LIBRARY OF THE
UNIVERSITY OF ILLINOIS
AT URBANA-CHAMPAIGN

510.84

I l 6 r

no.588-589

cop.2

Digitized by the Internet Archive
in 2013

<http://archive.org/details/frogdesignsimula588bose>

Il 64
no. 588 UIUCDCS-R-73-588

C00-1469-0231

ap. 2
FROG: DESIGN AND SIMULATION STUDY OF A
MECHANISM WHICH LEARNS SELF PRESERVATIVE
REACTIONS TO ITS ENVIRONMENT

by

DEBASISH BOSE

August, 1973

THE LIBRARY OF THE

OCT 8 1973

UNIVERSITY OF ILLINOIS
AT URBANA-CHAMPAIGN

DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN · URBANA, ILLINOIS

FROG: DESIGN AND SIMULATION STUDY OF A
MECHANISM WHICH LEARNS SELF PRESERVATIVE
REACTIONS TO ITS ENVIRONMENT

by

DEBASISH BOSE

August, 1973

DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN
URBANA, ILLINOIS 61801

This work was supported in part by Contract No. US AEC AT(11-1)1469 and was submitted in partial fulfillment of the requirements for the degree of Master of Science in Electrical Engineering.

FROG: DESIGN AND SIMULATION STUDY OF A
MECHANISM WHICH LEARNS SELF PRESERVATIVE
REACTIONS TO ITS ENVIRONMENT

BY

DEBASISH BOSE

B. Tech., Indian Institute of Technology, 1970

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Electrical Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 1973

Urbana, Illinois

ACKNOWLEDGEMENT

The author wishes to thank his advisor, Professor S. R. Ray, for suggesting this thesis topic and for his constant guidance and friendship. He is also very grateful to Professor W. J. Poppelbaum for the opportunity to work in his group and for his advice and understanding.

The author acknowledges the immense help which he derived from the work of Mr. John Kreger, who designed the first version of the system reported in this thesis.

Thanks are due to Mrs. Evelyn Huxhold for the patience and skill with which she typed the thesis, and to Mark Goebel and his staff at the drafting section for the drawings.

TABLE OF CONTENTS

	Page
1. INTRODUCTION - - - - -	1
2. THE BASIC LOGIC ELEMENTS - - - - -	4
2.1 The Basic Elements - - - - -	4
2.2 The Memory Elements - - - - -	5
3. LOGICAL DESIGN OF THE SYSTEM - - - - -	18
3.1 Structure of the Memory - - - - -	18
3.2 Memory Integration - - - - -	24
3.3 Decision Unit - - - - -	32
4. THE SIMULATOR - - - - -	36
4.1 The Stimuli - - - - -	36
4.2 Decay - - - - -	37
4.3 The Simulator - - - - -	39
5. RESULTS AND CONCLUSION - - - - -	45
5.1 Summary of a Run of the Simulator - - - - -	45
5.2 Results - - - - -	46
5.3 Conclusion - - - - -	53
LIST OF REFERENCES - - - - -	55
APPENDICES - - - - -	56
A. Trial by Trial Summary of a Run of the Simulator -	56
B. Simulator Source Deck Listing - - - - -	76
C. Sample Outputs - - - - -	96

LIST OF FIGURES

Figure		Page
2.1	T*-Element (Schematic) - - - - -	7
2.2	A Typical Chain of T*-Elements - - - - -	9
2.3	Logic Element Realization of a T*-Element - - - - -	10
3.1	Output Arrangement of a T*-Element Chain - - - - -	20
3.2	A Level of a Memory with Four Chains of T*-Elements -	21
3.3	Block Diagram of a Memory Plane - - - - -	25
3.4	Block Diagram of the Adaptive Filter - - - - -	26
3.5	Logic Element Realization of a MIS - - - - -	28
3.6	Block Diagram of a MIU - - - - -	30
3.7	Decision Logic - - - - -	33

LIST OF TABLES

Table		Page
3.1	Criteria for Decision by a MIS - - - - -	29
3.2	The Decisions - - - - -	35
4.1	Characteristics of the Stimuli - - - - -	38
4.2	List of Procedures - - - - -	40
4.3	Format of the Array RDPROB - - - - -	42

1. INTRODUCTION

FROG is an artificial intelligence scheme which behaves adaptively with its environment. The scheme was proposed by Professor S. R. Ray. The initial design and computer simulation of FROG was done by John Kreger. The results were reported in an unpublished memorandum. The work described here is based on simulation studies, by the author, of an improved design of the mechanism.

FROG, as a mechanism, is an attempt to simulate the feeding habits of a small animal. FROG is cohabited by some bugs (both good and bad tasting) and predators (encounters with whom are considered painful but not fatal). The environment of FROG is separated into a number of regions characterized by different surface properties. The machine learns to discriminate between the different stimuli by means of their visual characteristics.

According to Lettvin, Maturana et al. in "What the Frog's Eye Tells the Frog's Brain,"¹ the retina of a frog contains four groups of fibers which break down every image into four main characteristics which are then transmitted to the brain. The characteristics are:

1. Sustained contrast: The presence of a sharp boundary, moving or still, with much or little contrast.
2. Net convexity: The presence of a curved boundary, darker than the background and moving on it, showing most activity if the object moves intermittently with respect to a background.
3. Moving edge: Tells whether or not there is a moving boundary in a larger area within the field.

4. Net dimming: Tells how much dimming occurs in the largest area and so related to the size of the object.

In our system each stimulus is assigned a value, in the range 0 to 1, for each of the above four characteristic.* These are the characteristics (denoted as $visual_1$, $visual_2$, $visual_3$ and $visual_4$ respectively) from which FROG infers the good or bad taste or the pain it receives from a predator.

Each bug has a value n ($0 \leq n \leq 1$) for its good taste and its bad taste is put as $1 - n$. A good tasting bug is one whose good taste is greater than its bad taste. A bad tasting bug is one whose bad taste is greater than its good taste. A bug is assumed to inflict no pain. By contrast the predators it encounters inflict pain to a varying degree.

It is to be noted that FROG utilizes the concept of "continuous logic." According to P. N. Marinos³ "In deductive logic, it is invariably assumed that every proposition is either true or false... In real life, however, there are concrete situations which do not fit readily into such

* In a later publication by Maturana et al.² the number of classes of ganglion cells was revised to five. Four of them were found to act on the visual image to perform complex analytical operations that remain invariant under changes of general illumination; the fifth class measured the light intensity. The operations of ganglion cells are briefly described by their names:

- Class 1. Sustained edge detection--with non-erasable holding.
- Class 2. Convex edge detection--with erasable holding.
- Class 3. Changing contrast detection.
- Class 4. Dimming detection.
- Class 5. Darkness detection.

In spite of this finding we have decided to work with four visual characteristics in our simulation of FROG. Our effort is to devise a cognitive system which recognizes unique combinations of contexts. By not extending the dimensions of the stimulus (or context) space to five we will only restrict the complexity of the problem without affecting the difficulty of the problem. The approach we take is perfectly general and can be extended to any dimensions.

a simple scheme and the conventional dichotomy of true-false is highly inappropriate. In other words, it is quite possible that certain situations might have values other than falsehood and truth." Thus, while the only values a situation might have in deductive logic are 0 and 1, continuous logic allows a given situation to have the values 0, 1, or some number between them. This is particularly suitable for our system.

For each stimulus presented, FROG receives external data of two types, visual and tactile. The four visual characteristics can assume values in the range 0 to 1. The tactile data simply specifies the region in which the stimulus appears. When FROG attempts to feed on a stimulus, it is assumed that there are external agents to evaluate the taste or the pain inflicted and make it available to FROG. Internally FROG generates hunger, fatigue and alarm signals.

FROG acts like an adaptive filter. The environment sensory data, interpreted through the adaptive filter, and the somatic signals, which measure the requirements of FROG for feed and rest, are combined to decide upon a course of action. The modes of activity are (1) feed; (2) rest; (3) flee.

2. THE BASIC LOGIC ELEMENTS

2.1 The Basic Elements

The basic logic elements which constitute the adaptive filter are defined below. All inputs, x_i , and outputs, y , have values in the continuum $[0,1]$ unless contraindicated.

<u>Element Name</u>	<u>Symbol</u>	<u>Nearest Analog</u>	<u>Definition</u>
Complement (C)	$x \rightarrow \textcircled{C} \rightarrow y$	NOT	$y = 1-x$
Minimum (m)	$\begin{matrix} x_1 \nearrow \\ x_k \nearrow \end{matrix} \textcircled{m} \rightarrow y$	AND	$y = \min_{i=1}^k (x_i)$
Maximum (M)	$\begin{matrix} x_1 \nearrow \\ x_k \nearrow \end{matrix} \textcircled{P} \rightarrow y$	OR	$y = \max_{i=1}^k (x_i)$
Product (P)	$\begin{matrix} x_1 \nearrow \\ x_2 \nearrow \end{matrix} \textcircled{P} \rightarrow y$	--	$y = (x_1) \cdot (x_2)$
Equivalence (E)	$\begin{matrix} x_1 \nearrow \\ x_2 \nearrow \end{matrix} \textcircled{E} \rightarrow y$	Equiv. (fuzzy)	$y = x_1$ if $ x_1 - x_2 \leq 0.05$ $y = 0$ otherwise
Selective	$\begin{matrix} x_1 \nearrow \\ x_2 \nearrow \end{matrix} \textcircled{S} \rightarrow y$	--	$y = x_1$, if $x_2 \geq 0.5$ $y = 0$ otherwise
Sense Indicator	$x_i \rightarrow \textcircled{SI} \rightarrow y$	--	$y = 1$ if $x_1 > 0$ $y = 0$ otherwise
*Memory (MEM)	$x_1 \rightarrow \textcircled{\text{MEM}} \rightarrow y$	--	$y = 0$ when first powered "on." If x_1 is the first occurrence of $x_1 > 0$, $y = x_1'$

<u>Element Name</u>	<u>Symbol</u>	<u>Nearest Analog</u>	<u>Definition</u>
*Transmittance (T) $x_1 \rightarrow \textcircled{T}_{(n)} \rightarrow y$	--	<u>Time</u>	<u>Stored Value</u> <u>Current Output (y)</u>
		initially (t=0)	0 0
		t = t ₁	last non- zero input min (1, $\frac{i}{n}$). (Stored valued) i = no. of inputs till t ₁ .

* See the following section for further explanation.

2.2 The Memory Elements

1. MEM-element

The MEM-element is initially "unset" (no value stored) and its output is zero. It "sets" to (stores) and begins to output its first nonzero input. The output of a "set" MEM-element is invariant and constant: At any point in time the output will be the stored value. Subsequent inputs will neither change the stored value nor affect the output.

2. T-element

The T-element is also initially "unset" (i.e. no number stored) and its output zero. The T-element sets to the first nonzero input and begins to output the product $\frac{1}{n} \cdot (\text{stored value})$ in a constant manner. The "stepping parameter" n is a nonzero positive integer. The next time the T-element encounters a nonzero input it sets to the new value of the input and a stepping process occurs by which the T-element's output is stepped to $(\frac{2}{n}) \cdot (\text{stored value})$. This process of stepping by adding $\frac{1}{n}$ to the fraction multiplier $\frac{i}{n}$ occurs each time a nonzero value is input

to the T-element until the term $\frac{i}{n} = 1$. At this point the T-element outputs the full value of its currently stored contents and can be considered to have reached "full potential." Any subsequent inputs to this element will not cause the stepping process to occur but the stored value will set to every new input. At any point in time after a T-element has reached full potential its output will be its stored value.

The particular value of its stepping parameter, n , is an inherent invariant characteristic of a T-element. In the case of the simulation FROG it is useful to have four variations of T-elements, having stepping parameters of 8, 6, 4 and 2. This results in FROG having T-elements that come up to full potential very quickly ($n = 2$), very slowly ($n = 8$), or at moderate speed ($n = 4$ or $n = 6$).

3. T*-element

The principal macroelement of FROG's memory is a T*-element. A T*-element, shown schematically in Figure 2.1, is constructed out of the nine continuous logic elements described in section 2.1 and some delay elements.

A T*-element has a maximum of four inputs (A_1, A_2, A_3 to A_4) for the visual characteristics of the stimuli and an input B for a non-visual characteristic. A T*-element stores a set of values of a specific combination of the four visual characteristics taken one, two, three or four at a time. It also stores all the different values of the nonvisual characteristic of one kind (e.g. all the different bad taste values) which may be associated with the particular values of the combination of visual characteristics stored. The T*-element forms a product of the minimum of the visual input values stored with the maximum of all the different nonvisual inputs stored. The output of the element depends on

Figure 2.1 A T^* -Element (Schematic)

how many occurrences of the given combination of visual inputs FROG has come accross and will be described later.

Another input to a T*-element is a write regulating signal C. The element has two outputs: Output D is the main output of the T*-element and output E is a modification of input C.

The T*-elements are connected in a row. Figure 2.2 shows three of them connected together. Each of these elements retain a specific combination of values of two visual inputs. The visual inputs and the input B are the same for all the elements. However, input C, the regulating input NBHOOD, is set at 1.0 for the T*-element #1 (the leftmost T*-element) while input C for T*-element #2 is output E (the modified NBHOOD) of #1. The pattern is repeated.

To understand the operation of a T*-element consider a T*-element which is the leftmost one in the 'chain' described above.

Assume FROG has just "eaten" a bad tasting bug. The bug has specific values for its visual characteristics and bad taste. Assume FROG wants to record the specific combination of the values of the second and the third visual inputs ($visual_2$ and $visual_3$). The value of the second visual input appear on the A_1 line and that of the third visual input appear on the A_2 line. The value of the bad taste is placed on the B input line of the chain. Now, refer to Figure 2.3 for the following discussion. Consider what happens in block 1. The S-element on the B input gates in the bad taste only if it is greater than or equal to .5. This value is input to the m-element of block 1. Another input to this m-element is the signal NBHOOD, which has been set initially at 1.0. The third input to this m-element (the complement of the output of the now "unset" MEM-element in block 3) is also 1.0. The output of this m-

Figure 2.2 A Typical Chain of T*-Elements

Figure 2.3 Logic Element Realization of a T^* -Element

element (which is greater than .5, since it is equal to the input bad taste) becomes an input to the M-element of block 1; the other input (the output of the still "unset" MEM-element of block 3) is 0.0. The output, then, of the M-element is the same as the input bad taste value. This output, which is $\geq .5$, gates in the visual inputs through the S-elements in block 2 and the taste input through the S-elements in block 3. These stimuli then "set" the MEM-elements in their respective blocks. The MEM-elements, after being "set," begin to output the values of visual characteristics and taste that are still being input to them. These values reach the E-elements, and since each of the elements has two equal inputs, they now output the values of visual characteristics (in block 2) and the taste value (in block 3) that were input to the T*-element. The m-element in block 2 now select the minimum value of the visual inputs. This value is nonzero, so the lower SI-element output in block 4 will be 1.0, which is input to a m-element. Since the output of the E-element in block 3 is nonzero, the upper SI-element in block 4 will output 1.0. So the following C-element will output 0.0 which is the other input to the m-element in the same block. The m-element will now output 0.0. The taste input which is gated in at the S-element in block 3 is delayed till the output of the m-element is reliable. The delayed taste input is now applied as x_1 input to the S-element in block 5. Since the x_2 input is 0.0 the taste input is not gated into the MEM-element of block 5. The block 5 is similar in structure to block 3 and its function will be apparent later.

The MEM-element in block 5 is still unset. The inputs to the E-element of block 5 are both 0.0 so it outputs 0.0. The output of the E-element in block 3 is the taste value stored in block 3 and that of the

E-element in block 5 is 0.0. So the output of the lower M-element in block 6 is $\geq .5$. The output of the upper element is the maximum of the stored values in the MEM-elements in block 3 and 5; right now that would be the block 3 output since the other is unset. This value is now gated into block 7 through the S-element by the output of the lower M-element which is ≥ 0.5 .

One input to the block 7 is the minimum of all the visual input values stored in the MEM values of block 2. The other input is the maximum of the stored values in the MEM-elements of blocks 3 and 5. Each of these inputs go to a SI-element. Since the inputs are nonzero the SI-elements begin to output 1.0. The inputs also go to a P-element which forms their product. The product is the x_1 input to the upper S-element in block 7. The x_2 input to this S-element is the minimum of the outputs of the above two SI-elements. Since these outputs are each 1.0, the x_2 input to the S-element is 1.0 and the output of the P-element is gated through to the T-element.

At this point, the E output of the T*-element, the modified version of the NBHOOD signal becomes defined. The E output of the T*-element is 1.0 only if in block 8 the delayed C input to the element is 1.0 and there is no output from the upper S-element of block 7. The 1.0 on the C input (NBHOOD signal) is delayed for long enough span, μ , to allow a signal to be output from the upper S-element of block 7 (i.e. to allow it to become reliable). Until this happens, since the upper input to the m-element in block 8 is zero, the E output is zero. Now, when the upper S-element of block 7 has an output value > 0 , the side input to the m-element in block 8 becomes 0.0, so that even after the delayed NBHOOD signal reaches the top input of the m-element, the E output continues to remain zero.

To return to block 7, the output from the S-element has caused the T-element to "set," i.e. store this value and begin outputting $\frac{1}{n} \times$ this value. The T-element output becomes the x_1 input to the lower S-element in block 7. The x_2 input to this S-element is the minimum function of (1) the SI function of the visual input to block 7 which is 1.0 and (2) the complement of the SI function of the nonvisual input to block 7 which is 0.0. Thus the x_2 input to the lower S-element in block 7 is 0.0 with the result that the D output of the T*-element is 0.0, which is what it should be, for this is a "write" into the memory cell, not a "read" out of it.

This is a complete "write" cycle for T*-element #1, but now let us consider what happens to the still "unset" T*-element #2. Its A_1 , A_2 and B inputs are the same as that for the first cell, but the C input, NBHØØD (before and after modification by T*-element #1) is now 0.0. The effect of this will be to cause zero to be output by the m-element in block 1 and since the MEM-element in block 3 is unset, the output of the M-element will be 0.0. This zero is the x_2 input for each of the S-elements in blocks 1 and 2 with the result that the visual and nonvisual inputs to the T*-element will not be gated into the T*-element. However, let us look at the output E of this second T*-element, for it will be input for the T*-element #3. The complement of the SI function of the upper S-element in block 7 will be 1.0. This value will be one input to the m-element in block 8; the other will be the value coming out of the delay unit. However, this value, both before and after delay time, μ , has elapsed, will be 0.0. Therefore, after one memory cell in the chain has been written into, no "unset" T*-element to the right of that memory cell may be written into.

A T*-element can be written into a second time if the visual inputs match the values stored in the MEM-elements of block 2. The taste value need not necessarily match the value stored in block 3. A "set" T*-element will always gate through any visual input and any B input greater than .5 to blocks 2 and 3. This will be because the output of the MEM-element in block 3 will cause M-element in block 1 to output some value ≥ 0.5 , which, in turn, will cause the S-element gates to be constantly open in blocks 2 and 3. Since the visual inputs match the stored values in block 1, the E-elements will output nonzero values, and so the visual input to block 7 will be nonzero. On the otherside, the B input will be compared with the stored value in MEM-element of block 3. If they match, the E-element will output a nonzero value. The rest of the action will be exactly similar to what happened when the element was set for the first time. A nonzero product will be transmitted to the T-element which will increment the stepping parameter and set it to the new product which will essentially have the same value as the old product stored.

If the taste input does not match the value stored then the E-element output of block 3 will be zero. This causes the first input to the m-element of block 4 to be 1.0. The second input is also 1.0 since the visual inputs matched. The m-element output will thus be 1.0 and will gate in the delayed taste input to block 5 which will set the MEM-element in that block to the new B input value. Block 6 is arranged to output the maximum of the two stored values. The two inputs to block 7 will now be defined and the T-element will be set to the new product (which may not necessarily be the same as the old one) and increment the stepping parameter.

The purpose of block 5 is now apparent. It is to store a second value of the nonvisual input (bad taste in this case) which may be associated with a given combination of visual inputs. Block pairs like 4 and 5 may be repeated to provide for more different values of the nonvisual input. In our simulation we did not need to provide for more than two.

The T*-element will be written a third time, a fourth time and so on at the subsequent appearances of the same combination of visual inputs and a bad taste value (≥ 0.5) which may or may not match the stored values. Note that the bad taste value transmitted by block 6 to P-element in block 7 is always the maximum of all the different bad taste values encountered till then for the given visual inputs. Also note that the second input to the m-element in block 4 is always 0.0 except when all the visual inputs match the stored values in block 2. This prevents the input taste value (the B input) to be gated into block 5 unless the visual inputs match.

Now consider what will happen if a different set of A_1, A_2 input values are input to the chain. They will reach the T*-element #1, the one that is "set." As has been said previously, a "set" T*-element always gates the A and B inputs to block 2 and 3. Since the MEM-elements are already set, the new inputs will not affect them, but will be compared with the MEM-element outputs at the E-elements. Since we are assuming that at least one visual input is not within allowed tolerances for the E-elements, one or more E-elements in block 2 will output zero. The visual output from block 2 will then be zero which inhibits the taste signal to go to block 5. Block 3 could output some nonzero value if the B input matched the value stored in block 3. So the block 6 output may

or may not be zero. The block 2 output being zero both the S-elements in block 7 will be inhibited. So the D output of the leftmost T*-element will be zero. Also the E-output of this T*-element will be 1.0 after time μ . So input C of the T*-element #2 is 1.0 and this will become "set" in exactly the same manner as the leftmost T*-element.

The NBHØØD signal functions solely to tell the chain that a T*-element has been written into for a given combination of visual inputs and, hence, once this occurs, not to write into it for a different set of visual inputs. Note that for a given combination of visual inputs the T*-elements can hold more than one B-input values. In the present simulation provisions are there for two such values. The NBHØØD signal makes sure that the writing of a new combination proceeds sequentially down the chain from left to right.

All of the above discussion pertains only to "writing" into the T*-element memory cells. Now the question arises how is a T*-element caused to output the current output state of its T-element? The process occurs contextually, when FROG has just seen a stimulus and its memory cells must be consulted to see if FROG has had any experiences with this stimulus in its past and if so, whether they were good or bad experiences. Obviously, then, no nonvisual input is presented to the memory in this case, for feeding has not yet taken place; thus, the nonvisual input to a T*-element during a read cycle is zero.

The "reading" process occurs in the following manner. The visual stimulus is presented to all T*-element in the chain at once (reading, it will be seen, does not have to occur sequentially, as does writing). The visual inputs will be gated into block 2 of each of the 'set' T*-element in the chain. If the current visual input values are

close enough (to within ± 0.05) to that which are stored in the block 2 MEM-elements, then the E-elements in block 2 will output nonzero numbers; if not, no read will occur. Now recall that the x_2 input to the lower S-element in block 7 is the minimum of two inputs (1) the SI function of the visual output from block 2, which is 1.0 and (2) the complement of the nonvisual input from block 6, which (since the B input to the T*-element is zero) is 1.0, with the result that this second input to the m-element is also 1.0. Thus, since the m-element is outputting 1.0, the current output status of the T-element is caused to be output; no stepping of the T-element will occur for there is no input to it in the read cycle.

A write into an "unset" element cannot possibly occur by accident during a read cycle, for if a T*-element is "unset" and the nonvisual input is zero, then the visual stimulus input will not get past the S-elements in block 2. Also write into a block 5 of a 'set' T*-element is not possible as the B input is zero.

3. LOGICAL DESIGN OF THE SYSTEM

As has been mentioned earlier FROG, as a cognitive* system, is like an adaptive filter, such that when the input sensory data and the somatic signals are presented a course of action for FROG is decided. It consists of a 'memory' where the experiences with different stimuli are stored, i.e. which develops a 'model' of the world from which the stimuli are drawn. The memory is followed by a 'memory integration unit' (MIU), which is to handle the various memory outputs properly on presentation of a stimulus and to make them available to a 'decision unit' (DU), which decides on the mode of activity.

3.1 Structure of the Memory

Structurally the memory of FROG is divided into several parts (memory planes) equal in number to the number of regions in the environment. Experiences in the various regions are stored in separate memory planes. The memory plane for each region is again divided into three parts. The first is a good taste memory (GTM) which associates the distinguishing visual characteristics of good tasting bugs with their tastes. The second is the bad taste memory (BTM) which associates the visual characteristics of bad tasting bugs with their tastes. The third is a predator memory (PM) which associates the visual characteristics of a predators with the amount of pain they inflict.

The association of a nonvisual input (good taste, GT, bad taste, BT, or pain inflicted, PAIN) with visual characteristics or its

* In the sense of psychologists; meaning, a response judgment requiring an extended-time "consideration" of the stimulus data as opposed to a recognition task in which the response is relatively rapid and automatic.

combinations is done in a T*-element as described in Chapter 2. The question now arises as to how these T*-elements be arranged within a memory to simulate an efficient learning scheme. The strategy adapted in FROG is described below.

First of all, we see if any of the four visual characteristics might be a basis for discrimination between various stimuli. For this, we take three sets of four chains of T*-elements and place a set in each memory. These T*-element have one visual input each. The D outputs of all the T*-elements in each chain are connected to one M-element (see Figure 3.1). Thus the T*-elements in chain 1 in BTM would associate first visual characteristic of a particular stimulus with its bad taste. The second, the third and the fourth chains in each memory would perform similar functions with the second, the third and the fourth visual characteristic respectively. The outputs of each of these chains in a particular memory are then input to one M-element as shown in Figure 3.2. For the purpose of simulation study a chain is made up of three T*-elements. So 3 stimuli of each type can be remembered.

Let us examine the process of read out from and writing in to the memory. If a bug has just been eaten or a predator encountered, the four visual characteristics would be sent to each memory; visual characteristic 1 to chain 1, etc. The nonvisual inputs good taste (GT), bad taste (BT) and pain (PAIN) are also sent to the GTM, the BTM and the PM respectively. For a bad tasting bug, writing will take place in BTM only (since $BT > 0.5$, $GT < 0.5$, and $PAIN = 0.0$). Whereas, for a predator, writing takes place in PM only if $PAIN \geq 0.5$ ($GT = BT = 0$ and $PAIN \geq 0.5$).

When FROG has just seen a stimulus, a read is performed in each memory by sending the visual characteristics to their respective chains

Figure 3.1 Output Arrangement of a T*-Element Chain

Figure 3.2 A Level, of a Memory, with Four Chains of T*-Elements

along with zero nonvisual inputs to all memories. Any T*-element in which the visual input is numerically within a neighborhood of 0.05 of the value of visual characteristic stored in the MEM-element in block 2 will output the status of its T-element. Then the maximum function of all the T*-outputs in the memory will be output from each memory. That is, GTM will output the numerically greatest impulse possible for eating the bug, the BTM the numerically greatest possible impulse for avoidance of the bug, and PM the numerically greatest possible alarm. These outputs, one from each memory, are compared in a manner discussed later.

A problem arises when, for example, a good tasting bug and a predator have approximately equal values for their numerically largest visual characteristic and the good taste value of one is approximately equal to the pain inflicted by the other. If any of these two stimuli is presented the GTM and the PM outputs will be approximately equal and FROG will not be able to make a choice based on its memory outputs.

This problem arises because the four visual characteristics are too general to form a basis for discrimination. The natural solution is to take combinations of these four basic characteristics and use them as inputs to chains. The four visual characteristics taken two at a time forms the following six combinations:

- (1) $\text{visual}_1, \text{visual}_2$
- (2) $\text{visual}_1, \text{visual}_3$
- (3) $\text{visual}_1, \text{visual}_4$
- (4) $\text{visual}_2, \text{visual}_3$
- (5) $\text{visual}_2, \text{visual}_4$
- (6) $\text{visual}_3, \text{visual}_4$

Now we form six more chains of T*-elements in each of GTM, BTM and PM.

Each chain has two visual inputs and one nonvisual input, depending on the memory involved.

Should the above combinations prove insufficient as a basis for discrimination between certain stimuli, combinations of the visual characteristics taken three at a time are utilized. These combinations are:

- (1) $\text{visual}_1, \text{visual}_2, \text{visual}_3,$
- (2) $\text{visual}_1, \text{visual}_2, \text{visual}_4,$
- (3) $\text{visual}_1, \text{visual}_3, \text{visual}_4,$
- (4) $\text{visual}_2, \text{visual}_3, \text{visual}_4.$

Each of these combinations is input to a different chain of T*-elements in each memory, accompanied by a suitable nonvisual input.

The last level of discrimination is that using the four visual characteristics taken all at a time.

Thus each of the memories (GTM, BTM, PM) has 15 chains of T*-elements which may be divided into four groups referred to as "levels." Level 1 consists of four chains of T*-elements that have as their visual inputs the four basic visual characteristics of the stimulus. Level 2 consists of the six chains of T*-elements that have as their inputs the combinations of the four basic characteristics taken two at a time. Level 3 consists of the four chains of T*-elements that have as their visual inputs the basic characteristics taken three at a time. Level 4 consists of the chain which has as its visual inputs the combination of the four basic visual characteristics taken four at a time. Now we can justify why it is necessary to have upto four A inputs to a T*-element. Level 1 chains need T*-elements with one visual input (A_1). Level 2 chains need T*-element with two visual inputs and so on. The GTM, the

BTM and the PM all have the same structure with the only physical difference being the nonvisual input to the memory. The level output of any level is the maximum function of all the T*-element outputs of that level.

The different kinds of T*-elements not only differ in the number of visual inputs that it, can handle but also in the value of n . It is at this point that this variation becomes meaningful. It is apparent that, in going from level 4 to level 1, there is an increasing generality in the visual inputs to the chain, level 1 being the most general and least specific and level 4 being the least general and most specific. Thus, the most general level should learn the slowest in order to avoid making mistakes in discrimination of a very broad nature; but as the amount of generality decreases, faster learning can be permitted because a mistake in the lower levels would not be as disastrous as it would be in level 1.

In level 1, the T*-element contain T-elements which have $n = 8$. Thus it is necessary for the visual input to repeat (within the allowed neighborhood of ± 0.05) seven times after the initial setting of the T*-element in order to have the T-element come up to full output potential (full learning). The T*-elements in level 2 contain T-elements for which $n = 6$, so five repetitions are necessary for full learning. In level 3 T*-elements $n = 4$, so three repetitions are necessary for full learning. In level 4, the most specific level, the T*-elements contain T-elements for which $n = 2$; thus only 1 repetition of the stored visual inputs is necessary for full learning.

3.2 Memory Integration

When FROG has just seen a stimulus the visual characteristics are sent to all the memories without any taste or pain (i.e. nonvisual)

Figure 3.3 Block Diagram of a Memory Plane

Figure 3.4 Block Diagram of the Adaptive Filter

information. Let the stimulus be in region 1. There is a signal REGION which specifies to FROG the region in which stimulus has been presented. The read out should occur in the memory plane 1 which stores information regarding region 1. The GTM, the BTM and the PM in plane 1 now outputs twelve values; one per level, with four levels in each of the three memories. These outputs are compared and processed in a manner described below by a 'memory integration unit' (MIU) and made available to the 'decision unit' (DU). There is a MIU in each plane. The MIU in the plane compares three signals at a time, starting with the three level 1 outputs. This comparison is done in an element called the 'memory integration subunit' (MIS). The outputs from a memory plane are ALARMOUT, MIUOUT and NODEC⁴. (See Figures 3.3 and 3.4). If FROG senses a predator it will have an ALARMOUT. If FROG sees a bug there will be an output, MIUOUT, proportional to its desire to eat. But if no decision is reached the NODEC⁴ output of the plane is 1.0. When this happens FROG gates in the visual inputs to all the other planes for a readout from them. This enables FROG to come to a decision based on its knowledge of this stimulus in all the regions. The maximum of the MIUOUT of all the planes is the final MIU output (MIUOUTF). The ALARMOUT outputs of all the planes are also input to a M-element, the output of which is the final alarm output (ALARMF). If no decision is reached in all the planes NODEC⁴ from all the planes will be 1.0 so NODECF, the final indication of whether a decision has been made or not, would be 1.0.

The logic element realization of the MIS is shown in Figure 3.5. There are three inputs to the MIS; the three level outputs from the corresponding levels of GTM, BTM and PM in the plane (see Figure 3.3

and 3.6). There are three outputs MISOUT, ALARMIS, NODEC. If comparison of the input values result in a decision (the decision criteria are explained below) then either MISOUT or ALARMIS has a nonzero output; if no decision can be made, these outputs are zero. The third output, NODEC, tells the MIU whether a decision has been made or not; if a decision has been made, this output is zero, if not, it is one.

The criteria for decision by the MIS are detailed in Table 3.1. An input is called the largest if it is larger in value than that of all the other inputs by at least 0.05.

Condition	MIS OUTPUTS		
	MISOUT	ALARMIS	NODEC
GRMOUT is the largest and ≥ 0.5	GTMOUT	0	0
BTMOUT is the largest and ≥ 0.5	BTMOUT	0	0
PMOUT is the largest and ≥ 0.5	0	PMOUT	0
None of the input is largest by definition and/or all of them are < 0.5	0	0	1

Table 3.1 Criteria for Decision by a MIS

Thus it is necessary for the numerically largest input to be greater than 0.5 and greater than the next input value by at least 0.05, if a decision is to be made. If these conditions are not met, then the outputs MISOUT and ALARMIS are zero and the output NODEC is one.

A MIU consists of four MIS modules with some additional logic to connect the modules (see Figure 3.6). The first MIS compares the level 1 outputs of the three memories, if no decision is reached, based on the above criteria, then the outputs MISOUT1 and ALARMOUT1 are zero and NODEC1 is one. This allows the level 2 outputs to be gated into a second MIS. If comparison in this MIS yields no decision the outputs MISOUT2 and ALARMOUT2 are zero and NODEC2 is one. The process continues

Figure 3.6 Block Diagram of a MIU

for each of the MIS modules in a similar manner. If no decision is reached after comparison of the level 4 outputs the output NODEC₄ is one. Under this circumstances, the visual characteristics are gated into all other planes for a readout from them, and if no decisions are made in these memory planes as well, the individual NODEC₄ outputs will all be 1.0. Only when this happens that the NODECF output becomes 1.0, as mentioned previously.

The above discussion assumes that no decision can be made in any of the MIS modules. If, however, a decision is made at the nth MIS module of the plane in which the visual inputs are first gated in then NODEC_n will be zero. This, by means of the m-elements between each module, will cause zero to be the x_2 input for all of the S-elements, the yet unused modules, preventing (1) any further comparison and (2) the gating of the visual inputs to the other memory planes. Thus only one of the ALARMIS or one of the MISOUT lines of the plane will have a nonzero output. So either the ALARMOUT line or the MIUOUT line of the plane will have a nonzero output and NODEC₄ of the plane will be zero. The ALARMOUT, MIUOUT and NODEC₄ of all the other planes will be zero. Therefore, the final nonzero output on MIUOUTF (or ALARMOUTF) equals the output of the MIUOUT (or ALARMOUT) line which is activated. NODECF remains zero.

If, on the other hand, the decision is not reached in the plane in which the visual inputs have been gated in first, but is reached in one or more of the other planes, when the visual inputs are gated into these, then the NODEC₄ output from the planes in which decision has been reached will be zero. Either the ALARMOUT or the MIOUT of the plane(s) will carry a nonzero value. The NODECF will, therefore, be zero. The

final output MIUOUTF will be the maximum of all the individual MIUOUTs. Same is true for ALARMF (see Figure 3.4). Notice that the ALARMOUT and the MIUOUT of the same plane cannot have nonzero output at the same time. But in this circumstance, since more than one plane can have outputs, it is possible to have nonzero values on both ALARMF and MIUOUTF.

The comparison within a MIU is so sequenced that it proceeds from the most general level of learning to the most specific (level 1 to 4). Attempting to discriminate between the various classes of stimuli on the basis of level 1 output is a linear classification scheme while attempting the discrimination on the basis second, third or fourth level memory outputs is a nonlinear classification scheme.

3.3 Decision Unit

The decision unit is the subunit of FROG that makes all the decisions. Inputs to it are (1) ALARMF (2) MIUOUTF (3) HUNGER (4) the four basic visual inputs (5) NODECF (6) REGION (the signal which tells FROG in which region the stimulus has appeared) and (7) LOCATION (the signal which tells FROG its present location, i.e. the region in which it presently is). The outputs from it are the various decisions.

The decision logic is as shown in Figure 3.7. From the inputs we derive a signal called FEED. Whenever the MIU comparisons yield an output, NODECF is zero. So the second input to the left hand M-element is zero. The output of this element is therefore equal to MIUOUTF. But when the MIU comparisons are indecisive NODECF is 1.0 and MIUOUTF and ALARMF is zero. Under this circumstance the output of the M-element equals 1.0 if hunger is ≥ 0.5 or zero when hunger is < 0.5 .

Figure 3.7 Decision Logic

The output of the M-element is one of the input to the top m-element. The other inputs to it are HUNGER and the innate visual stimulus (INNATESTIM), which is simply the maximum function of the four visual characteristics. It serves in the early trials to tell FROG whether or not what it sees is a bug before its memories have learned enough about the bugs.

The output of this m-element goes to a M-element. The other input to this M-element is nonzero only when hunger reaches the point of starvation (hunger = 1.0); when FROG is starving this input assumes a value equal to the innate visual stimulus. Thus, when FROG is at the point of starvation, its feeding impulse would be fairly large, i.e. FROG will eat any bug regardless of what its memories say about its good or bad taste. When FROG is not starving, the output of the m-element determines the value of FEED.

FROG can assume one of five decisions as detailed in Table 3.2. The tabulation is fairly self-explanatory.

Decision No.	Decision	Condition	Comments
1	FROG has seen a predator	$\text{ALARMF} > \text{FEED}$ $\text{REGION} \neq \text{LOCATION}$	FROG has seen a predator. It is ALARMED but since it is in a different region FROG decides to ignore the stimulus.
2	FROG has seen a predator. The predator is too close to itself so it decides to flee.	$\text{ALARMF} > \text{FEED}$ $\text{REGION} = \text{LOCATION}$	The stimulus and FROG is in the same region. FROG has sensed a predator so it decides to flee to its safe resting place.
3	FROG has just seen a bug, but decides not to feed.	$\text{ALARMF} < \text{FEED}$ $\text{FEED} < \overline{0.5}$	FROG is not too eager to eat.
4	FROG decides to feed.	$\text{ALARMF} < \text{FEED}$ $\text{FEED} > \overline{0.5}$ $\text{FEED} > \text{FATIGUE}$	
5	FROG is too tired to move.	$\text{ALARMF} < \text{FEED}$ $\text{FEED} > \overline{0.5}$ $\text{FEED} < \text{FATIGUE}$	

Table 3.2 The Decisions

4. THE SIMULATOR

In this chapter various aspects of the simulation are discussed. First the various stimuli that form FROG's world are described. Following which a technique of simulating loss of memory is presented. A description of the simulator comes next.

4.1 The Stimuli

The simulation was done with three different classes of stimuli; good tasting bugs, bad tasting bugs and predators. Two each of each kind were chosen. These were: fly, grasshopper, honeybee, ladybug, beetle and scorpion. Table 4.1 lists the stimuli and their various visual and non-visual characteristics. The bugs that share a particular combination with the bug being listed (i.e. for the same set of visual characteristics, have values which are within 0.05 of those of the one listed) are indicated in parentheses after the value.

The table employs the abbreviations detailed below for the various combinations of visual inputs that are used as level inputs.

The assignment of the various visual and nonvisual characteristics to the stimuli was based on how the author felt a frog would react to them.

Level 1 Visual Inputs:

(1,1) = visual₁ (sustained contrast)

(1,2) = visual₂ (net convexity)

(1,3) = visual₃ (moving edge)

(1,4) = visual₄ (net dimming)

Level 2 Visual Inputs:

Input A ₁	Input A ₂
(2,1) = visual ₁	visual ₂
(2,2) = visual ₁	visual ₃
(2,3) = visual ₁	visual ₄
(2,4) = visual ₂	visual ₂
(2,5) = visual ₂	visual ₃
(2,6) = visual ₃	visual ₄

Level 3 Visual Inputs:

Input A ₁	Input A ₂	Input A ₃
(3,1) = visual ₁	visual ₂	visual ₃
(3,2) = visual ₁	visual ₂	visual ₄
(3,3) = visual ₁	visual ₃	visual ₄
(3,4) = visual ₂	visual ₃	visual ₄

Level 4 Visual Inputs:

Input A ₁	Input A ₂	Input A ₃	Input A ₄
(4,1) = visual ₁	visual ₂	visual ₃	visual ₄

4.2 Decay

Decay or loss of "stored" information from nonuse is a natural characteristics of learning processes. Effort has been made to simulate this condition by incorporating a decaying routine not using continuous logic elements. It functions in the following way: Everytime a "set" T*-element outputs its stored value a number, called the decay number, is subtracted from the stored contents. Every "set" T*-element has a decay number. The decay number is 0.0 initially. During every read cycle, each "set" T*-element which does not output its current output status,

	Fly	Grasshopper	Honeybee	Ladybug	Beetle	Scorpion
Good Taste	0.9	0.8	0.2	0.1	0.0	0.0
Bad Taste	0.1	0.2	0.8	0.9	0.0	0.0
Pain	0.0	0.0	0.0	0.0	0.8	0.9
(1,1)	0.8	0.6	0.9	0.7 (B,S)	0.7 (L,S)	0.7 (L,B)
(1,2)	0.7 (G,S,H)	0.7 (F,H,S)	0.7 (F,G,S)	0.9	0.6	0.7 (F,G,H)
(1,3)	0.7 (G,L,S)	0.7 (F,L,S)	0.8	0.7 (F,G,S)	0.9	0.7 (F,G,L)
(1,4)	0.6 (H)	0.8	0.8 (F)	0.7 (B)	0.7 (L)	0.9
(2,1)	0.7	0.6	0.7	0.7	0.6	0.7
(2,2)	0.7	0.6	0.8	0.7 (S)	0.7	0.7 (L)
(2,3)	0.6	0.6	0.6	0.7 (B)	0.7 (L)	0.7
(2,4)	0.7 (G,S)	0.7 (F,S)	0.7	0.7	0.6	0.7 (F,G)
(2,5)	0.6 (H)	0.7	0.6 (F)	0.7	0.6	0.7
(2,6)	0.6	0.7	0.6	0.7	0.7	0.7
(3,1)	0.7	0.6	0.7	0.7	0.6	0.7
(3,2)	0.6	0.6	0.6	0.7	0.6	0.7
(3,3)	0.6	0.6	0.6	0.7	0.7	0.7
(3,4)	0.6	0.7	0.6	0.7	0.6	0.7
(4,1)	0.6	0.6	0.6	0.7	0.6	0.7

Abbreviations for names of stimuli: F = Fly
 G = Grasshopper
 H = Honeybee
 L = Ladybug
 B = Beetle
 S = Scorpion

Table 4.1 Characteristics of the Stimuli

(i.e. its information is not used) would have its decay number incremented by 0.0005. On the other hand, if it were caused to output during that cycle, then the amount 0.0005 would be subtracted from the number. Thus if a T*-element after being "set" did not output for four cycles in a row, its output during the next read cycle, would have 0.0020, the decay number in this case, subtracted from the output value. Again, if the decay number for a T*-element at some point in time were 0.0025 and this T*-element is made to output its stored contents during the next two successive read cycles and is not caused to output anything for the following cycle, the decay number after these last three read cycles would be 0.0020. This value (0.0020) would then be subtracted from the output of this T*-element if it outputs during the next cycle. Thus provisions are made for both a "loss" of memory from nonuse and "refreshing" of memory when information is repeated.

A negative decay number is possible if a T*-element outputs for a sufficient number of time, however, this does not occur very frequently in an environment in which there are more than a few stimuli which are presented randomly.

4.3 The Simulator

The simulator is a PL/1 program. It consists of a main program FROG. Subroutines (or procedures) are written to simulate a general memory (so that under proper specification it simulates a GTM or a BTM or a PM), a MIU and a MIS. Function subroutines are written to simulate the logic elements S, E, P, T etc. Table 4.2 lists the procedures and additional information about each. The source deck listing in Appendix B provides further explanations about those.

<u>Procedure Name</u>	<u>Purpose</u>	<u>How Invoked</u>
FROG	calling program	main program
MIU	simulates MIU	by CALL statement
MEMORY	simulates a memory (GTM, BTM or PM)	"
TSTAR	simulates T*-element	"
T	simulates T-element	by function reference
MEM	simulates MEM-element	"
S	simulates S-element	"
E	simulates E-element	"
P	simulates P-element	"
SI	simulates SI-element	"
C	simulates C-element	"
MAX (*)	simulates M-element	"
MIN (*)	simulates m-element	"
MIS	simulates MIS	by CALL statement
DECAY	simulates decaying process of T*-elements	"
COMPUTE	simulates stepping process of T-elements	"
RANDOM	generates a random number on [0,1] for the random selection scheme	by function reference

(*) Both MAX and MIN are built-in generic routines and, here, are not actual procedures within FROG. They are included in the list to make the list of program components complete.

Table 4.2 List of Procedures

The main procedure, FROG, includes the program to generate the stimuli in the various regions in a random sequence, and the simulation of the 'decision unit' (DU).

The number of stimuli in FROG's environment is specified by initializing the variable HOWMANY. The names of the various stimuli, their visual characteristics, their good taste values, and the pain inflicted by each one of them is contained in the arrays NAMES, CHARS, TASTES and PAIN. These are dimensioned at execution time by HOWMANY. The array CHARS has as many rows as the number of stimuli and four columns to accomodate the values of $visual_1$ through $visual_4$ of the various stimuli. Thus CHARS (i,j) contain the jth visual characteristic of stimulus number i. The four attributes of the same stimulus occupy an identical place in the list, i.e. NAMES (2), CHARS (2,j) for $j = 1, \dots, 4$, TASTES (2) and PAIN (2) all refer to the same stimulus.

At the start of the program the array RDPROB is initialized with values which represent the desired frequencies in the random selection scheme of appearance of each of the stimuli in the different regions. The number of rows n equals the number of different stimuli. The number of columns m equals the number of regions in FROG's environment. The arrays is initially dimensioned by the variables HOWMANY and REGIONNO which specify n and m respectively. The format of the array is as shown in Table 4.3.

s \ r	1	2		m	Σ
1	P_{11}	P_{12}		P_{1m}	P_{1x}
2	P_{21}				P_{21}
.	.				
.	.				
.	.				
.	.				
n	P_{n1}			P_{nm}	P_{nx}
Σ	P_{x1}	P_{x2}		P_{xm}	P

Table 4.3 Format of the Array RDPROB

The quantity p_{ij} represents the probability of occurrence of stimulus i in region j . The partial sum $P_{ix} = \sum_{j=1}^m p_{ij}$ represents the total probability of occurrence of stimulus i over all the regions. The partial sum $P_{xj} = \sum_{i=1}^n p_{ij}$ represents the combined probability of occurrence of the various stimuli in the region j . Obviously

$$P = \sum_{i=1}^n \sum_{j=1}^m P_{ij} = \sum_{i=1}^n P_{ix} = \sum_{j=1}^m P_{xj} = 1.$$

After the first executable pair of statements which read in the array RDPROB and prints it, the statement labelled BEGIN invokes the subroutine RANDOM and assigns the generated random number to RANNUM. The following statement "IF COUNTER = n THEN GØ TØ TERM" terminates the program after $n-1$ trials. The next group of statements allow memory dumps at specified trials.

The group of statements following the memory dump indicator uses the random number RANNUM to select which bug is to be presented to FROG. Let $RANNUM = r$. This block of statements obtains the smallest values of i and j which satisfies the inequality

$$r \leq \sum_{k=1}^{i-1} P_{kx} + \sum_{\ell=1}^j p_{i\ell}$$

These values of i and j specifies the stimulus to be chosen for the trial and the region in which it is to appear, and are assigned to the variables BUGCHOICE and REGCHOICE respectively. The variables BUGNAME, GOODTST and PAIN are assigned BUGCHOICE_{th} members of the arrays NAMES, TASTES and PAININPT. The array VISUALCHAR is assigned the BUGCHOICE_{th} row of the array CHARS. The stimulus for the trial is now determined and its BADTST and INNATESTIM are evaluated. A message concerning the stimulus is now printed.

The program now initializes all the memory outputs from each plane to zero and also the overall memory outputs. Read occurs from the memory plane corresponding to the region in which the stimulus has been presented by calling MIU which in turn calls the three memories (GTM, BTM nad PM) for that plane. Only if the NODEC₄ output of this plane is 1, that the other planes are read.

From the memory outputs the final outputs ALARMF, NODECF, MIUOUTF are determined. The value of the quantity FEED is also calculated.

The statement following that simulates the "decision unit" (DU). This decides the course of action taken by FROG for the given stimulus. For each decision, the quantities HUNGER and FATIGUE are changed appropriately. The subroutine DECAY is called at the end to update the DECAYNOs. Appropriate messages are printed out in each case. If FROG decides to feed on the stimulus and no pain is inflicted by the stimulus on FROG, the visual and taste inputs are sent to the GTM and the BTM of the memory, for the plane, for recording purposes. The next statement is a call to ZERO-OUT, an entry point in the subroutine DECAY; this ensures that any

T*-element that may have output during the write cycle will not affect the array USE. Specifically, the statement executed after ZEROOUT is called sets the whole array to binary zero after which control is returned to FROG. The COUNTER is then incremented and the control is transferred to the statement BEGIN to start a new cycle.

5. RESULTS AND CONCLUSION

5.1 Summary of a Run of the Simulator

The run consisted of 250 trials. The assigned probabilities of occurrences of the various stimuli in different regions are tabulated below.

Stimulus \ Region				
	1	2	3	4
F	.06	.07	.07	.00
G	.08	.00	.07	.05
H	.09	.11	.00	.00
L	.00	.07	.00	.10
B	.00	.08	.00	.07
S	.00	.00	.10	.00

The actual number of occurrences of the different stimuli in the different regions were

Region					
	1	2	3	4	
Stimulus					
F	24	12	13	0	49
G	12	0	23	12	47
H	27	33	0	0	60
L	0	19	0	21	40
B	0	23	0	12	34
S	0	0	19	0	19
	63	87	55	45	250

So the actual frequencies were,

Region	1	2	3	4
Stimulus	<hr/>			
F	.096	.048	.052	.000
G	.048	.000	.089	.048
H	.108	.132	.000	.000
L	.000	.076	.000	.084
B	.000	.089	.000	.048
S	.000	.000	.076	.000

The frequency of good tasting bug .384, that of bad tasting bugs was .400 and that of predators was .216.

A trial by trial summary of the run is given in Appendix A.

5.2 Results

The reactions of FROG to the various stimuli are separately discussed in the following paragraphs.

Fly: A fly shares the visual combinations

(1,2), (1,3) and (2,4) with a grasshopper,

(1,2), (1,4) and (1,5) with a honeybee,

(1,3) with a ladybug,

(1,2), (1,3) and (2,4) with a scorpion.

Since flies are good tasting they were quite readily fed upon by FROG. Consequently they were learned fairly quickly compared to other stimuli. The first two times that a fly appeared in region 1, it was fed upon by FROG mostly out of curiosity and high level of hunger. This resulted in enough learning for it to show a memory directed approach in the third appearance (trial #23). In all the following encounters of a fly in region 1 FROG had no problems in correct recognition. In fact, a full learning was achieved by the end of trial #164 (the 12th time a fly

appeared in region 1). Learning was reinforced because of grasshoppers in region 1. Recall, a fly shares a few level 1 and level 2 combinations with a grasshopper.

In region 2 a memory directed approach was taken by FROG in the very first appearance of a fly (#20). This approach was based on what was learned about flies in region 1 in trial #s 1 and 3. Being tired FROG did not feed on it. But the same cause led FROG to feed on the flies which appeared in the trial #s 25 and 32. These feedings resulted in perceptible learning in the level 4 of GTM of region 2. The next time a fly appeared in region 2 (trial #75) FROG's decision was guided by the region 2 GTM output. Progressive feeding resulted in full learning after trial #135 (the 10th appearance of a fly in region 2).

In region 3 flies appeared with grasshoppers again. Initial learning took place when FROG was driven by region 1 GTM outputs to feed on flies. The first two times that they appeared in region 3 (trial #s 28, 36). From the third time FROG was guided by its experience in this region (trial #50). The knowledge was pretty quickly reinforced in level 1 and level 2 by the simultaneous appearance of the grasshoppers with which flies share three combinations of basic visual characteristics. Full learning was observed after the 8th appearance of a fly in region 3 (i.e., after trial #126).

It is to be noted that since the good tasting bugs are seldom avoided they are learned very fast. Therefore, these are hardly ever mistaken for anything else. In fact the problem is the other way round. The characteristics that some predators and bad tasting bugs share with some good tasting bugs cause FROG to make erroneous judgment about the former because GTM outputs for the shared visual combinations grow faster

to their full potential than the corresponding BTM or PM outputs.

Grasshopper: A grasshopper shares the visual combinations

(1,2), (1,3) and (2,4) with a fly,

(1,2) with a honeybee,

(1,3) with a ladybug,

(1,2), (1,3) and (2,4) with a scorpion.

It appears in region 1 with flies and honeybees, in region 3 with flies and scorpions and in region 4 with ladybugs and beetles.

In region 1, following a few hunger forced reactions to grasshoppers, FROG makes a favorable approach in the 5th appearance of region 1 (#71). The decision was mostly influenced by the GTM outputs corresponding to the characteristics shared with flies, which had appeared several times by this time. By the end of trial #250 grasshoppers were fed upon only seven times out of the twelve times that it was presented in region 1. So full learning was not achieved. But due to the shared characteristics between a fly and a grasshopper the GTM outputs were already quite close to the full potential value.

In region 2 first few memory directed approaches were guided by the GTM outputs of level 1 (trial #39, 45), and later by the level 2 GTM outputs of region 3 mostly due to the characteristics shared with a fly. At any rate, trends were similar to that observed in region 1 and full learning was demonstrated after the twelfth appearance of a grasshopper (trial #123) in this region.

In region 4 no flies appear with grasshoppers. Initial reactions to grasshopper are thus guided first by curiosity, then by the GTM outputs of region 1 (mostly resulting from eating of flies) and then by the GTM outputs of region 3. In the sixth appearance of a grasshopper in region 4

(trial #170) the experience in region 4 guide FROG to make its decisions. Grasshoppers were fully learned in region 4 after trial #189.

The interesting thing to note here is that the characteristics shared by a grasshopper with a fly reinforced FROG's learning favorably for both a fly and a grasshopper. However, the same characteristics were shared by the scorpions. The reinforcement of the idea that these characteristics essentially relate to good tasting bugs caused FROG to suffer a lot when it came to dealing with scorpions as will be seen later.

Honeybee: A honeybee shares the visual combinations

(1,2), (1,4), (2,5) with a fly,

(1,2) with a grasshopper,

(1,2) with a scorpion.

Honeybees appeared only in region 1 and 2.

In region 1 honeybees appeared with both flies and grasshoppers. Flies and grasshoppers are good tasting and are mostly eaten. Also the two together appear more frequently than honeybees. So for the shared combinations of visual characteristics development of an initial bias toward good taste is to be expected. The GTM output for the shared combinations should grow faster than the corresponding BTM outputs, causing error in FROG's judgment. The results confirm the predicted trends.

In the first few appearances of honeybees in region 1, they were mostly fed upon by FROG, directed solely by hunger. First memory directed approach occurred in trial #48 based on what had been learned about honeybees in region 1. This knowledge was reflected in the avoidance of honeybees in the subsequent appearances (trial #51, 59, 60, 60). These decisions were made on triple joint combinations (level 3

combinations) since not much was learned about single and double joint combinations (level 1 and 2). But as the number of trials increased the level 2 and level 1 outputs become significant. Since honeybees share some of its level 1 and level 2 characteristics with flies and grasshoppers, the reason stated earlier caused FROG to make some errors (#67, 73, 99, 117, 127, 150). Only after making six consecutive errors of judging honeybees as good tasting and feeding on them on five occasions FROG learned enough to recover from the spell as is evident in trial #151. Thereafter it made no mistakes with honeybees in region 1. At the end honeybees were not fully learned because they were mostly avoided.

In region 2 honeybees appeared with flies only. In other words honeybees were not overwhelmingly outnumbered by the two good tasting bugs with which it shares some visual characteristics. So mistakes by FROG due to shared characteristics happen a little late (trial #96). With the exception of a minor sign of recovery in the next appearance (#105) the spell continued through the subsequent appearances of honeybees (trial #124, 132, 149) when they were fed upon (three out of four times) being taken as good tasting bugs. This feeding resulted in further learning so that FROG's decisions were corrected from trial #158 onwards. But due to frequent avoidance, honeybees were not fully learned in region 2 either.

The behavior of FROG towards honeybees is a good example of how shared characteristics may confuse issues in the learning stage.

Ladybug: A ladybug shares the visual combinations

(1,3) with a fly

(1,3) with a grasshopper

(1,1), (1,4), (2.3) with a beetle,

(1,1) with a scorpion.

It appears in region 2 along with flies and beetle and in region 4 with grasshoppers and beetles.

In region 2 a memory directed approach is observed in the third appearance of a ladybug (trial #19) and they were avoided for a few subsequent appearances (trial #30, 56, 87, 90). As ladybugs were being avoided flies and grasshoppers were being fed upon. As the level 1 GTM output rose, the shared characteristic (1,3) with flies and grasshoppers inclined FROG to believe that ladybugs were probably good tasting too as is evident from FROG's reaction to ladybugs in trial #s 125, 144, 152, 157, 159. Ladybugs were fed upon in these trials except when the hunger was too low (trial #125). By feeding on them FROG found out they were not so tasty after all so it started avoiding ladybugs again (trial #s 178, 194, 212, 245, 248).

In region 4, for the first few appearances of ladybugs FROG was guided by its knowledge in other regions. The first time a ladybug was avoided, it was due to what was learned about it in the region 2. But ladybugs were fed upon the next two times when FROG was guided by the GTM output in region 3. These two feedings resulted in an appreciable learning in level 4 of BTM of region 4 which kept FROG away from ladybugs in region 4 for quite a few subsequent appearances (trial #s 83, 85, 92, 98, 103, 106, 155, 156, 166, 169, 175, 184). In the meanwhile grasshoppers appearing in this region were mostly fed upon. By trial #214 the GTM output for the shared characteristics (1,3) become more than 0.5 and dominate the decision. Thus ladybugs were taken as good tasting bugs in trial #s 214, 224, 225, 250. These four feedings were not

sufficient in bringing the level 1 BTM output of region 4 up enough to offset the corresponding GTM output. This would have happened if full learning was achieved.

Beetle: A beetle shares the visual combinations

(1,1), (1,4) and (2,3) with a ladybug,

(1,1) with a scorpion.

It appears in regions 2 and 4 with ladybugs but no scorpions.

In region 2 the first appearances of beetles were reacted to by FROG completely guided by its hunger. It knew nothing about them, so it had no particular basis for making a decision. Obviously, the attempts to feed on beetles were not very pleasant and the resultant learning caused FROG to correctly avoid them (trial #s 56, 122, 133, 140, 146, 147) only mildly believing them to be predators. But in the same region ladybugs also appeared with beetles. Because of the shared characteristics (1,1), (1,4), (2,3) between beetles and ladybugs FROG was slowly biased to believe the beetles as bad tasting bugs instead and to avoid them (trial #s 173, 183, 199, 205, 217, 232, 241, 247, 249). Although this was not a correct diagnosis of the features of a beetle, it kept FROG away from the dangerous stimulus anyway. It was good in one way, but, on the other hand, FROG did not get to know a beetle properly.

Of all the appearances of beetles in region 4, only once, the very first time, did FROG have an encounter with a beetle. This time FROG's decision was initiated by its hunger. But in all subsequent appearances FROG was guided by its experience in region 2. For some time FROG avoided the beetles thinking them to be predators (trial #s 72, 131, 133, 136, 143, 148), and rightly so, after which shared characteristics with ladybugs convinced FROG to believe that the beetles should be bad

tasting. Continuous avoidance caused almost no learning of this stimulus. Beetles were avoided but mostly for the characteristics they share with the bad tasting ladybugs.

Scorpion: A scorpion shares the visual combinations

(1,2), (1,3) and (2,4) with a fly,

(1,2), (1,3) and (2,4) with a grasshopper.

Scorpions appeared only in region 3 along with flies and grasshoppers. The first two times the curiosity of FROG led it into two unpleasant encounters so that level 4 learning became appreciable to make it stay away from the scorpions in the subsequent appearances (trial #s 24, 27, 33, 49). As the GTM level 1 and level 2 outputs rose with more and more feeding of flies and grasshoppers, the shared characteristics misled FROG to believe the scorpions, which appeared in the trial #s 53, 79, 84, 94, 95, 107, 134, to be good tasting. Resulting bad experiences (four times out of seven that FROG actually tried to eat on the scorpions) resulted in enough additional learning to make it stay away from scorpion for the next few appearances of them. Full learning was not achieved by the 250th trial.

5.3 Conclusion

The results show that FROG was able to learn the various stimuli more or less correctly. However, there were mistakes made in the initial trials. There were three distinct type of errors made by FROG. The first was deciding to feed on a bad tasting bug or on a predator. These errors were easily corrected, for the feeding resulted in additional learning which alleviated the problem. The second type of error occurred when a good tasting bug was avoided. This error was difficult to correct; with feeding inhibited additional learning was not possible. Only starvation

forced feeding helped reverse the situation for this error. The third type of error was made whenever a predator was avoided, being taken as a bad tasting bug. Although this error inhibited proper learning, its effect was not altogether undesirable. In the case of beetles, even at the end of the run, (i.e. after 250 trials) they were not learned in the proper perspective and, therefore, problems similar to that resulting from type 2 error can be evidenced.

Admittedly FROG is not perfect, it makes mistakes. But these errors are not undesirable. In fact, it makes the simulation more realistic and natural.

LIST OF REFERENCES

1. Lettvin, J. Y., Maturana, H. R., McCulloch, W. S. and Pitts, W. H.,
What the Frog's Eye Tells the Frog's Brain, Proceedings of the
IRE, November 1959, pp. 1940-1951.
2. Maturana, H. R., Lettvin, J. Y., McCulloch, W. S. and Pitts, W. H.,
Anatomy and Physiology of Vision in the Frog (Ranapipeus),
Jour. of Gen. Physiology, Vol. 43, No. 6, July 1960, pp. 129-
176.
3. Marinos, P. E., Fussy Logic and Its Application to Switching Systems,
IEEE Transactions on Computers, Vol. C-18, No. 4, April 1969,
pp. 343-348.

APPENDIX A

TRIAL BY TRIAL SUMMARY OF A RUN OF THE SIMULATOR

A trial by trial summary of the run reported is given in the following pages. A guide to the various columns is given below.

Trial number	- serial number of the trial.
Stimulus	- name of the stimulus presented in the trial concerned.
Region	- the region number in which the stimulus appeared.
Location	- the region number in which FROG was located at the time the stimulus was presented.
Innate stimulus	- value of the "innate stimulus" of the stimulus.
Hunger	- current value of FROG's hunger.
Fatigue	- current value of the signal FATIGUE.
Miuout	- value of the MIUOUTF signal for the trial.
Alarm	- value of the ALARMF signal.
Feed	- value of the FEED signal.
Decision	- mode of action decided upon by FROG for the stimulus presented. Abbreviations used are:
	F = FROG decided to feed.
	NF = FROG decided not to feed.
	T = FROG was too tired to move.
	P = FROG sensed a predator.
	F,R = FROG decided to feed on predator, and retreated to its safe resting place in region 1 on encounter.
	P,R = FROG sensed a predator too close to itself, so it decided to flee to its safe resting place.

Specific or Overall - A 'S' is put in this column if the decision was achieved from the memory outputs for one region and an 'O' is placed if memory outputs from all the regions were used to make the decision.

Primary reason - Entry to this column is the signal which was primarily responsible for the decision. Following abbreviations are used:

h = hunger

f = fatigue

n GTM m = level m GTM output of region n.

n BTM m = level m BTM output of region n.

n PM m = level m PM output of region n.

At times when the final decision was guided primarily by hunger or fatigue, the memory output is entered in this column along with the value of hunger or fatigue in order to show how well the cognitive memory worked.

Trial No.	Stimulus	Region	Location	Innate Stimulus	Hunger	Fatigue	Minout	Alarm	Feed	Decision	Specific or Overall	Primary Reason
001	F	1	1	0.80	1.00	0.00	0.00	0.00	0.80	F	0	h = 1.00
002	S	3	1	0.90	0.90	0.05	0.00	0.00	0.90	F,R	0	i = 0.90
003	F	1	1	0.80	0.95	0.25	0.00	0.00	0.80	F	0	i = 0.80
004	B	4	1	0.90	0.85	0.30	0.00	0.00	0.85	F	0	h = 0.85
005	G	3	1	0.80	0.90	0.50	0.00	0.00	0.80	F	0	i = 0.80
006	B	2	3	0.90	0.80	0.60	0.00	0.00	0.80	F,R	0	h = 0.80
007	G	1	1	0.80	0.85	0.80	0.00	0.00	0.80	F	0	i = 0.80
008	H	2	1	0.90	0.75	0.85	0.00	0.00	0.75	T	0	f = 0.85
009	L	2	1	0.90	0.80	0.65	0.00	0.00	0.80	F	0	h = 0.80
010	B	2	2	0.90	0.75	0.75	0.00	0.00	0.75	F,R	0	h = 0.75
011	G	1	1	0.80	0.80	0.90	0.00	0.00	0.80	T	0	f = 0.90
012	H	1	1	0.90	0.85	0.70	0.00	0.00	0.85	F	0	h = 0.85
013	L	2	1	0.90	0.80	0.75	0.00	0.00	0.80	F	0	h = 0.80
014	H	2	2	0.90	0.75	0.85	0.00	0.00	0.75	T	0	f = 0.85
015	H	2	2	0.90	0.80	0.65	0.00	0.00	0.80	F	0	h = 0.80
016	G	4	2	0.80	0.75	0.70	0.00	0.00	0.75	F	0	h = 0.75
017	G	1	4	0.80	0.65	0.80	0.00	0.00	0.65	T	0	f = 0.80

Trial No.	Stimulus	Region	Location	Innate Stimulus	Hunger	Fatigue	Minout	Alarm	Feed	Decision	Specific or Overall	Primary Reason
018	S	3	4	0.90	0.70	0.60	0.00	0.00	0.70	F,R	0	h = 0.70
019	L	2	1	0.90	0.75	0.80	0.37	0.00	0.37	NF	S	2BTM4 = 0.63
020	F	2	1	0.80	0.80	0.60	0.54	0.00	0.54	T	0	1GTM4 = 0.54, f = 0.60
021	H	2	1	0.90	0.85	0.40	0.00	0.00	0.85	F	0	h = 0.85
022	B	2	2	0.90	0.80	0.50	0.00	0.00	0.80	F,R	0	h = 0.80
023	F	1	1	0.80	0.85	0.65	0.54	0.00	0.54	T	S	f = 0.65
024	S	3	1	0.90	0.90	0.45	0.00	0.63	0.00	P	S	3PM4 = 0.63
025	F	2	1	0.80	0.95	0.25	0.54	0.00	0.54	F	0	1GTM4 = 0.54
026	L	4	2	0.90	0.85	0.35	0.37	0.00	0.37	NF	0	2BTM4 = 0.63
027	S	3	2	0.90	0.90	0.15	0.00	0.63	0.00	P	S	3PM4 = 0.63
028	F	3	2	0.80	0.95	0.00	0.54	0.00	0.54	F	0	1GTM4 = 0.54
029	G	3	3	0.80	0.85	0.10	0.00	0.00	0.80	F	0	h = 0.80
030	L	2	3	0.90	0.75	0.15	0.37	0.00	0.37	NF	S	2BTM4 = 0.63
031	H	2	3	0.90	0.80	0.00	0.00	0.00	0.80	F	0	h = 0.80
032	F	2	2	0.80	0.75	0.10	0.54	0.00	0.54	F	0	1GTM4 = 0.54
033	S	3	2	0.90	0.65	0.15	0.00	0.63	0.00	P	S	3PM4 = 0.63
034	H	1	2	0.90	0.70	0.00	0.00	0.00	0.70	F	0	h = 0.70

Trial No.	Stimulus	Region	Location	Innate Stimulus	Hunger	Fatigue	Minout	Alarm	Feed	Decision	Specific or Overall	Primary Reason
035	G	1	1	0.80	0.65	0.10	0.00	0.00	0.65	F	0	h = 0.65
036	F	3	1	0.80	0.55	0.15	0.54	0.00	0.54	F	0	1GTM4 = 0.54
037	G	4	3	0.80	0.45	0.25	0.00	0.00	0.00	NF	0	h 0.50
038	F	1	3	0.80	0.50	0.05	0.54	0.00	0.50	F	S	1GTM4 = 0.54, h = 0.50
039	G	3	1	0.80	0.40	0.15	0.53	0.00	0.40	NF	0	1GTM2 = 0.53, h = 0.40
040	B	2	1	0.90	0.45	0.00	0.00	0.00	0.00	NF	0	h 0.50
041	H	2	1	0.90	0.50	0.00	0.00	0.00	0.50	F	0	h = 0.50
042	H	2	2	0.90	0.45	0.10	0.44	0.00	0.44	NF	S	1BTM3 = 0.56
043	F	2	2	0.80	0.50	0.00	0.54	0.00	0.50	F	S	2GTM4 = 0.54
044	H	2	2	0.90	0.40	0.05	0.44	0.00	0.40	NF	S	h = 0.40, 2BTM3 = 0.56
045	G	3	2	0.80	0.45	0.00	0.52	0.00	0.45	NF	0	h = 0.45, 1GRM2 = 0.52
046	H	2	2	0.90	0.50	0.00	0.44	0.00	0.44	NF	S	2BTM3 = 0.56
047	H	2	2	0.90	0.55	0.00	0.44	0.00	0.44	NF	S	2BTM3 = 0.56
048	H	1	2	0.90	0.60	0.00	0.44	0.00	0.44	NF	S	2BRM3 = 0.56
049	S	3	2	0.90	0.65	0.00	0.00	0.63	0.00	P	S	3PM4 = 0.63

Trial No.	Stimulus	Region	Location	Innate Stimulus	Hunger	Fatigue	Minout	Alarm	Feed	Decision	Specific or Overall	Primary Reason
050	F	3	2	0.80	0.70	0.00	0.54	0.00	0.00	F	S	3GTM4 = 0.54
051	H	1	3	0.90	0.60	0.10	0.44	0.00	0.44	NF	0	2BTM3 = 0.56
052	H	2	3	0.90	0.65	0.00	0.44	0.00	0.44	NF	S	2BRM3 = 0.56
053	S	3	3	0.90	0.70	0.00	0.52	0.00	0.00	F,R	S	3GTM2 = 0.52
054	B	2	1	0.90	0.75	0.15	0.00	0.00	0.75	F,R	0	h = 0.75
055	H	2	1	0.90	0.80	0.35	0.44	0.00	0.44	NF	S	2BTM3 = 0.56
056	L	2	1	0.90	0.85	0.15	0.37	0.00	0.37	NF	S	2BTM4 = 0.63
057	G	3	1	0.80	0.90	0.00	0.52	0.00	0.52	F	S	3GTM2 = 0.52
058	F	2	3	0.80	0.80	0.10	0.54	0.00	0.54	F	S	2GTM4 = 0.54
059	H	1	2	0.90	0.70	0.20	0.44	0.00	0.44	NF	0	2BTM3 = 0.56
060	H	1	2	0.90	0.75	0.00	0.44	0.00	0.44	NF	0	2BTM3 = 0.56
061	H	1	2	0.90	0.80	0.00	0.44	0.00	0.44	NF	0	2BTM3 = 0.56
062	G	3	2	0.80	0.85	0.00	0.63	0.00	0.63	F	S	3GTM2 = 0.63
063	H	2	3	0.90	0.75	0.10	0.44	0.00	0.44	NF	S	2BTM3 = 0.56
064	F	2	3	0.80	0.80	0.00	0.63	0.00	0.63	F	S	2GTM3 = 0.63
065	H	2	2	0.90	0.70	0.10	0.44	0.00	0.44	NF	S	2BTM3 = 0.56
066	B	2	2	0.90	0.75	0.00	0.00	0.56	0.00	P	S	2PM3 = 0.56

Trial No.	Stimulus	Region	Location	Innate Stimulus	Hunger	Fatigue	Minout	Alarm	Feed	Decision	Specific or Overall	Primary Reason
067	H	1	1	0.90	0.80	0.10	0.55	0.00	0.55	F	0	3GTM1 = 0.55
068	F	3	1	0.80	0.75	0.15	0.55	0.00	0.55	F	S	3GTM1 = 0.55
069	L	4	3	0.90	0.65	0.25	0.63	0.00	0.63	F	0	3GTM1 = 0.63
070	B	2	4	0.90	0.60	0.35	0.00	0.56	0.00	P	S	2PM3 = 0.56
071	G	1	4	0.80	0.65	0.15	0.52	0.00	0.52	F	S	1GRM2 = 0.52
072	B	4	1	0.90	0.55	0.25	0.00	0.56	0.00	P	0	2PM3 = 0.56
073	H	1	1	0.90	0.60	0.05	0.63	0.00	0.60	F	0	3GTM1 = 0.63, h = 0.60
074	L	4	1	0.90	0.55	0.10	0.63	0.00	0.55	F	0	3GTM1 = 0.63, h = 0.55
075	F	1	4	0.80	0.50	0.20	0.63	0.00	0.50	F	S	1GTM2 = 0.63, h = 0.50
076	G	3	1	0.80	0.40	0.30	0.63	0.00	0.40	NF	S	3GTM1 = 0.63, h = 0.40
077	F	1	1	0.80	0.45	0.10	0.55	0.00	0.45	NF	S	1GTM1 = 0.55, h = 0.45
078	G	3	1	0.80	0.50	0.00	0.63	0.00	0.50	F	S	3GTM1 = 0.63, h = 0.50
079	S	3	3	0.90	0.40	0.10	0.63	0.00	0.40	NF	S	3GTM1 = 0.63, h = 0.40

Trial No.	Stimulus	Region	Location	Innate Stimulus	Hunger	Fatigue	Minout	Alarm	Feed	Decision	Specific or Overall	Primary Reason
080	F	2	3	0.80	0.45	0.00	0.52	0.00	0.45	NF	S	2GTM2 = 0.52, h = 0.45
081	H	2	3	0.90	0.50	0.00	0.44	0.00	0.44	NF	S	2BTM3 = 0.56
082	F	2	3	0.80	0.55	0.00	0.52	0.00	0.52	F	S	2GTM2 = 0.52
083	L	4	2	0.90	0.45	0.10	0.37	0.00	0.37	NF	S	4BTM4 = 0.63
084	S	3	2	0.90	0.50	0.00	0.63	0.00	0.50	F,R	S	3GTM1 = 0.63, h = 0.50
085	L	4	1	0.90	0.55	0.20	0.37	0.00	0.37	NF	S	4BTM4 = 0.63
086	F	3	1	0.80	0.60	0.00	0.63	0.00	0.60	F	S	3GTM1 = 0.63, h = 0.60
087	L	2	3	0.90	0.50	0.10	0.37	0.00	0.37	NF	S	2BTM4 = 0.63
088	F	3	3	0.80	0.55	0.00	0.63	0.00	0.55	F	S	3GTM1 = 0.63 h = 0.55
089	L	2	3	0.90	0.45	0.00	0.37	0.00	0.37	NF	S	2BTM4 = 0.63
090	L	2	3	0.90	0.50	0.00	0.37	0.00	0.37	NF	S	2BTM4 = 0.63
091	F	1	3	0.80	0.55	0.00	0.55	0.00	0.55	F	S	1GTM1 = 0.55
092	L	4	1	0.90	0.45	0.00	0.10	0.00	0.37	NF	S	4BTM4 = 0.63
093	F	1	1	0.80	0.50	0.00	0.63	0.00	0.50	F	S	1GTM1 = 0.63, h = 0.50

Trial No.	Stimulus	Region	Location	Innate Stimulus	Hunger	Fatigue	Minout	Alarm	Feed	Decision	Specific or Overall	Primary Reason
094	S	3	1	0.90	0.40	0.05	0.63	0.00	0.40	NF	S	3GTML = 0.63, h = 0.40
095	S	3	1	0.90	0.45	0.00	0.63	0.00	0.45	NF	S	3GTML = 0.63, h = 0.45
096	H	2	1	0.90	0.50	0.00	0.54	0.00	0.50	F	S	2GTML2 = 0.54, h = 0.50
097	G	4	2	0.80	0.45	0.10	0.63	0.00	0.45	NF	0	1GTML = 0.63, h = 0.45
098	L	4	2	0.90	0.50	0.00	0.37	0.00	0.37	NF	S	4BTML4 = 0.63
099	H	1	2	0.90	0.55	0.00	0.63	0.00	0.55	F	S	1GTML = 0.63, h = 0.55
100	G	4	1	0.80	0.50	0.10	0.63	0.00	0.50	F	0	1GTML = 0.63, h = 0.50
101	F	1	4	0.80	0.40	0.20	0.63	0.00	0.40	NF	S	1GTML = 0.63, h = 0.40
102	G	3	4	0.80	0.45	0.00	0.63	0.00	0.45	NF	S	3GTML = 0.63, h = 0.45
103	L	4	4	0.90	0.50	0.00	0.37	0.00	0.37	NF	S	4BTML4 = 0.63
104	F	1	4	0.80	0.55	0.00	0.63	0.00	0.55	F	S	1GTML = 0.63, h = 0.55
105	H	2	1	0.90	0.45	0.10	0.44	0.00	0.44	NF	S	2BTML3 = 0.56

Trial No.	Stimulus	Region	Location	Innate Stimulus	Hunger	Fatigue	Minout	Alarm	Feed	Decision	Specific or Overall	Primary Reason
106	L	4	1	0.90	0.50	0.00	0.37	0.00	0.37	NF	S	4BTM4 = 0.63
107	S	3	1	0.90	0.55	0.00	0.63	0.00	0.55	F, R	S	3GTM1 = 0.63, h = 0.55
108	G	3	1	0.80	0.60	0.20	0.63	0.00	0.60	F	S	3GTM1 = 0.63, h = 0.60
109	F	3	3	0.80	0.50	0.30	0.63	0.00	0.50	F	S	3GTM3 - 0.63, h = 0.50
110	F	1	3	0.80	0.40	0.35	0.63	0.00	0.40	NF	S	1GTM1 - 0.63, h = 0.40
111	F	1	3	0.45	0.15	0.00	0.63	0.00	0.45	NF	S	1GTM1 = 0.63, h = 0.45
112	F	3	3	0.80	0.50	0.00	0.63	0.00	0.50	F	S	3GTM1 = 0.63 h = 0.50
113	B	2	3	0.90	0.40	0.05	0.00	0.55	0.00	P	S	2)M3 = 0.55
114	H	2	3	0.90	0.45	0.00	0.44	0.00	0.44	NF	S	2BTM3 = 0.56
115	B	2	3	0.90	0.50	0.00	0.00	0.55	0.00	P	S	2PM3 = 0.55
116	H	2	3	0.90	0.55	0.00	0.44	0.00	0.44	NF	S	2BTM3 = 0.56
117	H	1	3	0.90	0.60	0.00	0.63	0.00	0.60	F	S	1GTM1 = 0.63
118	F	2	1	0.80	0.55	0.10	0.54	0.00	0.54	F	S	2GTM1 = 0.54
119	G	1	2	0.80	0.45	0.20	0.63	0.00	0.45	NF	S	1GTM1 = 0.63

Trial No.	Stimulus	Region	Location	Innate Stimulus	Hunger	Fatigue	Minout	Alarm	Feed	Decision	Specific or Overall	Primary Reason
120	G	4	2	0.80	0.50	0.00	0.63	0.00	0.50	F	0	3GTM1 = 0.63, h = 0.50
121	G	3	4	0.80	0.40	0.10	0.63	0.00	0.40	NF	S	3GTM1 = 0.63, h = 0.40
122	B	2	4	0.90	0.45	0.00	0.00	0.55	0.00	P	S	2PM3 = 0.55
123	G	3	4	0.80	0.50	0.00	0.63	0.00	0.50	F	S	3GTM1 = 0.63, h = 0.50
124	H	2	3	0.90	0.40	0.10	0.55	0.00	0.40	NF	S	2GTM1 = 0.55, h = 0.40
125	L	2	3	0.90	0.45	0.00	0.55	0.00	0.45	NF	S	2GTM1 = 0.55, h = 0.45
126	F	3	3	0.80	0.50	0.00	0.63	0.00	0.50	F	S	3GTM1 = 0.63, h = 0.50
127	H	1	3	0.90	0.40	0.05	0.63	0.00	0.40	NF	S	3GTM1 = 0.63, h = 0.40
128	F	3	3	0.80	0.45	0.00	0.72	0.00	0.45	NF	S	3GTM1 = 0.72, h = 0.45
129	F	3	3	0.80	0.50	0.00	0.72	0.00	0.50	F	S	3GTM1 = 0.72, h = 0.50
130	G	1	3	0.80	0.40	0.05	0.63	0.00	0.40	NF	S	1GTM1 = 0.63, h = 0.40

Trial No.	Stimulus	Region	Location	Innate Stimulus	Hunger	Fatigue	Minout	Alarm	Feed	Decision	Specific or Overall	Primary Reason
131	B	4	3	0.90	0.45	0.00	0.00	0.55	0.00	P	0	2PM3 = 0.55
132	H	2	3	0.90	0.50	0.00	0.55	0.00	0.50	F	S	2GTM1 = 0.55, h = 0.50
133	B	2	2	0.90	0.45	0.10	0.00	0.55	0.00	P,R	S	2PM3 = 0.55
134	S	3	1	0.90	0.50	0.20	0.63	0.00	0.50	F,R	S	3GTM1 = 0.63, h = 0.50
135	F	2	1	0.80	0.55	0.40	0.63	0.00	0.55	F	S	2GTM1 = 0.63, h = 0.55
136	B	4	2	0.90	0.45	0.50	0.00	0.55	0.00	P	0	2PM3 = 0.50
137	S	3	2	0.90	0.50	0.30	0.00	0.62	0.00	P	S	3PM3 = 0.62
138	G	1	2	0.80	0.55	0.10	0.63	0.00	0.55	F	S	1GTM1 = 0.63, h = 0.55
139	S	3	1	0.90	0.45	0.20	0.00	0.62	0.00	P	S	3PM3 = 0.62
140	B	2	1	0.90	0.50	0.00	0.00	0.55	0.00	P	S	2PM3 = 0.55
141	G	3	1	0.80	0.55	0.00	0.63	0.00	0.55	F	S	3GTM1 = 0.63, h = 0.45
142	G	3	3	0.80	0.45	0.10	0.63	0.00	0.45	NF	S	3GTM1 = 0.63, h = 0.45
143	B	4	3	0.90	0.50	0.00	0.00	0.55	0.00	P	0	2PM3 = 0.55
144	L	2	3	0.90	0.55	0.00	0.63	0.00	0.55	F	S	2GTM1 = 0.63, h = 0.55

Trial No.	Stimulus	Region	Location	Innate Stimulus	Hunger	Fatigue	Minout	Alarm	Feed	Decision	Specific or Overall	Primary Reason
145	G	1	2	0.80	0.50	0.10	0.63	0.00	0.50	F	S	1GTm1 = 0.63, h = 0.50
146	B	2	1	0.90	0.40	0.20	0.00	0.55	0.00	P	S	2PM3 = 0.55
147	B	2	1	0.90	0.45	0.00	0.00	0.55	0.00	P	S	2PM3 = 0.55
148	B	4	1	0.90	0.50	0.00	0.00	0.55	0.00	P	O	2PM3 = 0.55
149	H	2	1	0.90	0.55	0.00	0.63	0.00	0.55	F	S	2GTm1 = 0.63, h = 0.55
150	H	1	2	0.90	0.50	0.10	0.63	0.00	0.50	F	S	1GTm1 = 0.63, h = 0.50
151	H	1	1	0.90	0.45	0.20	0.37	0.00	0.37	NF	S	1BTm2 = 0.63
152	L	2	1	0.90	0.50	0.00	0.63	0.00	0.50	F	S	2GTm2 = 0.63, h = 0.50
153	G	1	2	0.80	0.45	0.10	0.63	0.00	0.45	NF	S	1GTm1 = 0.63, h = 0.45
154	H	1	2	0.90	0.50	0.00	0.37	0.00	0.37	NF	S	1BTm2 = 0.63
155	L	4	2	0.90	0.55	0.00	0.37	0.00	0.37	NF	S	4BTm4 = 0.63
156	L	4	2	0.90	0.60	0.00	0.37	0.00	0.37	NF	S	4BTm4 = 0.63
157	L	2	2	0.90	0.65	0.00	0.63	0.00	0.63	F	S	2GTm1 = 0.63
158	H	2	2	0.90	0.60	0.05	0.37	0.00	0.37	NF	S	2BTm2 = 0.63

Trial No.	Stimulus	Region	Location	Innate Stimulus	Hunger	Fatigue	Minout	Alarm	Feed	Decision	Specific or Overall	Primary Reason
159	L	2	2	0.90	0.65	0.00	0.63	0.00	0.63	F	S	2GTM1 = 0.63
160	F	3	2	0.80	0.60	0.05	0.72	0.00	0.60	F	S	3GTM1 = 0.72, h = 0.60
161	H	1	3	0.90	0.50	0.15	0.37	0.00	0.37	NF	S	1BTM2 = 0.63
162	S	3	3	0.90	0.55	0.00	0.00	0.63	0.00	P	S	3PM3 = 0.63
163	G	4	1	0.80	0.60	0.10	0.63	0.00	0.60	F	O	3GTM1 = 0.63
164	F	1	4	0.80	0.50	0.20	0.63	0.00	0.50	F	S	1GTM1 = 0.63, h = 0.50
165	G	3	1	0.80	0.40	0.30	0.63	0.00	0.40	NF	S	3GTM1 = 0.63 h = 0.40
166	L	4	1	0.90	0.45	0.10	0.37	0.00	0.37	NF	S	4BTM4 = 0.63
167	L	4	1	0.90	0.50	0.00	0.37	0.00	0.37	NF	S	4BTM4 = 0.63
168	H	2	1	0.90	0.55	0.00	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
169	L	4	1	0.90	0.60	0.00	0.37	0.00	0.37	NF	S	4BTM4 = 0.63
170	G	4	1	0.80	0.65	0.00	0.55	0.00	0.55	F	S	4GTM3 = 0.55
171	L	4	4	0.90	0.55	0.10	0.37	0.00	0.37	NF	S	4BTM4 = 0.63
172	H	1	4	0.90	0.60	0.00	0.37	0.00	0.37	NF	S	1BTM2 = 0.63
173	B	2	4	0.90	0.65	0.00	0.37	0.00	0.37	NF	S	2BTM2 = 0.63

Trial No.	Stimulus	Region	Location	Innate Stimulus	Hunger	Fatigue	Minout	Alarm	Feed	Decision	Specific or Overall	Primary Reason
174	F	3	4	0.80	0.70	0.00	0.72	0.00	0.70	F	S	3GTM1 = 0.72
175	H	2	3	0.90	0.60	0.10	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
176	B	4	3	0.90	0.65	0.00	0.37	0.00	0.37	NF	O	2BTM2 = 0.63
177	G	4	3	0.80	0.70	0.00	0.55	0.00	0.55	F	S	4GTM3 = 0.55
178	L	2	4	0.90	0.60	0.10	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
179	L	4	4	0.90	0.65	0.00	0.37	0.00	0.37	NF	S	4BTM4 = 0.63
180	B	4	4	0.90	0.70	0.00	0.37	0.00	0.37	NF	O	2BTM2 = 0.63
181	G	4	4	0.80	0.75	0.00	0.55	0.00	0.55	F	S	4GTM3 = 0.55
182	F	3	4	0.80	0.56	0.00	0.72	0.00	0.65	F	S	3GTM1 = 0.72
183	B	2	3	0.90	0.55	0.15	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
184	L	4	3	0.90	0.60	0.00	0.37	0.00	0.37	NF	S	4BTM4 = 0.63
185	H	2	3	0.90	0.65	0.00	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
186	B	4	3	0.90	0.70	0.00	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
187	G	3	3	0.80	0.75	0.00	0.63	0.00	0.63	F	S	3GTM1 - 0.63
188	H	1	3	0.90	0.65	0.05	0.37	0.00	0.37	NF	S	1BTM2 = 0.63
189	G	4	3	0.80	0.70	0.00	0.55	0.00	0.55	F	S	4GTM1 = 0.55

Trial No.	Stimulus	Region	Location	Innate Stimulus	Hunger	Fatigue	Minout	Alarm	Feed	Decision	Specific or Overall	Primary Reason
190	H	1	4	0.90	0.60	0.10	0.37	0.00	0.37	NF	S	1BTM2 = 0.63
191	F	1	4	0.80	0.65	0.00	0.72	0.00	0.65	F	S	1GTM1 = 0.72, h = 0.65
192	F	1	1	0.80	0.55	0.00	0.72	0.00	0.55	F	S	1GTM1 = 0.72, h = 0.65
193	H	2	1	0.90	0.45	0.15	0.37	0.00	0.37	NF	S	2GTM2 = 0.63
194	L	2	1	0.90	0.50	0.00	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
195	H	1	1	0.90	0.55	0.00	0.37	0.00	0.37	NF	S	1BTM2 = 0.63
196	G	3	1	0.80	0.60	0.00	0.63	0.00	0.60	F	S	1GTM1 = 0.63
197	S	3	3	0.90	0.50	0.10	0.00	0.62	0.00	P,R	S	3PM2 = 0.62
198	B	2	1	0.90	0.55	0.20	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
199	B	2	1	0.90	0.55	0.20	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
200	G	1	1	0.80	0.65	0.00	0.63	0.00	0.63	F	S	1GTM1 = 0.63
201	H	1	1	0.90	0.55	0.05	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
202	H	1	1	0.90	0.60	0.00	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
203	H	1	1	0.80	0.65	0.00	0.72	0.00	0.65	F	S	1GTM1 = 0.72, h = 0.65

Trial No.	Stimulus	Region	Location	Innate Stimulus	Hunger	Fatigue	Minout	Alarm	Feed	Decision	Specific or Overall	Primary Reason
204	F	1	1	0.80	0.55	0.05	0.72	0.00	0.55	F	S	1GTm1 = 0.72, h = 0.55
205	B	2	1	0.90	0.45	0.10	0.37	0.00	0.37	NF	S	2BTm2 = 0.62
206	S	3	1	0.90	0.50	0.00	0.00	0.62	0.00	P	S	3PM3 = 0.62
207	F	1	1	0.80	0.55	0.00	0.72	0.00	0.55	F	S	1GTm1 = 0.72, h = 0.55
208	G	3	1	0.80	0.45	0.05	0.63	0.00	0.45	NF	S	1GTm1 = 0.63, h = 0.45
209	H	2	1	0.90	0.50	0.00	0.37	0.00	0.37	NF	S	2BTm2 = 0.63
210	F	1	1	0.80	0.55	0.00	0.72	0.00	0.55	F	S	1GTm1 = 0.72, h = 0.55
211	H	2	1	0.90	0.45	0.05	0.37	0.00	0.37	NF	S	2BTm2 = 0.63
212	L	2	1	0.90	0.50	0.00	0.37	0.00	0.37	NF	S	2BTm2 = 0.63
213	F	1	1	0.80	0.55	0.00	0.72	0.00	0.55	F	S	1GTm1 = 0.72, h = 0.55
214	L	4	1	0.90	0.45	0.05	0.55	0.00	0.45	NF	S	1GTm1 = 0.55, h = 0.45
215	F	2	1	0.80	0.50	0.00	0.72	0.00	0.50	F	S	2GTm1 = 0.72, h = 0.50
216	H	2	2	0.90	0.40	0.10	0.37	0.00	0.37	NF	S	2BTm2 = 0.63

Trial No.	Stimulus	Region	Location	Innate Stimulus	Hunger	Fatigue	Minout	Alarm	Feed	Decision	Specific or Overall	Primary Reason
217	B	2	2	0.90	0.45	0.00	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
218	H	2	2	0.90	0.50	0.00	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
219	G	4	2	0.80	0.55	0.00	0.63	0.00	0.55	F	S	4GTML = 0.63
220	H	1	4	0.90	0.45	0.10	0.37	0.00	0.37	NF	S	1BTM2 = 0.63
221	S	3	4	0.90	0.50	0.00	0.00	0.62	0.00	P	S	3PM3 = 0.62
222	G	1	4	0.80	0.55	0.00	0.63	0.00	0.55	F	S	1GTML = 0.63, h = 0.55
223	F	1	1	0.80	0.45	0.10	0.72	0.00	0.45	NF	S	1GTML = 0.72, h = 0.45
224	L	4	1	0.90	0.50	0.00	0.55	0.00	0.50	F	S	4GTML = 0.55, h = 0.50
225	L	4	4	0.90	0.45	0.10	0.55	0.00	0.45	NF	S	4GTML = 0.55, h = 0.45
226	H	2	4	0.90	0.50	0.00	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
227	H	1	4	0.90	0.55	0.00	0.37	0.00	0.37	NF	S	1BTM2 = 0.63
228	G	3	4	0.80	0.60	0.00	0.63	0.00	0.60	F	S	3GTML = 0.63, h = 0.60
229	L	4	3	0.90	0.50	0.10	0.55	0.00	0.50	F	S	4GTML = 0.55, h = 0.50
230	B	4	4	0.90	0.45	0.20	0.37	0.00	0.37	NF	O	2BTM2 = 0.63

Trial No.	Stimulus	Region	Location	Innate Stimulus	Hunger	Fatigue	Minout	Alarm	Feed	Decision	Specific Or Overall	Primary Reason
231	H	1	4	0.90	0.50	0.00	0.37	0.00	0.37	NF	S	1BTM2 = 0.63
232	B	2	4	0.90	0.55	0.00	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
233	H	1	4	0.90	0.60	0.00	0.37	0.00	0.37	NF	S	1BTM2 = 0.63
234	G	3	4	0.80	0.65	0.00	0.63	0.00	0.63	F	S	3GTM1 = 0.63
235	G	4	3	0.80	0.55	0.10	0.63	0.00	0.55	F	S	4GTM1 = 0.63
236	B	4	4	0.90	0.45	0.20	0.37	0.00	0.37	NF	0	2BTM2 = 0.63
237	H	1	4	0.90	0.50	0.00	0.37	0.00	0.37	NF	S	1BTM2 = 0.63
238	G	3	4	0.80	0.55	0.00	0.63	0.00	0.55	F	S	3GTM1 = 0.63, h = 0.55
239	F	2	3	0.80	0.45	0.10	0.72	0.00	0.45	NF	S	2GTM1 = 0.72, h = 0.45
240	H	2	3	0.90	0.50	0.00	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
241	B	2	3	0.90	0.55	0.00	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
242	G	3	3	0.80	0.60	0.00	0.63	0.00	0.60	F	S	3GTM1 = 0.63, h = 0.60
243	F	1	3	0.80	0.50	0.05	0.72	0.00	0.50	F	S	1GTM1 = 0.72, h = 0.50
244	F	1	1	0.80	0.40	0.15	0.72	0.00	0.40	NF	S	1GTM1 = 0.72, h = 0.40

Trial No.	Stimulus	Region	Location	Innate Stimulus	Hunger	Fatigue	Minout	Alarm	Feed	Decision	Specific or Overall	Primary Reason
245	L	2	1	0.90	0.45	0.00	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
246	F	1	1	0.80	0.50	0.00	0.72	0.00	0.50	F	S	1GTM1 = 0.72, h = 0.50
247	B	2	1	0.90	0.40	0.05	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
248	L	2	1	0.90	0.45	0.00	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
249	B	2	1	0.90	0.50	0.00	0.37	0.00	0.37	NF	S	2BTM2 = 0.63
250	L	4	1	0.90	0.55	0.00	0.55	0.00	0.55	F	S	4GTM1 = 0.55

APPENDIX B

SIMULATOR SOURCE DECK LISTING


```
FROG: /* LOGIC SYSTEM SIMULATOR */ PROC OPTIONS(MAIN);
```

```
/* FROG IS THE MAIN CALLING PROGRAM. IN ADDITION, IT INCORPORATES  
THE SIMULATION OF THE DECISION UNIT. */
```

```
DCL (VISUALCHAR(4), /* FOUR VISUAL CHARACTERISTICS OF A BUG */  
GOODTST, /* GOOD TASTE OF A BUG */  
BADTST, /* BAD TASTE OF A BUG */  
PAIN, /* PAIN INFLICTED BY A PREDATOR */  
HUNGER INITIAL(1.0000000), /* THE INITIAL VALUE OF HUNGER */  
FATIGUE INITIAL(0.0000000), /* THE INITIAL VALUE OF FATIGUE */  
INNATESTIM, /* INNATE VISUAL STIMULUS OF A BUG OR A PREDATOR */  
FEED, /* IMPULSE OF FROG TO FEED */  
ALARMOUT(4), /* ALARM OUTPUTS OF THE FOUR MEMORY PLANES */  
MIUOUT(4), /* OUTPUTS FROM THE MIU'S OF THE FOUR MEMORY PLANES */  
NODEC4(4), /* THE NODEC4 OUTPUTS OF THE FOUR MEMORY PLANES */  
ALARME, /* THE FINAL ALARM OUTPUT OF THE MEMORY */  
MIUOUTF, /* THE FINAL MIU OUTPUT */  
NODECF, /* THE FINAL INDICATION OF WHETHER A DECISION CAN  
BE REACHED OR NOT */  
GTMEMOUT(4), /* FOUR LEVEL OUTPUTS OF A GTM */  
BTMEMOUT(4), /* FOUR LEVEL OUTPUTS OF A BTM */  
PMEMOUT(4), /* FOUR LEVEL OUTPUTS OF A PM */  
ZERO INITIAL(0.0000000),  
ONE INITIAL(1.0000000) DECIMAL FIXED (9,7),  
  
BUGNAME CHARACTER (20),  
  
ZERPOINT INIT(0) FIXED DECIMAL(1),  
(ONEINT INIT(1),TWOINT INIT(2),  
THREINT INIT(3), FOURINT INIT(4)) BINARY FIXED ,  
  
COUNTER INIT(1) FIXED DECIMAL(3),  
PRINT FIXED DECIMAL(1), /* PARAMETER TO TELL IF A MEMORY  
DUMP IS WANTED */  
(REGION, /* REGION IN WHICH THE STIMULUS APPEARS */  
LOCATION INIT(1), /* FROG IS INITIALLY IN REGION 1 */  
GTM INIT(1),BTM INIT(2), PM INIT(3) /* CODE NUMBERS FOR THE  
DIFFERENT MEMORIES */ ) BINARY FIXED,  
STARLINE CHAR(120) INIT((120)'*'), /* USED TO SEPERATE THE  
TRIALS IN THE PRINTOUT */  
(RANNUM, /* PARAMETER TO WHICH A RANDOM NUMBER IS ASSIGNED */  
PROB) BINARY FLOAT(31),  
(HOWMANY INIT(6), /* THIS IS THE NUMBER OF BUGS AND PRE-  
DATORS IN FROG'S ENVIRONMENT THIS RUN */  
REGIONNO INIT(4) /*THIS IS THE NUMBER OF REGIONS IN  
FROG'S ENVIRONMENT */ ) FIXED BINARY,  
  
/* ALL FUNCTION SUBROUTINES ARE DECLARED AS "ENTRIES" */  
T ENTRY( BINARY FIXED , DECIMAL FIXED (9,7),  
DECIMAL FIXED (9,7), DECIMAL FIXED (9,7) )  
RETURNS( DECIMAL FIXED (9,7) ),  
MEM ENTRY( DECIMAL FIXED (9,7), DECIMAL FIXED (9,7) )  
RETURNS( DECIMAL FIXED (9,7) ),  
SI ENTRY( DECIMAL FIXED (9,7) )  
RETURNS( DECIMAL FIXED (9,7) ),  
E ENTRY( DECIMAL FIXED (9,7), DECIMAL FIXED (9,7) )
```

```

        RETURNS( DECIMAL FIXED (9,7) ),
P      ENTRY( DECIMAL FIXED (9,7), DECIMAL FIXED (9,7) )
        RETURNS( DECIMAL FIXED (9,7) ),
C      ENTRY( DECIMAL FIXED (9,7) )
        RETURNS( DECIMAL FIXED (9,7) ),
S      ENTRY( DECIMAL FIXED (9,7), DECIMAL FIXED (9,7) )
        RETURNS( DECIMAL FIXED (9,7) ),
RDPROB(HOWMANY,REGIONNO) BINARY FLOAT (31),/* "RDPROB" IS AN
        ARRAY WHOSE MEMBERS ARE READ IN DURING EXECUTION. EACH ROW
        CORRESPONDS TO A PARTICULAR STIMULUS AND THERE ARE AS
        MANY ROWS AS THE NUMBER OF BUGS AND PREDATORS IN FROG'S
        ENVIRONMENT. EACH COLUMN CORRESPONDS TO A PARTICULAR
        REGION AND THERE ARE AS MANY COLUMNS AS THE NUMBER OF
        DIFFERENT REGIONS IN FROG'S ENVIRONMENT. EACH ELEMENT
        OF THE ARRAY REPRESENTS THE DESIRED PROBABILITY OF OCCU-
        RRENCE OF A PARTICULAR STIMULUS IN A PARTICULAR REGION. */

/* THE DIMENSIONS OF THE FOLLOWING FOUR ARRAYS ("NAMES","CHARS",
"TTASTES","PAININPT") CAN BE CHANGED IN ORDER TO INCREASE OR
DECREASE THE NUMBER OF STIMULI IN "FROG'S" ENVIRONMENT. THIS
IS ACCOMPLISHED BY ALTERING THE INITIAL VALUE OF "HOWMANY" */

NAMES( HOWMANY ) CHAR(20) INIT( /* THE NAMES OF THE BUGS */
        'FLY                                ',
        'GRASSHOPPER                        ',
        'HONEYBEE                           ',
        'LADYBUG                            ',
        'BEETLE                             ',
        'SCORPION                           '),

CHARS(HOWMANY,4) FIXED DECIMAL (9,7) INIT(
/* VISUAL CHARACTERISTICS OF THE STIMULI */
        0.8000000,0.7000000,0.7000000,0.6000000, /*FLY*/
        0.6000000,0.7000000,0.7000000,0.8000000, /*GRASSHOPPER*/
        0.9000000,0.7000000,0.8000000,0.6000000, /*HONEYBEE*/
        0.7000000,0.9000000,0.7000000,0.7000000, /*LADYBUG*/
        0.7000000,0.6000000,0.9000000,0.7000000, /*BEETLE*/
        0.7000000,0.7000000,0.7000000,0.9000000 /*SCORPION*/ ),

TASTES(HOWMANY) FIXED DECIMAL (9,7) INIT(
/* GOOD TASTES OF THE STIMULI */
        0.9000000,/* FLY */
        0.8000000,/* GRASSHOPPER */
        0.2000000,/* HONEYBEE */
        0.1000000,/* LADYBUG */
        0.0000000,/* BEETLE */
        0.0000000 /* SCORPION */ ),

PAININPT(HOWMANY) FIXED DECIMAL (9,7) INIT(
/* PAIN INFLICTED BY THE STIMULI */
        0.0000000,/* FLY */
        0.0000000,/* GRASSHOPPER */
        0.0000000,/* HONEYBEE */
        0.0000000,/* LADYBUG */
        0.8000000,/* BEETLE */
        0.9000000 /* SCORPION */ ),

CHOICE FIXED BINARY, /* CHOICE IS THE SUBSCRIPT OF THE
        MEMBER ARRAYS "NAMES", "CHARS","TASTES AND "PAININPT"

```

WHICH IS TO BE PRESENTED TO "FROG" IN A GIVEN TRIAL. */

/*THE FOLLOWING VARIABLES ARE ASSOCIATED WITH THE DECAY FUNCTION */

```
(USE1(4,3,4,3) INIT((144)(1)'0'B),
USE2(4,3,6,3) INIT((216)(1)'0'B),
USE3(4,3,4,3) INIT((144)(1)'0'B),
USE4(4,3,3) INIT(( 36)(1)'0'B)) BIT(1),/* TELLS WHETHER OR
NOT A GIVEN T*-ELEMENT HAS OUTPUT DURING ANY READ-CYCLE */
(DECAYNO1(4,3,4,3) INIT((144) 0.0000000),
DECAYNO2(4,3,6,3) INIT((216) 0.0000000),
DECAYNO3(4,3,4,3) INIT((144) 0.0000000),
DECAYNO4(4,3,3) INIT(( 36) 0.0000000)) DECIMAL FIXED (9,7),
/* NUMBERS TO BE SUBTRACTED TO SIMULATE DECAYING OF T*-
ELEMENTS */
MEMCNT(3) FIXED BINARY INIT((3) 0); /* INDICATES TO DECAY
HOWMANY T*-ELEMENT TO DECAY IN EACH CHAIN. MEMCNT(1) IS USED
FOR GTM, MEMCNT(2) IS USED FOR BTM, MEMCNT(3) FOR PM. */
```

```
GET LIST((RDPROB(I,J) DO J=1 TO REGIONNO) DO I=1 TO HOWMANY)); /* READ
IN THE LIST OF DESIRED FREQUENCIES OF OCCURRENCE OF T*ELEMENTS */
PUT EDIT((RDPROB(I,J) DO J=1 TO REGIONNO) DO I=1 TO HOWMANY))
(SKIP(5),6(4(X(10),F(4,2)),SKIP(2))); /* PRINT VALUES READ IN */
BEGIN: IF COUNTER=251 THEN GO TO TERM; /* CHECK FOR TERMINATION
CONDITION */
```

```
/* THE FOLLOWING GROUP OF STATEMENTS SPECIFIES THE MEMORY DUMPS */
IF COUNTER=125 | COUNTER=250
THEN PRINT=1;
ELSE PRINT=0;
```

```
RANUM=RANDOM; /* GET A RANDOM NUMBER */
```

```
/*THE FOLLOWING GROUP OF STATEMENTS DECIDES THE STIMULUS FOR
THIS TRIAL AND THE REGION IN WHICH IT SHOULD APPEAR */
```

```
PROB=0;
DO I=1 TO HOWMANY;
DO J=1 TO REGIONNO;
PROB=PROB+RDPROB(I,J);
IF RANUM <= PROB THEN DO;
CHOICE=I;
REGION=J;
GO TO ASSIGN;
END;
```

```
END ;
```

```
END;
```

```
ASSIGN: BUGNAME = NAMES(CHOICE);
VISUALCHAR = CHARS(CHOICE,*);
GOODTST = TASTES(CHOICE);
BADTST = 1- GOODTST;
INNATESTIM = MAX(VISUALCHAR(1),VISUALCHAR(2),VISUALCHAR(3),
VISUALCHAR(4));
```

```
PAIN = PAININPT(CHOICE);
```

```
PUT EDIT (STARLINE,STARLINE,'THIS IS TRIAL NUMBER ',COUNTER,
'"FROG" HAS JUST SEEN A BUG IN REGION NUMBER ',REGION,
'"FROG" IS PRESENTLY IN REGION NUMBER ',LOCATION,
'THIS BUG IS A ',BUGNAME,
'VISUAL CHARACTERISTICS OF THE BUG ARE:',
(VISUALCHAR(1) DO I=1 TO 4),'OTHER INFORMATION:',
'GENERAL INNATE STIMULUS IS:',INNATESTIM,
'HUNGER IS:',HUNGER,
'FATIGUE IS:',FATIGUE,
```

```

      ' "FROG" WILL NOW SEARCH ITS MEMORIES FOR',
      ' ANY EXPERIENCES WITH THIS BUG IN THE PAST.' )
    (PAGE,SKIP(2),A,SKIP,A,SKIP(2),X(45),A,F(3),SKIP(2),X(34),A,
      F(1),SKIP,X(38),A,F(1),SKIP(2),A,A(20),SKIP,A,
      4 F(12,7),SKIP,A,SKIP,X(5),A,F(10,7),SKIP,
      X(5),A,F(10,7),SKIP,X(5),A,F(10,7),
      SKIP(2),X(40),A,SKIP,X(39),A);

DO I=1 TO 4;
  ALARMOUT(I)=0.0000000;
  NODEC4(I) =0.0000000;
  MIUOUT(I) =0.0000000;
END;
ALARMF=0.0000000;
NODECF=0.0000000;
MIUOUTF=0.0000000;

/* READ INFORMATION OUT OF THE MEMORY PLANE FOR THE REGION
   IN WHICH THE STIMULUS HAS APPEARED */
CALL MIU(VISUALCHAR,ZERO,ZERO,ZERO,REGION,ALARMOUT(REGION),
  MIUOUT(REGION),NODEC4(REGION),PRINT);

/*IF NO DECISION IS OBTAINED FROM THIS MEMORY PLANE THEN ALL
   THE OTHER MEMORY PLANES SHOULD BE READ */
IF NODEC4(REGION) = 1 THEN DO;
  PUT EDIT(' "FROG" COULD NOT COME TO DECISION BASED ON WHAT IT',
    ' REMEMBERS ABOUT THE BUG PRESENTED IN THE GIVEN REGION.',
    ' SO IT WILL NOW SEARCH MEMORY FOR PREVIOUS',
    ' EXPERIENCES WITH THIS BUG IN OTHER REGIONS' )
    (SKIP(4),4 (X(32),A,SKIP));

  II=1;
  RECALC: IF II=REGION THEN GO TO INC;
  CALL MIU(VISUALCHAR,ZERO,ZERO,ZERO,II,ALARMOUT(II),
    MIUOUT(II),NODEC4(II),ZFROINT);
  INC: IF II=4 THEN GO TO FIN;
    II=II+1;
    GO TO RECALC;
  FIN: FND;
  ALARMF=MAX(ALARMOUT(1),ALARMOUT(2),ALARMOUT(3),ALARMOUT(4));
  NODECF=MIN(NODEC4(1),NODEC4(2),NODEC4(3),NODEC4(4));
  MIUOUTF=MAX(MIUOUT(1),MIUOUT(2),MIUOUT(3),MIUOUT(4));

/*THE FOLLOWING PORTION OF THE PROCEDURE SIMULATES THE DECISION
   UNIT */
  FEED=MAX(MIN(MAX(MIUOUTF,S(NODECF,SI(S(HUNGER,HUNGER))))),
    HUNGER,INNATESTIM),S(C(SI(C(HUNGER))),INNATESTIM));

  PUT EDIT('FINAL MEMORY INTEGRATION UNIT OUTPUT =',MIUOUTF,
    'ALARM OUTPUT OF MIU =',ALARMF)
    (SKIP(2),X(35),A,F(10,7),SKIP,X(43),A,F(10,7));

/* DECISION 1 */
IF ALARMF > FEED THEN
  DO;
    PUT EDIT('STATE OF ALARM =',ALARMF,
      'IMPULSE TO FEED =',MIUOUTF,
      'FROG HAS SEEN A PREDATOR.')
      (SKIP(2),X(47),A,F(10,7),SKIP,X(47),A,F(10,7),
        SKIP,X(47),A);
  
```

```

/* DECISION 2 */
  IF REGION = LOCATION THEN
    DO;
    PUT EDIT('THE PREDATOR IS CLOSE TO ITSELF SO IT DECIDES TO FLEE')
      (SKIP(2),X(34),A);
    LOCATION = 1;
    FATIGUE = MIN(FATIGUE+0.1000000,ONE);
  END;
  ELSE FATIGUE = MAX(FATIGUE-0.2000000,ZERO);
  COUNTER = COUNTER + 1;
  HUNGER = MIN(HUNGER+0.0500000,ONE);
  CALL DECAY;
  PUT EDIT(STARLINE,STARLINE)(SKIP(2),A,SKIP,A);
  GO TO BEGIN;
END;

/* DECISION 3 */
ELSE IF FEED < 0.5 THEN DO;
  PUT EDIT('IMPULSE OF "FROG" TO FEED',FEED,
    '"FROG" HAS JUST SEEN A BUG, BUT DECIDES NOT TO FEED',
    STARLINE,STARLINE)
    (SKIP(2),X(40),A,F(10,7),SKIP,X(33),A,SKIP(2),A,SKIP,A);
  FATIGUE=MAX(FATIGUE-0.2000000,ZERO);
  COUNTER=COUNTER+1;
  HUNGER=MIN(HUNGER+0.0500000,ONE);
  CALL DECAY;
  GO TO BEGIN;
END;

/* DECISION 4 */
ELSE IF FEED >= FATIGUE
  THEN DO;
  PUT EDIT('IMPULSE OF "FROG" TO FEED =',FEED,
    'FATIGUE LEVEL =',FATIGUE,
    '"FROG" DECIDES TO FEED',
    'RESULTING GOOD TASTE =',GOODTST,
    'RESULTING BAD TASTE =',BADTST,
    'PAIN INFLICTED =',PAIN,
    'FROG WILL NOW RECORD THE RESULTS OF THIS TRIAL IN ITS MEMORIES')
    (SKIP(2),X(40),A,F(10,7),SKIP,X(49),A,F(10,7),SKIP,
      X(49),A,3(SKIP,X(45),A,F(10,7)),SKIP,X(28),A);
  CALL DECAY;
  IF REGION=LOCATION THEN FATIGUE=MIN(FATIGUE+0.050000,ONE);
  ELSE DO;
    LOCATION=REGION;
    FATIGUE=MIN(FATIGUE+0.1000000,ONE);
  END;

  /*WRITE INFORMATION OF THIS TRIAL INTO MEMORIES */
  IF PAIN=0.0 THEN DO;
  CALL MEMORY(GTM,VISUALCHAR,GOODTST,GTMEMOUT,ZEROINT,REGION);
  CALL MEMORY(BTM,VISUALCHAR,BADTST,BTMEMOUT,ZEROINT,REGION);
  CALL ZEROOUT;
  PUT EDIT('"FROG"S MEMCRIES HAVE BEEN UPDATED',
    'AND "FROG" IS READY FOR ANOTHER STIMULUS',
    STARLINE,STARLINE)
    (SKIP(3),X(40),A,SKIP,X(40),A,SKIP,A,SKIP,A);
  IF GOODTST>=BADTST THEN HUNGER=MAX(HUNGER-0.1000000,ZERO);

```



```
ELSE HUNGER=MAX(HUNGER-0.0500000,ZERO);
END;
```

```
ELSE DO;
CALL MEMORY(PM,VISUALCHAR,PAIN,PMEMOUT,ZEROINT,REGION);
CALL ZFROOUT;
PUT EDIT('"FROG" HAS ENCOUNTERED A PREDATOR SO IT',
        'RETREATS TO ITS SAFE RESTING PLACE IN REGION 1',
        STARLINE,STARLINE)
        (SKIP(3),X(37),A,SKIP,X(37),A,SKIP(2),A,SKIP,A);
LOCATION=1;
HUNGER=MIN(HUNGER+0.0500000,ONE);
FATIGUE=MIN(FATIGUE+0.1000000,ONE);
END;
```

```
COUNTER=COUNTER+1;
GO TO BEGIN;
END;
```

```
/* DECISION 5 */
```

```
ELSE DO;
  PUT EDIT('IMPULSE OF "FROG" TO FEED =',FEED,
          'FATIGUE LEVEL =',FATIGUE,
          '"FROG" IS TOO TIRED TO MOVE',STARLINE,STARLINE)
          (SKIP(2),X(40),A,F(10,7),SKIP,X(40),A,F(10,7),
          SKIP,X(40),A,SKIP(2),A,SKIP,A);
  FATIGUE=MAX(FATIGUE-0.2000000,ZERO);
  COUNTER=COUNTER+1;
  HUNGER=MIN(HUNGER+0.0500000,ONE);
  CALL DECAY;
  GO TO BEGIN;
END;
```

```
MIU: PROCEDURE( VISIN, GTSTIN, BTSTIN, PAININ, MREGN,
                MALARM, MOUT, MNODEC, MPRNT );
```

```
/* SIMULATION OF A MIU.
```

THIS SUBROUTINE IS INVOKED BY A CALL STATEMENT. THE PARAMETER "VISIN" IS A FOUR-MEMBER ARRAY WHICH CONTAINS THE VISUAL CHARACTERISTICS OF THE STIMULUS; ITS MEMBERS HAVE ATTRIBUTES FIXED DECIMAL (9,7). "GTSTIN", "BTSTIN", AND "PAININ" ARE THE VALUES OF THE GOOD TASTE OF, THE BAD TASTE OF AND THE PAIN INFLICTED BY THE STIMULUS; EACH OF THE PARAMETERS HAVE THE ATTRIBUTES FIXED DECIMAL (9,7). "MREGN" SPECIFIES THE REGION IN WHICH THE STIMULUS HAS BEEN PRESENTED; IT HAS ATTRIBUTES BINARY FIXED. "MALARM", "MOUT" AND "MNODEC" ARE OUTPUTS ALARMOUT,MIUOUT AND NODEC4 OF THE MIU; THEIR ATTRIBUTES ARE FIXED DECIMAL (9,7). "MPRINT" IS THE PARAMETER WHICH TELLS THE "MIU" IF A MEMORY DUMP HAS BEEN REQUESTED; IT HAS ATTRIBUTES FIXED DECIMAL(1). IF "MPRNT"=1 A DUMP IS SUPPLIED; ELSE IF "MPRNT"=0, THEN THERE IS NO DUMP. */

```
DCL (VISIN(4),GTSTIN,BTSTIN,PAININ,MALARM,GTMISIN(4),BTMISIN(4),
    PMISIN(4),MOUT,MNODEC,MISOUT(4),ALARMISOUT(4),NODECISION(4))
```

```

DECIMAL FIXED (9,7),
MREGN BINARY FIXED,
MPRNT DECIMAL FIXED(1);

```

```

CALL MEMORY(GTM,VISIN,GTSTIN,GTMEMOUT,MPRNT,MREGN);
CALL MEMORY(BTM,VISIN,BTSTIN,BTMEMOUT,MPRNT,MREGN);
CALL MEMORY(PM ,VISIN,PAININ,PMEMOUT,MPRNT,MREGN);

```

```

GTMISIN(1)=GTMEMOUT(1);
BTMISIN(1)=BTMEMOUT(1);
PMISIN(1) =PMEMOUT(1);
CALL MIS(GTMISIN(1),PTMISIN(1),PMISIN(1),MISOUT(1),
        ALARMISOUT(1),NODECISION(1));
GTMISIN(2)=S(NODECISION(1),GTMEMOUT(2));
BTMISIN(2)=S(NODECISION(1),BTMEMOUT(2));
PMISIN(2) =S(NODECISION(1),PMEMOUT(2));
CALL MIS(GTMISIN(2),BTMISIN(2),PMISIN(2),MISOUT(2),
        ALARMISOUT(2),NODECISION(2));
NODECISION(2)=MIN(NODECISION(1),NODECISION(2));
GTMISIN(3)=S(NODECISION(2),GTMEMOUT(3));
BTMISIN(3)=S(NODECISION(2),BTMEMOUT(3));
PMISIN(3) =S(NODECISION(2),PMEMOUT(3));
CALL MIS(GTMISIN(3),BTMISIN(3),PMISIN(3),MISOUT(3),
        ALARMISOUT(3),NODECISION(3));
NODECISION(3)=MIN(NODECISION(2),NODECISION(3));
GTMISIN(4)=S(NODECISION(3),GTMEMOUT(4));
BTMISIN(4)=S(NODECISION(3),BTMEMOUT(4));
PMISIN(4) =S(NODECISION(3),PMEMOUT(4));
CALL MIS(GTMISIN(4),BTMISIN(4),PMISIN(4),MISOUT(4),
        ALARMISOUT(4),NODECISION(4));
NODECISION(4)=MIN(NODECISION(3),NODECISION(4));

MALARM=MAX(ALARMISOUT(1),ALARMISOUT(2),ALARMISOUT(3),
        ALARMISOUT(4));
MOUT=MAX(MISOUT(1),MISOUT(2),MISOUT(3),MISOUT(4));
MNODEC=NODECISION(4);

```

```

IF MPRNT=1 THEN PUT PAGE;
PUT EDIT('OUTPUTS OF LEVELS OF GOOD TASTE MEMORY FOR REGION ',
        MREGN,(GTMEMOUT(I) DO I=1 TO 4),
        'OUTPUTS OF LEVELS OF BAD TASTE MEMORY FOR REGION ',
        MREGN,(BTMEMOUT(I) DO I=1 TO 4),
        'OUTPUTS OF LEVELS OF PREDATOR MEMORY FOR REGION ',
        MREGN,(PMEMOUT(I) DO I=1 TO 4))
        (SKIP(3),X(10),A,F(1),X(5),4 F(12,7),
        SKIP(3),X(10),A,F(1),X(6),4 F(12,7),
        SKIP(3),X(10),A,F(1),X(7),4 F(12,7));

PUT EDIT('OUTPUTS OF MEMORY INTEGRATION SUBUNITS OF REGION ',
        MREGN,
        'LEVEL 1 SUBUNIT:',MISOUT(1),ALARMISOUT(1),
        NODECISION(1),
        'LEVEL 2 SUBUNIT:',MISOUT(2),ALARMISOUT(2),
        NODECISION(2),
        'LEVEL 3 SUBUNIT:',MISOUT(3),ALARMISOUT(3),
        NODECISION(3),
        'LEVEL 4 SUBUNIT:',MISOUT(4),ALARMISOUT(4),
        NODECISION(4),
        'OUTPUT OF MIU FOR REGION ',MREGN,':',MOUT,

```

```

      'ALARM OUTPUT OF MIU FOR REGION ',MREGN,':',MALARM)
      (SKIP(2),X(35),A,F(1),4(SKIP,X(37),A,3 F(10,7))),
      SKIP,X(41),A,F(1),A,F(10,7),
      SKIP,X(38),A,F(1),A,F(10,7));
RETURN;
END MIU;

```

```

MIS: PROCEDURE(GTLEVOUT,BTLEVOUT,PLEVOUT,MISOUT,ALARMIS,NODEC);

```

```

/* SIMULATION OF MEMORY INTEGRATION SUBUNIT.
ALL SIGNAL NAMES CORRESPOND TO THOSE IN THE LOGIC DIAGRAM
GIVEN IN FIGURE 3.5. MIS IS INVOKED BY THE USE OF A CALL
STATEMENT. THE SIX FORMAL PARAMETERS HAVE THE ATTRIBUTES
FIXED DECIMAL(9,7). THE PARAMETERS "GTLEVOUT", "BTLEVOUT"
AND "PLEVOUT" CORRESPONDS RESPECTIVELY TO THE OUTPUTS FROM
THE GTM, THE BTM AND THE PM. THE PARAMETER "MISOUT" IS THE
NUMERICAL VALUE OF THE DECISION IF A RUG IS SENSED. "ALARMIS"
IS THE ALARM OUTPUT IF A PREDATOR IS SENSED; "NODEC" IS THE
PARAMETER WHICH TELLS IF A DECISION HAS BEEN MADE OR NOT. */

```

```

DCL (GTLEVOUT,BTLEVOUT,PLEVOUT,MAX1,GTE,BTE,PE,CONTROLG,CONTROLB,
CONTROLP,MISOUT,ALARMIS,NODEC,SIGG,SIGB,SIGP)
DECIMAL FIXED (9,7);

```

```

MAX1=MAX(GTLEVOUT,BTLEVOUT,PLEVOUT);
GTE=F(MAX1,GTLEVOUT);
BTE=E(MAX1,BTLEVOUT);
PE =E(MAX1,PLEVOUT);
SIGG=C(SI(GTE));
SIGB=C(SI(BTE));
SIGP=C(SI(PE));
CONTROLG=MIN(SIGB,SIGP,S(GTE,GTE));
CONTROLB=MIN(SIGG,SIGP,S(BTE,BTE));
CONTROLP=MIN(SIGG,SIGB,S(PE,PE));
MISOUT=MAX(S(CONTROLG,GTE),(S(CONTROLB,C(BTE))));
ALARMIS=S(CONTROLP,PE);
NODEC=C(SI(MAX(CONTROLG,CONTROLB,CONTROLP)));
RETURN;
END MIS;

```

```

MEMORY: PROCEDURE(CODE,VISUAL,NONVISIN,MEMOUT,PRNT,REGN);

```

```

/* SIMULATION OF A MEMORY; THAT IS, A GTM OR A BTM OR A PM.
THIS SUBROUTINE IS INVOKED BY A CALL STATEMENT.
THE PARAMETER "CODE" IS A CODE WHICH SPECIFIES THE TYPE OF
MEMORY. "CODE"=1 REFERS TO A GTM. "CODE"=2 REFERS TO A BTM.
"CODE"=3 REFERS TO A PM. THE ATTRIBUTES OF "CODE" ARE FIXED
BINARY. THE PARAMETER "VISUAL" IS A FOUR-MEMBER ARRAY WHICH
CONTAINS THE VISUAL CHARACTERISTICS OF THE STIMULUS BEING
INPUT TO THE MEMORY; IT HAS ATTRIBUTES FIXED DECIMAL (9,7).
"MEMOUT" IS A FOUR-MEMBER ARRAY WHICH IS USED TO RETURN THE

```


OUTPUTS OF THE FOUR LEVELS; ITS MEMBERS ALSO HAVE ATTRIBUTES FIXED DECIMAL (9,7). "PRNT" IS THE PARAMETER WHICH TELLS "MEMORY" IF A DUMP HAS BEEN REQUESTED; IT HAS ATTRIBUTES FIXED DECIMAL (1). IF "PRNT"=1, A DUMP IS SUPPLIED; ELSE IF "PRNT"=0, THEN THERE IS NO DUMP. "REGN" SPECIFIES THE REGION IN WHICH THE STIMULUS HAS APPEARED; ITS PARAMETERS ARE FIXED BINARY.

THE SECTION OF CODE SIMULATING LEVEL 1 IS DOCUMENTED; THE OTHER LEVELS ARE SIMULATED IN A SIMILAR FASHION.

IN ORDER TO UNDERSTAND THE MEMORY DUMPS:
THE FOLLOWING ARE THE COMBINATORIAL INPUTS FOR THE FOUR LEVELS:

```

      LEVEL 1
COMBINATION 1=VISUAL(1)
COMBINATION 2=VISUAL(2)
COMBINATION 3=VISUAL(3)
COMBINATION 4=VISUAL(4)
      LEVEL 2
COMBINATION 1=VISUAL(1),VISUAL(2)
COMBINATION 2=VISUAL(1),VISUAL(3)
COMBINATION 3=VISUAL(1),VISUAL(4)
COMBINATION 4=VISUAL(2),VISUAL(3)
COMBINATION 5=VISUAL(2),VISUAL(4)
COMBINATION 6=VISUAL(3),VISUAL(4)
      LEVEL 3
COMBINATION 1=VISUAL(1),VISUAL(2),VISUAL(3)
COMBINATION 2=VISUAL(1),VISUAL(2),VISUAL(4)
COMBINATION 3=VISUAL(1),VISUAL(3),VISUAL(4)
COMBINATION 4=VISUAL(2),VISUAL(3),VISUAL(4)
      LEVEL 4
COMBINATION 1=VISUAL(1),VISUAL(2),VISUAL(3),VISUAL(4)  */

```

```

DCL CODE FIXED BINARY, /* SPACIFIES WHETHER A GTM, A BTM OR A PM
IS REFERRED TO */
(VISUAL(4), /* THE FOUR BASIC VISUAL CHARACTERISTICS */
NONVISIN, /* INPUT C AND OUTPUT E FOR T*-ELEMENTS */
MEMOUT(4), /* THE 4-TUPLE OUTPUT E FOR T*-ELEMENTS */
OUTPUT(4), /* THE OUTPUTS OF ALL THE LEVELS */
L1VISINPT(4,4) INIT((16) 0.0000000),
/* THE INPUTS TO THE FOUR CHAINS IN LEVEL 1 */
L2VISINPT(6,4) INIT((24) 0.0000000),
/* THE INPUTS TO THE SIX CHAINS IN LEVEL 2 */
L3VISINPT(4,4) INIT((16) 0.0000000),
/* THE INPUTS TO THE FOUR CHAINS IN LEVEL 3 */
L4VISINPT(4) INIT((4) 0.0000000),
/* THE INPUTS TO THE CHAIN IN LEVEL 4 */
/* THE ABOVE VARIABLES ARE SO DIMENSIONED THAT IT IS POSSIBLE
TO SPECIFY FOUR INPUTS TO EACH T*-ELEMENTS IN EACH LEVEL.
SINCE A LEVEL N T*-ELEMENT NEED N INPUTS (N=1,2,3 OR 4),
SOME VARIABLES ARE NOT USED */
LEV1OUT, /* THE OUTPUT OF LEVEL 1 */
LEV2OUT, /* THE OUTPUT OF LEVEL 2 */
LEV3OUT, /* THE OUTPUT OF LEVEL 3 */
LEV4OUT /* THE OUTPUT OF LEVEL 4 */ ) DECIMAL FIXED(9,7),
PRNT /* PARAMETER WHICH TELLS IF A MEMORY DUMP IS
REQUESTED */ FIXED DECIMAL (1),
REGN /* THE NUMBER OF THE REGION IN WHICH THE STIMULUS

```

```

HAS APPEARED */ FIXED BINARY,
/* THE FOLLOWING TWO VARIABLES ARE ASSOCIATED WITH THE DECAY
FUNCTION */
TEMP(3) /* SCRATCH AREA */ FIXED BINARY,
CNTR(3) /* TEMPORARY MARKER, ONE FOR EACH TYPE OF MEMORY,
TO SEE HOW MANY T*-ELEMENTS IN A CHAIN HAVE BEEN
WRITTEN INTO */ BIT (1),

```

```

/* THE FOLLOWING FOUR VARIABLES ARE FOR TEMPORARY STORAGE
OF THE OUTPUT STATUS OF EACH T*-ELEMENT IN THE BTM IN
IN CASE A DUMP IS REQUIRED */
(LASTOUT1(4,3,4,3) INIT((144) 0.0000000),
LASTOUT2(4,3,6,3) INIT((216) 0.0000000),
LASTOUT3(4,3,4,3) INIT((144) 0.0000000),
LASTOUT4(4,3,3) INIT((36) 0.0000000)) STATIC DECIMAL FIXED (9,7),

```

```

/* THE FOLLWING LIST OF STORED VALUES FOR ALL THE MEM-ELEMENTS
AND THE STORED VALUE AND FRACTION MULTIPLIERS OF ALL THE
T-ELEMENTS */
((L1TSTOVAL(4,3,4,3),L3TSTOVAL(4,3,4,3),L1TPFRAC(4,3,4,3),
L3TPFRAC(4,3,4,3)) INITIAL ((144) 0.0000000),
(L1MEM(4,3,4,3,6),L3MEM(4,3,4,3,6))
INITIAL ((864) 0.0000000),
(L2TSTOVAL(4,3,6,3),L2TPFRAC(4,3,6,3))
INITIAL ((216) 0.0000000),
L2MEM(4,3,6,3,6) INITIAL((1296) 0.0000000),
(L4TSTOVAL(4,3,3),L4TPFRAC(4,3,3)) INITIAL((36) 0.0000000),
L4MEM(4,3,3,6) INITIAL((216) 0.0000000))
STATIC DECIMAL FIXED (9,7);

```

```

/* LEVEL 1 */
L1VISINPT(*,1)=VISUAL; /* SET UP VISUAL INPUTS FOR EACH CHAIN */
OUTPUT(1)=ZERO; /* INITIALIZE LEVEL 1 OUTPUT TO ZERO */
DO I=1 TO 4; /* PARAMETER "I" IS THE NUMBER OF THE CHAIN */
NBHOOD=ONE; /*TURN ON SIGNAL "NBHOOD" AT BEGINNING OF
EACH CHAIN */
CNTR(CODE)='0'B; /* INITIALIZE TEMPORARY MARKER */
DO J=1 TO 3; /* PARAMETER "J" IS THE NUMBER OF THE T*-ELEMENT
IN CHAIN "I" */
CALL TSTAR(NONVISIN,ONEINT,L1VISINPT(I,*),
L1MEM(REGN,CODE,I,J,*),L1TSTOVAL(REGN,CODE,I,J),
L1TPFRAC(REGN,CODE,I,J),LEVIOUT,NBHOOD);
IF LEVIOUT=ZERO THEN USE1(REGN,CODE,I,J)='1'B; /* SET UP
"USE" ARRAY WHICH WILL INDICATE WHICH T*-ELEMENTS
OUTPUT SOME NONZERO VALUE DURING A READ CYCLE */
ELSE GO TO L1; /* IF THIS T*-ELEMENT HAS NOT OUTPUT SOME
NON-ZERO VALUE DURING A READ CYCLE */
OUTPUT(1)=MAX(OUTPUT(1),LEVIOUT-DECAYNO1(REGN,CODE,I,J));
/* IF THIS T*-ELEMENT HAS OUTPUT SOME NON-ZERO VALUE,
THEN DECAY IT BY THE CORRECT AMOUNT, AND KEEP A
RUNNING MAXIMUM TO FIND THE LARGEST T*-ELEMENT OUTPUT
FOR LEVEL 1 */
L1: LASTOUT1(REGN,CODE,I,J)=MAX(L1TSTOVAL(REGN,CODE,I,J)
*L1TPFRAC(REGN,CODE,I,J)-DECAYNO1(REGN,CODE,I,J),ZERO);
/* SET UP THE OUTPUT STATUS OF THIS T*-ELEMENT IN CASE
A DUMP HAS BEEN REQUESTED */
IF ~CNTR(CODE) & L1TSTOVAL(REGN,CODE,I,J)=ZERO THEN DO;
/* WHEN FIND THE FIRST T*-ELEMENT THE CHAIN THAT HAS NOT
BEEN WRITTEN INTO THEN SET "MEMCNT(CODE)" EQUAL TO

```

THE NUMBER OF THE T*-ELEMENT IMMEDIATELY TO ITS LEFT.
 THIS NUMBER (J-1) WILL TELL "DECAY" HOW MANY T*-
 ELEMENT IN EACH CHAIN HAVE BEEN WRITTEN INTO AND,
 HENCE, HOW MANY T*-ELEMENTS IN EACH ROW TO DECAY */
 CNTR(CODE)='1'B; /* ONCE A VALUE HAS BEEN SET FOR
 "MEMCNT(CODE)" IN THIS PASS THROUGH THE CHAIN THEN
 TURN ON THE SIGNAL THAT WILL PROHIBIT ENTRY INTO
 THIS DO-GROUP UNTIL THE NEXT ITERATION OF "I" */

TEMP(CODE)=J-1;

MEMCNT(CODE)=MAX(MEMCNT(CODE),TEMP(CODE));

END;

END;

END;

/* LEVEL 2 */

L2VISINPT(1,1)=VISUAL(1);

L2VISINPT(1,2)=VISUAL(2);

L2VISINPT(2,1)=VISUAL(1);

L2VISINPT(2,2)=VISUAL(3);

L2VISINPT(3,1)=VISUAL(1);

L2VISINPT(3,2)=VISUAL(4);

L2VISINPT(4,1)=VISUAL(2);

L2VISINPT(4,2)=VISUAL(3);

L2VISINPT(5,1)=VISUAL(2);

L2VISINPT(5,2)=VISUAL(4);

L2VISINPT(6,1)=VISUAL(3);

L2VISINPT(6,2)=VISUAL(4);

OUTPUT(2)=ZERO;

DO I=1 TO 6;

 NBHOOD=ONE;

 DO J=1 TO 3;

 CALL TSTAR(NJNVISIN,TWOINT,L2VISINPT(I,*),

 L2MEM(REGN,CODE,I,J,*),L2TSTOVAL(REGN,CODE,I,J),

 L2TPFRAC(REGN,CODE,I,J),LEV2OUT,NBHOOD);

 IF LEV2OUT /=ZERO THEN USE2(REGN,CODE,I,J)='1'B;

 ELSE GO TO L2;

 OUTPUT(2)=MAX(OUTPUT(2),LEV2OUT-DECAYNO2(REGN,CODE,I,J));

 L2: LASTOUT2(REGN,CODE,I,J)=MAX(L2TSTOVAL(REGN,CODE,I,J)*

 L2TPFRAC(REGN,CODE,I,J)-DECAYNO2(REGN,CODE,I,J),ZERO);

 END;

END;

/* LEVEL 3 */

L3VISINPT(1,1)=VISUAL(1);

L3VISINPT(1,2)=VISUAL(2);

L3VISINPT(1,3)=VISUAL(3);

L3VISINPT(2,1)=VISUAL(1);

L3VISINPT(2,2)=VISUAL(2);

L3VISINPT(2,3)=VISUAL(4);

L3VISINPT(3,1)=VISUAL(1);

```

L3VISINPT(3,2)=VISUAL(3);
L3VISINPT(3,3)=VISUAL(4);

L3VISINPT(4,1)=VISUAL(2);
L3VISINPT(4,2)=VISUAL(3);
L3VISINPT(4,3)=VISUAL(4);

OUTPUT(3)=ZERO;
DO I=1 TO 4;
  NBHOOD = ONE;
  DO J=1 TO 3;
    CALL TSTAR(NONVISIN,THREEINT,L3VISINPT(I,*),
               L3MEM(REGN,CODE,I,J,*),L3TSTOVAL(REGN,CODE,I,J),
               L3TPFRAC(REGN,CODE,I,J),LEV3OUT,NBHOOD);
    IF LEV3OUT /=ZERO THEN USE3(REGN,CODE,I,J)='1'B;
    ELSE GO TO L3;
    OUTPUT(3)=MAX(OUTPUT(3),LEV3OUT-DECAYNO3(REGN,CODE,I,J));
    L3: LASTOUT3(REGN,CODE,I,J)=MAX(L3TSTOVAL(REGN,CODE,I,J)*
                                     L3TPFRAC(REGN,CODE,I,J)-DECAYNO3(REGN,CODE,I,J),ZERO);
  END;
END;

/* LEVEL 4 */
L4VISINPT=VISUAL;

OUTPUT(4)=ZERO;
NBHOOD=ONE;
DO I=1 TO 3;
  CALL TSTAR(NONVISIN,FOURINT,L4VISINPT,
               L4MEM(REGN,CCODE,I,*),L4TSTOVAL(REGN,CODE,I),
               L4TPFRAC(REGN,CODE,I),LEV4OUT,NBHOOD);
  IF LEV4OUT /= ZERO THEN USE4(REGN,CODE,I)='1'B;
  ELSE GO TO L4;
  OUTPUT(4)=MAX(OUTPUT(4),LEV4OUT-DECAYNO4(REGN,CODE,I));
  L4: LASTOUT4(REGN,CODE,I)=MAX(L4TSTOVAL(REGN,CODE,I)*
                                L4TPFRAC(REGN,CODE,I)-DECAYNO4(REGN,CODE,I),ZERO);
END;

/* SEND OUTPUTS OF ALL LEVELS OF MEMORY BACK TO "FROG" */
MEMOUT=OUTPUT;

IF PRNT=1 THEN DO;
  DO K=1 TO 4;
    IF CODE=1 THEN DO;
      PUT EDIT('OUTPUT STATUS OF EACH T* ELEMENT IN REGION ',K,
               ' OF GOOD TASTE MEMORY IS NOW:', 'T* ELEMENT NUMBER',
               '1','2','3')
              (PAGE,SKIP(4),X(10),A,F(1),A,SKIP(2),X(64),A,
               SKIP,X(52),A,X(19),A,X(19),A,SKIP);
    END;
    ELSE IF CODE=2 THEN DO;
      PUT EDIT('OUTPUT STATUS OF EACH T* ELEMENT IN REGION ',K,
               ' OF BAD TASTE MEMORY IS NOW:', 'T* ELEMENT NUMBER',
               '1','2','3')
              (PAGE,SKIP(4),X(10),A,F(1),A,SKIP(2),X(64),A,
               SKIP,X(52),A,X(19),A,X(19),A,SKIP);
    END;
  ELSE DO;

```



```

      PUT EDIT('OUTPUT STATUS OF EACH T* ELEMENT IN REGION ',K,
        ' OF PREDATOR MEMORY IS NOW:', 'T* ELEMENT NUMBER',
        '1','2','3')
      (PAGE,SKIP(4),X(10),A,F(1),A,SKIP(2),X(64),A,
        SKIP,X(52),A,X(19),A,X(19),A,SKIP);
END;
DO I=1 TO 4;
  PUT EDIT('LEVEL 1',(LASTOUT1(K,CODE,I,J) DO J=1 TO 3),
    'COMBINATION ',I)
    (SKIP(2),X(25),A,X(15),F(10,7),X(10),
      F(10,7),X(10),F(10,7),SKIP,X(25),A,F(1));
END;
DO I=1 TO 6;
  PUT EDIT('LEVEL 2',(LASTOUT2(K,CODE,I,J) DO J=1 TO 3),
    'COMBINATION ',I)
    (SKIP(2),X(25),A,X(15),F(10,7),X(10),
      F(10,7),X(10),F(10,7),SKIP,X(25),A,F(1));
END;
DO I=1 TO 4;
  PUT EDIT('LEVEL 3',(LASTOUT3(K,CODE,I,J) DO J=1 TO 3),
    'COMBINATION ',I)
    (SKIP(2),X(25),A,X(15),F(10,7),X(10),
      F(10,7),X(10),F(10,7),SKIP,X(25),A,F(1));
END;
PUT EDIT('LEVEL 4',(LASTOUT4(K,CODE,I) DO I=1 TO 3),
  'COMBINATION 1')
  (SKIP(2),X(25),A,X(15),F(10,7),
    X(10),F(10,7),X(10),F(10,7),SKIP,X(25),A);
END;
END;
RETURN;
END MEMORY;

```

```

TSTAR: PROCEDURE(NONVISUAL,TLEVEL,VISSTIM,MEMSTOVAL,TSTOVAL,
  TPRODFRACTION,TSTAROUT,NBHOOD);

```

```

/* SIMULATION OF A T*-ELEMENT.

```

ALL SIGNAL NAMES CORRESPOND TO THOSE IN THE LOGIC DRAWING. TSTAR IS INVOKED BY USE OF A CALL STATEMENT. ALL OF THE FORMAL PARAMETERS EXCEPT "TLEVEL", HAVE ATTRIBUTES FIXED DECIMAL (9,7). "TLEVEL" IS BINARY FIXED. "NONVISUAL" IS THE NONVISUAL INPUT TO THE T*-ELEMENT. THE T*-ELEMENTS VARY IN THE NUMBER OF MEM ELEMENTS. THE PARAMETERS "VISSTIM" AND "MEMSTOVAL" ARE ARRAYS WHICH ARE DIMENSIONED SUCH THAT THEY CAN COPE WITH THE MAXIMUM VARIATION. "VISSTIM" IS A FOUR-MEMBER ARRAY WHICH CORRESPOND TO MAXIMUM OF FOUR VISUAL INPUTS TO THE T*-ELEMENT. IF THE T*-ELEMENT HAS LESS THAN FOUR INPUTS, THE HIGHER INDEXED MEMBERS OF THE ARRAY ARE SET TO ZERO. "MEMSTOVAL" IS A SIX MEMBER ARRAY WHICH CONTAINS THE STORED VALUES OF A MAXIMUM OF SIX MEM-ELEMENTS CONTAINED IN THE T*-ELEMENT. IF THE T*-ELEMENT CONTAIN LESS THAN SIX MEM-ELEMENTS THE CORRESPONDING MEMBERS OF THE ARRAY ARE SET TO ZERO. THE PARAMETERS "TSTOVAL" AND "TPRODFRACTION" ARE THE STORED VALUE AND FRACTION MULTIPLIER OF THE T-ELEMENT CONTAINED IN THE T*-ELEMENT. "TSTAROUT" CORRESPONDS TO THE

D OUTPUT, THE MAIN OUTPUT, "NBHOOD" CORRESPONDS TO THE C INPUT WHEN TSTAR IS INVOKED, IS MODIFIED, AND RETURNED AS THE E OUTPUT. "TLEVEL" TELLS WHICH LEVEL THIS T*-ELEMENT IS IN. */

```

DCL (NONVISUAL,TSTOVAL,TPRCDFRACTION,MODNONVISUAL,SIG,TIN,
     TSTAROUT,NBHOOD,PIN1,PIN2,MODPIN1,MODPIN2,MEMSTOVAL(6),
     MEMIN(6),VISSTIM(4),ECUT(6)) DECIMAL FIXED (9,7),
     TLEVEL BINARY FIXED;
MODNONVISUAL=S(NONVISUAL,NONVISUAL);
SIG=MAX(MEMSTOVAL(5),MIN(C(MEMSTOVAL(5)),
     NBHOOD,MODNONVISUAL));
PIN1=ONE;
DO NN=1 TO TLEVEL;
  MEMIN(NN)=S(SIG,VISSTIM(NN));
  EOUT(NN)=E(MEMIN(NN),MEM(MEMIN(NN),MEMSTOVAL(NN)));
  PIN1=MIN(PIN1,EOUT(NN));
END;
MEMIN(5)=S(SIG,MODNONVISUAL);
EOUT(5)=E(MEMIN(5),MEM(MEMIN(5),MEMSTOVAL(5)));
MEMIN(6)=S(MIN(PIN1,C(SI(EOUT(5)))),MEMIN(5));
EOUT(6)=E(MEMIN(6),MEM(MEMIN(6),MEMSTOVAL(6)));

PIN2=S(MAX(EOUT(5),EOUT(6)),MAX(MEMSTOVAL(5),MEMSTOVAL(6)));
MODPIN1=SI(PIN1);
MODPIN2=SI(PIN2);
TIN=S(MIN(MODPIN1,MODPIN2),P(PIN1,PIN2));
TSTAROUT=S(MIN(MODPIN1,C(MODPIN2)),
     T(TLEVEL,TIN,TSTOVAL,TPRODFRACTION));
NBHOOD=MIN(NBHOOD,C(SI(TIN)));
RETURN;
END TSTAR;

```

T: PROCEDURE(LEVELNO,INPUT,STOVAL,PRODFRACTION) DECIMAL FIXED (9,7);

/* SIMULATION OF THE T-ELEMENT.

THIS FUNCTION IS INVOKED BY THE USE OF
 "T(LEVEL NUMBER, INPUT TO T-ELEMENT, STORED VALUE OF T-
 ELEMENT, VALUE OF FRACTION MULTIPLIER)"
 IN AN EXPRESSION. THE FIRST ARGUMENT HAS THE ATTRIBUTES
 FIXED DECIMAL (1), THE OTHER THREE HAVE THE ATTRIBUTES FIXED
 DECIMAL (9,7); THE VALUE RETURNED TO THE POINT OF INVOCATION
 HAS THE ATTRIBUTES FIXED DECIMAL (9,7). THE PARAMETER
 "LEVELNO" CORRESPONDS TO THE LEVEL NUMBER; THE PARAMETER
 "INPUT" IS THE INPUT TO THE T-ELEMENT; THE PARAMETER
 "PRODFRACTION" IS THE VALUE OF THE FRACTION MULTIPLIER.
 SINCE THE OUTPUTS CONSTANTLY, IF "INPUT"=ZERO THEN THE CURRENT
 OUTPUT STATUS IS RETURNED. THE "STOVAL" SETS TO ANY NON-ZERO
 VALUE ON "INPUT" WHICH ALSO STEPS "PRODFRACTION" BY 1/N
 (IF POSSIBLE) AND THE NEW OUTPUT STATUS RETURNED. */

```

DCL (INPUT,STOVAL,PRODFRACTION) DECIMAL FIXED (9,7),
     LEVELNO BINARY FIXED;
IF INPUT=ZERO THEN RETURN(STOVAL*PRODFRACTION);
ELSE DO;

```

```

STOVAL=INPUT;
CALL COMPUTE(PRODFRACTION,LEVELNO);
RETURN(PRODFRACTION*STOVAL);
END;
END T;

```

MEM: PROCEDURE(INPUT,STOREDVALUE) DECIMAL FIXED (9,7) ;

```

/* SIMULATION OF THE MEM-ELEMENT.
THIS FUNCTION IS INVOKED BY THE USE OF
"MEM(INPUT TO MEM-ELEMENT, STORED VALUE OF MEM-ELEMENT)"
IN AN EXPRESSION. BOTH ARGUMENTS HAVE THE ATTRIBUTES FIXED
DECIMAL (9,7), AS DOES THE VALUE RETURNED TO THE POINT OF
INVOCATION. THE PARAMETER "INPUT" CORRESPOND TO THE INPUT
TO THE MEM-ELEMENT, "STOREDVALUE" CORRESPOND TO THE VALUE
STORED WITHIN THE MEM-ELEMENT. IF "INPUT" IS ZERO, THEN THE
OUTPUT STATUS OF THE MEM-ELEMENT IS RETURNED. OTHERWISE, IF
THE MEM-ELEMENT IS "UNSET", IT IS "SET" TO THE VALUE OF THE
"INPUT". "STOREDVALUE" IS THEN RETURNED TO THE POINT OF INVO-
CATION. */

```

```

DCL (INPUT,STOREDVALUE) DECIMAL FIXED (9,7);
IF STOREDVALUE=ZERO THEN STOREDVALUE=INPUT;
RETURN(STOREDVALUE);
END MEM;

```

E: PROCEDURE(EXTINPUT,TINPLT) DECIMAL FIXED (9,7);

```

/* SIMULATION OF E-ELEMENT.
THIS FUNCTION IS INVOKED BY THE USE OF "E(X2 INPUT,X1 INPUT)"
IN AN EXPRESSION. BOTH ARGUMENTS MUST HAVE THE ATTRIBUTES
FIXED DECIMAL (9,7); THE VALUE RETURNED TO THE POINT OF
INVOCATION HAS SIMILAR ATTRIBUTES. THE PARAMETER "EXTINPUT"
CORRESPONDS TO THE X2 INPUT TO THE E-ELEMENT; THE PARAMETER
"TINPUT" CORRESPOND TO THE X1 INPUT. IF THE ABSOLUTE VALUE
OF TINPUT - EXTINPUT <= 0.5, THEN THE VALUE OF TINPUT IS RE-
TURNED TO THE POINT OF INVOCATION; OTHER WISE ZERO IS RE-
TURNED. */

```

```

DCL (EXTINPUT,TINPUT,NBHD INIT(0.0500000)) DECIMAL FIXED (9,7);
IF ABS(TINPUT-EXTINPUT)<=NBHD THEN RETURN(TINPUT);
ELSE RETURN(ZERO);
END E;

```

P: PROCEDURE(INPUT1,INPUT2) DECIMAL FIXED (9,7);

```

/* SIMULATION OF P-ELEMENT.
THIS FUNCTION IS INVOKED BY THE USE OF "P(INPUT,INPUT)" IN
AN EXPRESSION. BOTH ARGUMENTS MUST HAVE THE ATTRIBUTES FIXED
DECIMAL (9,7); THE VALUE RETURNED TO THE POINT OF INVOCATION

```

HAS SIMILAR ATTRIBUTES. P FORMS THE PRODUCT OF THE PARAMETERS "INPUT1" AND "INPUT2" AND RETURNS TO THE POINT OF INVOCATION. */

```
DCL (INPUT1,INPUT2) DECIMAL FIXED (9,7);
RETURN(INPUT1*INPUT2);
END P;
```

S: PROCEDURE(SIGNAL,GATE) DECIMAL FIXED (9,7);

/* SIMULATION OF S-ELEMENT.

THIS FUNCTION IS INVOKED BY USE OF "S(X2 INPUT,X1 INPUT)" IN AN EXPRESSION. THE ARGUMENTS MUST HAVE THE ATTRIBUTES FIXED DECIMAL (9,7); THE ATTRIBUTES RETURNED TO THE POINT OF INVOCATION HAS SIMILAR ATTRIBUTES. THE PARAMETER "SIGNAL" CORRESPONDS TO THE X2 INPUT TO THE S-ELEMENT; THE PARAMETER "GATE" CORRESPONDS TO THE X1 INPUT. IF THE VALUE OF "SIGNAL" IS ≥ 0.5 THEN THE VALUE OF "GATE" IS RETURNED TO THE POINT OF INVOCATION; IF IT IS < 0.5 THEN ZERO IS RETURNED. */

```
DCL (SIGNAL,GATE,PFIVE INIT(0.5000000)) DECIMAL FIXED (9,7);
IF SIGNAL<PFIVE THEN RETURN(ZERO);
ELSE RETURN(GATE);
END S;
```

SI: PROCEDURE(SIGNAL) DECIMAL FIXED (9,7);

/* SIMULATION OF SI-ELEMENT.

INVOKED BY THE USE OF "SI(INPUT)" IN AN EXPRESSION, THIS FUNCTION RETURNS A VALUE TO THE POINT OF INVOCATION WITH ATTRIBUTES FIXED DECIMAL (9,7); THE ARGUMENT MUST HAVE THE SAME ATTRIBUTES. THE PARAMETER "SIGNAL" CORRESPONDS TO THE INPUT TO THE SI-ELEMENT. IF "SIGNAL" IS > 0 , THEN THE VALUE 1.0 IS RETURNED TO THE POINT OF INVOCATION; IF "SIGNAL" = 0, THEN ZERO IS RETURNED. */

```
DCL SIGNAL DECIMAL FIXED (9,7);
IF SIGNAL>ZERO THEN RETURN(ONE);
ELSE RETURN(ZERO);
END SI;
```

C: PROCEDURE(NUMBER) DECIMAL FIXED (9,7);

/* SIMULATION IF C-ELEMENT.

THIS FUNCTION IS INVOKED BY THE USE OF "C(INPUT TO C-ELEMENT)" IN AN EXPRESSION. THE ARGUMENTS MUST HAVE THE ATTRIBUTES FIXED DECIMAL (9,7); THE FUNCTION RETURNS A VALUE WITH THE SAME ATTRIBUTES. THE PARAMETER "NUMBER" CORRESPONDS TO THE INPUT TO THE C-ELEMENT; THE CONTINUOUS LOGIC COMPLEMENT 1-INPUT IS RETURNED TO THE POINT OF INVOCATION. */


```

DO K=1 TO 4;
  IF  $\neg$ USE3(J,I,K,L) THEN
    DECAYNO3(J,I,K,L)=DECAYNO3(J,I,K,L)+FACTOR;
  ELSE DECAYNO3(J,I,K,L)=DECAYNO3(J,I,K,L)-FACTOR;
END;

IF  $\neg$ USE4(J,I,L) THEN
  DECAYNO4(J,I,L)=DECAYNO4(J,I,L)+FACTOR;
ELSE DECAYNO4(J,I,L)=DECAYNO4(J,I,L)-FACTOR;

END;
END;
OUT: END LOOP;
ZEROOUT: ENTRY;
USE1='0'B;
USE2='0'B;
USE3='0'B;
USE4='0'B;
RETURN;
END DECAY;

```

COMPUTE: PROCEDURE(FRACTION ,LEVEL);

/* COMPUTE IS A SUBROUTINE WHICH SIMULATES THE STEPPING OF THE FRACTION MULTIPLIER OF THE T-ELEMENT OUTPUT. IT IS INVOKED BY THE STATEMENT "CALL COMPUTE(FRACTION,LEVEL NUMBER)"; THE PARAMETER "FRACTION" HAS ATTRIBUTES FIXED DECIMAL(9,7); THE PARAMETER "LEVEL" HAS ATTRIBUTES FIXED DECIMAL(1). "FRACTION" CORRESPONDS TO THE VALUE OF THE FRACTION MULTIPLIER, AND "LEVEL" TO THE NUMBER OF THE LEVEL WHICH CONTAINS THE T-ELEMENT WHOSE FRACTIONS MULTIPLIER IS BEING STEPPED. IF THE VALUE OF "FRACTION" IS 1.0, THEN THE T-ELEMENT WITH WHICH IT IS ASSOCIATED HAS REACHED FULL POTENTIAL SO THE VALUE OF "FRACTION" IS RETURNED UNCHANGED. OTHERWISE, "FRACTION" IS INCREMENTED ACCORDING TO THE LEVEL NUMBER AND RETURNED. */

```

DCL  FRACTION DECIMAL FIXED (9,7),
     LEVEL BINARY FIXED,
     ONEHALF DECIMAL FIXED (8,7) INIT(0.5000000),
     ONEFOURTH DECIMAL FIXED (8,7) INIT(0.2500000),
     ONESIXTH DECIMAL FIXED (8,7) INIT(0.1666667),
     ONEEIGHTH DECIMAL FIXED (8,7) INIT(0.1250000),
     ONE BINARY FIXED INIT(1),
     TWO BINARY FIXED INIT(2),
     THREE BINARY FIXED INIT(3);
IF FRACTION>=1.0000000 THEN RETURN;
IF LEVEL=ONE THEN DO;
  FRACTION=FRACTION+ONEEIGHTH;
  RETURN;
END;
ELSE IF LEVEL=TWO THEN DO;
  FRACTION=FRACTION+ONESIXTH;
  RETURN;
END;
ELSE IF LEVEL=THREE THEN DO;
  FRACTION=FRACTION+ONEFOURTH;

```

```
      RETURN;  
      END;  
    ELSE DO;  
      FRACTION=FRACTION+ONEHALF;  
      RETURN;  
      END;  
    END COMPUTE;  
  
TERM: END PROG;
```

APPENDIX C

SAMPLE OUTPUTS

Some sample outputs are given in the following pages. The outputs are for trial #s 69, 120 and 250. In trial #69 FROG had to consult its memory for all regions to come to a decision. The decision was specific to the region in both the trials 120 and 250. A memory dump was requested in trial #250.

THIS IS TRIAL NUMBER 69

"FROG" HAS JUST SEEN A BUG IN REGION NUMBER 4
"FROG" IS PRESENTLY IN REGION NUMBER 3

THIS BUG IS A LAOYBUG
VISUAL CHARACTERISTICS OF THE BUG ARE: 0.7000000 0.9000000 0.7000000 0.7000000
OTHER INFORMATION:
GENERAL INNATE STIMULUS IS: 0.9000000
HUNGER IS: 0.6500000
FATIGUE IS: 0.2500000

"FROG" WILL NOW SEARCH ITS MEMORIES FOR
ANY EXPERIENCES WITH THIS BUG IN THE PAST.

OUTPUTS OF LEVELS OF GOOD TASTE MEMORY FOR REGION 4 0.0679500 0.0000000 0.0000000 0.0000000

OUTPUTS OF LEVELS OF BAD TASTE MEMORY FOR REGION 4 0.0000000 0.0000000 0.0000000 0.0000000

OUTPUTS OF LEVELS OF PREDATOR MEMORY FOR REGION 4 0.0676000 0.0908333 0.0000000 0.0000000

OUTPUTS OF MEMORY INTEGRATION SUBUNITS OF REGION 4

LEVEL 1 SUBUNIT: 0.0000000 0.0000000 1.0000000

LEVEL 2 SUBUNIT: 0.0000000 0.0000000 1.0000000

LEVEL 3 SUBUNIT: 0.0000000 0.0000000 1.0000000

LEVEL 4 SUBUNIT: 0.0000000 0.0000000 1.0000000

OUTPUT OF MIU FOR REGION 4: 0.0000000

ALARM OUTPUT OF MIU FOR REGION 4: 0.0000000

"FROG" COULD NOT COME TO DECISION BASED ON WHAT IT
REMEMBERS ABOUT THE BUG PRESENTED IN THE GIVEN REGION.
SO IT WILL NOW SEARCH MEMORY FOR PREVIOUS
EXPERIENCES WITH THIS BUG IN OTHER REGIONS

OUTPUTS OF LEVELS OF GOOD TASTE MEMORY FOR REGION 1 0.3927000 0.0000000 0.0000000 0.0000000

OUTPUTS OF LEVELS OF BAD TASTE MEMORY FOR REGION 1 0.0000000 0.0000000 0.0000000 0.0000000

OUTPUTS OF LEVELS OF PREDATOR MEMORY FOR REGION 1 0.0000000 0.0000000 0.0000000 0.0000000

OUTPUTS OF MEMORY INTEGRATION SUBUNITS OF REGION 1

LEVEL 1 SUBUNIT: 0.0000000 0.0000000 1.0000000

LEVEL 2 SUBUNIT: 0.0000000 0.0000000 1.0000000

LEVEL 3 SUBUNIT: 0.0000000 0.0000000 1.0000000

LEVEL 4 SUBUNIT: 0.0000000 0.0000000 1.0000000

OUTPUT OF MIU FOR REGION 1: 0.0000000

ALARM OUTPUT OF MIU FOR REGION 1: 0.0000000

OUTPUTS OF LEVELS OF GOOD TASTE MEMORY FOR REGION 2 0.3918000 0.0000000 0.0000000 0.0000000

OUTPUTS OF LEVELS OF BAD TASTE MEMORY FOR REGION 2 0.2000500 0.2080500 0.3125500 0.6275500

OUTPUTS OF LEVELS OF PREDATOR MEMORY FOR REGION 2 0.2779000 0.3711334 0.0000000 0.0000000

OUTPUTS OF MEMORY INTEGRATION SUBUNITS OF REGION 2

LEVEL 1 SUBUNIT: 0.000000 0.000000 1.000000

LEVEL 2 SUBUNIT: 0.000000 0.000000 1.000000

LEVEL 3 SUBUNIT: 0.000000 0.000000 1.000000

LEVEL 4 SUBUNIT: 0.3724500 0.000000 0.000000

OUTPUT OF MIU FOR REGION 2: 0.3724500

ALARM OUTPUT OF MIU FOR REGION 2: 0.0000000

OUTPUTS OF LEVELS OF GOOD TASTE MEMORY FOR REGION 3 0.6293500 0.0000000 0.0000000 0.0000000

OUTPUTS OF LEVELS OF BAD TASTE MEMORY FOR REGION 3 0.0000000 0.0000000 0.0000000 0.0000000

OUTPUTS OF LEVELS OF PREDATOR MEMORY FOR REGION 3 0.2358500 0.3126000 0.0000000 0.0000000

OUTPUTS OF MEMORY INTEGRATION SUBUNITS OF REGION 3

LEVEL 1 SUBUNIT: 0.6293500 0.000000 0.000000

LEVEL 2 SUBUNIT: 0.000000 0.000000 0.000000

LEVEL 3 SUBUNIT: 0.000000 0.000000 0.000000

LEVEL 4 SUBUNIT: 0.000000 0.000000 0.000000

OUTPUT OF MIU FOR REGION 3: 0.6293500

ALARM OUTPUT OF MIU FOR REGION 3: 0.0000000

FINAL MEMORY INTEGRATION UNIT OUTPUT = 0.6293500

ALARM OUTPUT OF MIU = 0.0000000

IMPULSE OF "FROG" TO FEE0 = 0.6293500

FATIGUE LEVEL = 0.2500000

"FROG" DECIDES TO FEE0

RESULTING GOOD TASTE = 0.1000000

RESULTING BAD TASTE = 0.9000000

PAIN INFLICTED = 0.0000000

FROG WILL NOW RECORD THE RESULTS OF THIS TRIAL IN ITS MEMORIES

"FROG'S MEMORIES HAVE BEEN UPDATED
AND "FROG" IS READY FOR ANOTHER STIMULUS

.....

THIS IS TRIAL NUMBER 160

"FROG" HAS JUST SEEN A BUG IN REGION NUMBER 3
"FROG" IS PRESENTLY IN REGION NUMBER 2

THIS BUG IS A FLY
VISUAL CHARACTERISTICS OF THE BUG ARE: 0.8000000 0.7000000 0.7000000 0.6000000
OTHER INFORMATION:
GENERAL INNATE STIMULUS IS: 0.8000000
HUNGER IS: 0.6000000
FATIGUE IS: 0.0500000

"FROG" WILL NOW SEARCH ITS MEMORIES FOR
ANY EXPERIENCES WITH THIS BUG IN THE PAST.

OUTPUTS OF LEVELS OF GOOD TASTE MEMORY FOR REGION 3 0.7134000 0.6271001 0.6234000 0.5334000

OUTPUTS OF LEVELS OF BAD TASTE MEMORY FOR REGION 3 0.0800000 0.0000000 0.0000000 0.0000000

OUTPUTS OF LEVELS OF PREDATOR MEMORY FOR REGION 3 0.4713500 0.6273501 0.0000000 0.0000000

OUTPUTS OF MEMORY INTEGRATION SUBUNITS OF REGION 3

LEVEL 1 SUBUNIT: 0.7134000 0.0000000 0.0000000

LEVEL 2 SUBUNIT: 0.0000000 0.0000000 0.0000000

LEVEL 3 SUBUNIT: 0.0000000 0.0000000 0.0000000

LEVEL 4 SUBUNIT: 0.0000000 0.0000000 0.0000000

OUTPUT OF MIU FOR REGION 3: 0.7134800

ALARM OUTPUT OF MIU FOR REGION 3: 0.0000000

FINAL MEMORY INTEGRATION UNIT OUTPUT = 0.7134000

ALARM OUTPUT OF MIU = 0.0000000

IMPULSE OF "FROG" TO FEED = 0.6000000

FATIGUE LEVEL = 0.0500000

"FROG" DECIDES TO FEED

RESULTING GOOD TASTE = 0.9000000

RESULTING BAD TASTE = 0.1000000

PAIN INFLICTED = 0.0000000

FROG WILL NOW RECORD THE RESULTS OF THIS TRIAL IN ITS MEMORIES

"FROG'S MEMORIES HAVE BEEN UPDATED
AND "FROG" IS READY FOR ANOTHER STIMULUS

.....

THIS IS TRIAL NUMBER 250

"FROG" HAS JUST SEEN A BUG IN REGION NUMBER 4
"FROG" IS PRESENTLY IN REGION NUMBER 1

THIS BUG IS A LADYBUG
VISUAL CHARACTERISTICS OF THE BUG ARE: 0.7000000 0.9000000 0.7000000 0.7000000
OTHER INFORMATION:
GENERAL INNATE STIMULUS IS: 0.9000000
HUNGER IS: 0.5500000
FATIGUE IS: 0.0000000

"FROG" WILL NOW SEARCH ITS MEMORIES FOR
ANY EXPERIENCES WITH THIS BUG IN THE PAST.

OUTPUT STATUS OF EACH T* ELEMENT IN REGION 1 OF GOOD TASTE MEMORY IS NOW:

	1	T* ELEMENT NUMBER 2	3
LEVEL 1 COMBINATION 1	0.7105500	0.4100500	0.0000000
LEVEL 1 COMBINATION 2	0.6261500	0.0000000	0.0000000
LEVEL 1 COMBINATION 3	0.6234500	0.0000000	0.0000000
LEVEL 1 COMBINATION 4	0.5337500	0.5500500	0.0000000
LEVEL 2 COMBINATION 1	0.6205501	0.4700500	0.0000000
LEVEL 2 COMBINATION 2	0.6205501	0.4700500	0.0000000
LEVEL 2 COMBINATION 3	0.5305501	0.4700500	0.0000000
LEVEL 2 COMBINATION 4	0.6229501	0.0000000	0.0000000
LEVEL 2 COMBINATION 5	0.5337501	0.5500501	0.0000000
LEVEL 2 COMBINATION 6	0.5305501	0.5500501	0.0000000
LEVEL 3 COMBINATION 1	0.6205500	0.4700500	0.0000000
LEVEL 3 COMBINATION 2	0.5305500	0.4700500	0.0000000
LEVEL 3 COMBINATION 3	0.5305500	0.4700500	0.0000000
LEVEL 3 COMBINATION 4	0.5305500	0.5500500	0.0000000
LEVEL 4 COMBINATION 1	0.5305500	0.4700500	0.0000000

OUTPUT STATUS OF EACH T* ELEMENT IN REGION 2 OF GOOD TASTE MEMORY IS NOW:

	1	T* ELEMENT NUMBER 2	3
LEVEL 1 COMBINATION 1	0.7088500	0.0000000	0.0000000
LEVEL 1 COMBINATION 2	0.6235500	0.0000000	0.0000000
LEVEL 1 COMBINATION 3	0.6215500	0.0000000	0.0000000
LEVEL 1 COMBINATION 4	0.5326500	0.0000000	0.0000000
LEVEL 2 COMBINATION 1	0.6188501	0.0000000	0.0000000
LEVEL 2 COMBINATION 2	0.6188501	0.0000000	0.0000000
LEVEL 2 COMBINATION 3	0.5288501	0.0000000	0.0000000
LEVEL 2 COMBINATION 4	0.6197501	0.0000000	0.0000000
LEVEL 2 COMBINATION 5	0.5326501	0.0000000	0.0000000
LEVEL 2 COMBINATION 6	0.5288501	0.0000000	0.0000000
LEVEL 3 COMBINATION 1	0.6188500	0.0000000	0.0000000
LEVEL 3 COMBINATION 2	0.5288500	0.0000000	0.0000000
LEVEL 3 COMBINATION 3	0.5288500	0.0000000	0.0000000
LEVEL 3 COMBINATION 4	0.5288500	0.0000000	0.0000000
LEVEL 4 COMBINATION 1	0.5288500	0.0000000	0.0000000

OUTPUT STATUS OF EACH T+ ELEMENT IN REGION 3 OF GOOD TASTE MEMORY IS NOW:

	1	T+ ELEMENT NUMBER 2	3
LEVEL 1 COMBINATION 1	0.4711500	0.7095500	0.0000000
LEVEL 1 COMBINATION 2	0.6260500	0.0000000	0.0000000
LEVEL 1 COMBINATION 3	0.6250500	0.0000000	0.0000000
LEVEL 1 COMBINATION 4	0.6311500	0.5305500	0.0000000
LEVEL 2 COMBINATION 1	0.4711500	0.6195501	0.0000000
LEVEL 2 COMBINATION 2	0.4711500	0.6195501	0.0000000
LEVEL 2 COMBINATION 3	0.4711500	0.5295501	0.0000000
LEVEL 2 COMBINATION 4	0.6245501	0.0000000	0.0000000
LEVEL 2 COMBINATION 5	0.5511501	0.5305501	0.0000000
LEVEL 2 COMBINATION 6	0.5511501	0.5295501	0.0000000
LEVEL 3 COMBINATION 1	0.4711500	0.6195500	0.0000000
LEVEL 3 COMBINATION 2	0.4711500	0.5295500	0.0000000
LEVEL 3 COMBINATION 3	0.4711500	0.5295500	0.0000000
LEVEL 3 COMBINATION 4	0.5511500	0.5295500	0.0000000
LEVEL 4 COMBINATION 1	0.4711500	0.5295500	0.0000000

OUTPUT STATUS OF EACH T* ELEMENT IN REGION 4 OF GOOD TASTE MEMORY IS NOW:

	1	T* ELEMENT NUMBER 2	3
LEVEL 1 COMBINATION 1	0.4692000	0.0000000	0.0000000
LEVEL 1 COMBINATION 2	0.5509000	0.0000000	0.0000000
LEVEL 1 COMBINATION 3	0.5519000	0.0000000	0.0000000
LEVEL 1 COMBINATION 4	0.6292000	0.0000000	0.0000000
LEVEL 2 COMBINATION 1	0.4692000	0.0000000	0.0000000
LEVEL 2 COMBINATION 2	0.4692000	0.0000000	0.0000000
LEVEL 2 COMBINATION 3	0.4692000	0.0000000	0.0000000
LEVEL 2 COMBINATION 4	0.5498001	0.0000000	0.0000000
LEVEL 2 COMBINATION 5	0.5492001	0.0000000	0.0000000
LEVEL 2 COMBINATION 6	0.5492001	0.0000000	0.0000000
LEVEL 3 COMBINATION 1	0.4692000	0.0000000	0.0000000
LEVEL 3 COMBINATION 2	0.4692000	0.0000000	0.0000000
LEVEL 3 COMBINATION 3	0.4692000	0.0000000	0.0000000
LEVEL 3 COMBINATION 4	0.5492000	0.0000000	0.0000000
LEVEL 4 COMBINATION 1	0.4692000	0.0000000	0.0000000

OUTPUT STATUS OF EACH T* ELEMENT IN REGION 1 OF BAO TASTE MEMORY IS NOW:

	1	T* ELEMENT NUMBER 2	3
LEVEL 1 COMBINATION 1	0.6211500	0.0000000	0.0000000
LEVEL 1 COMBINATION 2	0.4858500	0.0000000	0.0000000
LEVEL 1 COMBINATION 3	0.5511500	0.0000000	0.0000000
LEVEL 1 COMBINATION 4	0.4138500	0.0000000	0.0000000
LEVEL 2 COMBINATION 1	0.5511501	0.0000000	0.0000000
LEVEL 2 COMBINATION 2	0.6311501	0.0000000	0.0000000
LEVEL 2 COMBINATION 3	0.4711500	0.0000000	0.0000000
LEVEL 2 COMBINATION 4	0.5511501	0.0000000	0.0000000
LEVEL 2 COMBINATION 5	0.4738500	0.0000000	0.0000000
LEVEL 2 COMBINATION 6	0.4711500	0.0000000	0.0000000
LEVEL 3 COMBINATION 1	0.5511500	0.0000000	0.0000000
LEVEL 3 COMBINATION 2	0.4711500	0.0000000	0.0000000
LEVEL 3 COMBINATION 3	0.4711500	0.0000000	0.0000000
LEVEL 3 COMBINATION 4	0.4711500	0.0000000	0.0000000
LEVEL 4 COMBINATION 1	0.4711500	0.0000000	0.0000000

OUTPUT STATUS OF EACH T* ELEMENT IN REGION 2 OF BAD TASTE MEMORY IS NOW:

	1	T* ELEMENT NUMBER 2	3
LEVEL 1 COMBINATION 1	0.4658500	0.6222500	0.0000000
LEVEL 1 COMBINATION 2	0.5975500	0.4848500	0.0000000
LEVEL 1 COMBINATION 3	0.4652500	0.5522500	0.0000000
LEVEL 1 COMBINATION 4	0.4657500	0.4136500	0.0000000
LEVEL 2 COMBINATION 1	0.6200501	0.5522501	0.0000000
LEVEL 2 COMBINATION 2	0.6201501	0.6322501	0.0000000
LEVEL 2 COMBINATION 3	0.6232501	0.4722500	0.0000000
LEVEL 2 COMBINATION 4	0.6200501	0.5522501	0.0000000
LEVEL 2 COMBINATION 5	0.6200501	0.4736500	0.0000000
LEVEL 2 COMBINATION 6	0.6200501	0.4722500	0.0000000
LEVEL 3 COMBINATION 1	0.6200500	0.5522500	0.0000000
LEVEL 3 COMBINATION 2	0.6200500	0.4722500	0.0000000
LEVEL 3 COMBINATION 3	0.6200500	0.4722500	0.0000000
LEVEL 3 COMBINATION 4	0.6200500	0.4722500	0.0000000
LEVEL 4 COMBINATION 1	0.6200500	0.4722500	0.0000000

OUTPUT STATUS OF EACH T* ELEMENT IN REGION 3 OF BAD TASTE MEMORY IS NOW:

	1	T* ELEMENT NUMBER 2	3
LEVEL 1 COMBINATION 1	0.0000000	0.0000000	0.0000000
LEVEL 1 COMBINATION 2	0.0000000	0.0000000	0.0000000
LEVEL 1 COMBINATION 3	0.0000000	0.0000000	0.0000000
LEVEL 1 COMBINATION 4	0.0000000	0.0000000	0.0000000
LEVEL 2 COMBINATION 1	0.0000000	0.0000000	0.0000000
LEVEL 2 COMBINATION 2	0.0000000	0.0000000	0.0000000
LEVEL 2 COMBINATION 3	0.0000000	0.0000000	0.0000000
LEVEL 2 COMBINATION 4	0.0000000	0.0000000	0.0000000
LEVEL 2 COMBINATION 5	0.0000000	0.0000000	0.0000000
LEVEL 2 COMBINATION 6	0.0000000	0.0000000	0.0000000
LEVEL 3 COMBINATION 1	0.0000000	0.0000000	0.0000000
LEVEL 3 COMBINATION 2	0.0000000	0.0000000	0.0000000
LEVEL 3 COMBINATION 3	0.0000000	0.0000000	0.0000000
LEVEL 3 COMBINATION 4	0.0000000	0.0000000	0.0000000
LEVEL 4 COMBINATION 1	0.0000000	0.0000000	0.0000000

OUTPUT STATUS OF EACH T* ELEMENT IN REGION 4 OF RAD TASTE MEMORY IS NOW:

	1	T* ELEMENT NUMBER 2	3
LEVEL 1 COMBINATION 1	0.3059000	0.0000000	0.0000000
LEVEL 1 COMBINATION 2	0.3949000	0.0000000	0.0000000
LEVEL 1 COMBINATION 3	0.3059000	0.0000000	0.0000000
LEVEL 1 COMBINATION 4	0.3059000	0.0000000	0.0000000
LEVEL 2 COMBINATION 1	0.4099000	0.0000000	0.0000000
LEVEL 2 COMBINATION 2	0.4099000	0.0000000	0.0000000
LEVEL 2 COMBINATION 3	0.4109000	0.0000000	0.0000000
LEVEL 2 COMBINATION 4	0.4099000	0.0000000	0.0000000
LEVEL 2 COMBINATION 5	0.4099000	0.0000000	0.0000000
LEVEL 2 COMBINATION 6	0.4099000	0.0000000	0.0000000
LEVEL 3 COMBINATION 1	0.6199000	0.0000000	0.0000000
LEVEL 3 COMBINATION 2	0.6199000	0.0000000	0.0000000
LEVEL 3 COMBINATION 3	0.6199000	0.0000000	0.0000000
LEVEL 3 COMBINATION 4	0.6199000	0.0000000	0.0000000
LEVEL 4 COMBINATION 1	0.6199000	0.0000000	0.0000000

OUTPUT STATUS OF EACH T* ELEMENT IN REGION 1 OF PREDATOR MEMORY IS NOW:

	1	T* ELEMENT NUMBER 2	3
LEVEL 1 COMBINATION 1	0.0000000	0.0000000	0.0000000
LEVEL 1 COMBINATION 2	0.0000000	0.0000000	0.0000000
LEVEL 1 COMBINATION 3	0.0000000	0.0000000	0.0000000
LEVEL 1 COMBINATION 4	0.0000000	0.0000000	0.0000000
LEVEL 2 COMBINATION 1	0.0000000	0.0000000	0.0000000
LEVEL 2 COMBINATION 2	0.0000000	0.0000000	0.0000000
LEVEL 2 COMBINATION 3	0.0000000	0.0000000	0.0000000
LEVEL 2 COMBINATION 4	0.0000000	0.0000000	0.0000000
LEVEL 2 COMBINATION 5	0.0000000	0.0000000	0.0000000
LEVEL 2 COMBINATION 6	0.0000000	0.0000000	0.0000000
LEVEL 3 COMBINATION 1	0.0000000	0.0000000	0.0000000
LEVEL 3 COMBINATION 2	0.0000000	0.0000000	0.0000000
LEVEL 3 COMBINATION 3	0.0000000	0.0000000	0.0000000
LEVEL 3 COMBINATION 4	0.0000000	0.0000000	0.0000000
LEVEL 4 COMBINATION 1	0.0000000	0.0000000	0.0000000

OUTPUT STATUS OF EACH T* ELEMENT IN REGION 2 OF PREDATOR MEMORY IS NOW:

	1	T* ELEMENT NUMBER 2	3
LEVEL 1 COMBINATION 1	0.2731000	0.0000000	0.0000000
LEVEL 1 COMBINATION 2	0.2309000	0.0000000	0.0000000
LEVEL 1 COMBINATION 3	0.3509000	0.0000000	0.0000000
LEVEL 1 COMBINATION 4	0.2730000	0.0000000	0.0000000
LEVEL 2 COMBINATION 1	0.3109000	0.0000000	0.0000000
LEVEL 2 COMBINATION 2	0.3642334	0.0000000	0.0000000
LEVEL 2 COMBINATION 3	0.3663334	0.0000000	0.0000000
LEVEL 2 COMBINATION 4	0.3109000	0.0000000	0.0000000
LEVEL 2 COMBINATION 5	0.3109000	0.0000000	0.0000000
LEVEL 2 COMBINATION 6	0.3642334	0.0000000	0.0000000
LEVEL 3 COMBINATION 1	0.4709000	0.0000000	0.0000000
LEVEL 3 COMBINATION 2	0.4709000	0.0000000	0.0000000
LEVEL 3 COMBINATION 3	0.5509000	0.0000000	0.0000000
LEVEL 3 COMBINATION 4	0.4709000	0.0000000	0.0000000
LEVEL 4 COMBINATION 1	0.4709000	0.0000000	0.0000000

OUTPUT STATUS OF EACH T* ELEMENT IN REGION 3 OF PREDATOR MEMORY IS NOW:

	1	T* ELEMENT NUMBER 2	3
LEVEL 1 COMBINATION 1	0.4644500	0.0000000	0.0000000
LEVEL 1 COMBINATION 2	0.4687500	0.0000000	0.0000000
LEVEL 1 COMBINATION 3	0.4677500	0.0000000	0.0000000
LEVEL 1 COMBINATION 4	0.5973500	0.0000000	0.0000000
LEVEL 2 COMBINATION 1	0.6198501	0.0000000	0.0000000
LEVEL 2 COMBINATION 2	0.6203501	0.0000000	0.0000000
LEVEL 2 COMBINATION 3	0.6198501	0.0000000	0.0000000
LEVEL 2 COMBINATION 4	0.6247501	0.0000000	0.0000000
LEVEL 2 COMBINATION 5	0.6198501	0.0000000	0.0000000
LEVEL 2 COMBINATION 6	0.6198501	0.0000000	0.0000000
LEVEL 3 COMBINATION 1	0.6198500	0.0000000	0.0000000
LEVEL 3 COMBINATION 2	0.6198500	0.0000000	0.0000000
LEVEL 3 COMBINATION 3	0.6198500	0.0000000	0.0000000
LEVEL 3 COMBINATION 4	0.6198500	0.0000000	0.0000000
LEVEL 4 COMBINATION 1	0.6198500	0.0000000	0.0000000

OUTPUT STATUS OF EACH T* ELEMENT IN REGION 4 OF PREDATOR MEMORY IS NOW:

	1	T* ELEMENT NUMBER 2	3
LEVEL 1 COMBINATION 1	0.0615500	0.0000000	0.0000000
LEVEL 1 COMBINATION 2	0.0491500	0.0000000	0.0000000
LEVEL 1 COMBINATION 3	0.0791500	0.0000000	0.0000000
LEVEL 1 COMBINATION 4	0.0614500	0.0000000	0.0000000
LEVEL 2 COMBINATION 1	0.0691500	0.0000000	0.0000000
LEVEL 2 COMBINATION 2	0.0824833	0.0000000	0.0000000
LEVEL 2 COMBINATION 3	0.0847833	0.0000000	0.0000000
LEVEL 2 COMBINATION 4	0.0691500	0.0000000	0.0000000
LEVEL 2 COMBINATION 5	0.0691500	0.0000000	0.0000000
LEVEL 2 COMBINATION 6	0.0824833	0.0000000	0.0000000
LEVEL 3 COMBINATION 1	0.1091500	0.0000000	0.0000000
LEVEL 3 COMBINATION 2	0.1091500	0.0000000	0.0000000
LEVEL 3 COMBINATION 3	0.1291500	0.0000000	0.0000000
LEVEL 3 COMBINATION 4	0.1091500	0.0000000	0.0000000
LEVEL 4 COMBINATION 1	0.2291500	0.0000000	0.0000000

OUTPUTS OF LEVELS OF GOOD TASTE MEMORY FOR REGION 4 0.5519000 0.0000000 0.0003333 0.0000000

OUTPUTS OF LEVELS OF BAD TASTE MEMORY FOR REGION 4 0.3949000 0.4109030 0.6199000 0.6199000

OUTPUTS OF LEVELS OF PREDATOR MEMORY FOR REGION 4 0.0815500 0.0847833 0.0000000 0.0000000

OUTPUTS OF MEMORY INTEGRATION SUBUNITS OF REGION 4

LEVEL 1 SUBUNIT: 0.5519000 0.0000000 0.0000000

LEVEL 2 SUBUNIT: 0.0000000 0.0000000 0.0000000

LEVEL 3 SUBUNIT: 0.0000000 0.0000000 0.0000000

LEVEL 4 SUBUNIT: 0.0000000 0.0000000 0.0000000

OUTPUT OF MIU FOR REGION 4: 0.5519000

ALARM OUTPUT OF MIU FOR REGION 4: 0.0000000

FINAL MEMORY INTEGRATION UNIT OUTPUT = 0.5519000

ALARM OUTPUT OF MIU = 0.0000000

IMPULSE OF "FROG" TO FEED = 0.5500000

FATIGUE LEVEL = 0.0000000

"FROG" DECIDES TO FEED

RESULTING GOOD TASTE = 0.1000000

RESULTING BAD TASTE = 0.9000000

PAIN INFLECTED = 0.0000000

FROG WILL NOW RECORD THE RESULTS OF THIS TRIAL IN ITS MEMORIES

"FROG'S MEMORIES HAVE BEEN UPDATED
AND "FROG" IS READY FOR ANOTHER STIMULUS

.....

U.S. ATOMIC ENERGY COMMISSION
UNIVERSITY-TYPE CONTRACTOR'S RECOMMENDATION FOR
DISPOSITION OF SCIENTIFIC AND TECHNICAL DOCUMENT

(See Instructions on Reverse Side)

1. AEC REPORT NO.

C00-1469-0231

2. TITLE

FROG: DESIGN AND SIMULATION STUDY OF A MECHANISM
WHICH LEARNS SELF PRESERVATIVE REACTIONS TO ITS ENVIRONMENT

3. TYPE OF DOCUMENT (Check one):

- ☒ a. Scientific and technical report
☐ b. Conference paper not to be published in a journal:

Title of conference _____

Date of conference _____

Exact location of conference _____

Sponsoring organization _____

- ☐ c. Other (Specify) _____

4. RECOMMENDED ANNOUNCEMENT AND DISTRIBUTION (Check one):

- ☒ a. AEC's normal announcement and distribution procedures may be followed.
☐ b. Make available only within AEC and to AEC contractors and other U.S. Government agencies and their contractors.
☐ c. Make no announcement or distribution.

5. REASON FOR RECOMMENDED RESTRICTIONS:

6. SUBMITTED BY: NAME AND POSITION (Please print or type)

Debasish Bose
Research Assistant

Organization

Digital Computer Laboratory
University of Illinois
Urbana, Illinois 61801

Signature

W. J. Ponnellbaum

Date

August 6, 1973

FOR AEC USE ONLY

7. AEC CONTRACT ADMINISTRATOR'S COMMENTS, IF ANY, ON ABOVE ANNOUNCEMENT AND DISTRIBUTION
RECOMMENDATION:

8. PATENT CLEARANCE:

- ☐ a. AEC patent clearance has been granted by responsible AEC patent group.
☐ b. Report has been sent to responsible AEC patent group for clearance.
☐ c. Patent clearance not required.

BIBLIOGRAPHIC DATA SHEET		1. Report No. UIUCDCS-R-73-588	2.	3. Recipient's Accession No.
4. Title and Subtitle FROG: DESIGN AND SIMULATION STUDY OF A MECHANISM WHICH LEARNS SELF PRESERVATIVE REACTIONS TO ITS ENVIRONMENT			5. Report Date August, 1973	
			6.	
7. Author(s) Debasish Bose			8. Performing Organization Rept. No.	
9. Performing Organization Name and Address Department of Computer Science University of Illinois Urbana, Illinois 61801			10. Project/Task/Work Unit No.	
			11. Contract/Grant No. 46-26-15-301	
12. Sponsoring Organization Name and Address Department of Computer Science University of Illinois Urbana, Illinois 61801			13. Type of Report & Period Covered Thesis	
			14.	
15. Supplementary Notes				
16. Abstracts FROG is the simulation of mechanism which learns to discriminate between various classes of stimuli. The stimuli are some bugs (good and bad tasting) and predators. Each class of stimulus is characterized by the particular values of a set of visual characteristics. FROG acts like an adaptive filter. The visual data, interpreted through the filter, and a few somatic signals, which measures the requirements of FROG for feed and rest, are combined to decide upon one of the following list of actions: (1) feed, (2) rest, (3) flee.				
17. Key Words and Document Analysis. 17a. Descriptors artificial intelligence cognitive system visual characteristics stimulus adaptive filter				
17b. Identifiers/Open-Ended Terms				
17c. COSATI Field/Group				
18. Availability Statement			19. Security Class (This Report) UNCLASSIFIED	21. No. of Pages 118
			20. Security Class (This Page) UNCLASSIFIED	22. Price

OCT 15 1973

JAN 23 1978

UNIVERSITY OF ILLINOIS-URBANA

510.84 IL6R no. C002 no.588-589(1973)

Digital computer internal report /

3 0112 088400855