Open access • Journal Article • DOI:10.1109/PGEC.1965.264063

On an Improved Diagnosis Program — Source link

Sundaram Seshu

Institutions: University of Illinois at Urbana–Champaign

Published on: 01 Feb 1965 - IEEE Transactions on Electronic Computers (IEEE)

Coordinated Science Laboratory

UNIVERSITY OF ILLINOIS - URBANA, ILLINOIS

ON AN IMPROVED DIAGNOSIS PROGRAM

S. Seshu

REPORT R-207 MAY, 1964

COORDINATED SCIENCE LABORATORY
UNIVERSITY OF ILLINOIS
URBANA, ILLINOIS

Contract DA-28-043-AMC-00073(E)

The research reported in this document was made possible by support extended to the University of Illinois, Coordinated Science Laboratory, jointly by the Department of the Army, Department of the Navy (Office of Naval Research), and the Department of the Air Force (Office of Scientific Research) under Department of Army Contract DA-28-043-AMC-00073(E).

ABSTRACT

In an earlier paper (Seshu, S., and Freeman, D. N., "Diagnosis of Asynchronous Switching Systems," IRE Trans. on Elec. Comp., August 1962, pp. 459-465) an IBM 7090 program for the diagnosis of asynchronous circuits was described. The present note describes an improved version written for the CDC-1604 computer. The principal improvements are in flexibility and intelligence. The present program has been written as a tool of research in the problem of self-diagnosis in electronic digital computers.

1. Purpose

The main problem at hand is self-diagnosis in electronic digital computers. In particular, one would like to know how computers should be designed (both at the system level and at the circuit level) to facilitate self-diagnosis. The current state of the art in self-diagnosis is so primitive that we do not even understand the problems involved in making a computer self-diagnosing. We have, therefore, chosen to approach the problem experimentally. Namely, we have chosen to study an existing computer from the point of view of self-diagnosis and examine the problems that arise. The particular example chosen is the CSX-1 computer which has been described elsewhere [1]. This machine is small enough to be studied in detail and yet complex enough to attempt self-diagnosis. Also, since it is a local product it is possible to insert machine modifications when they seem useful in the study. The principal experimental tool is a diagnosis program. The input to the program is a logical description of the sequential circuit and the output is a testing procedure for the circuit. This note is a description of the diagnosis program.

2. General

Seshu and Freeman [5] have described an IBM 7090 program which had essentially the same objectives. The 7090 program was not sufficiently flexible for use in an experimental study. Also, there was not enough intelligence provided as pointed out in [5]. Finally, the 7090 program is held proprietary. Hence this program.

The basic theoretical model used here is the same as in Seshu and Freeman [5]. The basic assumptions in this model are:

1. The class of possible failures is known and is finite.

2. Each failure transforms a sequential circuit into another sequential circuit. That is, only logical failures are to be considered.

3. It is possible to reset the feedback lines momentarily to a known initial state, even under failure conditions.

We think of each of the transformed circuits (including the original) as a "machine." Thus if N failures are to be considered, a given sample is one of N + 1 machines and the problem is to identify it. A "test" consists of applying an input and observing the output. A useful test thus partitions the class of possible machines into equivalence classes, the machines within an equivalence class having the same output. Now we apply another test to one of the equivalence classes, partitioning it further. The entire testing procedure is thus a "branching experiment" in the terminology of Moore [3]. Some of the tests may be "reset the feedback lines to an initial state."

The fundamental procedure for choosing tests is a simulated "multiple experiment" in the sense of Moore [3]. From the given description of a sequential circuit, we compile a simulator which can simulate the machines 1,2,...,N+1 (for N failures). At each point in the process, we try several courses of action. That is, we apply to the current subset of machines, tests or sequences of tests. Then we evaluate these tests and choose the "best" one in some sense.

The CDC-1604 program is written in machine language and is approximately 10,000 instructions. The size limitations on the sequential circuit are: 300 logical elements, 96 inputs, 96 outputs, 48 feedback loops, 1000 failures. The multiples of 48 arise from the fact that the 1604 has 48 bit words. All required data is kept in core memory. Tapes are used only for input and output. The entire program is always in core memory.

3. Program Structure

The general structure of the program is shown in Fig. 1. The main communication with the external world is through the control routine. This routine is somewhat similar to the CDC FORTRAN resident. Its main purpose is to read and obey control statements from any input medium. It is sufficiently flexible to permit one to control the flow of the program on-line if desired, from the console typewriter. Permissible control statements include normal tape handling, assignment of media, various problem oriented statements and an "execute" statement. The "execute" statement allows any subroutine to be executed. The main subroutines can be executed by name (symbolically).

As shown in Fig. 1, there are two driver routines in the program proper. One of these is the straight simulation driver. This driver will simulate any given sequence of inputs on any of the current subsets of machines. Optionally it will interpolate between inputs that differ in more than one bit. The other is the normal diagnosis driver. It is possible to switch back and forth between the drivers.

The diagnosis driver has two options. One option provides for complete diagnosis or stops the procedure as soon as the present subset is reduced as far as possible (check-out or failure detection only). The other provides for a "safety extension" on each branch after it has been followed as far as possible. This feature is a factor of safety introduced because of the questionable nature of the basic assumptions. This safety extension is a test designed for the particular machine, where possible, or a random sequence of preset length. The test is designed by the combinational test generator to be discussed later. The diagnosis driver makes a simple-minded indistinguishability test on each subset before proceeding.

There are four strategy subroutines which the diagnosis driver may call on, for choosing the next input or sequence of inputs. Each of these strategy subroutines maintains its own usefulness index (average gain per step). The strategy subroutines are always reordered according to this index. The diagnosis driver always calls on the first one on the ordered list, first. If it is unsuccessful, the driver goes to the next one etc. This "adaptive" feature is provided for computational efficiency. In general the same strategy is useful for circuits designed by the same engineer. These indices may be dumped and read in. The strategy subroutines operate on the following convention. The current status is first saved (by a subroutine discussed later). Then a sequence of events is followed on a trial basis. The steps taken are saved in a list. If it is found that this sequence is useful, the original status is restored and saved sequence of steps is simulated on a "use" basis--that is, with output, and updating data. If the sequence is not useful, the original status is restored and the subroutine exits. Communication parameters indicate the result to the calling program. Thus each strategy subroutine is "empowered to act," if useful.

The first (and generally the most useful) strategy is "best next or return to good input" strategy. It tries each of the next inputs (differing in one bit from the present input) to see if any of them are useful. If yes, the one with the largest gain (according to the current criterion) is used. It also looks to see if there is more than one good next input. If there is, the unused inputs are saved in a local list of good inputs (discarding duplicates). If no next input gives any gain, the subroutine searches the previously generated list of good inputs. If the list is non-empty, each input is tried (interpolating by one bit changes). They may not be useful now, because the feedback states are different or because the current subset of machines is different. If any of them is useful, it is used. Otherwise the subroutine exits.

The second strategy routine is "try wandering." This routine attempts to take a fixed number (specified by an option card) of psuedo-random steps in the hope of reaching a useful input. After each step, all next inputs are examined. If none of them partition primary outputs, feedback outputs are examined. If they are different, the input that gives the best gain on the feedback outputs is used. If not, a random step is taken. Again if there is no gain after the given number of steps, all the data lists are restored, no output occurs and the routine exits.

The third strategy is the "combinational" strategy. The basic subroutine (of about 1700 instructions) is a combinational test generation program. This program treats the circuit as combinational and generates a test for any given failure. It is a true "generator"; that is, it can be asked to give the next test for the given machine. The tests are given as 1, 0, X (don't care). Options are also available for producing only tests for which the test output is a primary output or for producing a test with a minimum number of specified (non-X) feedback states. The strategy subroutine calls on the test generator to produce all tests (sequentially) for all machines in the present subset (except the good machine). Each of these is tried (with interpolation). Since feedback states are not controllable, the tests may not be useful. If any one is useful, it is used. If not, no action is taken and the routine exits.

The last is the "reset" strategy. All available resets are always stored in memory. The strategy subroutine tries each of the resets followed by a fixed number of steps to see if any useful information is obtainable. Again if no reset is useful no action is taken.

This facility of trying a sequence of inputs before it is used is made possible by the data organization. All tables, with the exception of those used by the compiled simulator, are treated as lists arranged sequentially (the exception is made for speed). Thus, the name of the list, and not the table, is made available directly. To try a sequence one enters the subroutine "SAVEDATA" which merely changes the link portion of the name of each list to point to a psuedo-list. Another subroutine "RSTRDATA" restores the links thus recovering the original state.

This procedure necessitates many address substitutions in each subroutine, but memory limitations do not permit all tables to be duplicated. Only one "level" of saving is possible due to memory limitations.

Interpolation between inputs that differ in more than one bit is done by a subroutine with the following conventions. Only paths within the subcube defined by the initial and final inputs are tried. No bit is changed more than once. All paths (within these limits) are tried. The first available one (i.e., no malfunctions at any input) is stored in a list "PATH" as a sequence bits to change. Exit parameters state whether a path was found, whether there is any gain and if so the number of machines left in the subset. Another "straight sequence" subroutine can be used to simply simulate this sequence of inputs either all the way or only until the number of machines left reaches the prescribed number. Thus, once a course of action has been decided upon in the "try" mode, it can be simulated rapidly in the "use" mode.

Two criteria are currently available for computing the figure of merit of a test. One is the information gain in bits computed as follows. For a binary partition, if Σp_j is the sum of a priori probabilities of failure over one block of the partition and Σq_j the similar sum for the other block, normalized so that $\Sigma p_j + \Sigma q_j = 1$, the information gain is

$$I = - (\Sigma p_j \log \Sigma p_j + \Sigma q_j \log \Sigma q_j)$$

An n-ary partition is equivalent to a string of n-1 binary partitions. The subroutine actually consults the list of a priori probabilities. (If they are not known, they are initialized to equal probabilities.) the second criterion is the check-out or detection criterion, which is computed as (number of machines eliminated) / (original number in subset).

An elaborate list of options is available for controlling the flow of the program. As an example, the following course of action is possible. For rapid failure detection, set the criterion as check-out. Now the program is asked to follow the good machine branch as far as possible; then come to the typewriter. Now we examine the list of undetected failures. If we can (as is often the case) we generate, manually, the tests for these failures. Now call on the simulation driver to simulate these tests. Set the criterion to information gain, ask that the previous partition information be maintained and call on the diagnosis driver to complete the diagnosis.

4. Simulation Technique

The essentials of the simulation procedure remain the same as in [5]. The description of the sequential circuit consists of an identification, names of input, output and feedback variables followed by the list of logical elements. For each element we have a name, type (AND, OR etc.) and a list of inputs. The type may also be a two level "macro" (AND-OR, AND-NOR etc.). The description is first processed by an organizer program (which arranges it in levels and locates undefined feedback loops) before it is read by the diagnosis program. Provision is also made for including resets and a list of "previously-tested" logical elements on the input tape.

This information is translated into a string of machine instructions (the simulator) by the logic compiler. The compiler includes the instructions necessary for failure simulation and makes up a dictionary of failures as in [5]. The orientation of the failure simulation is toward transistor-diode logic. Thus we simulate open circuited diodes, short circuited diodes and transistor failures. The logical equivalents which are simulated are: input to an AND-type gate is 1, input to an OR-type gate is 0, output of a gate is 1, output of a gate is 0, output of a gate is the same as an input. Simulation of shorts and inverter failures (no inversion) are optional.

Since the 1604 has 48 bit words, 48 failures are simulated simultaneously. Appropriate driver routines separate the set of machines in the present subset into sets of 48 machines and set up the failure injection words. The simulator itself is driven by a Huffman [2] analysis program which contains race analysis and checks for oscillation. Races containing more than 48 branches are considered as malfunctions. There is no hazard analysis. By defining feedback loops appropriately, it can be made to do Muller [4] analysis. Ideal synchronous simulation is also available; however, the failure injection does not include failures in timing. Only static (and catastrophic) failures are included.

Through calling sequence parameters, it is possible to simulate on a trial or use (update feedback and partition information) basis, partition on the basis of primary or feedback outputs, stop simulation when a malfunction is discovered or complete the simulation. Malfunctioning machines are listed in memory according to type of malfunction (critical race, oscillation or race followed by oscillation). In the case of critical race, all possible final outcomes are also listed.

5. Comparison

The internal bookkeeping in the present program is completely different from the one used by Seshu and Freeman [5]. The list of all machines arranged according to the present partition, their ordered-pair indices, last feedback states and last used inputs are always in core memory. When several next inputs are tried, the simulation results are overwritten. It is thus necessary to simulate again on a "use" basis after the decision is made. Because of the availability of the "save" feature, no output occurs unless there is a useful result. Thus the "aimless wandering" of the previous program has been eliminated.

Comparison on the basis of running time is somewhat difficult. The tape handling time has been eliminated resulting in a large gain. On the other hand, the computing time has been increased by the addition of the combinational and reset strategies. Also reset is optimized to a specified depth (usually 3 steps). Furthermore, greater use is made of previous experience through the "return to good input" technique. Essentially the program "ponders" more but puts out a more efficient and more complete testing procedure. Overall, the computing time for a given problem is about the same as in the Seshu-Freeman program (disregarding the difference between machine speeds).

In very simple problems (such as a circuit containing one or two flip-flops) there are still cases where the program is unable to detect failures which can be detected. These are generally cases where the flip-flops must be set to some particular state before the test is applied. Unfortunately there is no memory space available for additional programming of any magnitude. The flexibility of the program makes it possible to join manually generated tests to program generated tests, a feature that was not previously possible. This "joining" may also be done on-line. "Almost combinational" circuits which presented problems earlier are now easily handled by the combinational test generation program.

A detailed technical report on the organizer and disgnosis programs is being issued separately.

The help of Mrs. Virginia Metze in the preparation of the program is gratefully acknowledged.

BIBLIOGRAPHY

1. Brown , R. M., Jenks, R. D., Stifle, J. E., and Trogdon, R. L., "The CSX-1 Computer," IEEE Trans. on Electronic Computers, vol. EC-13; June 1964.

2. Huffman, D. A., "Synthesis of Sequential Switching Circuits," Jour. Franklin Inst., vol. 257, pp. 161-190, 275-303; March and April 1954.

3. Moore, E. F., "Gedanken-Experiments on Sequential Machines," Automata Studies, Princeton University Press, pp. 129-153; 1956.

4. Muller, D. E., and Bartky, S., "A Theory of Asynchronous Circuits," Proc. Intl. Symp. on the Theory of Switching, in Ann. Harvard Comp. Lab., vol. 29, pp. 204-243; April 1957.

5. Seshu, S., and Freeman, D. N., "The Diagnosis of Asynchronous Sequential Switching Systems," IRE Trans. on Electronic Computers, vol. EC-11, pp. 459-465; August 1962.

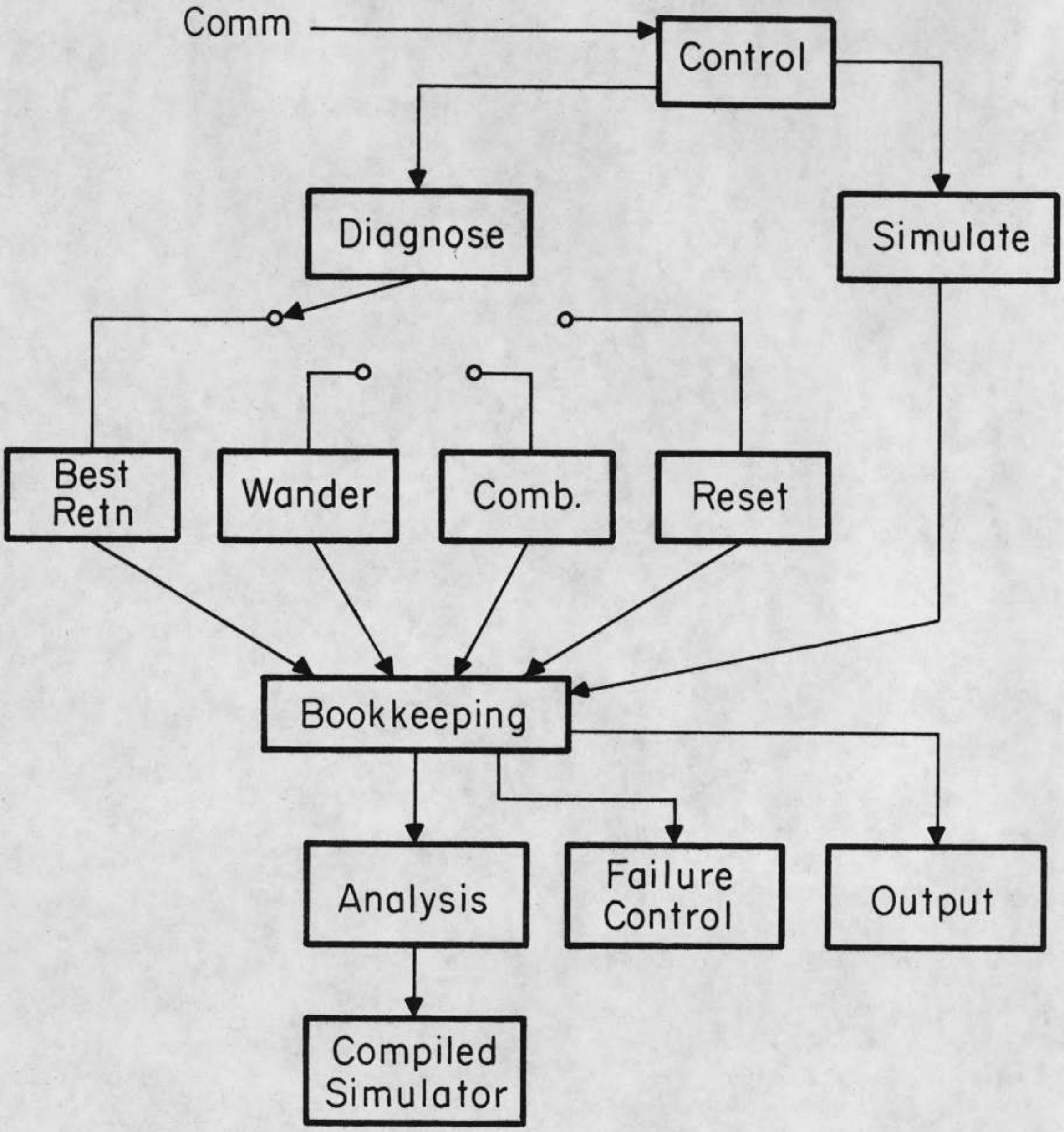

Figure 1. Sequential Circuit Analyzer

Distribution list as of February 1, 1964

1 Director
Air University Library
Maxwell Air Force Base, Alabama
Attn: CR-4803a

1 Redstone Scientific Information Center
U.S. Army Missile Command
Redstone Arsenal, Alabama

1 Electronics Research Laboratory (Unclassified)
University of California
Berkeley 4, California

2 Hughes Aircraft Company
Florence and Teale
Culver City, California
Attn: N. E. Devereux
Technical Document Center

3 Autonetics (Unclassified)
9150 East Imperial Highway
Downey, California
Attn: Tech. Library, 3041-11

1 Dr. Arnold T. Nordsieck
General Motors Corporation
Defense Research Laboratories
6767 Hollister Avenue
Goleta, California

1 University of California (Unclassified)
Lawrence Radiation Laboratory
P. O. Box 808
Livermore, California

1 Mr. Thomas L. Hartwick
Aerospace Corporation
P. O. Box 95085
Los Angeles 45, California

1 Lt. Colonel Willard Levin
Aerospace Corporation
P. O. Box 95085
Los Angeles 45, California

1 Professor Zorab Kaprelian
University of Southern California
University Park
Los Angeles 7, California

1 Sylvania Electronic Systems - West
Electronic Defense Laboratories
P. O. Box 205
Mountain View, California
Attn: Documents Center

1 Varian Associates
611 Hansen Way
Palo Alto, California
Attn: Dr. Ira Weissman

1 Huston Denslow (Unclassified)
Library Supervisor
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

1 Professor Nicholas George
California Institute of Technology
Electrical Engineering Department
Pasadena, California

1 Space Technology Labs., Inc.
One Space Park
Redondo Beach, California
Attn: Acquisitions Group
STL Technical Library

2 Commanding Officer and Director
U.S. Naval Electronics Laboratory
San Diego 52, California
Attn: Code 2800, C. S. Manning

1 Commanding Officer and Director
U.S. Navy Electronics Laboratory
San Diego 52, California
Attn: Library

1 Office of Naval Research Branch Office
1000 Geary Street
San Francisco, California

1 The RAND Corporation
1700 Main Street
Santa Monica, California
Attn: Library

1 Stanford Electronics Laboratories (Unclassified)
Stanford University
Stanford, California
Attn: SEL Documents Librarian

1 Dr. L. F. Carter
Chief Scientist Air Force
Room 4E-324, Pentagon
Washington 25, D. C.

1 Mr. Robert L. Feik
Associate Director for Research
Research and Technology Division
AFSC
Bolling Air Force Base 25, D. C.

1 Captain Paul Johnson (USN-Ret)
National Aeronautics and Space
Administration
1520 H Street, N. W.
Washington 25, D. C.

1 Major Edwin M. Myers
Headquarters USAF (AFRDR)
Washington 25, D. C.

1 Dr. James Ward
Office of Deputy Director
(Research and Info)
Department of Defense
Washington 25, D. C.

1 Dr. Alan T. Waterman
Director, National Science Foundation
Washington 25, D. C.

1 Mr. G. D. Watson
Defense Research Member
Canadian Joint Staff
2450 Massachusetts Ave., N. W.
Washington 8, D. C.

1 Mr. Arthur G. Wimer
Chief Scientist
Air Force Systems Command
Andrews Air Force Base
Washington 25, D. C.

1 Director, Advanced Research
Projects Agency
Washington 25, D. C.

1 Air Force Office of Scientific Branch
Directorate of Engineering Sciences
Washington 25, D. C.
Attn: Electronics Division

1 Director of Science and Technology
Headquarters, USAF
Washington 25, D. C.
Attn: AFRST-EL/GU

1 AFRST - SC
Headquarters, USAF
Washington 25, D. C.

1 Headquarters, R & T Division (Unclassified)
Bolling Air Force Base
Washington 25, D. C.
Attn: RTHR

1 Headquarters, U. S. Army Material Command
Research Division, R & D Directorate
Washington 25, D. C.
Attn: Physics & Electronics Branch
Electronics Section

1 Commanding Officer
Diamond Ordnance Fuze Laboratories
Washington 25, D. C.
Attn: Librarian, Room 211, Bldg. 92

1 Operation Evaluation Group
Chief of Naval Operations (OP-03EG)
Department of Navy
Washington, D. C. 20350

1 Chief of Naval Operations (Code OP-ONT)
Department of the Navy
Washington, D. C. 20350

1 Commanding Officer
U. S. Army Personnel Research Office
Washington 25, D. C.

1 Commanding Officer & Director
Code 142 Library
David W. Taylor Model Basin
Washington, D. C. 20007

1 Chief, Bureau of Ships (Code 686)
Department of the Navy
Washington, D. C. 20360

1 Chief, Bureau of Ships (Code 732)
Department of the Navy
Washington, D. C. 20360

1 Chief, Bureau of Naval Weapons
Technical Library, DLI-3
Department of the Navy
Washington, D. C. 20360

1 Director, (Code 5140)
U. S. Naval Research Laboratory
Washington, D. C. 20390

1 Chief of Naval Research (Code 437)
Department of the Navy
Washington, D. C. 20360

1 Dr. H. Wallace Sinaiko (Unclassified)
Institute for Defense Analyses
Research & Engineering Support Division
1666 Connecticut Ave., N. W.
Washington 9, D. C.

1 Data Processing Systems Division
National Bureau of Standards
Conn. at Van Ness
Room 239, Bldg. 10
Washington 25, D. C.
Attn: A. K. Smilow

1 National Bureau of Standards (Unclassified)
Research Information Center &
Advisory Service on Information
Processing
Data Processing Systems Division
Washington 25, D. C.

1 Exchange and Gift Division (Unclassified)
The Library of Congress
Washington 25, D. C.

1 NASA Headquarters
Office of Applications
400 Maryland Avenue, S. W.
Washington 25, D. C.
Attn: Mr. A. M. Greg Andrus
Code FC

1 APGC (PGAPI)
Eglin Air Force Base
Florida

1 Martin Company
P. O. Box 5837
Orlando, Florida
Attn: Engineering Library
MP-30

1 Commanding Officer
Office of Naval Research, Chicago Branch
6th Floor, 230 North Michigan
Chicago 1, Illinois

1 Laboratories for Applied Sciences
University of Chicago
6220 South Drexel
Chicago 37, Illinois

1 Librarian (Unclassified)
School of Electrical Engineering
Purdue University
Lafayette, Indiana

1 Donald L. Epley (Unclassified)
Department of Electrical Engineering
State University of Iowa
Iowa City, Iowa

1 Commanding Officer
U. S. Army Medical Research Laboratory
Fort Knox, Kentucky

2 Keats A. Pullen, Jr.
Ballistic Research Laboratories
Aberdeen Proving Ground, Maryland

1 Director
U. S. Army Human Engineering Laboratories
Aberdeen Proving Ground, Maryland

1 Mr. James Tippett
National Security Agency
Fort Meade, Maryland

1 Commander
Air Force Cambridge Research Laboratories
Laurence G. Hanscom Field
Bedford, Massachusetts

1 Dr. Lloyd Hollingsworth
Director, ERD
AFCRL
L. G. Hanscom Field
Bedford, Massachusetts

1 Data Sciences Laboratory
Air Force Cambridge Research Lab
Office of Aerospace Research, USAF
L. G. Hanscom Field
Bedford, Massachusetts
Attn: Lt. Stephen J. Kahne - CRB

1 Instrumentation Laboratory(Unclassified)
Massachusetts Institute of Technology
68 Albany Street
Cambridge 39, Massachusetts

1 Research Laboratory of Electronics (Unclassifi
Massachusetts Institute of Technology
Cambridge 39, Massachusetts
Attn: Document Room 26-327

1 Dr. Robert Kingston
Lincoln Laboratories
Lexington, Massachusetts

1 Lincoln Laboratory (Unclassified)
Massachusetts Institute of Technology
P. O. Box 73
Lexington 73, Massachusetts
Attn: Library, A-082

1 Sylvania Electric Products, Inc.
Electronic Systems
Waltham Labs. Library
100 First Avenue
Waltham 54, Massachusetts

1 Minneapolis-Honeywell Regulator Co.
Aeronautical Division (Unclassified)
2600 Ridgeway Road
Minneapolis 13, Minnesota
Attn: Dr. D. F. Elwell
Main Station: 625

1 Inspector of Naval Material
Bureau of Ships Technical Representative
1902 West Minnehaha Avenue
St. Paul 4, Minnesota

20 Activity Supply Officer, USAELRDL
Building 2504, Charles Wood Area
Fort Monmouth, New Jersey
For: Accountable Property Officer
Marked: For Inst. for Exploratory Research
Inspect at Destination
Order No. 576-PM-63-91

1 Commanding General
U. S. Army Electronic Command
Fort Monmouth, New Jersey
Attn: AMSEL-RE

1 Mr. A. A. Lundstrom
Bell Telephone Laboratories
Room 2E-127
Whippany Road
Whippany, New Jersey

1 AFMDC (MDSGP/Capt. Wright)
Holloman Air Force Base
New Mexico

1 Commanding General
White Sands Missile Range
New Mexico

1 Microwave Research Institute
Polytechnic Institute of Brooklyn
55 John Street (Unclassified)
Brooklyn 1, New York

1 Cornell Aeronautical Laboratory, Inc.
4455 Genesee Street
Buffalo 21, New York
Attn: J. P. Desmond, Librarian

1 Sperry Gyroscope Company (Unclassified)
Marine Division Library
155 Glen Cove Road
Carle Place, L. I., New York
Attn: Mrs. Barbara Judd

1 Major William Harris
RADC (RAWI)
Griffiss Air Force Base
New York

1 Rome Air Development Center
Griffiss Air Force Base
New York
Attn: Documents Library
RAALD

1 Library (Unclassified)
Light Military Electronics Department
General Electric Company
Armament & Control Products Section
Johnson City, New York

1 Columbia Radiation Laboratory
Columbia University (Unclassified)
538 West 120th Street
New York 57, New York

1 Mr. Alan Barnum
Rome Air Development Center
Griffiss Air Force Base
Rome, New York

1 Dr. E. Howard Holt (Unclassified)
Director
Plasma Research Laboratory
Rensselaer Polytechnic Institute
Troy, New York

3 Commanding Officer
U.S. Army Research Office (Durham)
Box CM, Duke Station
Durham, North Carolina
Attn: CRD-AA-IP, Mr. Ulsh

1 Battelle-DEFENDER
Battelle Memorial Institute
505 King Avenue
Columbus 1, Ohio

1 Aeronautical Systems Division
Navigation and Guidance Laboratory
Wright-Patterson Air Force Base
Ohio

1 Aeronautical Systems Division
Directorate of Systems Dynamic Analysis
Wright-Patterson Air Force Base
Ohio

1 Commander
Research & Technology Div.
Wright-Patterson Air Force Base
Ohio 45433
Attn: MAYT (Mr. Evans)

1 Commanding Officer (AD-5)
U.S. Naval Air Development Center
Johnsville, Pennsylvania
Attn: NADC Library

2 Commanding Officer
Frankford Arsenal
Philadelphia 37, Pennsylvania
Attn: SMUFA-1300

1 H. E. Cochran
Oak Ridge National Laboratory
P. O. Box X
Oak Ridge, Tennessee

1 U.S. Atomic Energy Commission
Office of Technical Information Extension
P. O. Box 62
Oak Ridge, Tennessee

1 President
U.S. Army Air Defense Board
Fort Bliss, Texas

1 U.S. Air Force Security Service
San Antonio, Texas
Attn: ODC-R

1 Director
Human Resources Research Office
The George Washington University
300 North Washington Street
Alexandria, Virginia

20 Defense Documentation Center
Cameron Station
Alexandria, Virginia
22314

1 Commander
U. S. Army Research Office
Highland Building
3045 Columbia Pike
Arlington 4, Virginia

1 U.S. Naval Weapons Laboratory
Computation and Analysis Laboratory
Dahlgren, Virginia
Attn: Mr. Ralph A. Niemann

2 Army Material Command
Research Division
R & D Directorate
Bldg. T-7
Gravelley Point, Virginia