The logic design of a centering device for a pattern recognition system.

Narayanan K. Natarajan

University of Windsor

Follow this and additional works at: https://scholar.uwindsor.ca/etd

Recommended Citation

https://scholar.uwindsor.ca/etd/6443

This online database contains the full-text of PhD dissertations and Masters’ theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email (scholarship@uwindsor.ca) or by telephone at 519-253-3000ext. 3208.
THE LOGIC DESIGN
OF A
CENTERING DEVICE FOR A PATTERN RECOGNITION SYSTEM

BY

NARAYANAN K. NATARAJAN

A Thesis
Submitted to the Faculty of Graduate Studies through the
Department of Electrical Engineering in Partial Fulfillment
Of the Requirements for the Degree of
Master of Applied Science at
University of Windsor

Windsor, Ontario

1966
INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

UMI

UMI Microform EC52624
Copyright 2008 by ProQuest LLC.
All rights reserved. This microform edition is protected against unauthorized copying under Title 17, United States Code.

ProQuest LLC
789 E. Eisenhower Parkway
PO Box 1346
Ann Arbor, MI 48106-1346

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Approved by

Dr. S.N. Kalra
Associate Professor

Dr. P.A.V. Thomas
Professor

Dr. E.W. Channen
Associate Professor

152220

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
ABSTRACT

A logic design has been proposed for a device which is part of a pattern recognition system\(^1\). The device centers a character projected on a grid of photodiodes. The centering is performed in two phases. During phase 1 a sequential network positions the character such that it lies symmetrically with respect to either of the two parallel sides of the grid or all the four sides. The sequential circuit operates in the fundamental mode and takes its primary inputs from the four sides of the grid. The sequential circuit realisation uses the minimum number of amplifiers in its feedback loops. If at the end of the sequential circuit operation, the character is not symmetrical with respect to all the four sides of the grid, phase 2 network goes into operation and shifts the character by the required amount in the required direction. The phase 2 network is essentially a combinational network (no feedback) which compares pairs of corresponding rows or columns of the grid, one at a time. The logic device was simulated on an IBM 1620 Model II computer and found to be satisfactory in its operation.

The device proposed here is novel for a pattern recognition system such as the one proposed by Dydyk.\(^1\)
ACKNOWLEDGMENTS

The author wishes to express appreciation to Dr. S.N. Kalra, who supervised this work, for his guidance and advice.

Acknowledgment is also due to the National Research Council for financial assistance provided for this project.

Thanks are also due to Miss Antoinette Kah who typed the manuscripts.
TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ABSTRACT</td>
<td>ii</td>
</tr>
<tr>
<td></td>
<td>ACKNOWLEDGMENTS</td>
<td>iii</td>
</tr>
<tr>
<td></td>
<td>TABLE OF CONTENTS</td>
<td>iv</td>
</tr>
<tr>
<td></td>
<td>LIST OF TABLES</td>
<td>v</td>
</tr>
<tr>
<td></td>
<td>LIST OF FIGURES</td>
<td>vi</td>
</tr>
<tr>
<td>Chapter</td>
<td>I. INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>II. THE SYSTEM</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>2.1 Review of Sequential Circuits</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>2.2 Outline of Proposed Method</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>III. LOGIC DESIGN OF THE CENTERING DEVICE</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>3.1 General Discussion</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>3.2 Flowtable</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>3.3 State Assignment</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>3.4 Output Matrix</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>3.5 Derivation of the Combinatorial Circuity</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>for the Sequential Network - Phase 1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.6 Combinatorial Network - Phase 2</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td>3.7 System Operation</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>3.8 Hardware Implementation</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>IV. SIMULATION OF THE CENTERING DEVICE</td>
<td>26</td>
</tr>
<tr>
<td></td>
<td>V. CONCLUSIONS</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td>BIBLIOGRAPHY</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td>APPENDIX A. DIAGRAMS</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>APPENDIX B. TABLES</td>
<td>56</td>
</tr>
<tr>
<td></td>
<td>APPENDIX C. SIMULATION PROGRAM LISTING</td>
<td>68</td>
</tr>
<tr>
<td></td>
<td>APPENDIX D. RESULTS OF SIMULATION</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td>VITA AUCTORIS</td>
<td>92</td>
</tr>
</tbody>
</table>

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
# LIST OF TABLES

<table>
<thead>
<tr>
<th>TABLE</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Output Codes.</td>
<td>57</td>
</tr>
<tr>
<td>2. Initial Flowtable.</td>
<td>58</td>
</tr>
<tr>
<td>3. Explanation of Internal States.</td>
<td>59</td>
</tr>
<tr>
<td>4. Condensed Flowtable.</td>
<td>61</td>
</tr>
<tr>
<td>5. Preliminary State Assignment.</td>
<td>62</td>
</tr>
<tr>
<td>6. Final State Assignment.</td>
<td>63</td>
</tr>
<tr>
<td>7. Partial Output Matrix.</td>
<td>64</td>
</tr>
<tr>
<td>8. Output Matrix.</td>
<td>65</td>
</tr>
<tr>
<td>9. Summary of Boolean Expressions to Realize Phase 1 Network</td>
<td>66</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>1.</td>
<td>Grid of Photocells</td>
</tr>
<tr>
<td>2.</td>
<td>Hazard from Complementary Commands.</td>
</tr>
<tr>
<td>3.</td>
<td>Representation of a Sequential Machine.</td>
</tr>
<tr>
<td>4.</td>
<td>State Device.</td>
</tr>
<tr>
<td>5.</td>
<td>Flow chart for Phase 2 Operation.</td>
</tr>
<tr>
<td>6.</td>
<td>Elements Used in Logic Drawings.</td>
</tr>
<tr>
<td>7.</td>
<td>Realization of $c_j$ and $r_i$.</td>
</tr>
<tr>
<td>8.</td>
<td>Phase 2 - Input Selector.</td>
</tr>
<tr>
<td>9.</td>
<td>Phase 2 - Circuitry #1.</td>
</tr>
<tr>
<td>10.</td>
<td>Phase 2 - Circuitry #2.</td>
</tr>
<tr>
<td>11.</td>
<td>Phase 2 - Circuitry #3.</td>
</tr>
<tr>
<td>12.</td>
<td>Phase 2 - Circuitry #4.</td>
</tr>
<tr>
<td>13.</td>
<td>Phase 1 - Network #1.</td>
</tr>
<tr>
<td>14.</td>
<td>Phase 1 - Network #2.</td>
</tr>
<tr>
<td>15.</td>
<td>Phase 1 - Network #3.</td>
</tr>
<tr>
<td>16.</td>
<td>Phase 1 - Network #4.</td>
</tr>
<tr>
<td>17.</td>
<td>Phase 1 - Network #5.</td>
</tr>
<tr>
<td>18.</td>
<td>Phase 1 - Network #6.</td>
</tr>
<tr>
<td>19.</td>
<td>Phase 1 - Network #7.</td>
</tr>
<tr>
<td>20.</td>
<td>Phase 1 - Network #8.</td>
</tr>
<tr>
<td>21.</td>
<td>Phase 1 - Network #9.</td>
</tr>
<tr>
<td>22.</td>
<td>Phase 1 - Network #10.</td>
</tr>
<tr>
<td>23.</td>
<td>Phase 1 - Network #11.</td>
</tr>
<tr>
<td>24.</td>
<td>General System Flow Chart for Simulation.</td>
</tr>
<tr>
<td>25.</td>
<td>Simulated Grid.</td>
</tr>
</tbody>
</table>
CHAPTER I

INTRODUCTION

An analogue - digital system for the recognition of hand printed capital block letters and numerals projected on a grid of photo cells was proposed by B.R. Dydyk\(^1\) at the Electrical Engineering Department of the University of Windsor. As further research proceeded on this, it was decided to develop the peripheral equipment of the system. The analogue portion of the system required letters of uniform size occupying as much of the grid as possible. This thesis presents the logic design of a centering device whose objectives are:

1. to make a character projected on the array of photo cells (figure 1) lie symmetrically with respect to the horizontal and vertical axes and

2. to make it occupy maximum area of the grid.

A character is considered partially centered when it lies entirely within the grid and on magnification simultaneously touches any two of the parallel sides of the grid. A character is considered fully centered when it lies entirely within the grid and on magnification touches all the four sides of the grid or when a partially centered character is shifted so as to lie symmetrically with respect to all the sides of the grid. To achieve this the device has to know the previous states of the character or in other words, it has to have a memory. The signal that determines the state of the character is obtained from the four sides of the array of diodes. It is called an input to the logic network. Commands to magnify, contract or shift the character would form the outputs of the network. The 'state' of the network corresponds to some memory of the past inputs. The dependence of the outputs on

---

\(^1\) Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
the present and past inputs can be expressed as a function of the present input and 'state' of the network. Such a network is called a sequential circuit. A character projected on the grid is shifted or contracted and magnified once or a number of times before it gets centered partially or fully by the sequential network. Partially centered characters need only a further shifting which can be accomplished by comparing the rows or columns the character spans on either side of the reference axis. The logic circuitry for this need only be of the combinatorial type (no feedback).

At present no information appears to be available on logic control devices used to position a group of characters in pattern recognition systems that use stimulus pictures.
2.1 Review of Sequential Circuits

Sequential circuits are usually classified as asynchronous or fundamental mode circuits and synchronous or pulse mode circuits. Inputs which can remain in one state or another for an indefinite length of time are termed level inputs. Asynchronous circuits have level inputs. Inputs to synchronous circuits are gated by clock signals. The outputs of sequential circuits can be of the pulse or fundamental mode.

The function realized by a sequential circuit can be described by a flowtable. The rows and columns of the flowtable correspond to the internal states and primary input states. The entries of the flowtable are the next internal state and output state. A total state is a combination of an input state and an internal state. A decimal total state number or simply a state number is defined as:

\[
q-1 \\
\sum_{j=0}^{q-1} x_j \cdot 2^j + \sum_{p=q}^{q+s-1} y_p \cdot 2^p
\]

where \( x_j, y_p = 0 \) or \( 1 \), \( q \) and \( s \) are the number of variables in the input and internal states respectively, 

\( x_{q-1} x_{q-2} \ldots x_0 \) represents an input state, and 

\( y_{q+s-1} y_{q+s-2} \ldots y_q \) represents an internal state.

In asynchronous circuits, a total state is stable if the next state entry is equal to the present internal state. When the circuit is in a stable total state, the internal state can change only after the input state is changed.

An asynchronous sequential circuit is said to operate in a
fundamental mode if the inputs are never changed unless the circuit is stable internally. If no more than one output change occurs for any input change in an asynchronous circuit, it is called a normal sequential circuit. A sequential circuit can be designed to operate meaningfully either as a synchronous or an asynchronous device.

If \((x_1, x_2, \ldots, x_n)\), are the inputs to a logic decision element and the output is

\[ Z = f (x_1, x_2, \ldots, x_n), \]

a delay occurs between the time the inputs are supplied and the time the output assumes the value designated by the function \(Z = f (x_1, \ldots, x_n)\). It is called an element delay. The delay in the transmission of a signal along a line is called a line delay. Line and element delays constitute stray delays. An inertial delay element \(\Delta\) is one that does not respond to input changes of duration less than \(\Delta\).

In asynchronous circuits there are no clock signals to regulate the circuit. Unger has shown that in level input asynchronous sequential circuits, if a closed loop exists and if the system variables are such that a signal may flow entirely around the loop, then that loop must contain an amplifier. Such a loop is called a feedback loop.

Sequential circuits can be represented in two ways—a Mealy model or a Moore model. In the Moore model the outputs are directly dependent on the internal states whereas in the Mealy model the outputs are dependent on the total state.

2.2 Outline of Proposed Method

The logic control unit centers a character projected on a grid of
photodiodes in two phases. Phase 1 consists of an asynchronous sequential circuit using RC networks as inertial delays. At the end of Phase 1 operation the character would have been centered either partially or fully. If the centering is only partial, Phase 2 network which is of the combinational type, shifts the character suitably.

The grid is composed of 80 photodiodes. It is proposed to mount photocells of low threshold values, on the sides of the grid as in figure 1, to determine the primary inputs to the sequential network. Such an arrangement permits the character on being centered to occupy a larger area of the grid. The low threshold values of these photo cells restrict over expansion of a character.

The grid of photocells is proposed to be mounted in the picture plane of a camera, to which the stimulus pictures can be shown. Contraction and magnification effect can be obtained using a zoom lens arrangement. The stimulus pictures are to be mounted in a light weight frame which can be shifted whenever shift operations are called for. Apart from these internal power supply, racks to hold the logic modules, push buttons and lamp indicators are required for the system.
3.1 General Discussion

In the design of a sequential circuit the first decision to be taken is the mode of operation of the circuit - synchronous or asynchronous. Asynchronous circuits are faster because the system operates at its own speed. Transient conditions during the change of state variables cannot be ignored with asynchronous operation, and several variables are allowed to change (referred to as a race) only if the resulting state does not depend on the order of change of these variables (referred to as non critical race). Though the problem of races is not present with synchronous operation, stray delays in combinational circuits and feedback lines can cause difficulties in synchronising the next input and the next state to obtain the intended behaviour of the system. In an asynchronous circuit if an input is applied and held on the input lines, the machine will change state only once, that is the next state will be a stable state under the input and if no critical race condition occur, the sequential circuit will remain stable until a new input is applied. In an asynchronous circuit, to change from the present to the next state, a sequence of internal variable changes may be required and the time between input changes must be sufficient to allow for the longest possible sequence.

Liu and Friedman have shown that normal fundamental mode circuits can be realized such that in a transition from the present to the next state, all internal state variables that need to be changed can be allowed to change simultaneously without the hazard of critical races. Such circuits are not only fast in operation but use the...
smallest possible number of amplifiers in the feedback loops. On this basis, the sequential network is designed to operate in the fundamental mode.

The primary inputs to the circuit will be taken from the four sides of the grid. So sixteen input combinations are possible. The bits in the input combination from left to right represent inputs from top, right, bottom and left sides of the grid. The inputs are coded as follows:

<table>
<thead>
<tr>
<th>Input</th>
<th>T R B L</th>
</tr>
</thead>
<tbody>
<tr>
<td>I1</td>
<td>0 0 0 0</td>
</tr>
<tr>
<td>I2</td>
<td>0 0 0 1</td>
</tr>
<tr>
<td>I3</td>
<td>0 0 1 0</td>
</tr>
<tr>
<td>I4</td>
<td>0 0 1 1</td>
</tr>
<tr>
<td>I5</td>
<td>0 1 0 0</td>
</tr>
<tr>
<td>I6</td>
<td>0 1 0 1</td>
</tr>
<tr>
<td>I7</td>
<td>0 1 1 0</td>
</tr>
<tr>
<td>I8</td>
<td>0 1 1 1</td>
</tr>
<tr>
<td>I9</td>
<td>1 0 0 0</td>
</tr>
<tr>
<td>I10</td>
<td>1 0 0 1</td>
</tr>
<tr>
<td>I11</td>
<td>1 0 1 0</td>
</tr>
<tr>
<td>I12</td>
<td>1 0 1 1</td>
</tr>
<tr>
<td>I13</td>
<td>1 1 0 0</td>
</tr>
<tr>
<td>I14</td>
<td>1 1 0 1</td>
</tr>
<tr>
<td>I15</td>
<td>1 1 1 0</td>
</tr>
<tr>
<td>I16</td>
<td>1 1 1 1</td>
</tr>
</tbody>
</table>

Thus input I18 means that the image touches the right, bottom and left sides of the grid.
The following basic output operations are involved in trying to center a character on the grid:

(a) contraction

(b) magnification

(c) shift.

Under the shift operation the character is moved up or down or right or left. The extent of shift is either full or one half the grid cell in the direction of shift. Thus the output command 'full down' will shift the whole character down by one grid cell. The command 'half right' will shift the character right by one half grid cell. It becomes necessary to shift by half grid cells, when the character spans an odd number of rows or columns on the grid. The contraction and magnification have to be uniform and their extent restricted to one half of a grid cell. Distinct command signals are required to indicate that the character is centered along the horizontal, vertical or both the axes. If the character is touching the top, right and left sides of the grid, it has to be moved down and contracted. So commands such as 'contract down', 'contract up' etc. are required. In all seventeen operations (indicated in Table I along with their binary and decimal codes) are required. These comprise the output states of the sequential network.

3.2 Flowtable

Table II indicates the initial flowtable for the asynchronous machine. Table III summarizes all the states involved in the initial flowtable. There are sixteen initial states. States JA, JB, JC, JD, KA, KB, KC and KD are introduced to prevent the hazard of the machine entering a never ending cycle from complementary commands associated
with states B, E and C, H (see figure 2).

Stable internal states in the flowtable are distinguished by underlining them. The decimal numbers besides the stable entries are the desired outputs for the corresponding total states.

Before the internal states of the sequential machine are coded, it is helpful to minimize the number of internal states. On the flowtable, this operation corresponds to minimizing the number of rows. In a Mealy type machine two or more rows can be merged if for each input state these rows do not have conflicting entries. An entry which appears in any one of the rows will appear in the composite row. Underlined entries in any one of the merged rows will be underlined in the composite row. In Moore model machines as the output is dependent only on the internal state, two rows can be considered for merging if and only if their outputs are the same. It is proposed to model the sequential machine as a Mealy type for two reasons:

1. It permits maximum row merger and hence minimum number of internal states.
2. The assignment technique developed by Friedman is particularly suited to Mealy type machines.

The following rows can be merged together:

<table>
<thead>
<tr>
<th>Rows Merged</th>
<th>Composite Row Designator</th>
</tr>
</thead>
<tbody>
<tr>
<td>AA, AB, AC, AD</td>
<td>A</td>
</tr>
<tr>
<td>DA, DB, DC, DD, DE, DF, DG, DH, DI, DJ, DK</td>
<td>D</td>
</tr>
<tr>
<td>JA, JB, JC, JD</td>
<td>J</td>
</tr>
<tr>
<td>KA, KB, KC, KD</td>
<td>K</td>
</tr>
</tbody>
</table>

The condensed flowtable appears as Table IV.

Two stable states in a flowtable associated with the same input
state and output state are considered equivalent unless there exists some sequence of input states which may start from either of the two underlined entries and give corresponding output sequences which differ from one another. It is seen that there are no equivalent states in the condensed flowtable. In the discussion that follows, the word flowtable is to be taken to mean the condensed flowtable.

3.3 State Assignment

The eight rows of the flowtable indicate the need for binary 3-tuples, $y_1 \ y_2 \ y_3$ to code the internal states. If the assignment is made such that any transition from one state to another changes only one $y_1$ variable, no races occur. A binary three tuple can have only 3 adjacent three tuples. It is seen from the flow table that a transition from state B can lead to states A, C, D, J or H depending on the input. It is only possible to assign to any three of these, states adjacent to B. Hence there is a necessity to employ non-critical races. An initial assignment is made as follows. As many of the next internal states as possible are made adjacent to the present states.

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>B</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>D</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>E</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>H</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>J</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>K</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>
Each column of the flowtable is examined separately for the nature of transitions involved. If noncritical races are to be set up, it is done by relaxing the excited state variables one by one from right to left changing only one $y_1$ variable at a time. This procedure is adopted to avoid repetition of states unnecessarily and to be consistent.

**Column 1**

The transition paths are:

<table>
<thead>
<tr>
<th>Initial State</th>
<th>Transition Path</th>
<th>Final State</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>000</td>
<td>A</td>
</tr>
<tr>
<td>B</td>
<td>001 $\rightarrow$ 000</td>
<td>A</td>
</tr>
<tr>
<td>C</td>
<td>010 $\rightarrow$ 000</td>
<td>A</td>
</tr>
<tr>
<td>D</td>
<td>100 $\rightarrow$ 000</td>
<td>A</td>
</tr>
<tr>
<td>E</td>
<td>110 $\rightarrow$ 100 $\rightarrow$ 000</td>
<td>A</td>
</tr>
<tr>
<td>H</td>
<td>111 $\rightarrow$ 110 $\rightarrow$ 100 $\rightarrow$ 000</td>
<td>A</td>
</tr>
</tbody>
</table>

Whatever be the intermediate states, the final state of 000 is always reached and so no hazards are involved.

**Column 2**

The transitions involved are:

<table>
<thead>
<tr>
<th>Initial State</th>
<th>Transition Path</th>
<th>Final State</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>000 $\rightarrow$ 001</td>
<td>B</td>
</tr>
<tr>
<td>B</td>
<td>001</td>
<td>B</td>
</tr>
<tr>
<td>C</td>
<td>010 $\rightarrow$ 011 $\rightarrow$ 001</td>
<td>B</td>
</tr>
<tr>
<td>D</td>
<td>100 $\rightarrow$ 101 $\rightarrow$ 001</td>
<td>B</td>
</tr>
<tr>
<td>E</td>
<td>110 $\rightarrow$ 111 $\rightarrow$ 101</td>
<td>K</td>
</tr>
<tr>
<td>H</td>
<td>111 $\rightarrow$ 101 $\rightarrow$ 001</td>
<td>B</td>
</tr>
<tr>
<td>J</td>
<td>011</td>
<td>J</td>
</tr>
<tr>
<td>K</td>
<td>101</td>
<td>K</td>
</tr>
</tbody>
</table>
Critical races are observed in paths starting from C, D, E and H and to avoid them the state assignment has to be augmented with additional variables. The minimum number of additional variables required, can be estimated using a method due to Liu.

A transition path can be represented using decimal total state numbers. The decimal total state numbers are found (using expression 1) as follows:

<table>
<thead>
<tr>
<th>Total State</th>
<th>State Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>E I2</td>
<td>$1x2^6 + 1x2^5 + 0x2^4 + 0x2^3 + 0x2^2 + 0x2^1 + 1x2^0 = 97$</td>
</tr>
<tr>
<td>K I2</td>
<td>$1x2^6 + 0x2^5 + 1x2^4 + 0x2^3 + 0x2^2 + 0x2^1 + 1x2^0 = 81$</td>
</tr>
<tr>
<td>H I2</td>
<td>$1x2^6 + 1x2^5 + 1x2^4 + 0x2^3 + 0x2^2 + 0x2^1 + 1x2^0 = 113$</td>
</tr>
</tbody>
</table>

Stable state numbers in the transition path, are circled. The primary input responsible for the transition is written over the first arrow.

The path from E to K in Column 2 would look like

\[ \text{I2} \quad 97 \rightarrow 113 \rightarrow \text{K} \].

A state number is said to repeat impermissibly if

1. an intermediate state number in a path is repeated in another path with different following state numbers
2. an initial or terminal state number of a path is repeated as intermediate state numbers of other paths.

The transition paths in Column 2 using state numbers are

\[ \text{I2} \]
\[ \text{17} \]
\[ \text{49} \]
\[ \text{81} \]

for starting states B, J and K wherein the final states are the same as starting states and
Two theorems due to Liu are given below without proof.

Theorem 1. In the transition of a flowtable if there is only one state number \( S \) that repeats impermissibly and it repeats exactly \( q > 1 \) times, then the necessary and sufficient number of additional variables is given by an integer that is the lowest to satisfy the inequality

\[
m > \log_2 q.
\]

Theorem 2. If more than one state number repeats impermissibly and if the set of paths in which these state numbers repeat are disjoint then the additional variables required is the lowest to satisfy the inequality

\[
m > \log_2 \left( \max_{i} (q_i) \right)
\]

where state \( S_1 \) repeats \( q_1 \) times, \( S_2 \) \( q_2 \) times and so on.

In the above cases, if an initial or terminal state number is repeating impermissibly, then the number of additional variables required is \( m + 1 \). The intermediate state numbers of paths that have states impermissibly repeating can be increased by the introduction of additional state variables. For example each of the state numbers 49, 81 and 113
in Column 2 repeat once impermissibly. State number 113 is also an initial state number of path

\[
I_2 \\
113 \rightarrow 81 \rightarrow 17
\]

If transition \(97 \rightarrow 113 \rightarrow 81\) did not exist, one additional state variable would have sufficed and the transition path would be set up as follows.

\[
\begin{align*}
241 & \rightarrow 209 \rightarrow 145 \\
113 & \rightarrow 81 \rightarrow 17
\end{align*}
\]

The dotted arrows indicate the old transition path and solid arrows indicate the new transition path. The number 128 (the new variable is \(y_j\) with a weight of \(2^7\) by expression 1) is added to every state number of the path with impermissibly repeating state numbers. The resulting sequence is made the path that leads from the initial to the final state. The various transitions become

\[
\begin{align*}
I_2 & \quad (1) \\
33 & \rightarrow 161 \rightarrow 177 \rightarrow 145 \rightarrow 17 \\
I_2 & \\
65 & \rightarrow 193 \rightarrow 209 \\
I_2 & \\
113 & \rightarrow 241 \\
I_2 & \\
97 & \rightarrow 113 \rightarrow 81
\end{align*}
\]

As state number 113 still repeats impermissibly, the addition of another variable becomes necessary.

\[
\begin{align*}
353 & \rightarrow 369 \rightarrow 337 \\
97 & \rightarrow 113 \rightarrow 81
\end{align*}
\]
Firm arrows above, indicate the path, free of critical races.

For Column 2 it is therefore seen that two additional variables are required to avoid critical races. Similar examination of other columns show that no more than two additional variables are required to avoid critical races and to effect the required transitions. Using this assignment technique the sequential machine for the centering device needs a sequence of up to four changes in state variables to accomplish a change of state.

If \( p \) is the number of states of an asynchronous machine and \( k_m \) the lowest integer to satisfy the inequality \( k_m \geq \log_2 p \), the machine can be realized with \( 2k_m - 1 \) state variables. Using this \( 2k_m - 1 \) assignment technique (due to Huffman) an eight state machine would need 5 internal variables. A set of 5-tuples is associated with each state. If the time required to effect one transition is taken as a unit time, this method would require utmost two units of time to effect a transition from any state to any other. The set of 5-tuples are so assigned that there is at least one 5-tuple in each that is adjacent to a 5-tuple in the set associated with the state to which the transition has to take place. Noncritical races to get into such adjacent 5-tuples can be allowed and a further change of one variable will complete the transition.

Table V shows a state assignment using Friedman’s technique. It is derived as follows. Each column of the flowtable is considered separately. If in the \( i \)th column, \( S_i \) distinct entries exist, \( a_i \) variables are required to code the column where \( a_i \) is the smallest integer to satisfy the inequality \( a_i \geq \log_2 S_i \). The \( a_i \) variables are
denoted by \( y_1^1 \ y_2^1 \ldots \ y_n^1 \). Here the superscript indicates the column and the subscript the number of variables required to code the entries of that column. Let \( \text{NS} (X, Ij) \) represent the next state when the present state is \( X \) and \( Ij \) is the input. Two states \( X \) and \( U \) in a column are given the same coding if and only if \( \text{NS} (X, Ij) = \text{NS} (U, Ij) \). The symbol \( \phi \) is entered to represent 'don't care' entries if the next state is unspecified. Table V shows the preliminary assignment thus made. Column 2 for example, contains three distinct states \( B, J \) and \( K \) which are encoded as follows:

\[
B \ 0 \ \phi \\
J \ 1 \ 0 \\
K \ 1 \ 1
\]

Column 2 is then filled. The other columns in the flowtable are filled similarly. An examination of Table V indicates the following:

1. Columns 1, 4, 7, 8, 10, 12, 13, 14 and 15 are identical and have only one entry-‘0’. These do not contribute to the uniqueness of the row assignment and hence can be discarded.
2. Columns 6, 11 and 16 are identical. Any two can be disregarded. The assignments would still be unique.
3. Subcolumns 2 of columns 2, 3, 5 and 9 can be merged as they have no conflicting entries.

When these modifications are carried out the row assignment of Table VI results. The variable \( y_{6,11,16}^{6,11,16} \) indicates that columns 6, 11 and 16 in Table V have been merged.

It is observed that this assignment has six state variables (compared to 5 variables of the previous schemes). It is shown in the next section, that no critical races are involved when this technique is
used. All variables that require to be changed in a transition are allowed to change simultaneously. Hence the speed of operation is fast as compared to the circuits derived using the other assignment methods. No more than two variables are required to code any column. This means that only two amplifiers are needed in the feedback paths. Additional amplifiers may however be needed for purposes such as wave shaping etc. Based on these considerations, the Friedman assignment technique is used for the sequential circuit.

3.4 The Output Matrix

Table VII shows the partial output matrix. Each row of the table corresponds to an internal state and each column to a primary input. The entries of the table are the output states associated with the corresponding total states. By consulting the output data in Table II, output states that correspond to the stable total states of the flowtable, are immediately assigned. This results in the partial output matrix (Table VI). One is concerned with the output only after the circuit has become internally stable. As the machine is being modelled after the Mealy type, each of the remaining unstable entries, is assigned the output associated with the corresponding stable entry in the flowtable. As the input is changed to its new value, the output is changed at once to the one corresponding to the ultimate circuit condition. An examination of the flowtable along with the output matrix (Table VIII) shows that there is no more than one output change for any input change. Such flowtables are called normal fundamental flowtables.⁴, ¹².

3.5 Derivation of the Combinatorial Circuitry for the Sequential Network-Phase I
Figure 3 shows a representation of an asynchronous circuit with feedback index $F$. LC is a logic circuit that has neither memory nor feedback loops. Me is a sequential circuit with memory (delay) but no feedback loops. As the operation is in the fundamental mode, for each stable state, the lines $q_1, q_2, \ldots, q_F$ have some values associated with them. The stable input states and stable $q_i$s determine uniquely, the state of MC and hence the system. $Z_1, Z_2, \ldots, Z_n$ represent the output bit combinations and is also determined by the input state and $q_i$s. It was seen earlier that the system has a feedback index of two. If $Q_i$ denotes the $i$th amplifier and $q_i$ its output

$$Q_i = \sum_{m=1}^{k} I_m y_i^m$$

where $k$ is the number of input columns in the flowtable. This means that the $i$th amplifier is used to amplify the $i$th state variable with superscript $m$ when the input is $I_m$. For, if the input is say $I_5$, $Q_i = y_i^5$.

The vector $(q_1, q_2, \ldots, q_F)$ completely determines for any input, the next values of output and state variables. Thus

$$y_j^n = \sum_{m=1}^{k} I_m \cdot g_{jm} (q_1, q_2, \ldots, q_F)$$

for $n = 1, 2, \ldots, k$

$$j = 1, 2, \ldots, F$$
\[ Z_i = \sum_{m=1}^{k} I_m G_m (q_1, q_2, \ldots, q_F) \]  
\[ i = 1, 2 \ldots l. \] 

1 is the number of bits in the output state.

Column \( I_m \) of the flowtable can be considered as a submachine \( MC_m \).

When the input is \( I_m \), there are no critical races in \( MC_m \). When input changes to \( I_m \), \( MC_m \) is stable and so there are no critical races in \( MC \).

Variable \( y^m_j \) changes only when the input is not \( I_m \), that is it changes only when not being used to prevent critical races.

To prevent combinational hazards involving \( q \) and \( I \), an RC network shown in figure 4 is used. This also serves as a state device. Since an electromechanical device is employed to shift, magnify or contract the character, the time constant of the state device must be slightly greater than the time required for the electromechanical device to complete a command such as contract-right.

Expressions for the feedback and state variables are derived as follows.

From expression 2

\[ Q_1 = I1+I4+I7+I8+I10+I12+I13+I14+I15 +I2 y_1^2+I3 y_1^3+I9 y_1^9 + (I6+I11+I16) y_1^6,11,16 \]  
\[ Q_2 = (I2+I3+I5+I9) y_2^{2,3,5,9} \]  
\[ y_1^9 \] is derived as follows:

\( y_1^9 \) is '1' if and only if the next state happens to be \( C \), \( J \) or \( K \).

The next state can be \( J \) or \( K \) only if the input is \( I2 \) or \( I3 \) or \( I5 \) or \( I9 \).
The next state can be C if the input is I3. All these inputs are considered one by one.

If the input is I2,
\[ Q_1 = y_1^2 \]
\[ Q_2 = y_2^2,3,5,9, \]

The next state can be J or K only if \( y_1^2 = 1 \).

Hence if I2 \( q_1 \) is true \( y_1^9 = 1 \) is true. If the input is I3, \( Q_1 = y_1^3 \)
and the next state is J or K if and only if \( y_1^3 = 1 \). The next state is C if \( y_1^3 = 0 \). So \( y_1^9 \) is true whenever I3 is true. Similarly for the inputs I5 and I9, the terms I5 \( q_1 \) and I9 \( q_1 \) are to be included.

Hence the expression for \( y_1^9 \) becomes \( y_1^9 = I3I2 + I5q_1 + I9q_1 \). (7)

Expressions for other state variables are derived in a similar manner and are given below.
\[ y_1^2 = I5q_1 + I2q_1q_2 + I3q_1q_2 + I5q_1q_2 + I9q_1q_2 \quad (8) \]
\[ y_1^3 = I9 + I2q_1 + I3q_1 + I5q_1 \quad (9) \]
\[ y_1^{6,11,16} = I1 + I6q_1 + I11q_1 + I16q_1 \quad (10) \]
\[ y_1^5 = I2 + I3q_1 + I5q_1 + I9q_1 \quad (11) \]
\[ y_2^2,3,5,9 = I5q_1 + I9q_1 + I2q_1q_2 + I3q_1q_2 \]
\[ + I5q_1q_2 + I9q_1q_2 \quad (12) \]

There are five bits in the output code viz, \( Z_1, Z_2, Z_3, Z_4, Z_5 \). Expression for \( Z_1 \) is derived as a typical example.
\[ Z_1 = 1 \text{ if the input is } I_2 \text{ and } y_1^2 y_2^2, 2, 2, 2 = 10. \text{ i.e. if } \]
\[ I_2 q_1 q_2 \text{ is true } Z_1 \text{ is true. If the input is } I_3 \text{ and } y_1^3 y_2^2, 2, 2, 2 = 10, \]
\[ Z_1 \text{ is '1'. Hence } Z_1 \text{ is true if } I_3 q_1 q_2 \text{ is true. If the input is } \]
\[ I_5 \text{ or } I_9 \text{ and } q_1 q_2 \text{ is true, } Z_1 \text{ is '1'. If the input is } I_{11} \text{ or } I_{16} \text{ or } I_6 \]
\[ \text{and } y_1^6 y_{11}^6 y_{16}^6 = 1 \text{ or } 0, \text{ } Z_1 \text{ is true. i.e. } Z_1 \text{ is true if } I_{11} \text{ or } I_6 \text{ or } I_{16} \]
\[ \text{is true. Hence the expression for } Z_1 \text{ is} \]
\[ Z_1 = I_6 + I_{11} + I_{16} + I_2 q_1 q_2 + I_3 q_1 q_2 + I_5 q_1 q_2 + I_9 q_1 q_2 \quad (13) \]

Expressions for the other output bits are derived in a similar manner and are given below.

\[ Z_2 = I_2 q_1 q_2 + I_5 q_1 q_2 + I_9 q_1 q_2 + I_4 + I_7 + I_8 + I_{10} + I_{12} + I_{13} + I_{14} + I_{15} \quad (14) \]
\[ Z_3 = I_3(q_1 q_2 + q_1) + I_5 q_1 + I_9 q_1 + I_4 + I_7 + I_8 + I_{10} + I_{13} + I_{14} + I_{15} \quad (15) \]
\[ Z_4 = I_2(q_1 + q_1 q_2) + I_3 q_1 + I_5 q_1 + I_{16} q_1 + I_9 q_1 + I_{11} q_1 + I_6 + I_{10} + I_{12} + I_{13} + I_{14} \quad (16) \]
\[ Z_5 = I_2(q_1 + q_1 q_2) + I_3(q_1 + q_1 q_2) + I_6 q_1 + I_{11} q_1 + I_1 + I_4 + I_7 + I_8 + I_{12} + I_{16} \quad (17) \]

Expressions 13 to 17 on simplification yield the following

\[ Z_1 = TBL q_1 q_2 + RBL q_1 q_2 + TRL q_1 q_2 + TRBL + TRB q_1 q_2 + TRL \quad (18) \]
\[ Z_2 = RL q_1 q_2 + TB q_1 q_2 + TBL + RBL + TRB q_1 q_2 + TRL \quad (19) \]
\[ Z_3 = TB q_1 + TB q_2 + TBL q_1 + TBL + TBL + TBL \quad (20) \]
\[ Z_4 = BL q_2 + RB q_1 + BL q_1 + TR q_1 + TRB + TRL + TL q_1 + TRL q_1 q_2 \quad (21) \]
\[ Z_5 = TR q_2 + RBL q_1 + BL q_2 + TL q_1 + TRB + TRB + TRL \quad (22) \]
3.6 Combinatorial Network - Phase 2

A flow chart for Phase 2 operation is given in figure 5. The elements used to implement this network are shown in figure 6.

A row or column of the grid is considered illuminated if one of the cells in the row or column is illuminated. Let $s_{ij}$ represent the output from a cell which is in the $i$th row and $j$th column of the grid.

$s_{ij} = 0$ or $1$ depending on whether the cell is illuminated or not. The binary variables $c_j$ and $r_i$ defined by expressions 23 and 24 give the information whether the $j$th column and $i$th row are illuminated.

$$c_j = \sum_{i=0}^{9} s_{ij} \quad j = 0, 7 \tag{23}$$

$$r_i = \sum_{j=0}^{7} s_{ij} \quad i = 0, 9 \tag{24}$$

Figure 7 shows a realization of $c_j$s and $r_i$s using NOR logic.

Figure 8 shows an input selector which is used to select pairs of rows or columns one at a time for comparison.

At the start of Phase 2 operation the state of each flipflop in the Phase 2 network is as follows.
Designation | Type | Location | State
--- | --- | --- | ---
FA1 | Reset - set (bistable) | Figure 9 | 0
FA2 | " | " | 0
FA3 | " | Figure 8 | 1
FA4 | " | " | 0
FA5 | " | " | 0
FA6 | " | " | 0
FA7 | " | " | 0
FA8 | unistable | Figure 10 | 0
FA9 | " | " | 0
FA10 | Reset - set (bistable) | Figure 11 | 0

It will be seen later how the initialisation of these flipflops takes place.

It is assumed for the sake of clarity in describing the Phase 2 operation, that the character has been centered in the right-left direction during Phase 1. This sets flipflop FA2 which in turn allows the binary valued $r_i$ signals to flow through (figures 9 and 12). The $c_j$ signals remain inhibited as FA1 is in the '0' state.

The binary variables $H_1, \ldots, L_5$ (figures 9 and 12) assume the logical values of the $r_i$ variables as follows.

$H_1 = r_0$
$L_1 = r_9$
$H_2 = r_1$
$L_2 = r_8$
$H_3 = r_2$
$L_3 = r_7$
$H_4 = r_3$
$L_4 = r_6$
$H_5 = r_4$
$L_5 = r_5$
The variables $H_1$ to $H_5$ and $L_1$ to $L_5$ represent the rows above and below the horizontal axis of the grid respectively.

In the input selector $FA_3$ is in the 1 state to start with. This allows only $H_1$ and $L_1$ to pass through while $H_2$ $L_2$ etc. are inhibited. Thus in figure 10, $H$ and $L$ assume the values of $H_1$ and $L_1$. The $H$ and $L$ signals are used in the comparison circuit (figure 11) to determine the proper circuit output. Listed below are the different possible outputs.

<table>
<thead>
<tr>
<th>H</th>
<th>L</th>
<th>Required operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Skip to next inner pair</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>Shift character (half-up)</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>Shift character (half-down)</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Character centered</td>
</tr>
</tbody>
</table>

The skip to next inner pair for further comparison is effected as follows. Since $HL$ is 00, $FA_8$ is set (figure 10). Shaper $S$ emits a pulse which sets $FA_9$. $FA_9$ returns to '0' state after $t$ seconds. The neat negative pulse on the $cc$ line resets $FA_3$ and sets $FA_7$. Therefore the input selector changes $H$ to $H_2$ and $L$ to $L_2$. If $HL$ is still '00', the pulse emerging from the delay line passes through to set $FA_9$ again. The delay $\Delta$ must be greater than the time constant $t$ of $FA_9$. During $\Delta-t$, $H$ and $L$ are allowed to take on the values corresponding to $H_2$ and $L_2$. The pulse would be recycled as many times as required.

If the Phase 1 network has centered the character in the top-bottom direction, $c^s$ would be passed through to become $H_2$, $H_3$, $H_4$, $H_5$, $L_2$, $L_3$, $L_4$, and $L_5$. $H_1$ and $L_1$ would be '00' (from figures 9 and 12) and the comparison would start from $H_2$ $L_2$. 

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
When the character is centered by the Phase 2 network, FA10 is set which resets FA1 and FA2. When both FA1 and FA2 are in the 0 state, signal in the R line of figure 8 sets FA3 and resets FA4, FA5, FA6 and FA7. Also, the Phase 1 network is returned to its starting state D, by holding the primary input at I4 (figure 1). The output associated with total state I4 D is inhibited. To process a new pattern the 'START' button is to be pushed once.

3.7 System Operation

The character is mounted on its frame and the initialise button is pushed once. This sets FA10 which initialises the Phase 1 and Phase 2 networks as described in section 3.6. Pressing the 'START' button resets FA10 and allows the sequential circuit to read the primary inputs and change state accordingly. If for any reason, the character on the grid while being centered is lost, the centering process is stopped and the networks are initialised. Pressing the START button would initiate the centering process again. Lamp annuncicators are used for displaying messages, like 'character partially centered' etc.

3.8 Hardware Implementation

A scheme to realize the logic circuits is shown in Appendix A. In arriving at this scheme, the following considerations have not been taken into account:

1) loading rules for interconnecting the logic elements
2) number of terminals available on each element.

On account of this, at the time of building the device proposed in this thesis, it would be necessary to modify the implementation scheme.

The boolean expressions for the internal and output state variables of the sequential network are summarized in Table 9.
CHAPTER IV

SIMULATION OF THE CENTERING DEVICE

The sequential network part of the logic device was simulated on an IBM 1620 Model II computer. The general system flow chart for simulation is given in figure 24. A listing of the program is attached as Appendix C. The purpose of the simulation has been twofold.

1. To check the behaviour of the sequential machine, for proper operation, as envisaged in the flowtable.

2. To serve as part of an overall simulation program of the entire pattern recognition system proposed by Dydyk. The program solves Boolean expressions 5 to 17 for finding the next states and output states. The program does not take into account the stray delays involved in the logic elements. While all operations to shift the character are performed automatically, magnification and contraction have to be performed manually. The program prints out on the typewriter the grid contents whenever such operations are called for. One has to enter the data of the magnified or contracted character for the program to carry on. This is a drawback of the simulation program.

Input to the program is from cards. A '1' is entered for a cell that is illuminated and a '0' for one that is not illuminated. Figure 25 shows the simulated grid. It has 32 rows and 28 columns. The simulated grid is larger in size than the real grid to enable one to enter all the data pertinent to the character.

Discussion of Results of Simulation:

Results of a typical simulation run is contained in Appendix D. The character T is shown at the start of the centering operation. It
is touching the left side of the grid and hence the primary input is '001' in decimal code. This corresponds to a primary input of 12 in the flowtable. As the sequential network is in state 'D' to start with, it changes to state B (coding is 000100). The output associated with stable state B is 00011 (full-right). The character is shifted two blocks to the right, whereupon the primary input becomes 15 and the internal state changes to J(101101). The output associated with total stable state 15, J is 'half left'. When the character is shifted half block to the left the primary input and next state change to 16 and D (000000). When the corresponding output command 'contract' is performed, the internal state changes to A(000010). This requires the character to be magnified and the grid contents as at this time of the program are printed out row by row. On magnification, the primary input changes back to 16 and the total state becomes 16, A. At this time the pattern has been partially centered (a message to this effect is printed out). Control is transferred to Phase 2 network, which shifts the character such that it is symmetrical with respect to the top and bottom sides of the grid. It is seen that the character occupies 13 rows on the simulated grid or about five and a half rows on the grid. A character as this one, can get centered only if the output commands of Phase 2 network call for shift operations over half a block or cell.
CHAPTER V

CONCLUSIONS

The use of photocells for the grid has certain drawbacks. When a character is projected on the grid, the cells near the projected character would be activated to some extent and erroneous inputs are likely to occur. Use of the additional pair of columns and rows on the sides of the grid to supply the primary inputs, has permitted maximum expansion of the character within the grid. To minimize erroneous inputs the operation of electromechanical drives to shift the pattern should be as fast as possible. Simulation of the device shows that its performance is satisfactory. In the absence of information regarding other logic devices to position characters or patterns on grids such as the one used here it is not possible to make any comparison. The logic device proposed here is however considered novel for a pattern recognition system such as the one proposed by Dydyk. It can also be made to handle a string of characters by modifying the grid size and suitably changing the Phase 2 design.


APPENDIX A

DIAGRAMS
Fig. 1 Grid of Photocells

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Fig. 2 Hazard from Complementary Commands
Fig. 3 Representation of a Sequential Machine
Fig. 4 State Device
Fig. 5 Flow Chart for Phase 2 Operation
Fig. 6 Elements Used in Logic Drawings
Fig. 7 Realization of $c_s$ and $r_s$
Fig. 8 Phase 2 - Input Selector
Fig. 9 Phase 2 - Circuitry #1
Fig. 10 Phase 2 - Circuitry #2
Fig. 11 Phase 2 - Circuitry #3
Fig. 12 Phase 2 - Circuitry #4

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Fig. 13 Phase 1 - Network #1
Fig. 14 Phase 1 - Network #2
Fig. 15 Phase 1 - Network #3
Fig. 16 Phase 1 - Network #4
Fig. 17 Phase 1 - Network #5
Fig. 18 Phase 1 - Network #6
Fig. 19 Phase 1 - Network #7
Fig. 20 Phase 1 - Network #8
Fig. 21 Phase 1 - Network #9
Fig. 22 Phase 1 - Network #10
Fig. 23 Phase 1 - Network #11

\[ \Delta t \text{ to be just greater than the time constant of state device.} \]
Fig. 24 General System Flow Chart for Simulation
Fig. 25 Simulated Grid
APPENDIX B

TABLES
TABLE I
Output Codes

<table>
<thead>
<tr>
<th>Decimal Coding</th>
<th>Operation</th>
<th>Binary Coding</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Magnify</td>
<td>000001</td>
</tr>
<tr>
<td>3</td>
<td>Full right</td>
<td>000111</td>
</tr>
<tr>
<td>4</td>
<td>Full left</td>
<td>001000</td>
</tr>
<tr>
<td>5</td>
<td>Full up</td>
<td>001011</td>
</tr>
<tr>
<td>6</td>
<td>Full down</td>
<td>001110</td>
</tr>
<tr>
<td>7</td>
<td>Half up</td>
<td>001111</td>
</tr>
<tr>
<td>8</td>
<td>Half down</td>
<td>010000</td>
</tr>
<tr>
<td>9</td>
<td>Half right</td>
<td>010001</td>
</tr>
<tr>
<td>10</td>
<td>Half left</td>
<td>010010</td>
</tr>
<tr>
<td>11</td>
<td>Contract-right</td>
<td>010111</td>
</tr>
<tr>
<td>12</td>
<td>Contract-left</td>
<td>011000</td>
</tr>
<tr>
<td>13</td>
<td>Contract-up</td>
<td>011001</td>
</tr>
<tr>
<td>14</td>
<td>Contract-down</td>
<td>011110</td>
</tr>
<tr>
<td>16</td>
<td>Pattern centred in top-bottom direction</td>
<td>100000</td>
</tr>
<tr>
<td>17</td>
<td>Pattern centred in all directions</td>
<td>100001</td>
</tr>
<tr>
<td>18</td>
<td>Pattern centred in right-left direction</td>
<td>100100</td>
</tr>
<tr>
<td>19</td>
<td>Contract</td>
<td>100111</td>
</tr>
</tbody>
</table>
## TABLE II

### Initial Flowtable

<table>
<thead>
<tr>
<th>Internal State</th>
<th>I2</th>
<th>I3</th>
<th>I4</th>
<th>I5</th>
<th>I6</th>
<th>I7</th>
<th>I8</th>
<th>I9</th>
<th>I10</th>
<th>I11</th>
<th>I12</th>
<th>I13</th>
<th>I14</th>
<th>I15</th>
<th>I16</th>
</tr>
</thead>
<tbody>
<tr>
<td>AA</td>
<td>A1</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>AB</td>
<td>DC</td>
<td>DD</td>
<td>H</td>
<td>DE</td>
<td>AC</td>
<td>DG</td>
<td>DH</td>
<td>DI</td>
<td>DJ</td>
</tr>
<tr>
<td>B</td>
<td>A</td>
<td>A3</td>
<td>C</td>
<td>DA</td>
<td>JA</td>
<td>DB</td>
<td>DC</td>
<td>DD</td>
<td>H</td>
<td>DE</td>
<td>DF</td>
<td>DG</td>
<td>DH</td>
<td>DI</td>
<td>DJ</td>
</tr>
<tr>
<td>C</td>
<td>AA</td>
<td>B</td>
<td>C5</td>
<td>DA</td>
<td>E</td>
<td>DB</td>
<td>DC</td>
<td>DD</td>
<td>JB</td>
<td>DE</td>
<td>DF</td>
<td>DG</td>
<td>DH</td>
<td>DI</td>
<td>DJ</td>
</tr>
<tr>
<td>DA</td>
<td>AA</td>
<td>A</td>
<td>C</td>
<td>DA13</td>
<td>E</td>
<td>DB</td>
<td>DC</td>
<td>DD</td>
<td>H</td>
<td>DE</td>
<td>DF</td>
<td>DG</td>
<td>DH</td>
<td>DI</td>
<td>DJ</td>
</tr>
<tr>
<td>E</td>
<td>AA</td>
<td>KA</td>
<td>C</td>
<td>DA</td>
<td>E4</td>
<td>DB</td>
<td>DC</td>
<td>DD</td>
<td>H</td>
<td>DE</td>
<td>DF</td>
<td>DG</td>
<td>DH</td>
<td>DI</td>
<td>DJ</td>
</tr>
<tr>
<td>DB</td>
<td>AA</td>
<td>B</td>
<td>C</td>
<td>DA</td>
<td>DB19</td>
<td>DC</td>
<td>DD</td>
<td>H</td>
<td>DE</td>
<td>DF</td>
<td>DG</td>
<td>DH</td>
<td>DI</td>
<td>DJ</td>
<td>DK</td>
</tr>
<tr>
<td>DC</td>
<td>AA</td>
<td>A</td>
<td>C</td>
<td>DA</td>
<td>E</td>
<td>DB</td>
<td>DC13</td>
<td>DD</td>
<td>H</td>
<td>DE</td>
<td>DF</td>
<td>DG</td>
<td>DH</td>
<td>DI</td>
<td>DJ</td>
</tr>
<tr>
<td>DD</td>
<td>AA</td>
<td>B</td>
<td>C</td>
<td>DA</td>
<td>E</td>
<td>DB</td>
<td>DC</td>
<td>DD13</td>
<td>H</td>
<td>DE</td>
<td>DF</td>
<td>DG</td>
<td>DH</td>
<td>DI</td>
<td>DJ</td>
</tr>
<tr>
<td>H</td>
<td>AA</td>
<td>B</td>
<td>KB</td>
<td>DA</td>
<td>E</td>
<td>DB</td>
<td>DC</td>
<td>DD</td>
<td>H6</td>
<td>DE</td>
<td>DF</td>
<td>DG</td>
<td>DH</td>
<td>DI</td>
<td>DJ</td>
</tr>
<tr>
<td>DE</td>
<td>AA</td>
<td>B</td>
<td>C</td>
<td>DA</td>
<td>E</td>
<td>DB</td>
<td>DC</td>
<td>DD</td>
<td>H</td>
<td>DE14</td>
<td>DF</td>
<td>DG</td>
<td>DH</td>
<td>DI</td>
<td>DJ</td>
</tr>
<tr>
<td>DF</td>
<td>AA</td>
<td>B</td>
<td>C</td>
<td>DA</td>
<td>E</td>
<td>DB</td>
<td>DC</td>
<td>DD</td>
<td>H</td>
<td>DE19</td>
<td>DF</td>
<td>DG</td>
<td>DH</td>
<td>DI</td>
<td>DJ</td>
</tr>
<tr>
<td>DG</td>
<td>AA</td>
<td>B</td>
<td>C</td>
<td>DA</td>
<td>E</td>
<td>DB</td>
<td>DC</td>
<td>DD</td>
<td>H</td>
<td>DE</td>
<td>DF</td>
<td>DG11</td>
<td>DH</td>
<td>DI</td>
<td>DJ</td>
</tr>
<tr>
<td>DH</td>
<td>AA</td>
<td>B</td>
<td>C</td>
<td>DA</td>
<td>E</td>
<td>DB</td>
<td>DC</td>
<td>DD</td>
<td>H</td>
<td>DE</td>
<td>DF</td>
<td>DG14</td>
<td>DI</td>
<td>DJ</td>
<td>DK</td>
</tr>
<tr>
<td>DI</td>
<td>AA</td>
<td>B</td>
<td>C</td>
<td>DA</td>
<td>E</td>
<td>DB</td>
<td>DC</td>
<td>DD</td>
<td>H</td>
<td>DE</td>
<td>DF</td>
<td>DG14</td>
<td>DJ12</td>
<td>DK</td>
<td></td>
</tr>
<tr>
<td>DJ</td>
<td>AA</td>
<td>B</td>
<td>C</td>
<td>DA</td>
<td>E</td>
<td>DB</td>
<td>DC</td>
<td>DD</td>
<td>H</td>
<td>DE</td>
<td>DF</td>
<td>DG</td>
<td>DH</td>
<td>DI</td>
<td>DJ</td>
</tr>
<tr>
<td>DK</td>
<td>AA</td>
<td>B</td>
<td>C</td>
<td>DA</td>
<td>E</td>
<td>DB</td>
<td>DC</td>
<td>DD</td>
<td>H</td>
<td>DE</td>
<td>DF</td>
<td>DG</td>
<td>DH</td>
<td>DI</td>
<td>DJ</td>
</tr>
</tbody>
</table>

**Notes:**
- 

**Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
<table>
<thead>
<tr>
<th>State</th>
<th>Nature</th>
<th>Disposition of character on grid</th>
</tr>
</thead>
<tbody>
<tr>
<td>AA</td>
<td>Initial</td>
<td>Touches none of the sides</td>
</tr>
<tr>
<td>B</td>
<td>Initial</td>
<td>Touches left side only</td>
</tr>
<tr>
<td>C</td>
<td>Initial</td>
<td>Touches bottom side only</td>
</tr>
<tr>
<td>DA</td>
<td>Initial</td>
<td>Touches bottom and left sides</td>
</tr>
<tr>
<td>E</td>
<td>Initial</td>
<td>Touches right side only</td>
</tr>
<tr>
<td>DB</td>
<td>Initial</td>
<td>Touches right and left sides</td>
</tr>
<tr>
<td>DC</td>
<td>Initial</td>
<td>Touches right and bottom sides</td>
</tr>
<tr>
<td>DD</td>
<td>Initial</td>
<td>Touches right, bottom and left sides</td>
</tr>
<tr>
<td>H</td>
<td>Initial</td>
<td>Touches top side only</td>
</tr>
<tr>
<td>DE</td>
<td>Initial</td>
<td>Touches top and left sides</td>
</tr>
<tr>
<td>DF</td>
<td>Initial</td>
<td>Touches top and bottom sides</td>
</tr>
<tr>
<td>DG</td>
<td>Initial</td>
<td>Touches top, bottom and left sides</td>
</tr>
<tr>
<td>DH</td>
<td>Initial</td>
<td>Touches top and right sides</td>
</tr>
<tr>
<td>DI</td>
<td>Initial</td>
<td>Touches top, right and left sides</td>
</tr>
<tr>
<td>DJ</td>
<td>Initial</td>
<td>Touches top, right and bottom sides</td>
</tr>
<tr>
<td>DK</td>
<td>Initial</td>
<td>Touches all the sides</td>
</tr>
<tr>
<td>AB</td>
<td>Final</td>
<td>Centered in right-left direction</td>
</tr>
<tr>
<td>AC</td>
<td>Final</td>
<td>Centered in top-bottom direction</td>
</tr>
<tr>
<td>AD</td>
<td>Final</td>
<td>Centered in all directions</td>
</tr>
<tr>
<td>JA</td>
<td>Intermediate</td>
<td>Touches right side only</td>
</tr>
<tr>
<td>JB</td>
<td>Intermediate</td>
<td>Touches top side only</td>
</tr>
<tr>
<td>State</td>
<td>Nature</td>
<td>Disposition of character on grid</td>
</tr>
<tr>
<td>-------</td>
<td>---------</td>
<td>----------------------------------------</td>
</tr>
<tr>
<td>JC</td>
<td>Final</td>
<td>Centered in right-left direction</td>
</tr>
<tr>
<td>JD</td>
<td>Final</td>
<td>Centered in top-bottom direction</td>
</tr>
<tr>
<td>KA</td>
<td>Intermediate</td>
<td>Touches left side only</td>
</tr>
<tr>
<td>KB</td>
<td>Intermediate</td>
<td>Touches bottom side only</td>
</tr>
<tr>
<td>KC</td>
<td>Final</td>
<td>Centered in top-bottom direction</td>
</tr>
<tr>
<td>KD</td>
<td>Final</td>
<td>Centered in right-left direction</td>
</tr>
</tbody>
</table>
### TABLE IV

Condensed Flowtable

<table>
<thead>
<tr>
<th>Internal State</th>
<th>I1</th>
<th>I2</th>
<th>I3</th>
<th>I4</th>
<th>I5</th>
<th>I6</th>
<th>I7</th>
<th>I8</th>
<th>I9</th>
<th>I10</th>
<th>I11</th>
<th>I12</th>
<th>I13</th>
<th>I14</th>
<th>I15</th>
<th>I16</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A</td>
<td>1</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>A</td>
<td>18</td>
<td>D</td>
<td>D</td>
<td>H</td>
<td>D</td>
<td>A</td>
<td>16</td>
<td>D</td>
<td>D</td>
</tr>
<tr>
<td>B</td>
<td>A</td>
<td>B</td>
<td>3</td>
<td>C</td>
<td>D</td>
<td>J</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>H</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>D</td>
</tr>
<tr>
<td>C</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>J</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>D</td>
</tr>
<tr>
<td>D</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>13</td>
<td>D</td>
<td>19</td>
<td>D</td>
<td>13</td>
<td>D</td>
<td>13</td>
<td>H</td>
<td>D</td>
<td>14</td>
<td>D</td>
</tr>
<tr>
<td>E</td>
<td>A</td>
<td>K</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>4</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>H</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>D</td>
</tr>
<tr>
<td>H</td>
<td>A</td>
<td>B</td>
<td>K</td>
<td>D</td>
<td>E</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>H</td>
<td>6</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>D</td>
</tr>
<tr>
<td>J</td>
<td>J</td>
<td>18</td>
<td>J</td>
<td>16</td>
<td>J</td>
<td>10</td>
<td>D</td>
<td></td>
<td></td>
<td>J</td>
<td>8</td>
<td>D</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>K</td>
<td>K</td>
<td>9</td>
<td>K</td>
<td>7</td>
<td>K</td>
<td>18</td>
<td>D</td>
<td></td>
<td></td>
<td>K</td>
<td>16</td>
<td>D</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>N</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
<td>G</td>
<td>H</td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>0 φ</td>
<td>0 φ</td>
<td>0 φ</td>
<td>0 φ</td>
<td>0 φ</td>
<td>0 φ</td>
<td>0 φ</td>
<td>0 φ</td>
<td>0 φ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>1 φ</td>
<td>0 φ</td>
<td>1 φ</td>
<td>0 φ</td>
<td>1 φ</td>
<td>0 φ</td>
<td>1 φ</td>
<td>0 φ</td>
<td>1 φ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>1 φ</td>
<td>1 φ</td>
<td>0 φ</td>
<td>1 φ</td>
<td>0 φ</td>
<td>1 φ</td>
<td>0 φ</td>
<td>1 φ</td>
<td>0 φ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>0 φ</td>
<td>1 φ</td>
<td>0 φ</td>
<td>1 φ</td>
<td>0 φ</td>
<td>1 φ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>0 φ</td>
<td>1 φ</td>
<td>0 φ</td>
<td>1 φ</td>
<td>0 φ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>0 φ</td>
<td>1 φ</td>
<td>0 φ</td>
<td>1 φ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>0 φ</td>
<td>1 φ</td>
<td>0 φ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>0 φ</td>
<td>1 φ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>0 φ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td>1 φ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Preliminary state assignment
<table>
<thead>
<tr>
<th>Final State Assignment</th>
<th>Y_1</th>
<th>Y_2</th>
<th>Y_3</th>
<th>Y_4</th>
<th>Y_5</th>
<th>Y_6</th>
<th>Y_7</th>
<th>Y_8</th>
<th>Y_9</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>B</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>D</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>E</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>F</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>G</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>H</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>I</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>J</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>K</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
### TABLE VII

**Partial Output Matrix**

<table>
<thead>
<tr>
<th>Internal State</th>
<th>I1</th>
<th>I2</th>
<th>I3</th>
<th>I4</th>
<th>I5</th>
<th>I6</th>
<th>I7</th>
<th>I8</th>
<th>I9</th>
<th>I10</th>
<th>I11</th>
<th>I12</th>
<th>I13</th>
<th>I14</th>
<th>I15</th>
<th>I16</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>00001</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10010</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>00011</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>00101</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10001</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>01101</td>
<td>01011</td>
<td>01101</td>
<td>01101</td>
<td>01110</td>
<td>10011</td>
<td>01011</td>
<td>01110</td>
<td>01110</td>
<td>01110</td>
<td>01100</td>
<td>10011</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>00100</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>00110</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td>00110</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>01000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td>10010</td>
<td>10000</td>
<td>01010</td>
<td>01000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>K</td>
<td>01001</td>
<td>00111</td>
<td>10010</td>
<td>10000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
# TABLE VIII

Output Matrix

<table>
<thead>
<tr>
<th></th>
<th>I1</th>
<th>I2</th>
<th>I3</th>
<th>I4</th>
<th>I5</th>
<th>I6</th>
<th>I7</th>
<th>I8</th>
<th>I9</th>
<th>I10</th>
<th>I11</th>
<th>I12</th>
<th>I13</th>
<th>I14</th>
<th>I15</th>
<th>I16</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>00001</td>
<td>00101</td>
<td>01101</td>
<td>01100</td>
<td>10010</td>
<td>01101</td>
<td>01110</td>
<td>01100</td>
<td>01110</td>
<td>01100</td>
<td>10000</td>
<td>01011</td>
<td>01110</td>
<td>01110</td>
<td>01100</td>
<td>10001</td>
</tr>
<tr>
<td>B</td>
<td>00001</td>
<td>00101</td>
<td>01101</td>
<td>01100</td>
<td>10010</td>
<td>01101</td>
<td>01110</td>
<td>01100</td>
<td>01110</td>
<td>01100</td>
<td>10000</td>
<td>01011</td>
<td>01110</td>
<td>01110</td>
<td>01100</td>
<td>10011</td>
</tr>
<tr>
<td>C</td>
<td>00001</td>
<td>00101</td>
<td>01101</td>
<td>00100</td>
<td>10011</td>
<td>01101</td>
<td>01101</td>
<td>01000</td>
<td>01110</td>
<td>10011</td>
<td>01011</td>
<td>01110</td>
<td>01110</td>
<td>01100</td>
<td>10011</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>00001</td>
<td>00101</td>
<td>01101</td>
<td>00100</td>
<td>10011</td>
<td>01101</td>
<td>01101</td>
<td>01110</td>
<td>01110</td>
<td>10011</td>
<td>01011</td>
<td>01110</td>
<td>01110</td>
<td>01100</td>
<td>10011</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>00001</td>
<td>01001</td>
<td>01101</td>
<td>01000</td>
<td>10011</td>
<td>01101</td>
<td>01101</td>
<td>01110</td>
<td>01110</td>
<td>10011</td>
<td>01011</td>
<td>01110</td>
<td>01110</td>
<td>01100</td>
<td>10011</td>
<td></td>
</tr>
<tr>
<td>H</td>
<td>00001</td>
<td>00111</td>
<td>01101</td>
<td>00100</td>
<td>10011</td>
<td>01101</td>
<td>01101</td>
<td>01110</td>
<td>01110</td>
<td>10011</td>
<td>01011</td>
<td>01110</td>
<td>01110</td>
<td>01100</td>
<td>10011</td>
<td></td>
</tr>
<tr>
<td>J</td>
<td>00001</td>
<td>10010</td>
<td>10000</td>
<td>01010</td>
<td>10011</td>
<td>01000</td>
<td>10011</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>K</td>
<td>00001</td>
<td>01001</td>
<td>00111</td>
<td>10010</td>
<td>10011</td>
<td>01000</td>
<td>10011</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>


TABLE IX

Summary of Boolean Expressions to Realize Phase 1 Network

\[ Q_1 = I1+I4+I7+I8+I10+I12+I13+I14+I15 \]
\[ +I2 \ y_1^2 y_1^3 y_1^9 + (I6+I11+I16) \ y_1^{6,11,16} \]  
(5)

\[ Q_2 = (I2+I3+I5+I9) \ y_2^{2,3,5,9} \]  
(6)

\[ Y_1^9 = I3+I2 \ q_1+I5 \ q_1+I9 \ q_1 \]  
(7)

\[ Y_1^2 = I5 \ q_1 + I2 \ q_1 q_2 + I3 \ q_1 q_2 + I5 \ q_1 q_2 + I9 \ q_1 q_2 \]  
(8)

\[ Y_1^3 = I9 + I2 \ q_1 + I3 \ q_1 + I5 \ q_1 \]  
(9)

\[ Y_1^{6,11,16} = I1 + I6 \ q_1 + I11 \ q_1 + I16 \ q_1 \]  
(10)

\[ Y_1^5 = I2 + I3 \ q_1 + I5 \ q_1 + I9 \ q_1 \]  
(11)

\[ Y_2^{2,3,5,9} = I5 \ q_1 + I9 \ q_1 + I2 \ q_1 q_2 + I3 \ q_1 q_2 \]
\[ + I5 \ q_1 q_2 + I9 \ q_1 q_2 \]  
(12)

\[ Z_1 = \bar{TBL} \ q_1 \bar{q}_2 + \bar{RBL} \ q_1 \bar{q}_2 + \bar{TBL} + \bar{RBL} + TBL + TRB \]  
(18)

\[ Z_2 = \bar{RL} \ q_1 q_2 + \bar{TBL} \ q_1 \bar{q}_2 + \bar{RBL} + TRB + RL \ q_1 \bar{q}_2 + TRL \]  
(19)

\[ Z_3 = \bar{TB} \ q_1 \bar{q}_2 + \bar{RL} \ q_1 \bar{q}_2 + \bar{TBL} + TBL + TRL + \bar{TBL} \]  
(20)
\[ Z_4 = \bar{B}L \bar{q}_2 + R\bar{B} q_1 + \bar{B}L \bar{q}_1 + T\bar{R} \bar{q}_2 + \bar{TRB} + TRL + TL \bar{q}_1 + \bar{TRBL} q_1 q_2 \] (21)

\[ Z_5 = \bar{TR} q_2 + \bar{RBL} \bar{q}_1 + BL q_2 + TL \bar{q}_1 + \bar{TBRBL} + \bar{TRB} + TBL \] (22)
APPENDIX C

SIMULATION PROGRAM LISTING
SPS PROGRAM....SIMULATION OF CENTERING DEVICE

SIMULATED GRID SIZE 32 ROWS x 28 COLUMNS.
ENTER GRID DATA OF UNCENTERED PATTERN ROW BY ROW
THROUGH CARDS USING COLUMNS 1 TO 28 ONLY.
IF TYPICAL SIMULATED GRID CELL IS DENOTED BY
GRID(I,J) THE PRIMARY INPUT CELLS ARE IN FORTRAN
NOTATION....TOP ((GRID(I,J)+J=7,22),I=5,6)
RIGHT ((GRID(I,J)+J=7,26),J=23,24)
BOTTOM ((GRID(I,J)+J=7,22),J=27,23)
LEFT ((GRID(I,J)+J=7,26),J=5,6)

PROGRAM PRINTS OUT GRID CONTENTS FOR CONTRACTION AND
MAGNIFICATION. CONTRACT/MAGNIFY THE PATTERN TO THE
EXTENT OF ONE SIMULATED GRID CELL. AND ENTER GRID DATA
ROW BY ROW THROUGH TYPEWRITER.
VECTOR ZVAR CONTAINS THE OUTPUT VARIABLES.
VECTOR YVAR CONTAINS THE INTERNAL STATE VARIABLES.
VECTOR QUE CONTAINS THE O VARIABLES

ENTER DSA BEGIN
ENTER DSA BEGIN
LINE DS 2
DC 1
AREA DS 23

REMARK DC 1,
CONST DAC 26:OPERATION IS CONTRACTION
CONST DAC 26:OPERATION IS CONTRACTION UP
CONST DAC 30:OPERATION IS CONTRACTION LEFT
CONST DAC 31:OPERATION IS CONTRACTION RIGHT
CONST DAC 33:OPERATION IS CONTRACTION DOWN
CONST DAC 33: PATTERN ON GRID BEFORE CENTERING
CTR DS 5
FINAL DAC 26: PATTERN HAS BEEN CENTERED
GO DSA TABLE
GDUMY DSC 26:0
GDUMY DSC 26:0
GRID DSS 696
GDUMY DSS 64
GDUMY DSS 80
MAG DAC 27:OPERATION IS MAGNIFICATION
NPUT DS 16
NO DAC 17: UNDEFINED OUTPUT
NEW DAC 35: PRESS START TO PROCESS NEW PATTERNS
POUT1 DAC 22: THE PRIMARY INPUT IS 0
POUT2 DAC 41: PATTERN CENTERED IN TOP BOTTOM DIRECTION
POUT3 DAC 41: PATTERN CENTERED IN RIGHT LEFT DIRECTION
DC 1
QUE DS 2
RIGHT DS 5
RUN DSA JUMPY
SCRIN DSA GRID
STOR DS 5
TEMP DS 5
TEST DS 5
UPSETDSA RIGHT, TEMP
    DC 1.0
UPSET2DSA TEMP, RIGHT
    DC 1.0
WEIGHT DS 3
    DC 1.0
XPLEN1DAC 49, PRESS START TO ENTER GRID DATA TOP ROW TO BOTTOM
YSTAT DAC 24, THE INTERNAL STATE IS Y
YVAR DS 6
    DC 1.0
ZERO DC 20,0
    DC 1.0
ZERO1 DC 6.0
ZERO2 DC 20.0
    DC 1.0
ZERO3 DC 10.0
ZOUT DS 5
ZSTAT DAC 16, THE OUTPUT IS Z
ZVAR DS 5
    DC 1.0
ZWAIT DS 5
*
*INITIALISATION
*
COMENS
K 100102
TFM CTRL
TF YVAR, ZERO1
TFM ZVAR, 0
TF AREA, ZERO
SF GRID
B6A*+12
BLX*+12,0 (1)
BLX*+12,0 (2)
BLX*+12,0 (3)
BLX*+12,0 (4)
BLX*+12,0 (5)
BLX*+12,0 (6)
BLX*+12,0 (7)
MF UPSET1, UPSET1+1
MF UPSET2+1, UPSET2+1
*
*READING IN DATA
*
*PHASE 1 THE SEQUENTIAL CIRCUIT
*
BLX*+12+32 (e)
TF STOR, SCRIN
RNCDGCUMY
TDM GOUX+26, REMARK
TRN*STOR, GOUX+6
TDM GOUX+28, 0
AM STOR, 28
3CX*+60, 1 (2)
TDM.CTR,0
SLXMX*-12+20(1)
3D:ITD,TEST,1
AM:TEST,28
SCX:*-24+1(1)
3D:*-48+CTR
TDM.CTR,1
TFM:TEST,GRID+173
D:INPUT3+24
TFM:CTR,0
B:ITD+12

*ITAS:WEIGTS,9
B:INPUT1
ITDS:WEIGTS,4,9
B:INPUT2
ITCS:WEIGTS,2,9
B:INPUT3
ITDS:WEIGTS,1,9
B:INPUT4
SX:*+12+CC(7)
MM:WEIGTS,24+10
BX:*+12+CC(7)
B:BSX:TABLE-12, TABLE-9(7)
B:

*TABLE TDM:INPUT+1
B:SRC101
TDM:INPUT-1+1
B:SRC101
TDM:INPUT-2+1
B:SRC101
TDM:INPUT-3+1
B:SRC101
TDM:INPUT-4+1
B:SRC101
TDM:INPUT-5+1
B:SRC101
TDM:INPUT-6+1
B:SRC101
TDM:INPUT-7+1
B:SRC101
TDM:INPUT-8+1
B:SRC101
TDM:INPUT-9+1
B:SRC101
TDM:INPUT-10+1
B:SRC101
TDM:INPUT-11+1
B:SRC101
TDM:INPUT-12+1
B:SRC101
TDM:INPUT-13+1
B:SRC101

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
* ROUTINE TO FIND C1

SRCH01 WATYP=OUT1
  WNTYWEIGT=2
  K *CO102
  K *CO104
  DD LPHA01+24*NPUT-1
  DD LPHA01+49*NPUT-2
  DD LPHA01+72*NPUT-4
  DD LPHA01+95*NPUT-5
  DD LPHA01+98*NPUT-10
  DD LPHA01+99*NPUT-15
  DD LPHA01+120*NPUT-8
  DD S3TA01*NPUT
  DD S3TA01*NPUT-3
  DD S3TA01*NPUT-6
  DD S3TA01*NPUT-7
  DD S3TA01*NPUT-9
  DD S3TA01*NPUT-11
  DD S3TA01*NPUT-12
  DD S3TA01*NPUT-13
  DD S3TA01*NPUT-14

LPHA01 DOM QUE=1,0
  3 SRCH02
  DD S3TA01*YVAR-5
  DD LPHA01
  DD S3TA01*YVAR-3
  DD LPHA01
  DD S3TA01*YVAR-2
  DD LPHA01
  DD S3TA01*YVAR-1
  DD LPHA01
  DD S3TA01*YVAR
  DD LPHA01

S3TA01 DOM QUE=1,1

* ROUTINE TO FIND G2

SRCH02 DD LPHA02+24*NPUT-1
  DD LPHA02+24*NPUT-2
  DD LPHA02+24*NPUT-4
  DD LPHA02+24*NPUT-3

LPHA02 DOM QUE=0
  3 SRCH03
  DD S3TA02*YVAR-4
  DD LPHA02

S3TA02 DOM QUE=1

* ROUTINE TO FIND Y12

*
SRCH035D  SETHA3, INPU T-4
  6D  LPHA03+24, INPU T-1
  6D  LPHA03+24, INPU T-2
  6D  LPHA03+24, INPU T-3
LPHA03TMD  YVAR-5.0
  3  SRCH04
  6D  SETHA3, GUE-1
  3  LPHA03
SETHA3TMD  YVAR-5.1
*  *ROUTINE TO FIND Y2359
  *  *
SRCH048D  LPHA04+24, INPU T-4
  6D  LPHA04+24, INPU T-8
  6D  LPHA04+43, INPU T-1
  6D  LPHA04+43, INPU T-2
LPHA04TMD  YVAR-4.0
  6  SRCH05
  6D  *+24, GUE-1
  3  SETHA4
  6D  *+24, GUE
  3  LPHA04
  6D  SETHA4, GUE-1
  3  LPHA04
SETHA4TMD  YVAR-4.1
*  *ROUTINE TO FIND Y13
  *  *
SRCH055D  SETHA5, INPU T-3
  6D  LPHA05+24, INPU T-1
  6D  LPHA05+24, INPU T-2
  6D  LPHA05+24, INPU T-4
LPHA05TMD  YVAR-3.0
  3  SRCH06
  6D  SETHA5, GUE-1
  3  LPHA05
SETHA5TMD  YVAR-3.1
*  *ROUTINE TO FIND Y13
  *  *
SRCH055D  LPHA05+24, INPU T-2
  6D  LPHA05+24, INPU T-4
  6D  LPHA05+24, INPU T-8
  6D  SETHA5, INPU T-1
LPHA05TMD  YVAR-2.0
  3  SRCH07
  6D  SETHA5, GUE-1
  3  LPHA06
SETHA5TMD  YVAR-2.1
*  *ROUTINE TO FIND Y6, 11, 16
  *  *
SRCH076D  LPHA07+24, INPU T-5
  6D  LPHA07+24, INPU T-10
  6D  LPHA07+24, INPU T-15

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
* 
* *ROUTINE TO FIND Y19
* 
SRCH08D LPHA03+24,INPUT-1
3D LPHA03+24,INPUT-4
3D LPHA03+24,INPUT-3
3D BETA08,INPUT-2

LPHA07TDM YVAR=1+1

* 
* *ROUTINE TO FIND Z1
* 
SRCH09D LPHA09+24,INPUT-1
3D LPHA09+24,INPUT-4
3D LPHA09+24,INPUT-4
3D LPHA09+72,INPUT-5
3D LPHA09+72,INPUT-8
3D BETA09,INPUT-10
3D BETA09,INPUT-15
3D BETA09,INPUT-6

LPHA09TDM ZVAR=4+0
3 SRCH10
3D *+24,QUE=1
3 LPHA09
3D LPHA09,QUE
3 BETA09
3D *+24,QUE=1
3 LPHA09
3D BETA09,QUE
3 LPHA09

BETA09TDM ZVAR=4+1

* 
* *ROUTINE TO FIND Z2
* 
SRCH10D LPHA10+24,INPUT-4
3D LPHA10+24,INPUT-8
3D LPHA10+24,INPUT-1
3D BETA10,INPUT-3
3D BETA10,INPUT-6
3D BETA10,INPUT-7
3D BETA10,INPUT-9
3D BETA10,INPUT-11
3D BETA10,INPUT-12
3D BETA10,INPUT-13
3D BETA10,INPUT-14

LPHA10TDM ZVAR=3+0

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
BETA10: ZVAR-3,1
*
*ROUTINE TO FIND Z3
*
SRCH1100 LPHA11+24*NPUT-2
50 LPHA11+72*NPUT-4
50 LPHA11+72*NPUT-8
50 BETA11*NPUT-3
50 BETA11*NPUT-6
50 BETA11*NPUT-7
50 BETA11*NPUT-9
50 BETA11*NPUT-12
50 BETA11*NPUT-13
50 BETA11*NPUT-14
LPHA11: ZVAR-2,3
0 SRCH12
0 **24*CUE-1
5 BETA11
50 BETA11*CUE
5 LPHA11
50 LPHA11*CUE-1
BETA11: ZVAR-2,1
*
*ROUTINE TO FIND Z4
*
SRCH1200 LPHA12+103*NPUT-2
50 LPHA12+24*NPUT-4
50 LPHA12+43*NPUT-10
50 LPHA12+43*NPUT-13
50 LPHA12+43*NPUT-8
50 LPHA12+72*NPUT-1
50 BETA12*NPUT-5
50 BETA12*NPUT-9
50 BETA12*NPUT-11
50 BETA12*NPUT-12
50 BETA12*NPUT-13
LPHA12: ZVAR-1,0
0 SRCH13
50 BETA12*CUE-1
5 LPHA12
50 LPHA12*CUE-1
5 BETA12
5 **24*CUE-1
3 BETA12
3 BETA12
3 **24*CUE-1
B LPHA12
B D ETA12,QUE
B LPHA12

B ETA12D  ZVAR=1,1
*

*ROUTINE TO FIND ZB
*

SRCH13DD
B D LPHA13+24,NPUT-1
B D LPHA13+24,NPUT-2
B D LPHA13+72,NPUT-5
B D LPHA13+72,NPUT-10
B D ETA13,NPUT
B D ETA13,NPUT-3
B D ETA13,NPUT-6
B D ETA13,NPUT-7
B D ETA13,NPUT-11
B D ETA13,NPUT-15

B LPHA13D  ZVAR=1

B PUNCH
B D *24,QUE-1
B D ETA13
B D ETA13,QUE
B D LPHA13
B D LPHA13,QUE-1
B D ETA13

B ETA13D  ZVAR=1

*PUNCH WATYYSTAT
WNTYYVAR=3
K 00102
K 00104

WATYZSTAT
WNTYZVAR=4
K 00102
K 00104

B D ZA+ZVAR
B 2+24
ZA
AM ZOUT=1
B D ZA+ZVAR-1
B 2+24
ZB
AM ZOUT=2
B D ZB+ZVAR-2
B 2+24
ZC
AM ZOUT=3
B D ZC+ZVAR-3
B 2+24
ZD
AM ZOUT=3
B D ZD+ZVAR-4
B 2+24
ZE
AM ZOUT=15
B EXIT+48

EXIT WATYNEX
K 00102
K 00104
B COMENS
*
*OPERATION ROUTINE
*
DLX #+12, RUN(7)
MM ZOUT.12
SF 00095
S X #+12, J0055(7)
BLX M #+12, -32(2)
TF STOR, SCRIN
H
SSX JUMP Y-12, JUMP Y-6(7)
3
ERROR WATYING
K 00102
K 00104
K COMENS
JUMP Y ERROR
M A S E Y
S ERROR
G R I G T
B FL R I G T
B FL LEFT
B FL UP
B FLOOD
H F UP
H F DOWN
H F R I G T
H FL EFT
H CN R I G T
H CN LEFT
H CNUP
H CN DOWN
ERROR
PCTU
PC
PCNL
CNTAT
PC
K K WATYFINAL
K 00102
K 00104
EXIT
PC NL WATYP OUT3
K 00102
K 00104
PHAS 21
PCTU WATYP OUT2
K 00102
K 00104
PHAS 2 2
*
FL R I G T E B I R R S H I T I C
B 350
FL LEFT E B I L L S H I T I C

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
B  CNTRAT
*
*
CNTRAT\WATYCCN
B  *+24
MAGFY  WATYHAG
K  *0102
K  *0104
STM  PUT\0
WATYXPLEN1
K  *0102
K  *0104
H
BLXM)+12,-32(1)
TF  STOR:SCRIN
RNTYAREA=27
K  *0102
TANKSTOR\AREA=27,6
AM  STOR\ES
BCXH=+43,1(1)
B  SEG
*
*
\HALF LEFT ROUTINE
*

HFLT  NOP
TF  AREA\25,6
AM  STOR\1
SF  STOR\76
AM  STOR\26
TF  AREA\1,STOR\11
TF  STOR\AREA\6
SM  STOR\27
CF  STOR\6
AM  STOR\26
BCXH=HFLT+24,1(2)
SS
*
*
\HALF RIGHT ROUTINE
*

HFRT  NOP
TF  AREA\25,6
SF  STOR\6
AM  STOR\26
TF  AREA\STOR\11
SM  STOR\26
CF  AREA\26
TANKSTOR\AREA\27,6
AM  STOR\26
BCXH=HFRT+24,1(2)
SS
*
* SHIFT UP/DOWN ROUTINE

NOP
DUP H
TF SETI+11;DUP-1
SF SETI+6
SF STOR+6
TF TEMP;STOR
TF RIGHT;STOR
AM TEMP+36
SF TEMP+6
AM TEMP+27
AM RIGHT+27

SET1 TF
AM STOR+28
CF TEMP+6
CF RIGHT+6
BCX:DP+36;1(2)
33

* *SHIFT LEFT ROUTINE
* * NOP

LSSHITH
SF STOR+6
AM STOR+27
TF AREA;STOR+11
SM STOR+27
TRNMSR;AREA-25;6
AM STOR+25
TRNMSR;ZERO-1;6
CF STOR+6
AM STOR+3
BCXM*-108;1(2)
33

* *SHIFT RIGHT ROUTINE
* * NOP

RSSHITH
TF AREA;ZERO
SF STOR+6
AM STOR+23
TF AREA;STOR+11
SM STOR+23
CF AREA-25
TRNMSR;AREA-27;6
AM STOR+23
BCXM*-84;1(2)
33

* *GRID PRINT OUT ROUTINE
* *

NOP

PUT H
BLX:+12,-32(2)
TF LINE:ZERO2
TF STOR:SCRIN
SF STOR:*6
AM LINE:11,10
AM STOR:27
TF AREA:STOR:*11
WNTYARZA:27
SPTY
SPTY
WNTYLINE:1
K 00102
SM STOR:27
CF STOR:*6
AM STOR:28
SCX*X-144,1(2)
SB
*
*
PHASE 2 OPERATION  FINAL SHIFTING OF PATTERN
*
PHASE 1
TFM ZOUT:GRID+706
TFM CTR:GRID+753
TFM TEST:GRID+174
TFM RIGHT:GRID+202
TF STOR:TEST
TF TEMP:RIGHT
BLX:*+12,-16(2)
TFM WEIGHT:8,9
3D ENT1:STOR:*11
3D ENT1:TEMP:*11
AM STOR:*1
AM TEMP:*1
SCX*X-45,1(2)
LOW1 BLX:*+12,-16(2)
TF ZVAR:ZOUT
TF ZWAIT:CTR
3D ENT2:ZVAR:*11
3D ENT2:ZWAIT:*11
AM ZVAR:*1
AM ZWAIT:*1
SCX*X-45,1(2)
B ENT2:+12
ENT1 AM WEIGHT:2,9
B LOW1
ENT2 AM WEIGHT:1,9
TF STOR:SCRIN
BLX:*+12,-32(2)
BLX:*+12,ENTER1(6)
MM WEIGHT:12,10
SF 0095
SX:*+12,0099(6)
SX BEGIN1-12,BEGIN1-6(6)
B
BEGIN

1 HL00
2 HL01
3 HL02
4 BTM
5 PUT
6 PC

HL00: SM: ZOUT+66
SM: CTR+56
AM: TEST+56
AM: RIGHTS+36
B: PHAS21+43

HL01: SM: DUP+03+28
BT: DUP+UPSET+5
AM: DUP+83+23
B: PHAS21+72

HL02: AM: STCR+540
AM: SET+1+11,10
SM: DUP+03+28
BT: DUP+UPSET+2+5
AM: DUP+83+25
SM: SET+1+1+11,10
3 PHAS21+72

PHAS22: TF: ZOUT+GRID+766
TF: CTR+GRID+767
TF: TEST+GRID+189
TF: RIGHT+GRID+198
TF: STOR+TEST
TF: TEMP+RIGHT
BLX:*+12+18(2)
TF: WEIGT+3+9
SD: ENT+STCR+11
SD: ENT+TEMP+11
AM: STOR+28
AM: TEMP+26
BLX:*-48+1(2)
LOW2: BLX:*+124-20(2)
TF: ZVAR+ZOUT
TF: ZWAIT+CTR
SD: ENT4+ZVAR+11
SD: ENT4+ZWAIT+11
SM: ZVAR+3
SX: ZWAIT+28
SCX:*-40+1(2)
G: ENT+4+12
G: AM: WEIGT+2+9
G: AM: LOW2
ENT+4: AM: WEIGT+1+7
TF: STOR+SCRIN
BLX:*+124-32(2)
BLX:*+124+ENTER(3)
AM: WEIGT+12+10
SF: 00095
SX: *+122+3039(3)
JSX: BEGIN+2-12+BEGIN+2-6(B)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
BEGIN23  RLCO
B   RLCO1
3   RLCO2
3TM  PUT+0
B   PC
RLCO1 3TM  HFR+0
B   PHAS22+72
RLCO2 3TM  HFL+0
B   PHAS22+72
RLCO3 3M  TEST+2
3M  RIGHT+2
AM  ZOUT+2
AM  CTR+2
B   PHAS22+43
DENDCOMENS
APPENDIX D

RESULTS OF SIMULATION
END OF ASSEMBLY.
12540 CORE POSITIONS REQUIRED
00826 STATEMENTS PROCESSED

EXECUTION

PATTERN ON GRID BEFORE CENTERING

00000000000000000000000000000000 01
00000000000000000000000000000000 02
00000000000000000000000000000000 03
00000000000000000000000000000000 04
00000000000000000000000000000000 05
00000000000000000000000000000000 06
00000000000000000000000000000000 07
00000000000000000000000000000000 08
00000000000000000000000000000000 09
00000000000000000000000000000000 10
00000000000000000000000000000000 11
00000000000000000000000000000000 12
00000000000000000000000000000000 13
00000000000000000000000000000000 14
00000000000000000000000000000000 15
00000000000000000000000000000000 16
00000000000000000000000000000000 17
00000000000000000000000000000000 18
00000000000000000000000000000000 19
00000000000000000000000000000000 20
00000000000000000000000000000000 21
00000000000000000000000000000000 22
00000000000000000000000000000000 23
00000000000000000000000000000000 24
00000000000000000000000000000000 25
00000000000000000000000000000000 26
00000000000000000000000000000000 27
00000000000000000000000000000000 28
00000000000000000000000000000000 29
00000000000000000000000000000000 30
00000000000000000000000000000000 31
00000000000000000000000000000000 32
THE PRIMARY INPUT IS 001
THE INTERNAL STATE IS 000100
THE OUTPUT IS 00011

THE PRIMARY INPUT IS 004
THE INTERNAL STATE IS 101101
THE OUTPUT IS 01010

THE PRIMARY INPUT IS 005
THE INTERNAL STATE IS 000000
THE OUTPUT IS 10011

OPERATION IS CONTRACTION

00000000000000000000000000000000 01
00000000000000000000000000000000 02
00000000000000000000000000000000 03
00000000000000000000000000000000 04
00000000000000000000000000000000 05
00000000000000000000000000000000 06
00000000000000000000000000000000 07
00000000000000000000000000000000 08
00000000000000000000000000000000 09
00000000000000000000000000000000 10
00000000000000000000000000000000 11
00000000000000000000000000000000 12
00000000000000000000000000000000 13
00000000000000000000000000000000 14
00000000000000000000000000000000 15
00000000000000000000000000000000 16
00000000000000000000000000000000 17
00000000000000000000000000000000 18
00000000000000000000000000000000 19
00000000000000000000000000000000 20
00000000000000000000000000000000 21
00000000000000000000000000000000 22
00000000000000000000000000000000 23
00000000000000000000000000000000 24
00000000000000000000000000000000 25
00000000000000000000000000000000 26
00000000000000000000000000000000 27
00000000000000000000000000000000 28
00000000000000000000000000000000 29
00000000000000000000000000000000 30
00000000000000000000000000000000 31
00000000000000000000000000000000 32

PRESS START TO ENTER GRID DATA TOP ROW TO BOTTOM
THE PRIMARY INPUT IS 000
THE INTERNAL STATE IS 000010
THE OUTPUT IS 00001
Contents of grid before magnification

OPERATION IS MAGNIFICATION

PRESS START TO ENTER GRID DATA TOP ROW TO BOTTOM
Magnified character entered via typewriter as shown below

0000000000000000000000000000

S

RS

R

g

RSR

000001111111111111110000

00000000000011000000000000

B

1

0000000000000000000000000000

R

5

p

5

R

T H E  P R I M A R Y  I N P U T  I S FF0 5
T H E  I N T E R N A L  S T A T E  I S  000010
T H E  O U T P U T  I S  10010
Program print out of centered character

PATTERN HAS BEEN CENTERED

PRESS START TO PROCESS NEW PATTERN

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
VITA AUCTORIS

1937  Born on January 10, in Madras, India.

1952  Completed higher secondary education at Ramakrishna Mission High School, Madras, India.

1956  Graduated from Madras University, Madras, India with degree of B.Sc.

1961  Graduated from Delhi University, Delhi, India with degree of B.E. in Electrical Engineering.

1966  Candidate for degree of M.A.Sc. degree in Electrical Engineering at the University of Windsor.