150	Scan left
	0845	98	0846	0847	Test for P (i.e. colon)
	0846	97	0959	0847	D: Go to 6 Rec.
	0847	16	0710	0848	Test for q mn
	0848	45	0844	0849	

Title - Recursion

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0849	69	0850	0100	Scan rt. to N (qmn)=0XXX9
	0850	69	0851	1201	Remove '9' and shift
	0851	20	0804	0852	Store N (q mn) in 0804
	0852	69	1512	0853	
	0853	24	0109	0854	$Sc_1 \rightarrow Sc$ ie. # q mn
	0854	69	0855	0100	
	0855	98	0856	0854	Scan rt. to first =
	0856	95	0860	0854	
0827	0857	69	0858	0150	Scan left
	0858	98	0859	0857	test for =
	0859	95	0860	0857	
	0860	69	0861	0150	Scan left
	0861	69	0868	0100	Scan right
	0862	16	0863	0864	
	0863	= (00	0000	7048)	Test for Po4 i.e. for)
	0864	45	0865	0886	I: Go to 4 Rec.
	0865	65	0109	0866	
	0866	16	0790	0867	Test Sc=P
	0867	45	0861	0886	I: Go to 4 Rec
	0868	98	0869	0862	test for =
	0869	95	0884	0862	
	0870	65	1505	1967	β ctr
	0871	30	0003	0872	form α_S
	0872	15	0980	0873	'9' tag
	0873	10	0109	0874	
	0874	11	0068	0875	# (=)-2 $\rightarrow$ U.A.
	0875	69	0876	0200	$\alpha_S 9 \rightarrow$ #(=)-2
	0876	65	1504	0877	(I.C.)= $\alpha\pi$
	0877	30	0003	0878	form $\alpha\pi 9$
	0878	15	0980	0879	
0870	1967	15	0063	0871	

Title - Recursion

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0879	10	0109	0880	
	0880	98	0881	0882	
	0881	10	0121	0883	#(=)-1 → U.A.
	0882	11	0221	0883	
	0883	69	0861	0200	α_π 9 → #(=)-1
	0884	65	1505	0885	
	0885	16	0804	1982	Subtract $\bar{M}$ from Bctr
0864 0867	0886	69	0887	1198	$B^i \rightarrow B^{i+1}$
	0887	69	0888	0889	record: end-vector -
	0888	= (00	0001	0002)	rec. mark in B°
	0889	24	1550	0890	
	0890	69	0891	0150	Scan left
	0891	98	0892	0898	Test for =
	0892	95	0893	0898	
	0893	69	0894	1198	$B^i \rightarrow B^{i+1}$
	0894	65	0109	0895	
	0895	30	0003	0896	#(=)2 → B°
	0896	15	1033	0897	
	0897	20	1550	0890	
0892	0898	16	0322	0899	test for Po3=∀
	0899	45	0890	0900	
	0900	69	0901	1198	$B^i \rightarrow B^{i+1}$
	0901	69	0902	0903	
	0902	= (00	0002	0002)	Record Begin-Vector-
	0903	24	1550	0904	Rec. mark in B°
	0904	69	0905	0100	Scan right
	0905	96	0908	0906	Test for a of deg. 1
	0906	95	0909	0907	Test for a of deg. 2
	0907	69	0910	0100	Scan rt. to q mn
	0908	69	0907	0100	
0885	1982	20	1505	0870	

Title - Recursion

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0909	69	0908	0100	Scan rt
	0910	69	0109	0911	Sc → Sc_1
	0911	24	1512	0917	
	0912	69	1210	0913	# TAI first cell → Sc
	0913	24	0109	0914	
	0914	69	0915	0100	
	0915	91	0916	0914	test for end of TAI
	0916	65	0793	0918	
0911	0917	20	0793	0912	q mn → temp storage
	0918	10	0109	0919	
	0919	69	0920	0200	μq 7 → TAI
	0920	69	0921	0100	Scan rt
	0921	65	1497	0922	
	0922	10	0109	0923	Pol → TAI
	0923	69	0924	0200	
	0924	69	0925	0100	
	0925	65	0804	0926	$\bar{M}$ = no. of initial values +1
	0926	30	0003	0927	
	0927	15	0980	0928	'9' tag
	0928	10	0109	0929	
	0929	69	0930	0200	$\bar{M}$9 TAI
	0930	69	1512	0931	Sc_1 → Sc
	0931	24	0109	0932	
	0932	69	0933	0100	Scan rt.
	0933	16	0863	0934	Test for Po4 ≡)
	0934	45	0935	0938	
	0935	65	0109	0936	
	0936	16	0790	0937	Test Sc = P
	0937	45	0932	0941	
	0938	69	0939	0150	Scan left

Title - Recursion

Came from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0939	16	1497	0940	Test for comma
	0940	45	0938	0941	
	0941	69	0942	0150	Scan left
	0942	98	0943	0941	Test for =
	0943	95	0951	0944	
	0944	16	1497	0945	Test for ,
	0945	45	0941	0946	
	0946	69	0947	1198	$B^i \longrightarrow B^{i+1}$
	0947	65	0109	0948	
	0948	30	0003	0949	#(,)9 $\longrightarrow$ B°
	0949	15	0980	0950	
	0950	20	1550	0941	
	0951	69	0952	0150	Scan left
	0952	16	0322	0953	Test for Po3 $\equiv \forall$
	0953	45	0954	0957	
	0954	15	0322	0955	
	0955	16	1497	0956	Test for Pol $\bar{=}$,
	0956	45	0951	0941	
	0957	65	1550	0958	
	0958	69	0326	0209	B° $\longrightarrow$ Sc I: Go to 1B
0846	0959	01	0959	0000	Error

D. PHASE 2b

Title – Interpreter

Came. from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
2B	0300	69	0301	0302	See 0960, 12B
	0301	(69	0544	0000)	Set exit of 0543, 11 I
	0302	24	0543	0303	to be 0544
	0303	69	0304	0305	
	0304	(69	0327	0100)	Set exit of 1093, 29 I
	0305	24	1093	0306	to be 0327
	0306	69	1513	0307	#(comma) → Sc
	0307	24	0109	0308	
	0308	69	0309	0150	Scan left
	0309	91	0310	0308	test for f
	0310	69	0109	0311	#(ϕ,) → G
	0311	24	0904	0312	
	0312	21	0906	0313	Zero → L. Go to 2 I
1R	0416	69	0417	0418	See 0964, 12B
	0417	(69	1115	0000)	Set exit of 543, 11 I
	0418	24	0543	0419	to be 1115
	0419	69	0420	0305	Set exit of 1093, 29 I
	0420	(69	0601	0100)	to be 0601 and go to 0305
0324	0421	95	0325	1129	test for =
	0422	16	0423	0424	
	0423	(00	0000	7018)	test for comma
	0424	45	0326	0325	Go to 2 I
0642	0425	01	0425	0000	Error
0421	1129	65	0140	0422	

Title – Interpreter

Came. from	Loc. of Inst.	Op. Code	Data Address	Instr. Address	Notes
	0313	69	0314	0150	Scan left
	0314	24	0901	0315	Distrib. →0901 (Temp)
	0315	60	0316	0317	
	0316	(00	0001	0000)	Add 1 to L
	0317	10	0906	0318	
	0318	21	0906	0319	
	0319	69	0901	0320	(0901) → Distrib. See
	0320	99	0321	0323	Test S for 0314 address
	0321	69	0322	0150	Scan left
	0322	96	0313	0314	Test S for a of deg.1
	0323	91	1975	0324	Test S for f
	0324	98	0421	0326	Test for = or ; . D: go to 1 I
0424	0325	20	0906	0339	Zero →L.Go to 3I
0424	0326	01	0326	0000	Error
	1975	65	0140	0327	
	0327	16	0328	0329	Test S for aux f
	0328	(00	0001	4511)	
	0329	46	0330	0335	
	0330	15	0328	0331	
	0331	30	0001	0332	Go to Li(f)
	0332	69	0333	8002	Deg f →U.A.
	0333	11	0906	0334	Test L > deg f
	0334	46	0335	0339	
	0335	60	1506	0336	
	0336	10	0316	0337	Add 1 to Int. ctr
	0337	21	1506	0338	
	0338	21	0907	0342	Int.ctr. →V

Title – Interpreter

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	0339	69	0340	0341	
	0340	(00	0337	0000)	$\rho \rightarrow$V
	0341	24	0907	0342	
0338	0342	69	0904	0343	G → Sc
	0343	24	0109	0344	
	0344	69	0345	0150	Scan left
	0345	69	0346	0100	Scan right
	0346	16	0328	0347	Test S for aux f
	0347	46	0348	0900	I:go to error stop,27 I
	0348	15	0328	0349	
	0349	30	0001	0350	Go to Li(f). 34 I
	0350	69	0351	8002	Deg f → U.A.
	0351	21	0905	0352	Deg f → D
	0352	20	0706	0353	Zero → su Aux

Title – Interpreter

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	0353	69	0904	0354	G → Sc
	0354	24	0109	0355	
	0355	69	0356	0357	
	0356	(20	0908	0371)	Initialize 0370
	0357	24	0370	0358	
	0358	69	0359	0360	
	0359	(20	0917	0379)	Initialize 0378
	0360	24	0378	0361	
	0361	69	0362	0363	
	0362	(21	0917	0384)	Initialize 0383
	0363	24	0383	0364	
	0364	60	0706	0365	Zero test Su Aux
	0365	44	0900	0366	D:Go to stop,27 I
	0366	69	0367	0100	Scan rt.
	0367	24	0901	0368	Distrib → 0901(Temp)
	0368	69	0370	1201	truncate and shift S → OK
	0370	20	9999	0371	D = 0908 – 0916
	0371	60	0109	0372	Zero → cell which has
	0372	69	0373	0200	just been scanned
	0373	69	0901	0374	(0901) → Distrib.See 0367
	0374	96	0375	0381	Test S for a of deg 1
	0375	69	0376	0100	Scan right
	0376	69	0378	1201	
					S → MK
	0378	20	9999	0379	D = 0917 – 0925
	0379	60	0109	0380	Zero → cell which has
	0380	69	0384	0200	just been scanned

Title – Interpreter

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	0381	99	0383	0382	Test for address
	0382	01	0382	0000	Error
	0383	21	0917	0384	Zero→MK. D=0917-0925
	0384	60	0905	0385	
	0385	11	0316	0386	Decrease value of D by 1
	0386	21	0905	0387	
	0387	44	0388	0397	Zero test D. I:Go to 6 I
	0388	60	0370	0389	
	0389	10	0316	0390	Increment 0370
	0390	21	0370	0391	
	0391	60	0383	0392	
	0392	10	0316	0393	Increment 0383
	0393	21	0383	0394	
	0394	60	0378	0395	
	0395	10	0316	0396	Increment 0378
	0396	21	0378	0366	Go to 4 I

Title – Interpreter

Came from	Loc.of Instr.	Op. Code	Data Address	Instr. Addresss	Notes
0387	0397	69	0904	0398	G→Sc
	0398	24	0109	0399	
	0399	69	0400	0150	
	0400	69	0401	0100	
	0401	16	0328	0402	Test for aux f
	0402	46	0403	0900	I:Go to stop, 27 I
	0403	15	0404	0405	
	0404	(00	0001	3511)	Go to meta generator of f
	0405	30	0001	8002	7, 35, 36 I
meta gen.	0406	65	0907	0407	
	0407	30	0003	0408	
	0408	15	0409	0410	replace f by ↜ or Int ctr
	0409	(00	0000	0009)	
	0410	10	0904	0411	
	0411	69	0412	0200	
	0412	60	0906	0413	Zero test L
	0413	44	0414	0426	I: Go to 9 I
	0414	69	0904	0415	G→Sc
	0415	24	0109	0308	Go to 1 I
0509	0369	69	0377	0100	scan rt for = sign
	0377	95	0479	0369	
	0479	60	0109	0997	
	0997	11	0063	0998	00009 →#(=)↜
	0998	15	0409	0999	
	0999	69	0523	0200	D: Go to 10 I

Title – Interpreter:
Set up meta generator for binary library functions

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
+	1302	69	1030	1031	
	1030	(65	0000	0225)	S.C.C. "Add" → Op.
	1031	24	0464	0450	I: Go to 7a I
x	1304	69	1032	1031	S.C.C. "Mult" → Op.
	1032	(65	0000	0428)	
–	1303	60	0909	1033	
	1033	11	0340	1034	Test 02 = ρ
	1034	44	1035	1037	
	1035	69	1036	1031	S.C.C. "Subt" → Op
	1036	(66	0000	0225)	
	1037	69	1038	1031	S.C.C. "Rev Subt" → Op
	1038	(69	0000	0423)	
÷	1305	60	0909	1039	
	1039	11	0340	1040	Test 02 = ρ
	1040	44	1041	1043	
	1041	69	1042	1031	S.C.C. "Div" → Op
	1042	(65	0000	0529)	
	1043	69	1044	1031	S.C.C. "Inv.Div" → Op
	1044	(69	0000	0226)	
foo	1300	00	0000	0426	Go to 9 I

Title – Interpreter:
Meta-generator for binary library functions: $\pm$, ., .

Came from	Loc. of Instr.		Op. Code	Data Address	Instr. Address	Notes
1072						
1031	0450		60	0909	0451	
	0451		11	0340	0453	Test $O_2 = \rho$
	0452		45	0467	0473	
	0453		44	0474	0454	
	0454		69	0908	0455	$O_1 \rightarrow O_2$
	0455		24	0909	0456	
	0456		65	0917	0457	$M_1 \rightarrow M_2$
	0457		20	0918	0458	
	0458		45	0459	0462	Zero test M_2
	0459		15	0460	0461	
	0460	=	(65	0000	0093)	Punch "Mod (M_2)"
	0461		69	0462	0000	
	0462		65	0909	0463	
	0463		15	0464	0465	Punch "Op(O_2)"
	0464	=	(	OP	)	
	0465		69	0466	0000	
	0466		60	0907	0452	
	0467		11	0340	0468	Test (V) = ρ
	0468		44	0469	0473	
	0469		65	0907	0470	
	0470		15	0471	0472	
	0471	=	(46	0000	0155)	Punch "Store (V)"
	0472		69	0473	0000	
	0473		00	0000	0406	Go to 6 I or to 1073,28I
0453	0474		60	0908	0475	if set by Br. See 1070, 28I
	0475		11	0340	0476	Test $O_1 = \rho$
	0476		44	0477	0486	
	0477		65	0917	0478	Zero test M_1
	0478		45	0480	0482	

Title – Interpreter
Meta generator for binary library functions

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	0480	15	0460	0481	Punch "Mod (M_1)"
	0481	69	0482	0000	
	0482	65	0908	0483	
	0483	15	0484	0485	Punch "Ins(O_1)"
	0484	(60	0000	0180)	
	0485	69	0486	0000	
	0486	65	0918	0458	$M_2 \rightarrow$L.A. Go to 7a I

Title – Interpreter:
Non-Rec End Procedure

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
0413	0426	60	0904	0427	
	0427	11	0428	0429	test G =W_1 i.e. aux f
	0428	(00	9429	0000)	W_1=9429 Wi in 0430–0434
	0429	44	1100	0900	D:Go to 14I I:Go to 27I
1102	0435	69	0436	0150	Scan left
	0436	95	0437	0705	Test for =. I:Go to 19I
	0437	16	0438	0439	
	0438	(00	0000	5008)	Zero test n_1n_2 of = n_1n_2
	0439	45	0760	0440	D:Go to 21I
	0440	69	0441	0150	Scan left
	0441	98	0444	0442	Test alphabetic
	0442	93	0568	0443	Test for r D:Go to 11 I
	0443	92	0635	0440	Test for b(min) D:Go to 14I
	0444	94	0643	0445	Test for d D:Go to 14 I
	0445	16	0423	0446	Test for ,
	0446	45	0447	0501	
	0447	15	0423	0448	
	0448	16	0449	0500	Test for $\forall$, i.e. P_{03}
	0449	(00	0000	7038)	
	0500	45	0440	0738	I:Go to 15 I
0446	0501	69	0507	0502	
	0502	24	0506	0503	
	0503	60	1550	0504	
	0504	30	0001	0505	isolate tag of B^0
	0505	65	8002	0506	
	0506	99	9999	9999	
	0507	16	0508	0509	
	0508	(40	0000	0000)	test for phase tag
	0509	45	0510	0369	I: Go to 6 I
	0510	60	1519	0511	$T_s \rightarrow$U.A.

Title – Interpreter:
Non Rec end procedure

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	0511	44	0512	0515	Test T_S= 0
	0512	11	0513	0514	Test T_S= e 6.1
	0513 =	(00	1327	0000)	
	0514	44	0571	0515	D: Go to 12 I
	0515	69	0516	0100	
	0516	98	0517	0515	Scan rt to = sign
	0517	95	0518	0515	
	0518	69	0519	0150	Scan left
	0519	69	0520	0150	Scan left
	0520	98	0560	0521	test (=)–2 for α_s. See 1145, 30I
	0521	69	0522	0100	Scan rt for = sign
	0522	95	0523	0521	
	0523	65	1504	0524	
	0524	30	0003	0525	replace = sign by λ
	0525	10	0109	0526	
	0526	69	0893	0200	D: Go to 14I
0894	0527	65	1511	0528	BI = 1511
	0528	30	0005	0529	
	0529	10	0109	0530	
	0530	98	0531	0532	
	0531	10	0221	0533	0221 = 00 1001 0000
	0532	11	0121	0533	0121 = 00 1000 0000
	0533	69	0534	0200	BI → #(=)+1, #(=)+2
	0534	60	1511	0535	
	0535	35	0005	0536	
	0536	30	0005	0537	
	0537	65	8003	0538	
	0538	10	0109	0539	
	0539	10	0316	0540	
	0540	69	0895	0200	Go to 0541 (See 14I)

Title – Interpreter:
Non Rec. end procedure

Came from	Loc. of Instr.		Op. Code	Data Address	Instr. Address	Notes
	0541		65	0542	0543	
	0542	=	(00	0000	0080)	S.C.C. "No Op" $\rightarrow \lambda$
	0543		69	9999	0000	D: 0544 or 1115 See 1 I
	0544		65	1550	0545	
	0545		69	0546	1201	Test B^0 for stop
	0546		45	0547	1132	I: Go to 20 I
	0547		20	1512	0548	$B^0 \rightarrow Sc_1$
	0548		69	0549	0502	Isolate tag of B^0
	0549		16	0550	0551	
	0550	=	(30	0000	0000)	Test B^0 min tag
	0551		45	0552	1996	I: go to 33 I
	0552		69	0553	1199	$B^{i+1} \rightarrow B^i$
	0553		69	1512	0554	$Sc_1 \rightarrow Sc$
	0554		24	0109	0555	
	0555		69	0556	0150	Scan left to # bmn-1
	0556		69	0557	0558	
	0557	=	(00	0000	0340)	Read in phase 2a and
	0558		24	0024	0559	go to 2B
	0559		70	0007	0000	
0520	0560		69	0063	0561	
	0561		24	0562	0563	$1 \rightarrow \overline{S}$
	0562	=	(	$\overline{S}$	)	
	0563		69	0564	0565	
	0564		(24	0109	0521)	Set exit 0704, 18I
	0565		24	0704	0566	to be 0521
	0566		69	0109	0567	$Sc \rightarrow Sc_1$
	0567		24	1512	0686	I: Go to 17 I
0442	0568		69	0569	0570	Set exit 0704,18I
	0569	=	(24	0109	0440)	to be 0440

Title – Interpreter:
Non rec. end procedure

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	0570	24	0704	0646	I: Go to 17 I
1110					
0514	0571	65	1505	1113	I: Go to 20 I
1115	0572	30	0001	0573	test R^o for blank
	0573	45	0587	0574	
	0574	60	1519	0575	Zero test T_S
	0575	45	0578	0576	
	0576	69	0577	0558	Read in phase 2a and
	0577	(00	0000	1058)	go to 1058, 2A
0575	0578	11	0579	0580	
	0579	(00	1327	0000)	Test T_S=e6.1
	0580	45	0600	0581	D: Go to 13 I
	0581	20	1519	0582	$0 \rightarrow T_S$
	0582	69	1518	0583	Scq→Sc
	0583	24	0109	0584	
	0584	69	0585	0150	Scan left to # r-1
	0585	69	0586	0558	Read in phase 2a and
	0586	(00	0000	1304)	go to 1304, 1Q
0573	0587	60	1519	0588	Zero test T_S
	0588	45	0598	0589	
	0589	65	1531	0590	
	0590	69	0591	1201	$R^o \rightarrow Sc_1$
	0591	24	1512	0592	
	0592	69	0593	1249	$R^{i+1} \rightarrow R^i$
	0593	69	1512	0594	$Sc_1 \rightarrow Sc$
	0594	24	0109	0595	
	0595	69	0596	0150	scan left to # r-1
	0596	69	0597	0558	read in phase 2a and
	0597	(00	0000	0627)	go to 0627, 1R
0588	0598	11	0579	0599	Test T_S= e6.1

Title – Interpreter:
Non Rec. end procedure

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	0599	44	1117	0589	D: Go to 20 I I: Go to 12 I
1117					
0580	0600	69	1210	0601	#TAI – 1→Sc
	0601	24	0109	0602	
	0602	69	0603	0100	Scan rt. for blank
	0603	91	0604	0602	in TAI
	0604	67	1509	0605	
	0605	10	0109	0606	St→TAI
	0606	69	0607	0200	i.e.#(reg$_0$)3→TAI
	0607	69	0608	0100	Scan rt
	0608	65	1502	0609	
	0609	10	0109	0610	I_R→TAI
	0610	69	0611	0200	
	0611	69	0612	0100	Scan rt
	0612	65	1503	0613	
	0613	10	0109	0614	I_C→TAI
	0614	69	0615	0200	
	0615	69	0616	0100	Scan rt
	0616	65	1511	0617	
	0617	30	0005	0618	
	0618	10	0109	0619	BI(upper half)→TAI
	0619	69	0620	0200	
	0620	69	0621	0100	Scan rt
	0621	65	1511	0622	
	0622	35	0005	0623	
	0623	11	8003	0624	BI (lower half)→TAI
	0624	30	0005	0625	
	0625	10	0109	0626	
	0626	69	0627	0200	
	0627	69	0628	0100	Scan rt
	0628	65	1505	0629	β ctr

Title – Interpreter:
non-rec end procedure

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	0629	30	0003	0630	
	0630	15	0409	0631	
	0631	10	0109	0632	β ctr (√9) → TAI
	0632	69	1116	0200	
1116	0633	69	0634	0558	read in phase 2a and
	0634	(00	0000	1160)	go to 1160, 6A
0443	0635	69	0109	0636	Sc → Sc_1
	0636	24	1512	0637	
	0637	69	0638	1198	$B^i \longrightarrow B^{i+1}$
	0638	65	1512	0639	
	0639	30	0003	0640	
	0640	15	0641	0642	#(b)3 → B^0
	0641	(00	0000	0003)	
	0642	20	1550	0425	Go to incomplete stop 1, I
0444	0643	69	0644	0645	
	0644	(00	0000	0440)	set exit of 1011,16I to be 0440
	0645	24	1011	1000	Go to 16I
0632	1116	69	0633	0040	Decrement βctr by 1
0526	0893	69	0541	0894	
	0894	24	0895	0527	Go to 10 I
	0895	(	exit	)	
0429	1100	10	1101	1102	
	1101	(00	0195	0000)	Test G = $\overline{W}o$ + 1 i.e. storage r
	1102	45	0435	1103	
	1103	65	1509	1104	St → L.A.
	1104	69	1105	1201	shift and truncate. Go to 15 I

Title – Interpreter: ∀ end procedure for non-rec b

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
0500	0738	69	0109	0739	$Sc \rightarrow Sc_1$
	0739	24	1512	0740	
	0740	69	0741	0100	Scan rt
	0741	98	0742	0740	Test for =
	0742	95	0743	0740	
	0743	69	0744	0150	scan left
	0744	69	0745	0150	scan left
	0745	99	0746	0756	Test for αs in #(=) → a
	0746	69	1512	0747	$Sc_1 \rightarrow Sc$
	0747	24	0109	0748	i.e. #∀ → Sc
	0748	69	0749	0100	scan rt
	0749	97	0751	0750	Test for M_q7
	0750	01	0750	0000	Error
	0751	69	0752	1201	00 M_q0000 L.A.
	0752	15	0753	0754	
	0753	= (47	0000	0183)	Punch: Increment M_q"
	0754	69	0755	0000	
	0755	69	0440	0150	Scan left. Go to 9I
0745	0756	69	0757	0502	
	0757	16	0508	0758	test B^0 for phase mark
	0758	45	0759	0746	
	0759	01	0759	0000	Error
1104	1105	20	0109	1106	# req → Sc See 14I
	1106	69	1107	0150	scan left
	1107	98	1108	1106	
	1108	94	1111	1109	Test for d
	1109	16	0423	1110	Test for comma
	1110	45	1106	0571	I: Go to 12I
	1111	69	1112	0645	Go to 14I
	1112	(00	0000	0571)	

Title – Interpreter: Dump end procedure

Came from	Loc. of Instr.		Op. Code	Data Address	Instr. Address	Notes
0807 0645 0863	1000		16	1001	1002	obtain subscript of dmn
	1001	=	(00	0000	4008)	
	1002		35	0008	1003	
	1003		20	0902	1004	n →0902
	1004		44	1014	1005	Zero test m
	1005		16	1006	1007	
	1006	=	(10	0000	0000)	test n=1
	1007		45	1008	1012	
	1008		65	1009	1010	
	1009	=	(00	0025	0151)	Punch: "Punch and skip 2 lines"
	1010		69	1011	0000	
	1011		00	0000	9999	0854, I:0440,0571,0808 (See 0643, 1111,0863,0805)
	1012		65	1013	1010	
	1013	=	(00	0015	0151)	"Punch and skip 1 line"
	1014		11	1015	1016	
	1015	=	(00	0000	0006)	test m > 5
	1016		46	1018	1017	
	1017		01	1017	0000	error
	1018		10	1015	1019	
	1019		65	8003	1020	
	1020		35	0004	1021	
	1021		15	1022	1023	
	1022	=	(46	0326	0155)	Punch: "Store in (0326+m)"
	1023		69	1024	0000	
	1024		65	0902	1025	test n = 0
	1025		45	1005	1011	

Title – Interpreter: Store End–Procedure

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
0859 0570	0646	69	0109	0647	$Sc \rightarrow Sc_1$
	0647	24	1512	0648	
	0648	65	0140	0649	r → L.A.
	0649	69	0650	1202	Search table for r
	0650	45	0651	0652	
	0651	01	0651	0000	error
	0652	65	0109	0653	
	0653	15	0655	0654	Add 2 to sc. i.e. scan
	0654	20	0109	0668	rt. 4 places
	0655	00	0002	0000	
	0668	69	0669	0100	scan rt to $\alpha_s(r)=\mu_r$ 9
	0669	69	0670	1201	Truncate and shift
	0670	15	0460	0671	(0460) = S.C.C. "Modify"
	0671	69	0672	0000	Punch: "Mod. μ_r"
	0672	69	1500	0673	#comma → Sc
	0673	24	0109	0674	
	0674	69	0675	0150	scan left
	0675	16	1217	0676	test for r
	0676	45	0677	0682	
	0677	60	0109	0678	
	0678	11	0679	0680	
	0679	= (00	8600	0000)	
	0680	45	0674	0681	
	0681	01	0681	0000	error
	0682	69	0683	0100	scan rt
	0683	99	0684	0681	obtain N(r)
	0684	69	0685	1201	truncate and shift
	0685	20	0562	0686	$N(r) \rightarrow \overline{S}$
0883 0567	0686	66	0562	0687	test $\overline{S} = 1$
	0687	15	o316	0688	

Title – Interpreter: Store End-Procedure

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	0688	45	0689	0691	test $\overline{S} = 1$
	0689	15	1505	0690	$\overline{S} \neq 1$
	0690	20	1505	0692	sub $(\overline{S}-1)$ from βctr
	0691	65	1515	0692	obtain α_s from βctr
	0692	15	0471	0693	(0471)=S.C.C. "store"
	0693	69	0695	0000	Punch: "Store in α_s"
	0694	69	0703	0040	Decrement βctr
	0695	69	1512	0696	Sc'→Sc
	0696	24	0109	0697	
	0697	69	0698	0100	scan rt
	0698	95	0699	0697	
	0699	65	1505	0700	
	0700	30	0003	0701	
	0701	15	0409	0702	
	0702	10	0109	1130	record $\alpha_s 9$ in # (=)–2
	1130	11	0063	1131	
	1131	69	0694	0200	
	0703	69	1512	0704	Sc'→Sc
	0704	24	0109	9999	I: go to 0521,10I or to 0440, 9I or to 0854,25I or to 0868,26I. See 0563,0568,11 I; 0856, 25I, 0881, 26I.

Title – Interpreter:
End–Procedure for Initial Values Recursion

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
0436	0705	21	0706	0707	Zero $\rightarrow \overline{M}$
	0706	(	$\overline{M}$	)	
	0707	69	0708	0150	scan left
	0708	98	0709	0710	
	0709	95	0715	0710	
	0710	16	0423	0711	test for comma
	0711	45	0707	0712	
	0712	60	0706	0713	
	0713	10	0063	0714	Add 1 to $\overline{M}$
	0714	21	0706	0707	
	0715	69	0716	0150	scan left
	0716	69	0717	0150	scan left
	0717	69	0718	1201	truncate and shift
	0718	15	0706	0719	add α_s to $\overline{M}$
	0719	15	0471	0720	Punch: "store ($\overline{M}$)"
	0720	69	0721	0000	
	0721	69	0722	1199	$B^{i+1} \rightarrow B^i$
	0722	65	1550	0723	
	0723	16	0724	0725	test B^o for marks indicating
	0724	(00	0002	0002)	first equation
	0725	45	0726	0730	
	0726	65	1550	0727	$B^o \rightarrow$ Sc
	0727	69	0728	0209	
	0728	69	0729	0558	Read in phase 2a, and
	0729	(00	0000	0326)	go to 0326, 1B
0725	0730	69	0731	1199	$B^{i+1} \rightarrow B^i$
	0731	65	1550	0732	$\neq$ (=) $\rightarrow$ Sc
	0732	69	0733	0209	
	0733	69	0734	0150	scan left
	0734	69	0735	0100	scan rt to = $n_1 n_2$

Title – Interpreter:
End procedure for initial values recursion

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	0735	20	1522	0736	00000$\equiv n_1 n_2 \rightarrow$sub
	0736	69	0737	0558	Read in phase 2a and
	0737	(00	0000	0310)	go to 0310, 1B
0571	1113	15	0471	1114	
	1114	69	1115	0000	Punch: "Store β" for r equations
	1115	65	1531	0572	I: Go to 12I
0599	1117	20	1519	0600	Set T_S= 0. I: Go to 13I
0546	1132	65	1133	1134	
	1133	(01	1111	1111)	Punch:"Stop"
	1134	69	1135	0000	
	1135	71	0027	0954	I: Go to 38I

Title – Interpreter:
Simple Rec. End–Procedure for = sign

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
0439	0760	16	0761	0762	
	0761	(00	0000	0180)	Test for Branch
	0762	45	1136	1094	D: go to 22I I: go to 30I
1138	0763	15	0761	0764	Store 0—0$n_1n_2$0 in 0902
	0764	20	0902	0765	
	0765	69	0766	0150	Scan left
	0766	69	0767	0150	Scan left to α_s9
	0767	20	0901	0768	$\alpha_s \rightarrow$0901
	0768	65	0902	0769	(0902)=n_1n_2
	0769	30	0002	0770	
	0770	16	0771	0772	
	0771	(00	0000	0005)	Test n_1= 5
	0772	45	0777	0773	
	0773	65	0901	0774	0—0α_s9
	0774	69	0775	1201	
	0775	15	0471	0776	Punch: "Store α_s"
	0776	69	0799	0000	Go to 23I
0772	0777	69	0778	0150	
	0778	16	0449	0779	Scan left for $\forall$ = P03
	0779	45	0777	0780	
	0780	69	0781	0100	Scan rt. to μ_q7
	0781	69	0782	1202	Search TAI for μ_q7
	0782	45	0783	0784	Test for μ_q7 or end of table
	0783	01	0783	0000	Error
	0784	69	0785	0100	Scan rt
	0785	16	0786	0787	test for P02
	0786	(00	0000	7028)	
	0787	45	0788	0790	
	0788	65	1217	0789	μ_q7→L.A.
	0789	69	0782	1282	resume scan

Title – Interpreter:
Simple Rec. End–Proc. for = sign

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	0790	69	0791	0100	Scan rt to $\bar{\mu}_q 9$
	0791	20	0237	0792	00000 $\bar{\mu}_q 9 \rightarrow \bar{Z}$
	0792	69	0793	1201	
	0793	15	0460	0794	Punch "Mod $\bar{\mu}_q$"
	0794	69	0795	0000	
	0795	65	0901	0796	α_s
	0796	69	0797	1201	
	0797	15	0471	0798	Punch: "Store α_s"
	0798	69	0799	0000	Go to 23I
0762	1136	16	1137	1138	
	1137	(00	0000	0100)	
	1138	45	0763	1185	D:Go to 21I. I:Go to 32I

Title – Interpreter:
Simple Rec. End Procedure

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
0798 0776	0799	65	1550	0800	# (= of b-eq) → Sc
	0800	69	0813	0209	
0813	0801	69	0802	0150	scan left to b
	0802	92	0803	0801	
	0803	69	0804	0100	Test for d
	0804	94	0805	0808	
	0805	69	0806	0807	set exit of 1011,16I to be 0808
	0806	(00	0000	0808)	Go to 16I
	0807	24	1011	1000	
1011	0808	69	0809	1199	$B^{i+1} \rightarrow B^{i}$
	0809	65	1550	0810	
	0810	16	0811	0812	
	0811	(00	0001	0002)	Test B^{0} for End-Rec.Mark
	0812	45	0731	0814	D: Go to 19I
0800	0813	20	1512	0801	$Sc \rightarrow Sc_1$
	0814	69	1512	0815	$Sc_1 \rightarrow Sc$
	0815	24	0109	0816	
	0816	69	0817	0150	scan left
	0817	16	0449	0818	Test for $\forall = P_{03}$
	0818	45	0816	0819	
	0819	69	0820	0100	scan rt to $\mu_q 7$
	0820	69	0821	0100	scan rt
	0821	98	0834	0822	test for bij of minimization
	0822	92	0830	0834	
0831	0823	69	0109	0824	$Sc \rightarrow Sc_1$
	0824	24	1512	0825	
	0825	69	0826	1198	$B^{i} \rightarrow B^{i+1}$
	0826	65	1512	0827	# (bij)
	0827	30	0003	0828	
	0828	15	0641	0829	Min Tag = 0——03

Title – Interpreter:
Simple Rec End Procedure

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	0829	20	1550	0832	#(bij)3→B^0
0822	0830	69	0831	1201	#(bij)0→E
	0831	20	1510	0823	
	0832	69	0833	0558	Read in phase 2a and
	0833	(00	0000	0300)	go to 0300,1B
0821 0822 1978	0834	69	0836	0150	scan left to $\mu_q 7$
	0835	69	0836	1201	
	0836	15	0753	0837	Punch:"Increment μ_q"
	0837	69	0838	0000	Go to 25I

Title – Interpreter:
Simple Rec End Procedure

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	0838	69	0839	0100	scan rt
	0839	98	0840	0838	test for =
	0840	95	0844	0841	
	0841	16	0842	0843	
	0842	(00	0000	7048)	Test for P04 i.e. for)
	0843	45	0838	0884	I: Go to 27I
0840	0844	20	0907	0845	= $n_1 n_2 \rightarrow$ 0907
	0845	65	0237	0846	0...0 $\bar{\mu}_q 9 \rightarrow$ L.A. See 0791
	0846	69	0847	1201	
	0847	15	0460	0848	Punch: "Mod.$\bar{\mu}_q$"
	0848	69	0849	0000	
	0849	69	0850	0150	scan left from =
	0850	69	0851	0150	scan left to α_s
	0851	69	0852	1201	
	0852	15	0484	0853	Punch: "Ins α_s"
	0853	69	0854	0000	
	0854	69	0855	0150	scan left
	0855	93	0856	0860	Test for r
	0856	69	0857	0858	
	0857	(24	0109	0854)	set exit of 0704,18I to be 0854
	0858	24	0704	0859	
	0859	24	0903	0646	$\neq 0 \rightarrow$X Go to 17I
	0860	94	0861	0864	Test for d
	0861	69	0862	0863	
	0862	(00	0000	0854)	set exit of 1011,16I to be 0854
	0863	24	1011	1000	Go to 16I
	0864	16	0423	0865	test for comma
	0865	45	0854	0866	
	0866	65	0903	0867	zero test X
	0867	45	0874	0879	

Title – Interpreter:
Simple Rec End Procedure

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
0874	0868	65	0907	0869	$=n_1n_2 \rightarrow$ L.A. See 0844
	0869	16	0438	0870	(0438) = 0 ...05008
	0870	30	0002	0871	(0641) = 0 ...03
	0871	16	0641	0872	test $n_1 > 2$
	0872	46	0873	0877	
	0873	69	0875	1199	$B^{i+1} \rightarrow B^i$
0867	0874	21	0903	0868	Zero $\rightarrow$ X
	0875	69	0876	0100	Scan rt to =
	0876	95	0523	0875	Go to 10 I
	0877	69	0878	0100	Scan rt to =
	0878	95	0838	0877	Go to 25 I
0867	0879	69	0063	0880	
	0880	24	0562	0881	Store 000001 0000 in $\overline{S}$
	0881	69	0882	0883	
	0882	(24	0109	0868)	Set exit of 0704,18I to be 0868
	0883	24	0704	1977	
	1977	69	0109	1976	$Sc \rightarrow Sc_1$
	1976	24	1512	0686	Go to 17I

Title – Interpreter:
Simple–Rec End Procedure

Came from	Loc. of Instr.	Op. Code	Data address	Instr. Address	Notes
0843	0884	65	0542	0885	S.C.C. "No Op"→L.A.
	0885	69	0886	0000	Punch: "No OP"→λ
	0886	69	0887	0150	Scan left
	0887	98	0888	0897	Test for =
	0888	95	0889	0897	
	0889	65	1504	0890	λ = (I.C.)→L.A.
	0890	30	0003	0891	
	0891	10	0109	0892	Replace = by λ0
	0892	69	0896	02oo	
	0896	69	0886	0894	B I → (=)+1, (=)+2 See 10 I and 14 I for subroutine
	0897	16	0449	0898	Test for ∀=P03
	0898	45	0886	0899	
	0899	69	0544	1199	$B^{i+1} \rightarrow B^{i}$ Go to 11 I
0347	0900	01	0900	0000	Error

Title – Interpreter: Branching

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
35 I	1052	24	0471	1053	S.C.C. "Branch on ..." → 0471 See 7a, I ; 35 I
	1053	69	1054	0100	Scan rt from f_{BR}
	1054	16	0423	1055	test for comma
	1055	45	1056	1057	
	1056	01	1056	0000	Error
	1057	65	1059	1058	
	1058	10	0109	1060	
	1059	(00	0000	5188)	Replace comma by = 18
	1060	69	1061	0200	
	1061	69	1505	1062	(βctr) = γ
	1062	24	0907	1063	$\gamma \rightarrow$ V
	1063	69	1036	1064	S.C.C. "Subtract" → 0464= OP
	1064	24	0464	1065	see 7a, I
	1065	69	1065	0150	scan left to f_{BR}
	1066	65	1505	1067	
	1067	30	0003	1068	Replace f_{BR} by γ_o
	1068	10	0109	1069	
	1069	69	1998	0200	
1998	1070	69	1071	1072	
	1071	(00	0000	1073)	set exit of 0473,7a I to be
	1072	24	0473	0450	1073. Go to 7a I
	1073	69	1074	1075	Reset exit of 0473, 7a I to
	1074	(00	0000	0406)	be 0406
	1998	69	1070	0040	Decrement βctr

Title – Interpreter: Branching

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	1075	24	0473	1076	
	1076	69	1077	1078	
	1077	(46	0000	0155)	Restore 0471,7aI.See 1052,
	1078	24	0471	1196	28I
	1079	60	1511	1080	
	1080	45	1084	1081	
	1081	10	1006	1082	1006= 10 0000 0000
	1082	21	1511	1083	
	1083	69	1997	0100	
	1084	60	1511	1085	Replace 6_{j+1} by 1 in B.I.
	1085	30	0001	1086	
	1086	21	1511	1087	
	1087	11	8003	1088	
	1088	45	1089	1084	
	1089	10	1511	1090	
	1090	35	0001	(1999) 1091	see 1196 and 1182
	1091	15	1006	1092	
	1092	36	0000	1082	
	1093	69	9999	0100	scan rt to f of first formula
1083	1997	69	1093	0558	read in phase 2a and go to 0327, 1B or 0601,1R. See 1 I and 11 I.
	1999	10	1197	1092	change 1 to 2
	1197	(00	0000	0001)	
1078	1196	69	1195	1194	
	1195	(35	0001	1091)	
	1194	24	1090	1079	

Title – Interpreter: Branching

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
0762	1094	65	1504	1095	(I.C.) = γ
	1095	30	0003	1096	
	1096	10	0109	1097	
	1097	69	1098	0200	γ replaces $=_{18}$
	1098	69	1099	1139	
	1099	(24	0109	1148)	
	1139	24	0564	1140	
	1140	69	1141	1142	
	1141	(45	0510	1148)	
	1142	24	0509	1143	
	1143	69	1144	1145	Change exits of 0564,0509, 0520
	1144	(98	0560	1148)	9–11 I to be 1148
	1145	24	0520	1146	
	1146	69	1147	0150	
	1147	69	0440	0150	Go to 9 I
	1148	69	1149	1150	
	1149	(24	0109	0521)	
	1150	24	0564	1151	
	1151	69	1152	1153	
	1152	(45	0510	0521)	reset exits of 0564,0509,0520
	1153	24	0509	1154	9–11 I to be 0521
	1154	69	1155	1156	
	1155	(98	0560	0521)	
	1156	24	0520	1157	

Title – Interpreter: Branching

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	1157	69	1158	0100	scan rt to =
	1158	95	1159	1157	
	1159	69	1160	0100	scan rt to γ_0
	1160	60	8002	1161	
	1161	35	0003	1162	
	1162	15	1504	1163	
	1163	15	0063	1164	Data Punch:"Transfer to λ_1+1"
	1164	15	1165	1166	$\longrightarrow \gamma$
	1165	(00	0000	0129)	= S.C.C."Transfer"
	1166	69	1167	0050	
	1167	69	1168	0100	scan rt to γ
	1168	69	1169	0100	scan rt to first f in ψ_1(i.e.ρ)
	1169	65	1505	1170	
	1170	15	1165	1171	Punch:"Transfer to γ_1"
	1171	69	1172	0000	
	1172	65	1505	1173	
	1173	30	0003	1174	Replace ρ by γ_1
	1174	10	0109	1175	
	1175	69	1176	0200	
	1176	69	1177	0100	scan rt to comma after ψ_1
	1177	65	1178	1179	
	1178	(00	0000	5288)	$=_{28}$

Title – Interpreter: Branching

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	1179	10	0109	1180	replace comma
	1180	69	1181	0200	by $=_{28}$
	1181	69	1182	1183	
	1182	(35	0001	1999)	set exit of 1090, 30I
	1183	24	1090	1184	to be 1999
	1184	69	0656	0150	scan left
	0656	69	1079	0040	Decrement to βctr
22I 1138	1185	69	1186	0150	Scan left to γ_1
	1186	60	8002	1187	
	1187	35	0003	1188	
	1188	15	1504	1189	
	1189	15	0063	1190	Data punch:"Trans λ2+1"→γ_1
	1190	15	1165	1191	
	1191	69	1192	0050	
	1192	69	1193	0150	scan left to γ
	1193	35	0003	0487	
	0487	15	1165	0488	Punch "Trans γ"
	0488	69	0489	0000	
	0489	60	1511	0490	
	0490	30	0001	0491	
	0491	21	1511	0492	
	0492	11	8003	0493	Replace last '2' in BI
	0493	45	0494	0489	by zero
	0494	60	1511	0495	
	0495	45	0496	0497	
	0496	36	0000	0497	
	0497	21	1511	0498	
	0498	69	0499	0150	scan left to γ
	0499	69	0523	0150	scan left to =. Go to 10 I

Title – End – Minimization

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
0551	1996	69	1512	1995	$\neq$ b → Sc
	1995	24	0109	1994	
	1994	69	1993	0150	scan to b
	1993	69	1992	0100	
	1992	69	1991	0209	Initialize sc with $\neq$(b eq)
	1991	69	1990	0100	scan rt to α_s
	1990	69	1989	1201	
	1989	15	0484	1988	Punch: "Ins.α_s"
	1988	69	1987	0000	
	1987	65	1504	1986	I.C. = α_π
	1986	15	1985	1984	
	1985	(00	0002	0000)	Punch:"Br on $\geq$ 0 to α_π+2"
	1984	15	1983	1982	
	1983	(00	0000	0121)	
	1982	69	1981	0000	
	1981	69	1980	1199	$B^{i+1} \rightarrow B^i$
	1980	60	1512	1979	$\neq$ b → Sc
	1979	21	0109	1978	
	1978	69	0834	0200	replace b by zero and go to 24 I

Title – Interpreter: Li(f) for degree degree U.A.

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	1402	24	1028	1026	+ [go from fmn to 1400 +mn
	1026	60	1027	1028	see 0350, 3I]
	1027	(00	0002	0000)	
	1028	99	9999	9999	exit
	1403	24	1028	1026	–
	1404	24	1028	1026	X
	1405	24	1028	1026	÷
	1407	24	1028	1029	Ins Abs
	1029	60	0063	1028	(0063) = 00 0001 0000
	1412	24	1028	1029	Sq rt
	1413	24	1028	1029	Neg Abs
	1415	24	1028	1029	e^x
	1416	24	1028	1029	ln
	1417	24	1028	1029	sin
	1418	24	1028	1029	cos
	1401	24	1028	1029	ident
	1406	24	1028	1029	neg ident
	1409	24	1028	1026	Br on =
	1410	24	1028	1026	Br on <
	1411	24	1028	1026	Br on ≥

Title – Interpreter: set up meta generator for unitary functions

Came from	Loc. of Instr.		Op. Code	Data Address	Instr. Address	Notes
	1301		69	0484	1045	Ident.
	1045		24	0464	1046	"Ins Abs"→OP
	1046		65	0917	1047	Zero test M_1
	1047		45	1048	1050	
	1048		15	0460	1049	Punch "Mod (M_1)"
	1049		69	1050	0000	
	1050		65	0908	0990	
	0990		15	0464	0991	Punch "OP 0_1"
	0991		69	0466	0000	Go to 7a I
	1306		69	0992	1045	Neg Ident
	0992		(61	0000	0180)	= S.C.C. "Ins.Minus"
	1307		69	0993	1045	Ins Abs
	0993		(67	0000	0280)	= S.C.C. "Ins Abs"
	1313		69	0994	1045	Neg Abs
	0994		(68	0000	0280)	= S.C.C. "Ins Neg Abs"
	1312		69	0995	1045	Sq Rt
	0995		(66	0000	0380)	= S.C.C. "Sq Rt"
			Branch Functions			
	1309		69	0996	1052	Br on =
	0996	=	(00	0000	0120)	S.C.C. "Br on = "→ 0471. Go to 28 I
	1311		69	1983	1052	Br on ≥
	1983	=	(00	0000	0121)	
	1310		69	1051	1052	Br on –
	1051	=	(00	0000	0126)	

Title – Interpreter: Meta generator for functions needing key cards

Came from	Loc. of Instr.		Op. Code	Data Address	Instr. Address	Notes
e^x	1315		65	0926	0951	R(f) → L.A.
	0926	= –	(00	0079	0000)	initially R(f)=–ℓ(f)(i.e.
	0951		46	0952	0937	minus the length of the subroutine)
	0952		20	0910	0953	–ℓ(f)→0910
	0953		65	1505	0930	"βctr"→ R(f)
	0930		20	0926	0931	
ℓ_n x	1316		65	0927	0978	R(f) → L.A.
	0927	= –(	00	0100	0000)	
	0978		46	0979	0937	test L.A. for –ℓ(f) or +βctr
	0979		20	0910	0980	–ℓ(f)→0910
	0980		65	1505	0981	
	0981		20	0927	0931	
sin x	1317		65	0928	0982	
	0928	= –	(00	0088	0000)	R(f)→ L.A.
	0982		46	0983	0937	
	0983		20	0910	0984	–ℓ(f)→ 0910
	0984		65	1505	0985	
	0985		20	0928	0931	
cos x	1318		65	0929	0986	R(f)→ L.A. Go to 37 I
	0929	= –	(00	0088	0000)	
	0931		15	0932	0933	
	0932	=	(45	0000	0198)	(0932)=S.C.C."subroutine"="63"
	0933		20	0464	0934	"63βctr"→ OP
	0934		65	1505	0935	Decrement βctr by ℓ(f)
	0935		15	0910	0936	
	0936		20	1505	0939	Go to 37 I
	0937		15	0932	0938	"63 βctr"→ OP
	0938		20	0464	0939	Go to 37 I

Title – Interpreter: Meta generator for functions needing key cards

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	0939	60	0908	0940	
	0940	11	0340	0941	Test $O_1 = \rho$
	0941	45	0942	0949	
	0942	65	0917	0943	zero test M_1
	0943	45	0944	0946	
	0944	15	0460	0945	Punch "Mod M_1"
	0945	69	0946	0000	
	0946	65	0908	0947	
	0947	15	0484	0948	Punch "Ins O_1"
	0948	69	0949	0000	
	0949	65	0464	0950	Punch: "OP"
	0950	69	0466	0000	Go to 7a I
1318	0986	46	0987	0937	
	0987	20	0910	0988	$-\ell(f) \rightarrow 0910$
	0988	65	1505	0989	"βctr" $\rightarrow$ R(f)
	0989	20	0929	0931	I: Go to 36 I

Title – Interpreter: Punch out needed key cards

Came from	Loc. of Instr.	Op. Code	Data Address	Instr. Address	Notes
	0954	69	0957	0956	"Non–8" in pos.10 →0036
	0956	24	0036	0957	
	0957	60	0926	0958	Minus test R(f). D:See 0974
	0958	46	0967	0959	
	0959	11	0932	0960	for (0932) see 36 I
	0960	21	0032	0961	00 α0000 →0032
	0961	69	0962	0963	
	0962 =	(00	0001	0000)	Initially. See 0971–0973
	0963	24	0034	0964	
	0964	61	8001	0965	–Zero →0027
	0965	20	0027	0966	
	0966	71	0027	0967	Punch
	0967	60	0962	0968	
	0968	11	0969	0970	test 0962=00 0004 0000
	0969	(00	0004	0000)	
	0970	45	0971	0977	
	0971	10	0972	0973	
	0972	(00	0005	0000)	increment 0962
	0973	21	0962	0974	
	0974	60	0957	0975	
	0975	10	0316	0976	increment 0957
	0976	21	0957	8001	I: Go to 0957
	0977	01	7171	7171	STOP. Formulation has been programmed

E. ERROR INDICATIONS

	Error Indications (Phase 1)	Page
0254:	numeric should be alphabetic	1, I.A.
0257:	non-permissible alphabetic or omission of P02 or P01	1, I.A.
0267:	alphabetic should be numeric	1, I.A.
0271:	zero should not be in computer table	1, I.A.
0673:	duplicate a's in computer table	2, I.A.
0449:	integer a not followed by q	7, I.A.
0498:	subscript of qmn or rmn $\geqslant$ 50	8, I.A.
0466	last symbol of formulation is not comma	1, F.S.
0505:	error on left side of = sign	3, F.S.
0532:	duplicate b's that are not a vector	4 F.S.
0542:	no right parenthesis in vector equation, or right parenthesis in wrong position	4 F.S.
0549:	different subscripts on b's to left of = in vector equation or vector recursion	5 F.S.
0560:	error in quantification of vector recursion	5 F.S.
0566:	no comma to left of $\forall$ or more than one $\forall$ in vector equation	5 F.S.
0577:	duplicate b in double recursion	6 F.S.
0599:	right side of r equation doesn't begin with f	7 F.S.
0618:	duplicate r-equation	7 F.S.
0594:	right side of r-equation is not a well-formed formula	8 F.S.
0636:	fmn defined by f-equation is a library function	9 F.S.
0641:	duplicate f-equation and not vector	9 F.S.
0707:	b,r,f undefined or a not in computer table	2 S.S.
0747:	numeric + α is negative initial address	3 S.S.
0804:	more than 50 numerical constants in formulation	5 S.S.
0839:	no b00 equation	1 of End Phase 1

	Error Indications (Phase 2a and b)	Page
1317:	improper quantifier bound; a of degree 1.	1, Q
1326:	quantifier bound is non-setup Q or non-permissible symbol	2, Q
1330:	right side of r-equation does not beginwith f	2, Q
1350:	error in upper bound of minimization	3, Q
1361:	error in b of minimization	3, Q
1377:	error in Q of minimization	3, Q
1394:	Q was twice quantified	4, Q
1494:	no Q in TAI	7, Q
0957:	double recursion or recursion index not in computer table	6, Rec
0798:	En_1n_2 has $n_2 = 0$, $n_1 \neq 0$.	1, Rec
0359:	vector equation of vector recursion (Not permitted in model 1)/B-equation contains special library f	11, B
0306:	right side of b-equation does not begin with f	1, B
0318:	left side of b-equation has non-permissible symbol or symbols out of order	1, B
0421:	a non-recursion b occurs implicitly, i.e. is referred to twice before being programmed	5, B
0528:	b of degree 1 or 2 with improper index	9, B
0557:	improper argument in b-equation	10, B
0600:	$n_2 \neq 9$, 1, or 2 in vector equation	12, B
0966:	Stop for model 1, incompleteness	12, B
0608:	no TLU minimization permitted in Model 1 will stop with 45 2002 0609	1, R
0621:	r is a quantifier bound and involves a non-setup q	1, R
0636:	q in r-equation not setup in (aq) with $T \neq 0$.	2, R
0650:	storage r involves a'r	2, R
0655:	incorrect operand in r-equation	2, R
0726:	(1) storage r equation does not have C01 or $_{\text{after}}{}_f$q	5, R
	or (2) no q or C02 if TS = e 17	
	or (3) a involves other q	
0770:	q not setup in store b equation	6, R

Error Indications (Phases 2a and b) continued

1072:	entry from b-schema with q not set up	4, A
1090:	improper index: not q, r, nor numeric	5, A
0973:	recursive Q not properly set up	8, A
0425:	incomplete stop: bmm is not permitted in model 1	1, I
0326:	b-equation not well formed formula	2, I
0900:	auxiliary f-equations not permitted in model 1	27, I
0382:	incorrect number of operands for some f	5, I
0750:	error in quantifier schema	15, I
0759:	store r not written or written to left of quantifier	15, I
1017:	m of dmn > 5	16, I
0651:	encoder error: storage r for b equation	17, I
0681:	storage r not in computer table	17, I
0783:	error in Q schema or recursion Schema. No $\mu q7$ in TAI	21, I
1056:	error in formulation of branch equation	28, I

Aeroballistic Research Department
External Distribution List for Applied Mathematics (X3)

No. of Copies	
	Chief, Bureau of Ordnance Department of the Navy Washington 25, D. C.
1	Attn: Ad3
1	Attn: Ad6
1	Attn: Ree
1	Attn: Re9a
	Chief, BuAer Washington 25, D. C.
3	Attn: TD-414
	Commander, U. S. NOTS Inyokern, China Lake, Calif.
1	Attn: Technical Library
1	Attn: Code 503
	Commander, NAMTC Point Mugu, California
2	Attn: Technical Library
	Superintendent U. S. Naval Postgraduate School Monterey, California
1	Attn: Tech Rpts Section
	Director, NRL Washington 25, D. C.
1	Attn: Code 2021
	Officer in Charge, NPG Dahlgren, Virginia
2	Attn: Technical Library
	Office, Chief of Ordnance Washington 25, D. C.
1	Attn: ORDTU
	Chief, AFSWP Washington 25, D. C.
1	Attn: Document Library Br.
	Commander, WADC Wright-Patterson AF Base Ohio
3	Attn: WCORC

No. of Copies	
	Commanding General Aberdeen Proving Ground, Md.
2	Attn: Tech Info Br.
2	Attn: Director, BRL
	Commanding General Redstone Arsenal Huntsville, Alabama
1	Attn: Aero. Lab, GMDD
5	ASTIA Document Service Center Knott Building Dayton 2, Ohio
	NACA High Speed Flight Station Box 273 Edwards AF Base, Calif.
1	Attn: Mr. W. C. Williams
	NACA Ames Aeronautical Laboratory Moffett Field, California
1	Attn: Librarian
	NACA Langley Aeronautical Laboratory Langley Field, Virginia
1	Attn: Librarian
1	Attn: Adolf Busemann
2	Attn: John Stack
	NACA Lewis Flight Propulsion Lab. 21000 Brookpark Rd. Cleveland 11, Ohio
1	Attn: Librarian
1	NACA 1512 H Street, N. W. Washington 25, D. C.
	Commanding Officer, DOFL Washington 25, D. C.
1	Attn: Lib., Rm 211, Bldg. 92

Aeroballistic Research Department
External Distribution List for Applied Mathematics (X3a)

No. of Copies	
1	Chief, Fluid Mechanics Section National Bureau of Standards Washington 25, D. C.
	National Bureal of Standards Washington 25, D. C.
1	Attn: Applied Math Div.
6	Commanding Officer Office of Naval Research Br. Off. Box 39, Navy 100 Fleet Post Office New York, New York
	Langley Aeronautical Laboratory Langley Field, Virginia
1	Attn: Theoretical Aerodynam. Division
	Naval Research Laboratory Washington 25, D. C.
1	Attn: Dr. H. M. Trent Code 6230
	Case Institute of Technology Cleveland 6, Ohio
1	Attn: G. Kuerti
	Massachusetts Institute of Tech. Cambridge 39, Mass.
1	Attn: Prof. Joseph Kaye Room 1-212
	The Johns Hopkins University Charles and 34th Streets Baltimore 18, Maryland
1	Attn: Dr. Francis H. Clauser
2	Director Inst. for Fluid Dynamics and Applied Mathematics University of Maryland College Park, Maryland
	Cornell University Graduate School of Aero. Eng. Ithaca, New York
1	Attn: Prof. W. R. Sears

No. of Copies	
1	AERCON, INC. 560 Punahou St. Altadena, California
	University of Michigan Randall Laboratory Ann Arbor, Michigan
1	Attn: Prof. Otto Laporte
	Brown University Providence 12, Rhode Island
1	Attn: Prof. William Prager
	University of Michigan Engineering Research Institute Ann Arbor, Michigan
1	Attn: Mr. H. E. Stubbs Research Associate
	Lewis Flight Propulsion Lab. 21000 Brookpark Road Cleveland 11, Ohio
1	Attn: Chief, Supersonic Propulsion Div.
	Hughes Aircraft Corp. Culver City, California
1	Attn: Assistant Director GMRD Division
1	McDonnell Aircraft Corp. P. O. Box 516 St. Louis 3, Missouri
	General Electric Company 2900 Campbell Avenue Schenectady 5, New York
1	Attn: Library Guided Missiles Dept.
	Eastman Kodak Company Navy Ordnance Division 50 West Main Street Rochester 14, New York
2	Attn: Mr. W. B. Forman

No. of Copies

Office of Naval Research
Room 2709, T-3 Building
Washington 25, D. C.
1 Attn: Head, Mechanics Br.

Director of Intelligence
Headquarters, USAF
Washington 25, D. C.
1 Attn: AOIN-3B

Director, DTMB
Aerodynamics Laboratory
Washington 7, D. C.
1 Attn: Library

University of California
Berkeley 4, California
2 Attn: Dr. S. A. Schaaf

The University of Texas
P. O. Box 8029
Austin 12, Texas
1 Attn: Defense Research Lab.

1 Applied Math and Statistics Lab.
Stanford University
Stanford, California

University of Michigan
Willow Run Research Center
Willow Run Airport
Ypsilanti, Michigan
1 Attn: Librarian

APL/JHU
8621 Georgia Ave.
Silver Spring, Maryland
2 Attn: Technical Rpts Group

The Ohio State University
Research Foundation
Nineteenth Avenue
Columbus 10, Ohio
1 Attn: Security Officer

No. of Copies

CIT
Pasadena 4, California
1 Attn: Aeronautics Dept.
1 Attn: Jet Propulsion Lab.

University of Minnesota
Minneapolis 14, Minn.
1 Attn: Mechanical Eng. Dept.

RAND Corp.
1700 Main St.
Santa Monica, California
1 Attn: Lib., USAF Project RAND

Douglas Aircraft Co., Inc.
Santa Monica Division
3000 Ocean Park Blvd.
Santa Monica, California
1 Attn: Chief Engineer

1 CONVAIR Corp.
A Div. of Gen. Dynamics Corp.
Daingerfield, Texas

United Aircraft Corporation
400 Main Street
East Hartford 8, Connecticut
1 Attn: Chief Librarian

Guggenheim Aeronautical Lab.
California Inst. of Technology
Pasadena 4, California
1 Attn: Aeronautics Library

Cornell Aeronautical Lab., Inc.
P. O. Box 235, 4455 Genessee St.
Buffalo 21, New York
1 Attn: Librarian

The Avco Manufacturing Corp.,
1 Research Laboratories
2385 Revere Beach Parkway
Everett 49, Massachusetts

External Distribution List for Applied Mathematics

No. of Copies	
1	Oak Ridge National Laboratory Oak Ridge, Tennessee Attn: Dr. A. S. Householder
1	University of Michigan Engineering Research Institute Willow Run Airport Ypsilanti, Michigan Attn: Mr. C. C. Elgot
1	Los Alamos Scientific Laboratory Los Alamos, New Mexico Attn: Dr. S. M. Ulam
1	University of Wisconsin Madison 6, Wisconsin Attn: Dr. S. C. Kleene
1	National Bureau of Standards Washington 25, D. C. Attn: Dr. H. Antosiewicz
1	National Bureau of Standards Washington 25, D. C. Attn: Dr. F. L. Alt
1	Pennsylvania State College State College, Pa Attn: Dr. H. B. Curry
1	Electronic Computer Project Institute for Advanced Study Princeton, N. J. Attn: Dr. H. H. Goldstine
1	Applied Mathematics Laboratory David Taylor Model Basin Washington 7, D. C. Attn: Dr. H. Polachek
1	University of California Berkeley 4, California Attn: Dr. D. H. Lehmer
1	Raytheon Company Waltham 54, Mass. Attn: Dr. R. F. Clippinger
1	Division of Computing Services Institute of Math. Sciences New York University New York 3, N.Y. Attn: Dr. E. Bromberg
1	Westinghouse Electric Corp., Pittsburgh 30, Pa Attn: Dr. C. W. Adams
1	Digital Computer Laboratory Massachusetts Institute of Technology Cambridge 39, Mass. Attn: Dr. Jay Forrester
1	Sperry Rand Corporation Philadelphia, Pa Attn: Dr. Grace Hopper
1	University of Pennsylvania Philadelphia, Pa. Attn: Dr. S. Gorn
1	Data Processing Department Computing Devices of Canada P.O.Box 508 Ottawa 4, Ontario Attn: Mrs. M. Larmour
1	Applied Mathematics University of Illinois Graduate College 168 Engineering Research Laboratory Urbana, Illinois Attn: Mr. J. P. Nash
1	University of Michigan Engineering Research Institute Willow Run Airport Ypsilanti, Michigan Attn: Mr. J. B. Wright
1	University of Michigan Engineering Research Institute Willow Run Airport Ypsilanti, Michigan Attn: Mr. J. W. Carr III

No. of Copies

1 Methods Division
The Prudential Ins. Co. of America
Newark 1, New Jersey
Attn: Mr. J. E. Coachman

1 Director
National Security Agency
3801 Nebraska Ave., N.W.
Washington 16, D.C.

1 Electronic Data Processing Development
General Electric Company
One River Road
Schenectady 5, New York
Attn: Mr. J. W. Pontius

1 Air Force Cambridge Research Center
L. G. Hanscom Field
Bedford, Massachusetts
Attn: Mr. Bert F. Krauss, CRRI

1 Computer Control Co.,
10966 LeConte St.,
W. Los Angeles, California
Attn: Mr. Frank Stockmal

1 The Equitable Life Assurance Society
393 Seventh Ave.,
New York 1, N.Y.
Attn: Mr. Walter L. DeVries

1 Metropolitan Life Insurance Co.,
One Madison Ave.,
New York 10, N.Y.
Attn: Mr. Robt. D. Acker

1 Digital Computer Laboratory
Convair
San Diege 12, California
Attn: Mr. Ben Ferber

1 Computer Techniques Development
Investigations Section
AGT Development Department
Building 300
General Electric Co.,
Cincinnati 15, Ohio.
Attn: Mr. D. L. Shell

No. of Copies

1 New York University
College of Engineering
University Heights
New York 53, N.Y.
Attn: Mr. Emanuel Mehr

1 Burroughs Corporation
Research Center
Paoli, Penna
Attn: Mr. Jos. Deutsch

1 Programming and Operation
Research
Hughes Aircraft Co.,
Culver City, California
Attn: Mr. Leon Gainen

1 Systems Analysis Dept.,
Remington Rand
Engineering Research
Association Division
1902 West Minnehaha Ave.,
St. Paul W4, Minnesota
Attn: Mr. B. F. Cheydleur

1 American Standard Atomic
Energy Division
1682 Broadway
Redwood City, California
Attn: Mr. Edward J. Leshan

1 General Kinetics Inc.,
555 – 23rd St. South
Arlington 2, Virginia
Attn: Mr. A. E. Roberts

1 Carnegie Institute of Technology
Schenley Park
Pittsburgh 13, Penna
Attn: Prof. A. J. Perlis

1 Office of Naval Research
Logistics Branch
Washington, D. C.
Attn: Dr. M. E. Rose

1 Melpar, Inc.,
3000 Arlington Blvd.,
Falls Church, Virginia
Attn: Mr. L. Lerner

No. of Copies	
1	Avion Division – A.C.F. 800 No. Pitt St., Alexandria, Va. Attn: Mr. Aaron Halperin
1	United Merchants and Manufacturers 1407 Broadway New York 18, N.Y. Attn: Mr. S. Greshin
1	Computer Services Lockheed Aircraft Corp., Missile Systems Division Van Nuys, California Attn: Mr. W. W. Leutert
1	Remington Rand Eckert–Mauchly Division 1900 West Allegheny Ave., Philadelphia 29, Penna Attn: Mr. Charles Katz Box 5616, Room 701
1	Naval Research Laboratory Washington 25, D.C. Attn: Dr. Ben Lipson Code 6232
1	International Business Machines 590 Madison Ave., New York 22, N.Y. Attn: Mr. Harlan Herrick
1	Bell Telephone Laboratories Murray Hill, N.J. Attn: Dr. R. W. Hamming
1	Lockheed Aircraft Corp., Mathematical Analysis Dept., Burbank, California Attn: Mr. Harvey Bratman
1	General Electric Co., Vallecitor Atomic Laboratory 1763 South First St., San Jose, California Attn: Mr. R. H. Stark

No. of Copies	
1	Northrop Aircraft 2417 N 165 St., Gardena, Calif. Attn: Mr.Seymour Ginsburg
1	Computing Operation Bldg. 305 FPLD General Electric AGT Cincinnati 15, Ohio Attn: Mr.Donald J. Hahn
1	Boeing Airplane Co., Physical Research Staff Plant II Seattle, Washington Attn: Miss Mandalay Grems
1	Dartmouth College Dept. of Mathematics Hanover, New Hampshire Attn: Prof.John McCarthy
1	The Rand Corporation 1452 Fourth St., Santa Monica, California Attn: Mr.Harry A. Pappo
1	Ramo–Woolridge Corp., Computing Center 5730 Arbor Vitae St., Los Angeles 45, Calif. Attn: Dr.Walter F. Bauer
1	Stanford Research Institute Mathematics Division Menlo Park, California Attn: Dr. George Evans
1	Babcock and Wilcox Computational Services Sec: 1201 Kemper St., Lynchburg, Va. Attn: Mr. R.F.Reiss
1	Ramo–Wooldridge Corp., Computing Center 5730 Arbor Vitae St., Los Angeles 45, Calif. Attn: Mr.Robt.Perkins

No. of
Copies

1 Dept. of Interior
Washington 25, D. C.
Attn: Mr. Frank Reilly
Room 1449

1 Servomechanisms Inc.,
23821 Madison St.,
Torrance, California
Attn: Mr. Neil Block

1 Ramo-Wooldridge Corp.,
Computing Center
5730 Arbor Vitae St.,
Los Angeles 45, California
Attn: Dr. E. K. Blum

1 Ramo-Wooldridge Corp.,
Computing Center
5730 Arbor Vitae St.,
Los Angeles 45, Calif.
Attn: Dr. David Young

1 National Bureau of Standards
Washington, D. C.
Attn: Mr. Max Klein
Room 108A
West Building

1 Michelson Lab. Code 507
China Lake, California
Attn: Mr. H. Hauer

1 Columbia University
Mechanical Engineering Dept.,
Broadway and116 St.,
New York, N.Y.
Attn: Prof. J.H.Weiner

1 Pennsylvania State College
Mathematics Dept.,
State College, Penna
Attn: Prof. George Raney

1 International Business Machines
590 Madison Ave.,
New York 22, New York
Attn: Mr. J. C. McPherson

1 International Business Machines
1111 Connecticut Ave.,N.W.
Washington, D.C.
Attn: Mr. T. Horton