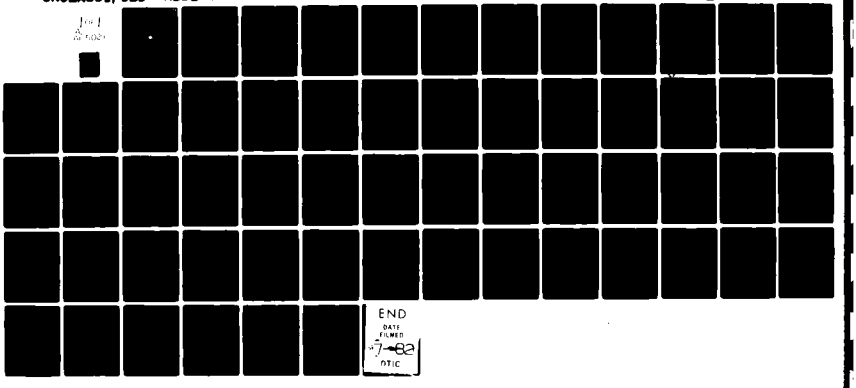
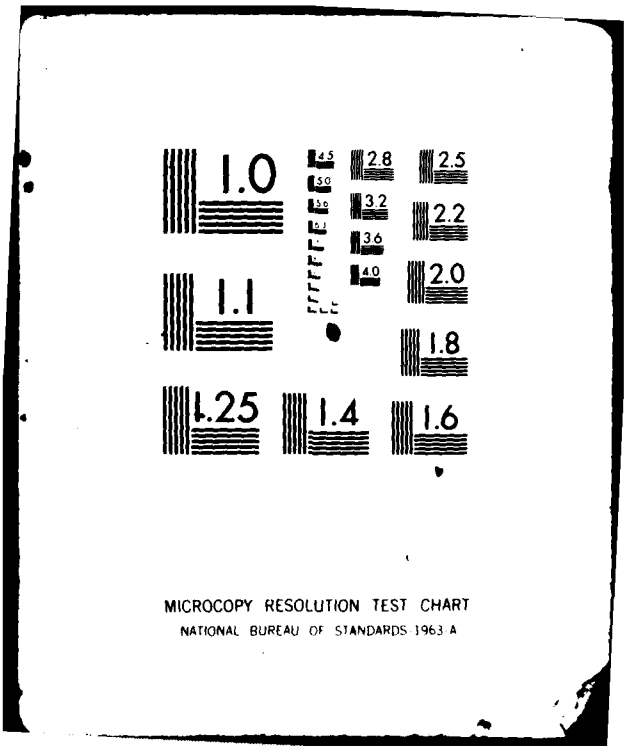
AD-A115 021 NAVAL BIODYNAMICS LAB NEW ORLEANS LA F/G 5/10
HUMAN PERFORMANCE TESTS FOR REPEATED MEASUREMENTS: ALTERNATE FO--ETC(U)
JAN 82 R C CARTER, H E SBISA
UNCLASSIFIED NBDL-82R003 NL




1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

NBDL - 82R003

Human Performance Tests for Repeated Measurements: Alternate Forms of Eight Tests by Computer

Robert C. Carter, Naval Biodynamics Laboratory

and

Harvey E. Sbisa, Q.E.I., Inc.

January 1982

NAVAL BIODYNAMICS LABORATORY
New Orleans, Louisiana

Approved for public release. Distribution unlimited.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NBDL-82R003	2. GOVT ACCESSION NO. AD-A115 021	3. RECIPIENT'S CATALOG NUMBER
4. TITLE *(and Subtitle)* Human Performance Tests for Repeated Measurements: Alternate Forms of Eight Tests by Computer		5. TYPE OF REPORT & PERIOD COVERED 6. PERFORMING ORG. REPORT NUMBER NBDL-82R003
7. AUTHOR(s) Robert C. Carter and Harvey E. Sbisa		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Biodynamics Laboratory Box 29407 New Orleans, La. 70189		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS MF58.524-002-5027
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Medical Research and Development Command Bethesda, MF 20014		12. REPORT DATE January 1982 13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS(*if different from Controlling Office*)		15. SECURITY CLASS. *(of this report)* Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited

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

18. SUPPLEMENTARY NOTES

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

Repeated Measurement, Human Performance, Alternate Forms, Computer Generated Tests

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

Eight performance tests for repeated measurements based on published tests were generated on paper by computer programs. The computer programs can be used to sample equivalent forms of the tests for any number of occasions of repeated measurement. The programs also print any specified number of copies of the alternate forms to provide for multiple subjects. The logic of the item-sampling procedures and block diagrams of the item-generated method are discussed. The Fortran computer programs are given as Appendices. The tests are Addition, Number Comparison, Code Substitution, Grammatical Reasoning,

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601 |

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

ABSTRACT (Block 20) Continued

Fitts' Histoform Recognition, Klien's Pattern Comparison, Neisser's Letter Search, and randomly-placed Number Search.

NBDL - 82R003

Human Performance Tests for Repeated Measurements: Alternate Forms of Eight Tests by Computer

Robert C. Carter, Naval Biodynamics Laboratory

and

Harvey E. Sbisa, Q.E.I., Inc.

January 1982

Bureau of Medicine and Surgery
Work Unit No. MF58.524-002-5027

Approved by

Channing L. Ewing, M. D.
Chief Scientist

Released by

Captain J. E. Wenger MC USN
Commanding Officer

Naval Biodynamics Laboratory
Box 29407
New Orleans, LA 70189

SUMMARY PAGE

PROBLEM

Sometimes it is necessary to measure some aspect of human performance capability repeatedly. For example, repeated measurements are necessary to track the time-course of onset or recovery from the effects of stressors such as drugs, diseases, and exposure to toxic materials or unusual environments (e.g., vibration, high altitude, undersea compression, heat, cold, etc.). These repeated measurements usually cannot be made with the same human performance test because the subjects are influenced by their previous responses. The problem is to make different but equivalent test forms for each occasion of measurement.

FINDINGS

Computers can be programmed to sample alternate forms of a test from the population of all possible test items of the type in the test. These test-generation methods have proven to be useful, economical, and rapid. Eight illustrative tests are provided.

RECOMMENDATIONS

Use sampling techniques to generate alternate forms of human performance tests. The sampling techniques can be implemented on a computer for printing of multiple copies of multiple forms of tests.

The work was funded by the Naval Medical Research and Development Command and by the Biological Sciences Division of the Office of Naval Research.

INTRODUCTION

Human performance tests are samples of behavior. They are used to infer capabilities which may be reflected in many other types of behavior, including occupational performance. Sometimes it is necessary to sample the same subjects repeatedly, for example, to monitor recovery from an affliction (e.g., Bell, Jurek, & Wilson, 1976), to assess the effect of an environmental stressor (e.g., Carter, 1979), or to evaluate the effectiveness of training (Goldstein, 1974). In these repeated-measures applications, different and equivalent forms of tests must be used for each occasion of measurement because a subject might recall the answers if the same form of the test were reused.

Equivalence of numerous "alternate forms" of some tests has been demonstrated empirically. The Wondelic Test of general mental ability can be bought in 14 forms. Moran, Kimble, and Mefferd (1964) have published 20 alternate forms of tests of five specific mental abilities, and Horne (1972) has similarly developed alternate forms for repeated measurements. Empirical verification of the equivalence of such tests is a tedious and controversial procedure involving examination of test characteristics like the number of items, item difficulties, item variances, item covariances, and item-criterion covariances (Horst, 1968). A much easier, and theoretically appealing approach to alternate forms of tests can be based on sampling techniques (Cochran, 1977). Alternate forms of tests will be created by sampling randomly from a homogeneous population of test items. This report will describe the application of sampling techniques implemented on a computer to generate alternate printed forms of performance tests. Use of the computer enables one to create any number of copies of any number of forms of a test.

Relevant Sampling Concepts (Cochran, 1977)

In this report, alternate forms of tests are considered to be samples of test items. A sample is a part of an aggregate. The most fundamental concept in sampling theory is the population. The population is the aggregate from which a sample is chosen. For example, in sampling arithmetic items the population could be all possible arithmetic items, all addition problems, or all problems involving addition of three positive two-digit integers arrayed vertically. It is important to define the population precisely, so that the sample will include only items that are relevant to the purpose of the sample.

Error of measurement is another important concept in sampling theory. Errors of measurement occur when what is measured is different from what was intended to be measured. If the population is not precisely defined, the sample may include items that would not have been intended. For example, if any addition problem involving one or two-digit numbers is allowed, then zero may occasionally be sampled as one of the numbers and the problem will not really require addition skill.

The most useful concept of sampling theory is the random sample. If a sample of n items from among N in the population is random, then each of the $\frac{N!}{(N-n)!\,n!}$ possible samples of n from among N has an equal chance of being selected. ($N! = N \cdot (N-1) \cdot (N-2) \cdot \ldots \cdot 1$). If the n items are drawn one

at a time from among the N original items in the population, then a random sample gives an equal chance of selection to any item in the population not already drawn. A random sample has two properties of major importance: representativeness, and analytic variances of sample statistics. The representativeness of a random sample is obvious; every segment of the population has as much chance of being included as another comparable segment. Random samples also enable one to calculate the variance of the sample statistics (like the mean, total, or proportion) without making repetitive samples. Random samples enable one to calculate variances from analytic expressions rather than from empirical exercises requiring vastly more resources. Random sampling of test items requires that each possible test item has an equal chance of selection. For example, there are 90 two-digit numbers, so there are 90.90 = 8100 possible addition problems involving two two-digit numbers; in a random sample of one item, each problem has one chance in 8100 of being selected.

The possibility of expressing analytically the variance of sample statistics enables us to identify the conditions leading to the most precise (minimum variance) samples. A result that is useful in test construction tells us how to allocate test effort (time or number of items) among various types of test items which measure the same thing, assuming that the items are randomly sampled within each type. The optimum allocation will produce a great increase in precision if the types of test items produce far different means and variances of performance. For example, in Neisser's letter search task (Neisser, Novick, & Lazar, 1964), the mean and variance increase with the number of targets for which the subject is scanning. An optimum allocation of sampled test items is proportional to the standard deviation of test scores for each type of item, and is inversely proportional to the square root of the cost of testing (or test time) for that type of item, assuming that cost increases linearly with the number of items (Cochran, 1977, pp. 98). A practical rule is sample more of a particular type of item if performance on that type is more variable, or sampling on that type is cheaper. The score for a test made up of parts would be the weighted average of the scores for each part; the weight for each part is the number of items in that part.

These sampling techniques (defining the population, reducing errors of measurement by excluding peculiar items, random sampling of test items, and optimum allocation of test effort to similar types of items with differing mean performance or dispersion) are employed in the computer programs described in the remainder of this report.

The Tests

The tests were generated on paper by computerized sampling of items and were modeled after tests that have been reported as useful in the literature of performance testing. The tests of arithmetic computation (Number Facility) were like those described by Ekstrom, French, Harman, and Dermen (1976). The number comparison test (Perceptual Speed) was also taken from Ekstrom, French, Harman, and Dermen. The Code Substitution test was like that in the Weschler Intelligence Scale (1958). The Grammatical Reasoning test was from Baddeley (1968). The Pattern Recognition test was similar to that used by Alluisi and Thurmond (1970), based on "metric figures" or histoforms invented by P. M. Fitts (Fitts, Weinstein, Rappaport, Anderson, & Leonard, 1956). The pattern comparison test, which is procedurally similar

to the number comparison test, was reported by Klein and Armitage (1979). The letter search test is an adaptation by Rose (1974) of Neisser's experiment (Neisser, Novick, & Lazar, 1963). Finally, the number search test was similar to an experimental psychology task used by Green and Anderson (1956), and others. It requires a subject to find a target number located among other numbers randomly dispersed on a page of computer paper. The subject's task, the method of sampling items, and some representative items for each test will be discussed in the next section.

Test Procedures and Items

Addition Test

The addition test requires the subject to perform and record addition of three two-digit numbers, arranged vertically. Some representative items are:

34	14	91	46
32	85	27	61
+73	+41	+26	+35

The test is conducted in two parts, each lasting two minutes, with a brief rest between parts. Each part includes 4 lines of 12 items presented on a single page. The preferred score is the total number of correct items. Data obtained with this test were reported by Bittner and Carter (1981). A block diagram showing the construction of a single item and the Fortran IV computer program for this test are given in Appendix A.

Other computational tests involving addition of three-digit numbers arranged horizontally, division, subtraction, and multiplication have also been programmed for item sampling. The arithmetic tests produce highly correlated scores, so only the simplest arithmetic test, addition, was presented here.

Number Comparison

The number comparison test requires the subject to compare two adjacent strings of three to nine digits. The strings will be the same, or (with probability .5), they will differ in one of the digits (chosen at random). The subject is to write "S" on a line between the strings if they are the same, or "D" if they are different. Some representative items are:

930 _ _ _ _ _ _ 930 63983496 _ _ _ _ _ _ 63903496

The test is conducted in one part lasting three minutes. There are 14 lines of 3 items on each page, and the computer prints 5 pages of items. The preferred score is the number of correct responses minus the number of incorrect responses. Data obtained with this test were reported by Bittner and Carter (1981) and a block diagram showing construction of a single item is presented in Appendix B. The Fortran IV computer program for this test is in Appendix B.

Code Substitution

The code substitution test requires the subject to refer repeatedly to a table of nine digit-letter pairs to find the digits which correspond to

randomly selected letters. The letters of the nine code pairs associated with each form of the test are chosen, at random, from among the 26 letters of the alphabet. Given a letter, the subject responds by writing the corresponding digit under the letter. Some representative items are:

CODE	Q	E	H	G	U	P	J	M	C
DIGIT	(2)	(1)	(3)	(8)	(4)	(9)	(6)	(5)	(7)
	M	J	H	M	Q	H			
	( )	( )	( )	( )	( )	( )			

The test is conducted in one part lasting three minutes. The preferred score is the number of correct responses. Data obtained with this test were reported by Pepper, Kennedy, Bittner, and Wiker (1980). A block diagram showing how to construct a single item and the Fortran IV computer program for this test are given in Appendix C.

Grammatical Reasoning

The grammatical reasoning test requires the subject to read and comprehend a simple statement about the order of two letters, A and B. Then the subject observes AB or BA printed next to the statment. If the statement correctly describes the order of the letters, then the subject marks "T". Otherwide, the subject marks "F". The statements about the order of the letters use the verbs "preceeds", or "follows", the active or passive voice, and negative or affirmative phrasing. In addition, the letters can be in either order, and the statement can be true or false. These five dichotomies lead to 32 possible test items. Representative items are:

```
A  preceeds B              AB        T  F
B  is not followed by A    BA        T  F
```

All items are used in each form of the test; the forms differ in the order of the items. There are 32! possible forms. ($32! = 2.6 \times 10^{35}$). Forms are constructed by chosing the order of the items at random. The test lasts one minute. The preferred score is the number correct minus the number of incorrect responses. Data obtained with this test were reported by Carter, Kennedy, and Bittner, (1981). Appendix D gives the Fortran IV computer program which generates this test.

Pattern Recognition

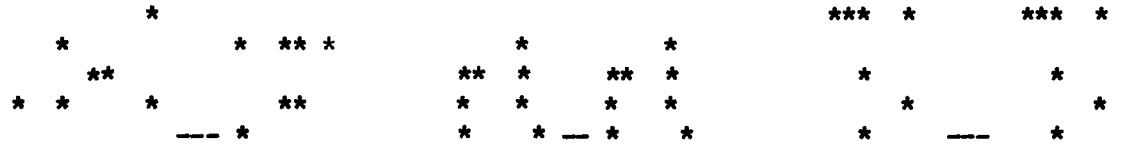
The pattern recognition test requires the subject to look at a histogram pattern, and then recognize it among nine other patterns arrayed in a row to the right of it. The subject underlines the pattern which matched the target pattern at the left end of the row. A representative item is:

```
x   x    x  x                      x    x   x   x                      x
x   x    x  x              x       x  x x   x   x   x          x       x            x
x x x    x  xx      x     x  x     x  x x   x x x   x  x       x       x  xx      xxx
x xxx    x xxx    xxxxx   xxxx     x  x x   x xxx   x xxx     xxx      x  xx    xxxxx
xxxxx    x xxx    xxxxx   xxxxx    x xx x   xxxxx   xxxxxx  x xxx      x xxx    xxxxx
xxxxxx   xxxxxx   xxxxxx  xxxxxx   xxxxxx   xxxxxx  xxxxxx  xxxxxx   xxxxxx    xxxxxx
```

The test is conducted in two parts, each lasting 90 seconds. Each part consists of five pages of five items. The preferred score is the number of correct responses. Data obtained with this test were reported by Shannon and Carter (1981) and a block diagram showing construction of a single item is presented in Appendix E. The Fortran IV computer program which generates this test is in Appendix E.

Pattern Comparison Test

The pattern comparison test is procedurally similar to the number comparison test. The pattern comparison test requires the subject to compare two adjacent patterns of astrisks. The subject is to write "S" on a line between them if they are the same, or "D" if they are different. Some representative items are:



The test is conducted in one part lasting two minutes. There are six lines of three items on each of eight pages of the test. The preferred score is the number of correct responses minus the number of incorrect responses. Data obtained with this test were described by Shannon and Carter (1981) and a block diagram showing construction of a single item is presented in Appendix F. The Fortran IV computer program for this test is in Appendix F. An improved version of the test, in which the patterns may differ only in the placement of a single asterisk, has been programmed in Basic language.

Letter Search

The letter search test has two parts. In the first part the subject is required to look for a particular target letter or number in an array with many rows and five columns of numbers or letters. A mark is to be made next to any row having the target letter in it. In the second part of the test there are four target letters or numbers. In this part of the test a mark is to be made next to any row having any of the four target letters or numbers in it. Some representative items are:

Part 1	Part 2
Target: G	Targets: M G T X
G L N R 7	K T N L 7
T M T R L	K L F R H
G L 7 H N	N F R K H
M 7 M 7 G	M N 7 F R

Subjects are allowed 90 seconds for part 1 and 3 minutes for part 2. The test times for the two parts approximate an optimum sampling allocation because the standard deviation of performance in part 2 is double that for part 1. The preferred score is the time per correct response. Data obtained

with this test were described by Shannon and Carter (1981) and a block diagram showing construction of a single item is presented in Appendix G. The Fortran IV computer program for this test is in Appendix G.

Number Search Test

The number search test has four parts. All four parts require the subject to look for a target number among other numbers scattered at random locations on a page (the search page). The specifications for each part are:

Part	Number of Targets	Number of Numbers on the Search Page
1	1	10
2	4	10
3	1	40
4	4	40

The target or four possible targets are printed at the top of the page preceding the search page. Only one target number will appear on the search page, and it appears only once, in a randomly chosen location. The subject is required to find and mark the target on the search page. A representative item, with targets 1,5,2,9 is presented on the following page. The test has 6, 12, 8 and 16 items in parts 1 through 4, respectively, to allocate test resources in a near optimum way, considering the variance of performance in each part.

The preferred score is time per correct response. Time to complete all items is recorded for each part. Data obtained with this test were described by Shannon and Carter (1981) and a block diagram showing construction of a single item is presented in Appendix H. The Fortran IV computer program for this test is in Appendix H.

Summary

Eight performance tests for repeated measurements are presented, along with computer programs to generate the tests. The computer programs can be used to sample equivalent forms of the tests for any number of occasions of repeated measurement. The programs also print any specified number of copies of the alternate forms to provide for multiple subjects. The logic of the item-sampling procedures and block diagrams of the item-generation method were discussed.

References

Alluisi, E. A., & Thurmond, J. B. Relative effects of figural noise and rotation on the visual perception of form. Perceptual and Motor Skills, 1970, 31, 547-554.

Baddeley, A. D. A three-minute reasoning test based on grammatical transformation. Psychonomic Science, 1968, 10, 341-342.

Bell, E., Jurek, K., & Wilson, T. Hand skill measurement: A gauge for treatment. The American Journal of Occupational Therapy, 1976, 30, 80-86.

Bittner, A. C., Jr., Carter, R. C., & Krause, M. Performance Tests for Repeated Measures: Moran and Computer Batteries. (Research Report No. NBDL-81R012) New Orleans: Naval Biodynamics Laboratory, 1981.

Carter, R. C. Mental abilities during a simulated dive to 427 meters underwater. Journal of Applied Psychology, 1979, 64, 449-454.

Carter, R. C., Kennedy, R. S., & Bittner, A. C., Jr. Grammatical Reasoning: A stable performance yardstick. Human Factors, 1981, 23, 589-591.

Cochran, W. G. Sampling techniques. New York: Wiley, 1977.

Ekstrom, R. B., French, J. W., Harman, H. H., & Dermen, D. Manual for kit of factor-referenced cognitive tests. Princeton, New Jersey: Educational Testing Service, August 1976.

Fitts, P. M., Weinstein, M., Rappaport, M., Anderson, N., & Leonard, J.A. Stimulus correlates of visual pattern recognition: A probability approach. Journal of Experimental Psychology, 1956, 51, 1-11.

Goldstein, I. L. Training: Program development and evaluation. Monterey, CA.: Brooks/Cole, 1974.

Green, B. F., & Anderson, L. K. Color coding in a visual search task. Journal of Experimental Psychology, 1956, 51, 19-24.

Horne, J. L. State, trait, and change dimensions of intelligence. The British Journal of Educational Psychology, 1972, 42, 159-185.

Horst, P. Psychological measurement and prediction. Belmont, CA.: Brooks/Cole, 1968.

Klein, R., & Armitage, R. Rhythms in human performance: 1½-hour oscillations in cognitive style. Science, 1979, 204, 1326-1328.

Moran, L. J., Kimble, J. P., Jr., & Mefferd, R. G., Jr. Repetitive psychometric measures: equating alternate forms. Psychological Reports, 1964, 14, 335-338.

Neisser, U., Novick, R., & Lazar, R. Searching for ten targets simultaneously. Perceptual Motor Skills, 1963, 17, 955-961.

Pepper, R. L., Kennedy, R. S., Bittner, A. C., Jr., & Wiker, S. F. Performance Evaluation Tests for Environmental Research (PETER): Code Substitution test. Proceedings of the Seventh Psychology in the DoD Symposium. Colorado Springs, Colorado: USAF Academy, 1980.

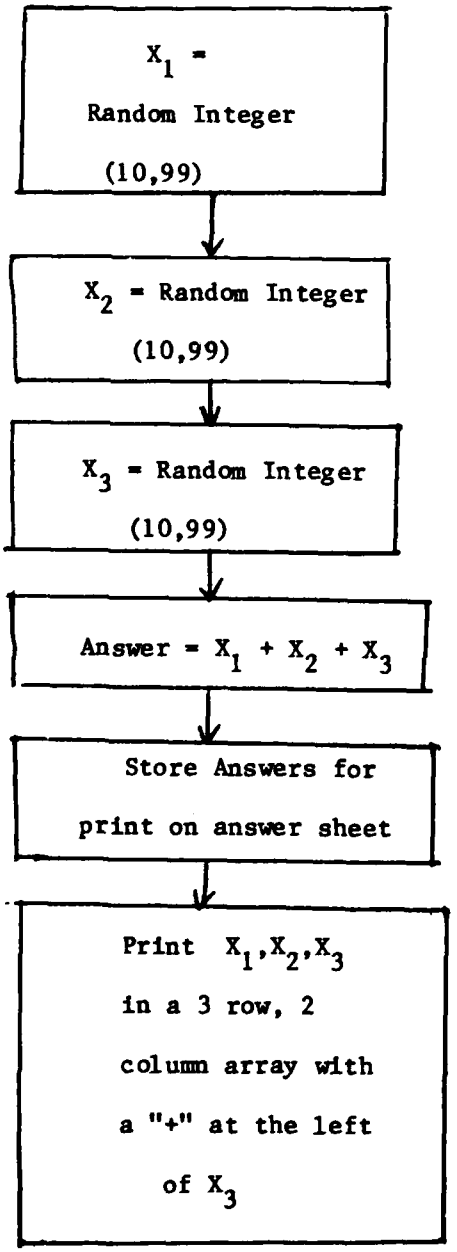
Rose, A. M. Human information processing: An assessment and research battery. Technical Report No. 46, University of Michigan, Ann Arbor, 1974.

Shannon, R. H., & Carter, R. C. A systematic approach to performance test battery development. In J. C. Guignard (Ed.) Proceedings of the International Workshop on Research Methods in Human Motion and Vibration Studies. New Orleans: Naval Biodynamics Laboratory, 1981, in press.

Wechsler, D. The measurement and appraisal of adult intelligence. Baltimore: Williams & Wilkins, 1958.

APPENDIX A
ADDITION TEST

Addition Test Block Diagram for a Single Item



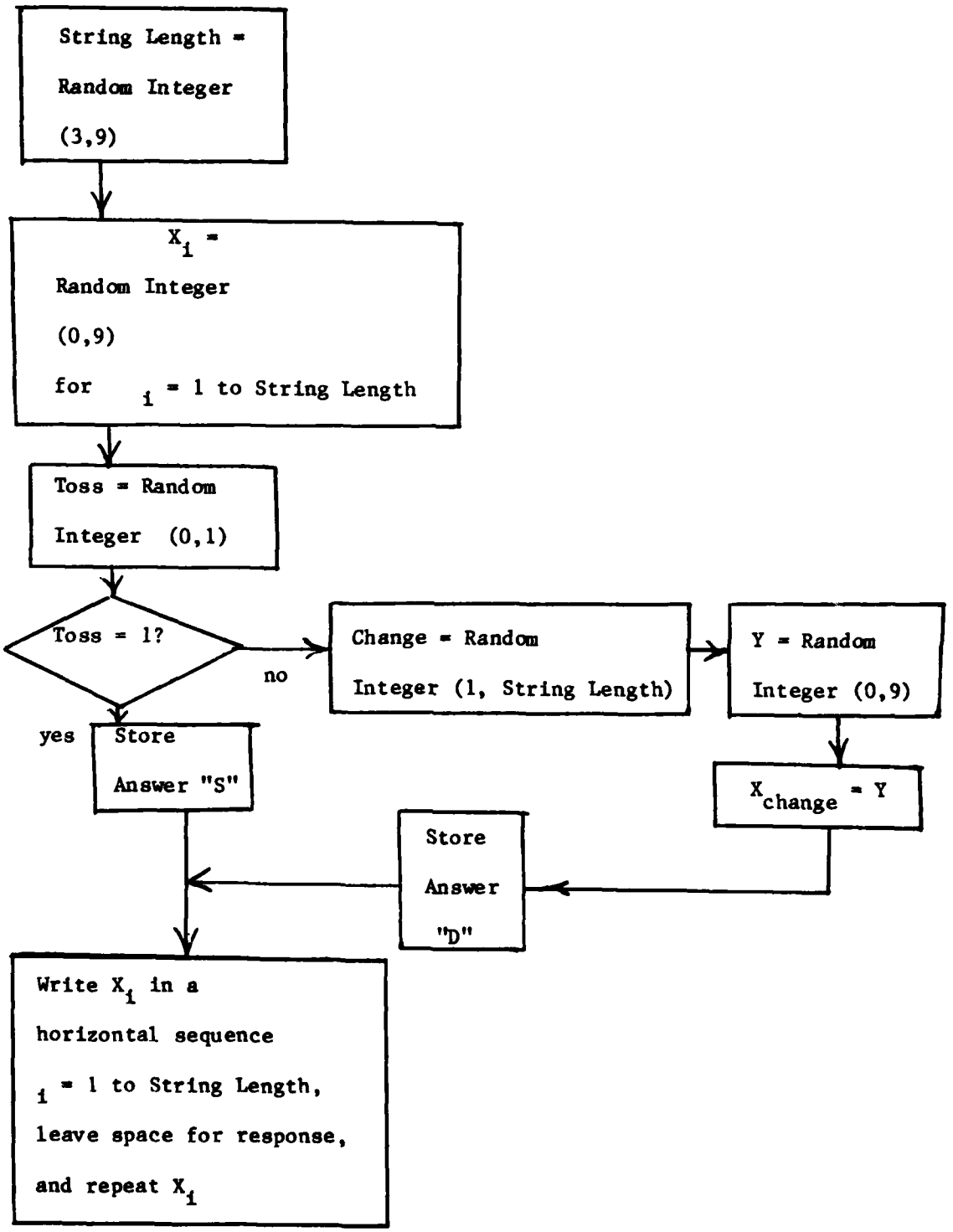
```
@ELT,SLID     .ADDITION/VERT
ELT 8R1 S74QIC 01/15/82 15:08:45 (->0)
      1.          00      @FOR,IS  MAIN,MAIN
      2.          00           DIMENSION ADDNUM(133,3),SUM(133)
      3.          00           INTEGER ADDNUM,SUM
      4.          00           NFORMS=75
      5.          00           NCOPY=1
      6.          00           DO 107 KK=1,NFORMS
      7.          00           DO 100 I=1,132
      8.          00           SUM(I)=0
      9.          00           DO 100 J=1,3
     10.          00      300  JRA=IRAND(X)
     11.          00        102 ADDNUM(I,J)=JRA-(JRA/100)*100
     12.          00           IF(ADDNUM(I,J).LT.10) GO TO 300
     13.          00           SUM(I)=SUM(I)+ADDNUM(I,J)
     14.          00        100 CONTINUE
     15.          00           DO 106 JJ=1,NCOPYS
     16.          00           WRITE(6,203) KK
     17.          00        203 FORMAT(1H1,'NAME:',25X,'SSN:',10X,'DATE/TIME:',/,/,' ADDITION TEST
     18.          00          *, FORM ',I2,/,/,/,5X,' THIS',
     19.          00          *IS A TEST TO SEE HOW QUICKLY AND ACCURATELY YOU CAN ADD.  IT IS NO
     20.          00          *T EXPECTED THAT YOU',/,' WILL FINISH ALL THE PROBLEMS IN THE TIME
     21.          00          *TIME ALLOWED.'/,5X,' YOU ARE TO WRITE THE ANSWERS BELOW THE PROBLE
     22.          00          *MS. SEVERAL PRACTICE PROBLEMS ARE ARE',/,' GIVEN BELOW WITH THE  F
     23.          00          *IRST ONE CORRECTLY WORKED. PRACTICE FOR SPEED ON THE OTHERS.',/,
     24.          00          *' THIS PRACTICE MAY HELP YOUR SCORE.',/,/,' PRACTICE PROBLEMS:')
     25.          00        104 WRITE(6,209) ((ADDNUM(IBEAR,J) ,IBEAR=121,132),J=1,3),SUM(121)
     26.          00        209 FORMAT(1H,/,/,2(12(8X,I2)/),12(7X,'+',I2),/,7X,I3,/,/,/,/)
     27.          00           WRITE(6,205)
     28.          00        205 FORMAT(1H,5X,'YOUR SCORE ON THIS TEST WILL BE THE NUMBER OF PROBL
     29.          00          *EMS THAT ARE ADDED CORRECTLY.',/,' WORK AS RAPIDLY AS YOU CAN WITH
     30.          00          *OUT SACRIFICING ACCURACY.',/,' YOU WILL HAVE TWO MINUTES FOR EACH
     31.          00          * OF THE TWO PARTS OF THIS TEST.',/,' EACH PART HAS ONE PAGE. WHEN
     32.          00          *YOU HAVE FINISHED PART ONE,STOP.',/,' PLEASE DO NOT GO ON TO PART
     33.          00          * TWO UNTIL YOU ARE ASKED TO DO SO.',/,/,/,' DO NOT TURN THIS PAGE
     34.          00          *UNTIL YOU ARE ASKED TO BEGIN,')
     35.          00           DO 105 II=1,2
     36.          00           WRITE(6,204) II
     37.          00      204  FORMAT(1H1,/,1H1,'PART',I2,'  (2 MINUTES)')
     38.          00           DO 103 L=1,49,12
     39.          00           M=L
     40.          00           IF (II.EQ.2)  M=L+60
     41.          00           NZNEB=M+11
     42.          00           WRITE(6,200) ((ADDNUM(MOOH,I),MOOH=M,NZNEB),I=1,3)
     43.          00        200 FORMAT(1H,/,/,12(8X,I2),/,12(8X,I2),/,12(7X,'+',I2),/,/,/,/)
     44.          00        103 CONTINUE
     45.          00           WRITE(6,202)
     46.          00        202 FORMAT(1H ,15('   STOP'))
     47.          00        105 CONTINUE
     48.          00        106 CONTINUE
     49.          00           WRITE(6,201) KK,(SUM(I),I=1,120)
     50.          00        201 FORMAT(1H1,'ANSWERS,FORM = ',I2,/,/,10(12(7X,I3),/,/))
     51.          00        107 CONTINUE
     52.          00           STOP
     53.          00           END
     54.          00      @FOR,IS RANINT,RANINT
     55.          00           INTEGER FUNCTION RANINT(I,M)
     56.          00           IF(I.EQ.1)  GO TO 101
     57.          00      100  J=IRAND(X)
     58.          00           L=MOD(34359738367,I)
     59.          00           K=34359738367-L
     60.          00           IF(J.GT.K) GO TO 100
     61.          00           RANINT=MOD(J,I)
```

```
 62.          00            IF(RANINT.EQ.0.AND.M.EQ.1) GO TO 100
 63.          00            GO TO 102
 64.          00       101  RANINT=1
 65.          00       102  RETURN
 66.          00            END
 67.          00      *FOR,IS IRAND,IRAND
 68.          00      C FUNCTION IRAND.....................................................
 69.          00      C
 70.          00      C OCTOBER 17, 1977
 71.          00      C
 72.          00      C GENERATE PSEUDO-RANDOM POSITIVE INTEGER WITH RECTANGULAR DISTRIBUTION
 73.          00      C IN FULL RANGE OF POSITIVE INTEGERS.
 74.          00      C THE ONE ARGUMENT IS DUMMY, AND HAS NO EFFECT.
 75.          00      C
 76.          00      C.....................................................................
 77.          00      C
 78.          00      C
 79.          00            FUNCTION IRAND(X)
 80.          00            LOGICAL AA,BB,LCOMPJ,LCOMPK                                    00000200
 81.          00            INTEGER A,B,P,Q,INTSIZ,M(98)                                   00000300
 82.          00            DATA J/0/,P/98/,Q/27/                                          00000400
 83.          00            DATA INTSIZ/35/                                                00000500
 84.          00            DATA M/ 55401076!6, 9465F96368,3109010275!,18771747577,        00000600
 85.          00           A 28429589995,17910665383, 2755776712,1577210837+,24868503636,  00000700
 86.          00           B 24050196658,10406361912,2589433048B,10239025078,23172152762,  00000800
 87.          00           C 16749650049,10546720580,16558947567, 4477013940, 5230881612,  00000900
 88.          00           D  5872042252,10375682927,3055574152!, 9897021507,13528500273,  00001000
 89.          00           E  4676199511, 4790613885,34237141705,16092853401, 6242230728,  00001100
 90.          00           F 18576454781,32549771790, 5345127795, 9050133044, 1996740055,  00001200
 91.          00           G 33666709376,32156029504,31994998514, 1331568448,25461263460,  00001300
 92.          00           H 10347229411,30965797332,12720210185,10885710696,28347389480,  00001400
 93.          00           I 21198968033,29917444937, 5707930159,16990730598, 9839223853,  00001500
 94.          00           J  4190023379,11263804815,12759423169, 2851834982,26267332518,  00001600
 95.          00           K 24635128900,21358828836,15048763990,31322345056,30055861835,  00001700
 96.          00           L 32542479077,27377696085,34215017208,10363747822,24895847765,  00001800
 97.          00           M 19361735837,28691085376,29117273251,12029489027, 55[illegible]2408,  00001900
 98.          00           N  9207135596,26254700186,22067063391,25885314502, 8687457660,  00002000
 99.          00           O 20893679244,23627546845,23468987177,29476161617,26666237334,  00002100
100.          00           P  3472118440,10963653676,14433404634,  518915085,12635407907,  00002200
101.          00           Q 11244663113,17297788094,25153870389, 5144201994,10310609825,  00002300
102.          00           R 13823834309,21578907839, 1350079164,31293115939,15063593696,  00002400
103.          00           S 15755048953,28707348220,27273213702,20062610167/              00002500
104.          00            EQUIVALENCE (AA,A),(BB,B),(MCOMPJ,LCOMPJ),(MCOMPK,LCOMPK)      00002600
105.          00            N=(2**(INTSIZ-1)-1)*2+1                                        00002700
106.          00            J=J+1                                                          00002800
107.          00            IF(J.GT.P) J=1                                                 00002900
108.          00            K=J+Q                                                          00003000
109.          00            IF(K.GT.P)K=K-P                                                00003100
110.          00            MCOMPJ=N-M(J)                                                  00003200
111.          00            MCOMPK=N-M(K)                                                  00003300
112.          00            A=M(K)                                                         00003400
113.          00            B=M(J)                                                         00003500
114.          00            BB=LCOMPJ.AND.AA.OR.LCOMPK.AND.BB                              00003600
115.          00            M(J)=B                                                         00003700
116.          00            IRAND=M(J)                                                     00003800
117.          00            RETURN                                                         00003900
118.          00            END                                                            00004000
119.          00      *MAP,IS   THIS,MP0060/ADDVT
120.          00       LIB SCC$*RLIB.

END ELT.  ERRORS: NONE.  TIME:  0.634 SEC.  IMAGE COUNT: 120
```

APPENDIX B

NUMBER COMPARISON

Number Comparison Test Block Diagram for a Single Item

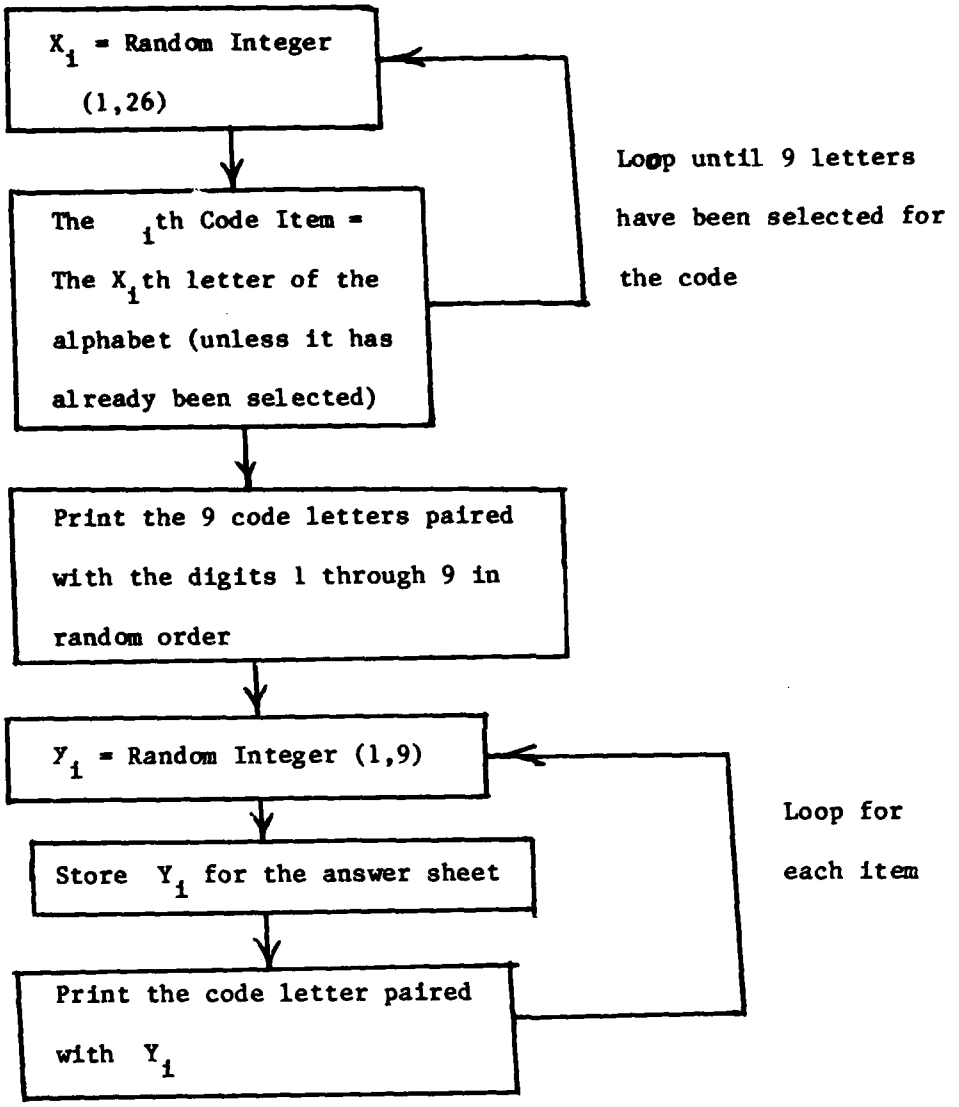


```
@ELT,SL ID    .ADDITION/HOR
ELT 8RI S74Q1C 01/11/82 15:16:31 (->0)
      1.            00      @FOR,IS  MAIN,MAIN
      2.            00            COMMON IANS(112,3)
      3.            00            NFORM=75
      4.            00            NCOPY=1
      5.            00            DO 1000 IFORM=1,NFORM
      6.            00            DO 1001 ICOPY=1,NCOPY
      7.            00            WRITE(6,201) IFORM
      8.            00      201   FORMAT(1H1,/,1H1,' NAME:',25X,'SUBJECT NUMBER:',15X,'DAY/DATE/TIME
      9.            00           *:',/,
     10.            00           */,' NUMBER COMPARISON TEST,FORM',I3,/,/,20X,'INSTRUCTIONS',/,/,/
     11.            00           $                              ,5X,'THIS IS A TEST OF HOW QUICKLY YOU
     12.            00           *CAN COMPARE TWO NUMBERS',/,' AND DECIDE WHETHER THEY ARE THE SAME.
     13.            00           *IF THE NUMBERS ARE THE SAME, PUT  S   ON THE LINE BETWEEN THEM.',
     14.            00           $/,' IF THE NUMBERS ARE DIFFERENT, PUT  D  ON THE LINE BETWEEN',  '
     15.            00           * THEM.',/,' ',
     16.            00           *'SEVERAL EXAMPLES ARE GIVEN BELOW WITH THE FIRST THREE MARKED CORR
     17.            00           *ECTLY.  PRACTICE FOR SPEED ON THE OTHERS.',/,' WORK FROM LEFT TO R
     18.            00           $IGHT, ROW BY ROW.',/)
     19.            00            WRITE(6,202)
     20.            00      202   FORMAT( /,11X,'123456789--S--123456789',13X,'135465--D--125465',15
     21.            00           *X,'7013487--S--7013487',///)
     22.            00            CALL ITEMS(7)
     23.            00            WRITE(6,205)
     24.            00      205   FORMAT(1H0,5X,'YOUR SCORE WILL BE THE NUMBER OF PAIRS MARKED CORRE
     25.            00           *CTLY MINUS A FRACTION OF THE NUMBER MARKED',/,' INCORRECTLY.  THER
     26.            00           *EFORE IT WILL NOT BE TO YOUR ADVANTAGE TO GUESS UNLESS YOU HAVE SO
     27.            00           *M',                                                          '
     28.            00           *E',/,' IDEA OF WHETHER OR NOT THE NUMBERS ARE THE SAME.',/,' Y',  '
     29.            00           *'OU WILL HA',                                                '
     30.            00           *VE THREE (3) MINUTES FOR THIS TEST.  IT HAS SEVERAL PAGES.',/,' SO
     31.            00           * WHEN YOU FINISH ONE PAGE PLEASE GO ON TO THE NEXT UNTIL YOU ARE A
     32.            00           *SKED TO STOP.',//,' DO NOT TURN THIS PAGE UNTIL ASKED TO BEGIN.',/
     33.            00           *,1H1,/,1H1)
     34.            00            CALL ITEMS(69)
     35.            00            WRITE(6,207)
     36.            00      207   FORMAT(/9(10X,'STOP'))
     37.            00            WRITE(6,206) (((IANS(L,M),M=1,3),L=1,69)
     38.            00      206   FORMAT(1H1, 69(3(21X,A1,11X)////))
     39.            00      1001  CONTINUE
     40.            00      1000  CONTINUE
     41.            00            STOP
     42.            00            END
     43.            00      @FOR,IS ITEMS,ITEMS
     44.            00            SUBROUTINE ITEMS(NLINES)
     45.            00            COMMON IANS(112,3)
     46.            00            DIMENSION X(9),STRING(10), L(3),ISTRIN(2,10,3)
     47.            00           $,IVAR(10)
     48.            00           *,IAN(2)
     49.            00            DATA IVAR/'0','1','2','3','4','5','6','7','8','9'/
     50.            00            DATA IAN/'S','D'/
     51.            00            DO 101 J=1,NLINES
     52.            00            DO 100 K=1,3
     53.            00            DO 103 I=1,10
     54.            00            ISTRIN(1,I,K)='       '
     55.            00            ISTRIN(2,I,K)='       '
```

```
 56.          00      103   CONTINUE
 57.          00            IANS(J,K)=IAN(I)
 58.          00            CALL ERTRAN(3,IDATE,ITIME)
 59.          00            IF (J*K .EQ. 1) X(1)=ABS(FLOAT(ITIME))
 60.          00            CALL RANDU(X,6)
 61.          00            L(K)=X(1)*7+3  * LENGTH OF THE STRING STRING
 62.          00            LL=L(K)
 63.          00            CALL RANDU(STRING,L(K))
 64.          00            IN=L(K)*X(3)+1.0  * INSERT A CHANGE AT STRING(IN)
 65.          00            ICHANG=X(4)*10.0+1.0
 66.          00            DO 102 I=1,LL
 67.          00            ISTRIN(2,I,K)=STRING(I)*10.0
 68.          00            NN=ISTRIN(2,I,K)+1
 69.          00            ISTRIN(2,I,K)=IVAR(NN)
 70.          00            M=9-L(K)+1
 71.          00            ISTRIN(1,M,K)=ISTRIN(2,I,K)
 72.          00      102   CONTINUE
 73.          00            IF(X(2).LE.0.5) ISTRIN(2,IN,K)=IVAR(ICHANG)
 74.          00            JN=9-L(K)+IN
 75.          00            IF(ISTRIN(1,JN,K).NE.ISTRIN(2,IN,K)) IANS(J,K)=IAN(2)
 76.          00      100   CONTINUE
 77.          00            WRITE(6,200)(((ISTRIN(II,IJ,IK),IJ=1,9),II=1,2),IK=1,3)
 78.          00      200   FORMAT(1H ,3(10X,9A1,'-- --',9A1)///)
 79.          00            IF (J.EQ.14.OR.J.EQ.28.OR.J.EQ.42.OR.J.EQ.56.OR.J.EQ.70.OR.J.EQ.84
 80.          00           *.OR.J.EQ.98.OR.J.EQ.112) WRITE(6,201)
 81.          00      201   FORMAT(1H1,/,1H1)
 82.          00      101   CONTINUE
 83.          00            RETURN
 84.          00            END
 85.          00      *FOR.IS  RANDU,RANDU
 86.          00            SUBROUTINE RANDU(X,N)
 87.          00            DIMENSION X(N)
 88.          00            DATA IA/0000000002001/
 89.          00            DATA IT/02000000000000/
 90.          00            DATA IFLAG/0/
 91.          00            XDIV=FLOAT(IT)
 92.          00       25     FORMAT(1X,I12,2X,I10,2X,E12.5)
 93.          00            IF (IFLAG .EQ. 0) IX=X(1)
 94.          00            IF (IFLAG .NE. 0) IX=ILAST
 95.          00            DO 10 I=1,N
 96.          00            ICON=IX*IA+1
 97.          00            IX=ABS(MOD(ICON,IT))
 98.          00            X(I)=FLOAT(IX)/XDIV
 99.          00      10    CONTINUE
100.          00      20    FORMAT(1X,10(F10.5))
101.          00            IFLAG=1
102.          00            ILAST=IX
103.          00            RETURN
104.          00            END
105.          00      *MAP.IS  THIS,MP0060/ADDHOR
106.          00       LIB SCC$*RLIB.

END ELT.  ERRORS: NONE. TIME:  0.493 SEC. IMAGE COUNT: 106

*BRKPT PRINT$/SF
```

APPENDIX C
CODE SUBSTITUTION

Code Substitution Test Block Diagram for a Single Item



```
*FOR,IS    MAIN,MAIN
FOR E3AB-01/12/82-10:42:40 (,0)

   MAIN PROGRAM

   STORAGE USED: CODE(1) 000235; DATA(0) 001137; BLANK COMMON(2) 000000


   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   RANINT
    0004   NINTR$
    0005   NWDU$
    0006   NIO3$
    0007   NIO1$
    0010   NIO2$
    0011   NSTOP$


   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000022 10L       0001   000051 100L      0000   001062 1000F     0001   000013 111G      0001   000021 116G
    0001   000043 126G      0001   000076 144G      0001   000113 156G      0001   000114 161G      0001   000140 176G
    0001   000051 20L       0000   001074 2000F     0001   000164 212G      0001   000171 216G      0001   000212 234G
    0001   000217 240G      0001   000116 30L       0000   001104 3000F     0001   000104 360L      0000   001122 4000F
    0000   001115 987F      0000 I 000000 I         0000 I 000022 IANS      0000 I 001046 II        0000 I 001047 ISUM
    0000 I 000011 J         0000 I 001057 JJ        0000 I 000450 K         0000 I 000416 LETTER    0000 I 001045 NCOPYS
    0000 I 001044 NFORMS    0000 I 001060 NQ        0000 I 001061 NR        0000 I 001051 N1        0000 I 001054 N2
    0000 I 001053 N3        0000 I 001055 N5        0000 I 001056 N6        0000 I 001050 N7        0000 I 001052 N8
    0003 I 000000 RANINT


00101      1*            DIMENSION I(9),J(9)      ,IANS(14,18),LETTER(26),K(14,18)                    000000
00103      2*            INTEGER RANINT                                                               000001
00104      3*            DATA LETTER /'A','B','C','D','E','F','G','H','I','J'    ,'K','L','          000001
00104      4*           *M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z' /                     000001
00106      5*            NFORMS=29                                                                    000001
00107      6*            NCOPYS=1                                                                     000003
00110      7*            DO 450 II=1,NFORMS                                                           000006
00113      8*            ISUM=0                                                                       000013
00114      9*            N7=0                                                                         000014
00115     10*            DO 350 N1=1,9                                                                000022
00120     11*       10   N8=RANINT(27,1)                                                              000022
00121     12*            I(N1)=LETTER(N8)                                                             000026
00122     13*            IF(N1.EQ.1) GO TO 100                                                        000031
00124     14*            N3=N1-1                                                                      000034
00125     15*            DO 200 N2=1,N3                                                               000037
00130     16*            IF(I(N2).EQ.I(N1)) GO TO 10                                                  000043
00132     17*       200  CONTINUE                                                                     000051
00134     18*       100  CONTINUE                                                                     000051
00135     19*        20  J(N1)=RANINT(10,1)                                                          000051
00136     20*            IF(N1.EQ.1) GO TO 360                                                        000055
```

```
00140     21*           N3=N1-1                                                    000060
00141     22*           IF(N1.EQ.9) J(N1)=45-ISUM                                  000063
00143     23*           DO 300 N2=1,N3                                             000076
00146     24*           IF(J(N1).EQ.J(N2)) GO TO 20                                000076
00150     25*     300   CONTINUE                                                   000104
00152     26*     360   ISUM=ISUM+J(N1)                                           000104
00153     27*     350   CONTINUE                                                   000114
00155     28*           DO 400 N5=1,13                                             000114
00160     29*           DO 500 N6=1,18                                             000114
00163     30*           N8=N7                                                      000114
00164     31*     30    N7=RANINT(10,1)                                            000116
00165     32*           IF(N8 .EQ.N7) GO TO 30                                     000122
00167     33*           K(N5,N6)=      I(N7)                                       000124
00170     34*           IANS(N5,N6)=J(N7)                                          000127
00171     35*     500   CONTINUE                                                   000140
00173     36*     400   CONTINUE                                                   000140
00175     37*           DO 550 JJ=1,NCOPYS                                         000140
00200     38*           WRITE(6,1000) (I(NQ),NQ=1,9),II                            000140
00204     39*     1000  FORMAT(1H1,'CODE',6X,9(A1,6X),20X,'NAME:',20X,'FORM ',I2)  000151
00205     40*           WRITE(6,2000)(J(NR),NR=1,9)                                000151
00210     41*     2000  FORMAT(1X ,'DIGIT',3X,9( '(',1X,I1,1X,')',2X)//)           000164
00211     42*           DO 700 N5=1,13                                             000164
00214     43*           WRITE(6,3000)(K(N5,N6),N6=1,18)                            000164
00222     44*     3000  FORMAT(1X,18(    ,2X,A1,2X,    ,2X)/1X,18('(',3X,')',2X)//) 000201
00223     45*     700   CONTINUE                                                   000201
00225     46*     550   CONTINUE                                                   000201
00227     47*           WRITE(6,987)  II                                           000201
00232     48*     987   FORMAT(1H1, 'ANSWERS FOR FORM ',I2)                       000212
00233     49*           DO 600 N5=1,13                                             000212
00236     50*           WRITE(6,4000) (IANS(N5,N6),N6=1,18)                        000212
00244     51*     4000  FORMAT(1X ,18(2X,I1,4X)/)                                  000230
00245     52*     600   CONTINUE                                                   000230
00247     53*     450   CONTINUE                                                   000230
00251     54*           STOP                                                       000230
00252     55*           END                                                        000234

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS RANINT,RANINT
FOR E3AB-01/12/82-10:42:44 (,0)

   FUNCTION RANINT      ENTRY POINT 000052

   STORAGE USED: CODE(1) 000061; DATA(0) 000016; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   IRAND
    0004   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000003 100L      0001   000037 101L      0001   000041 102L      0000   000006 INJP$     0003 I 000000 IRAND
    0000 I 000001 J         0000 I 000004 K         0000 I 000003 L         0000 I 000000 RANINT   0000 R 000002 X


00101      1•            INTEGER FUNCTION RANINT(I,M)                                    000000
00103      2•            IF(I.EQ.1)  GO TO 101                                           000000
00105      3•      100   J=IRAND(X)                                                      000003
00106      4•            L=MOD(34359738367,I)                                            000006
00107      5•            K=34359738367-L                                                 000012
00110      6•            IF(J.GT.K) GO TO 100                                            000014
00112      7•            RANINT=MOD(J,I)                                                 000017
00113      8•            IF(RANINT.EQ.0.AND.M.EQ.1) GO TO 100                            000023
00115      9•            GO TO 102                                                       000035
00116     10•      101   RANINT=1                                                        000037
00117     11•      102   RETURN                                                          000041
00120     12•            END                                                             000060

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
*FOR,IS IRAND,IRAND
FOR E3AB-01/12/82-10:42:46 (.0)

   FUNCTION IRAND        ENTRY POINT 000100

   STORAGE USED: CODE(1) 000104; DATA(0) 000165; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   XPII
    0004   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    C000 I 000151 A        0000 L 000151 AA       0000 I 000152 B        0000 L 000152 BB       0000   000156 INJPS
    0000 I 000003 INTSIZ   0000 I 000000 IRAND    0000 I 000146 J        0000 I 000150 K        0000 L 000153 LCOMPJ
    0000 L 000154 LCOMPK   0000 I 000004 M        0000 I 000153 MCOMPJ   0000 I 000154 MCOMPK   0000 I 000147 N
    0000 I 000001 P        0000 I 000002 Q


00100      1*     C FUNCTION IRAND.....................................................          000000
00100      2*     C                                                                              000000
00100      3*     C OCTOBER 17, 1977                                                             000000
00100      4*     C                                                                              000000
00100      5*     C GENERATE PSEUDO-RANDOM POSITIVE INTEGER WITH RECTANGULAR DISTRIBUTION        000000
00100      6*     C IN FULL RANGE OF POSITIVE INTEGERS.                                          000000
00100      7*     C THE ONE ARGUMENT IS DUMMY, AND HAS NO EFFECT.                                000000
00100      8*     C                                                                              000000
00100      9*     C.....................................................................         000000
00100     10*     C                                                                              000000
00100     11*     C                                                                              000000
00101     12*           FUNCTION IRAND(X)                                                       000000
00103     13*           LOGICAL AA,BB,LCOMPJ,LCOMPK                                  00000200   000000
00104     14*           INTEGER A,B,P,Q,INTSIZ,M(98)                                 00000300   000000
00105     15*           DATA J/0/,P/2147483647/,Q/27/                                00000400   000000
00111     16*           DATA INTSIZ/55/                                              00000500   000000
00113     17*           DATA M/ 5540107616, 9465596368,3109010275l,1877174757,       00000600   000000
00113     18*          A 28429589985,17910665383, 2755776712,15772108374,24868563636,           00000700   000000
00113     19*          B 24050196658,10406361912,25894330488,10239025078,23172152762,           00000800   000000
00113     20*          C 16749650049,[illegible]546720580,16558947567, 4477013940, 5230881612,   00000900   000000
00113     21*          D  5872042252,[illegible]475682927,3055574l521, 9897021507,13528500273,   00001000   000000
00113     22*          E  [illegible]676199511, [illegible]90613885,34237141705,16092853401, 6242230728,   00001100   000000
00113     23*          F 18576454781,32549771790, 5345127795, 9050133044, 1996740055,           00001200   000000
00113     24*          G 33666709376,32156029504,31994998514, 1331568448,25461263460,           00001300   000000
00113     25*          H 10347229411,30965797332,12720210185,10885710696,28347382480,           00001400   000000
00113     26*          I 21198968033,29917444937, 5707930159,16990730598, 9839223853,           00001500   000000
00113     27*          J  4190023379,11263804815,12759423169, 2851834982,26267332518,           00001600   000000
00113     28*          K 24635128900,21358828836,15048763990,31322345056,30055861835,           00001700   000000
```

```
00113     29*          L 32542479077,27377696085,34215017208,10363747822,24895847765,     00001800  000000
00113     30*          M 19361735837,28691085376,29117273251,12029489027, 5529942408,     00001900  000000
00113     31*          N  9207135596,26254700186,22067063391,25885314502, 8687457660,     00002000  000000
00113     32*          O 20893679244,23627546845,23468987177,29476161617,26666237334,     00002100  000000
00113     33*          P  3472118440,10963653676,14433404634,  518915885,12635407907,     00002200  000000
00113     34*          Q 11244663113,17297788094,25153870389, 5144201994,10310609825,     00002300  000000
00113     35*          R 13823834309,21578907839, 1350079164,31293115939,15063593696,     00002400  000000
00113     36*          S 15755048953,28707348220,27273213702,20062610167/                 00002500  000000
00115     37*           EQUIVALENCE (AA,A),(BB,B),(MCOMPJ,LCOMPJ),(MCOMPK,LCOMPK)         00002600  000000
00116     38*           N=(2**(INTSIZ-1)-1)*2+1                                           00002700  000000
00117     39*           J=J+1                                                             00002800  000012
00120     40*           IF(J.GT.P) J=1                                                    00002900  000015
00122     41*           K=J+Q                                                             00003000  000023
00123     42*           IF(K.GT.P)K=K-P                                                   00003100  000026
00125     43*           MCOMPJ=N-M(J)                                                     00003200  000037
00126     44*           MCOMPK=N-M(K)                                                     00003300  000043
00127     45*           A=M(K)                                                            00003400  000047
00130     46*           B=M(J)                                                            00003500  000051
00131     47*           BB=LCOMPJ.AND.AA.OR.LCOMPK.AND.BB                                 00003600  000053
00132     48*           M(J)=B                                                            00003700  000062
00133     49*           IRAND=M(J)                                                        00003800  000064
00134     50*           RETURN                                                            00003900  000065
00135     51*           END                                                               00004000  000103

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

APPENDIX D
BADDELEY TEST

```
SBISA*QEIDOC(1).TBADTST
      C     A THREE MINUTE REASONING TEST BASED ON GRAMMATICAL TRANSFORMATION
      C
      C
      C     A QEI PROJECT FOR HUMAN FACTORS , LT R. CARTER
      C
      C                          BY HARVEY SBISA, Q.E.I., INC.
      C
      C
            COMPILER (DATA=IBM)
            INTEGER RAND(32)
            DIMENSION STAT(7,32), MASK(2,32)
            DIMENSION MSK(64)
            EQUIVALENCE (MASK(1,1),MSK(1))
            DATA MSK /'T                F           F           F     T
           1     ','T                F           F     T               F
           2  ','T                F     T           T               F     '
           3,'      F     T           T           T           T           ','T
           4              F           F     T           T           ','
           5  F           F     T                 F     T           ','     F
           6           F     */
      C
      C
            DATA STAT(:,1) /'A IS NOT FOLLOWED BY B (BA]    T     F     ','A DO
           1ES NOT PRECEDE B   (AB]    T     F     ','A FOLLOWS B            (A
           2B]    T     F     ','B PRECEDES A           (AB]    T     F     ','A
           3 IS PRECEDED BY B     (BA]    T     F     ','A FOLLOWS B
           4 (BA]    T     F     ','B IS FOLLOWED BY A     (AB]    T     F     '
           5,'B IS NOT PRECEDED BY A (AB]    T     F     ','B PRECEDES A
           6    (BA]    T     F     ','A IS PRECEDED BY B     (AB]    T     F
           7  ','A DOES NOT FOLLOW B    (AB]    T     F     ','A IS NOT FOLLOWE
           8D BY B (AB]    T     F     ','A IS NOT PRECEDED BY B (AB]    T
           9F    ','B FOLLOWS A            (AB]    T     F     ','A IS NOT PREC
           AEDED BY B (BA]    T     F     ','B IS PRECEDED BY A     (BA]    T
           B   F     */
            DATA STAT(1,17) /'B IS NOT PRECEDED BY A (BA]    T     F     ','A I
           1S FOLLOWED BY B     (AB]    T     F     ','B IS NOT FOLLOWED BY A (
           2AB]    T     F     ','A PRECEDES B           (AB]    T     F     ','
           3A DOES NOT PRECEDE B   (BA]    T     F     ','B DOES NOT PRECEDE A
           4  (BA]    T     F     ','B IS NOT FOLLOWED BY A (BA]    T     F
           5','B DOES NOT PRECEDE A   (AB]    T     F     ','B DOES NOT FOLLOW
           6A    (BA]    T     F     ','A PRECEDES B           (BA]    T     F
           7   ','B FOLLOWS A            (BA]    T     F     ','B IS FOLLOWED B
           8Y A     (BA]    T     F     ','A IS FOLLOWED BY B     (BA]    T
           9 F    ','B IS PRECEDED BY A     (AB]    T     F     ','B DOES NOT F
           AOLLOW A    (AB]    T     F     ','A DOES NOT FOLLOW B    (BA]    T
           B    F     */
      C
         10 WRITE (6,90)
            WRITE (6,60)
            READ (5,70) INBR
      C
            WRITE (6,80) INBR
            READ (5,70) ICOPY
            IF (ICOPY.EQ.0) ICOPY=1
            WRITE (6,90)
      C     INITIALIZE RANDOM TABLE (INDEX TABLE INTO THE BASIC TEST SET)
      C     TO 1 THRU 32 IN ORDER TO ALLOW A BASIC SET PRINT OUT
      C     IF SO DESIRED.
      C
            DO 20 I=1,32
         20   RAND(I)=I
      C
```

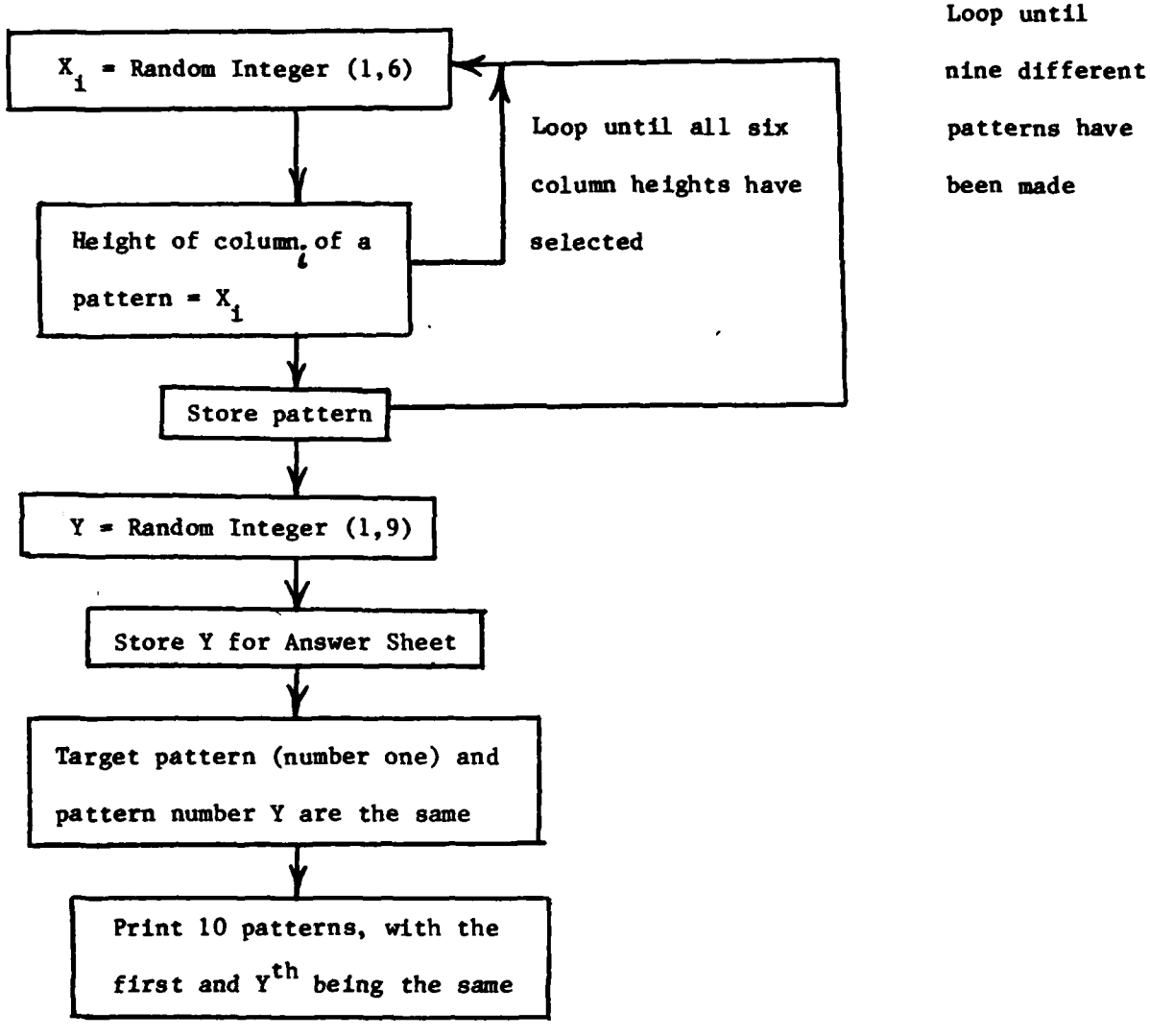
```
      C     INITIALIZE DATE AND TIME CELLS TO 'BASIC', 'SET'. ALL TESTS
      C     GENERATED, AND THEIR CORRESPONDING ANSWER MASKS WILL
      C     HAVE THE SAME, NUMBER, DATE, AND TIME OF DAY.
      C
            IDATE='BASIC '
            ITIME='SET   '
      C     $99 TO JUMP AROUND 'DO 4'
            IF (INBR.EQ.0) CALL BTG (RAND,IDATE,ITIME,INBR,ICOPY,$40)
      C     GENERATE DESIRED 'INBR' NUMBER OF BADDELEY TESTS !
            DO 30 I=1,INBR
              NBR=I
              CALL RANDOM (RAND,IDATE,ITIME)
              CALL BTG (RAND,IDATE,ITIME,NBR,ICOPY,$50)
         30   CONTINUE
              WRITE (6,90)
         40 WRITE (6,90)
              WRITE (6,100)
              READ (5,110) ANS
              IF (ANS.EQ.'YES') STOP
              IF (ANS.EQ.'NO ') GO TO 10
              WRITE (6,120) ANS
         50 STOP
      C
      C
         60 FORMAT (T10,'HOW MANY BADDELEY TESTS SHALL I GENERATE FOR YOU?'/,T
           110,'PLEASE TYPE IN A 2 DIGIT NUMBER FROM 00 TO 99      '/)
         70 FORMAT (I2)
         80 FORMAT (T10,'HOW MANY COPIES OF THE ',I2,' UNIQUE SETS OR FORMS OF
           1 THE BADDELEY TEST SHALL I GENERATE FOR YOU?'/,T10,'PLEASE TYPE IN
           2 A 2 DIGIT NUMBER FROM 01 TO 99 '/)
         90 FORMAT (1H1)                          • GENERATE A BLANK PAGE
        100 FORMAT (T10,'ALL THROUGH?'/,T10,'PLEASE RESPOND:  YES  OR  NO ',/)
        110 FORMAT (A3)
        120 FORMAT (T10,'YAH NYEH PANYIMAYOU SHTOW VIE SKASAL, SAYO NARA YALL!
           1 ')
      C
            SUBROUTINE BTG (RAND,IDATE,ITIME,INBR,ICOPY,$)
            INTEGER RAND(32)
      C     PRINT THE HEADING
            DO 30 L=1,ICOPY
              IF (INBR.NE.0) ITIME=ITIME+1
              WRITE (6,40)
              WRITE (6,50) IDATE,ITIME,INBR,L
              WRITE (6,60)
      C
      C
      C     PRINT OUT THE BASIC SET
      C
      C
              WRITE (6,70) INBR,L,IDATE,ITIME
              DO 10 K=1,32
                I=RAND(K)
                WRITE (6,80) K,(STAT(J,I),J=1,7)
         10     CONTINUE
              WRITE (6,60)                        • TOP OF FORM
              IF (L.NE.ICOPY) GO TO 30
              WRITE (6,90) INBR,L,IDATE
              DO 20 K=1,32
                I=RAND(K)
                WRITE (6,100) K,(MASK(J,I),J=1,2)
         20     CONTINUE
         30   CONTINUE
              IF (INBR.EQ.0) RETURN 6
```

```
      RETURN
C
C
   40 FORMAT (1H1,T47,'NAVAL AEROSPACE MEDICAL RESEARCH LAB'/,T60 'BOX 2
     1 19407',/,T57,'MICHOUD STATION',/,T50,'NEW ORLEANS LOUISIANA /0189',
     2//,T53,'HUMAN FACTORS DIVISION'///,T2,'NAME: ',T39,'SSN: ',T68,'DA
     3TE/TIME: '//,T17,'THIS IS A TEST TO SEE HOW QUICKLY AND ACCURATELY
     4 YOU CAN UNDERSTAND COMPLEX'/,T17,'STATEMENTS. IT IS NOT EXPECTED
     5THAT YOU WILL FINISH ALL THE PROBLEMS IN THE TIME'/,T17,'ALLOWED.'
     6//,T17,'YOU ARE TO PLACE A SLASH THROUGH THE CORRECT ANSWER, T OR
     7F, DEPENDING ON'/,T17,'WHETHER EACH STATEMENT IS TRUE OR FALSE. SE
     8VERAL PRACTICE PROBLEMS ARE GIVEN','/,T17,'AND THE FIRST TWO ARE CO
     9RRECTLY WORKED FOR YOU.'/,/,T17,'THIS PRACTICE MAY HELP YOUR SCORE
     A.'//)
   50 FORMAT (T17,'PRACTICE PROBLEMS:'//,T42,' 1.  A FOLLOWS B ---------
     1- (BA)    T     F',T78,'/'/,T42,' 2.  B PRECEDES A --------- (AB)
     2   T     F',T84,'/'/,T42,' 3.  A IS FOLLOWED BY B --- (AB)    T
     3  F'/,T42,' 4.  B IS NOT FOLLOWED BY A (BA)    T     F'/,T42,' 5.
     4 B IS PRECEDED BY A --- (BA)    T     F'/,T42,' 6.  A DOES NOT PRE
     5CEDE B - (BA)    T     F'/,//,T17,'     YOUR SCORE ON THIS TEST WI
     6LL BE THE NUMBER OF PROBLEMS ANSWERED CORRECTLY.',//,T17,'DO NOT T
     7URN THIS PAGE UNTIL YOU ARE ASKED TO DO SO.',///////////////,T100,
     8'TEST GENERATION ID'/,T100,'DATE= ',A6,' QE1#  ',A6,/,T100,'NBR= '
     9,I2,' COPY= ',I2,/)
   60 FORMAT (1H1)                          • TOP OF FORM
   70 FORMAT (1H1,T50,'BADDELEY TEST NBR ',I2,' COPY ' I2,T100,'DATE= ',
     1A6,' QE1#  ',A6,/////////////)
   80 FORMAT (T42,I2,'.  ',7A6)
   90 FORMAT (1H1,T46,'ANSWER MASK FOR BADDELEY TEST NBR ',I2,' COPIES G
     1ENERATED ',I2,' DATE: ',A6,/////////////)
  100 FORMAT (T42,I2,'.  ',T78,2A6)
C
      SUBROUTINE RANDOM (Z,IDATE,ITIME)
      INTEGER X,Z(32),Y
      DATA ICON /0777777777777/
      DATA ICHK /0777777777760/          • MASK FOR 32
      DATA IFLAG /0/
      WRITE (6,60)
      IF (IFLAG.NE.0) GO TO 10
      CALL ERTRAN (9,IDATE,ITIME) • USE TIME OF DAY AS INITIAL VALUE
   10 CONTINUE
      IF (IFLAG.NE.0) GO TO 20
      Y=FLD(26,10,ITIME)
      IFLAG=1
   20 IA=0
      J=0
      I=1
      K1=5**4+1
      KMOD=5**10
   30 Y=AMOD(K1*Y+3,KMOD)
      X=AMOD(Y,32)+1
      L=X-1
      I=I+1
      IF (I.GT.1000) GO TO 50      • IF UNABLE TO GEN A NEW NR IN 1K TRIE
      IF (FLD(L,1,IA).EQ.FLD(1,1,ICON)) GO TO 30 • IF THIS NO PREV GEN'D
      J=J+1                                • THEN GENERATE ANOTHER!
      Z(J)=BOOL(X)                         • A NEW NBR SO ADD TO THE ARRAY !
      FLD(L,1,IA)=FLD(1,1,ICON)
      IF (IA.EQ.ICHK) GO TO 40
      GO TO 30
   40 PRINT 70, Z
      RETURN
   50 PRINT 80, IA
```

```
      RETURN 0
C
   60 FORMAT (1H1)                          • TOP OF FORM - GEN A BLANK PAGE
   70 FORMAT (T58,'RANDOM TABLE FOR THIS SET '/,1X,3214)
   80 FORMAT (1X,012)
C
      END

•XQT QE1DOC.MP0060/BADDELEY
```

APPENDIX E
PATTERN RECOGNITION

Pattern Recognition Test Block Diagram for a Single Item



```
SBISA*QEIDCC(1).PATREC
      1       *DELETE,C      1.
      2       *DELETE,C      2.
      3       *ASG,CP            1.,F40///10000
      4       *ASG,CP            2.,F40///10000
      5       * .
      6       * .
      7       * .
      8       * .
      9       *FOR,SBISA PATMACHER
     10       C      A QEI PROJECT FOR HUMAN PERFORMANCE SCIENCE DEPARTMENT, NBL
     11       C
     12       C                TASK ORIGINATOR :  LT. R. CARTER
     13       C                 TASK ASSIGNED TO : H. E. SBISA
     14              DIMENSION IANS(52,10)
     15              COMMON IPAT(6,6,10), IFLAG,IDATE,ITIME
     16       1       WRITE(6,400)
     17       400          FORMAT(T10,'HOW MANY UNIQUE PATTERN RECOGNITION TESTS SHALL I
     18            $I GENERATE?',/,T10,'PLEASE TYPE IN A 2 DIGIT NUMBER FROM 01 TO ',
     19            $'99.'/)
     20              READ(5,500)ITST
     21       500    FORMAT(I2,
     22              WRITE(6,401)ITST
     23       401    FORMAT(T10,'HOW MANY COPIES OF THE ',I2,' UNIQUE TESTS SHALL I
     24            $GENERATE?',/,T10,'PLEASE TYPE IN A 2 DIGIT NUMBER FROM 01 TO 99
     25            $.'/)
     26              READ(5,500)ICPY
     27       C       SET UP NBR OF TESTS LOOP
     28              DO 2 NT=1,ITST
     29       C
     30                          IP = NT
     31       C     SET IP TO TEST # FOR  SECTOR WRITING  AND THEN READING
     32       C       CLEAR AND INSERT SPACES INTO THE ANS ARRSAY
     33              DO 111 IJ=1,10
     34              DO 111 JI=1,52
     35              IANS(JI,IJ)='
     36       111    CONTINUE
     37              IFLAG=0        * RESET 1ST TIME THRU THIS TEST FLAG
     38       C
     39       C       SET UP NBR OF LINES/TEST LOOP
     40              NANS=1
     41              ISET=((IP-1)*780)        * SECTOR ADR OF THIS IP
     42       C    MAKE IT EVEN SECTORS BY WRITE TO NOT BY WRITE UPTO...!
     43              CALL SETADR(1,ISET)          * BEGINING OF CURRENT TEST TO BE PRINTED TAPE
     44              DO31L=1,52
     45              CALL PATMAK(NANS,IANS,IP)
     46       5      CONTINUE
     47              WRITE(2)IANS        *WRITE THE ANSWER ARRAY OUT TO TAPE UNIT 2 (REALLY A DRUM)
     48       2       CONTINUE
     49              END FILE 1
     50              END FILE 2
     51              REWIND 1
     52              REWIND 2
     53       C
     54       C       SET UP SECOTR ADR OF THE CURRENT TEST YO BE PRINTED.
     55       C       AND START THE SHOW!
     56              DO 4 IP =1,ITST        * NBR OF TESTS LOOP
     57              ISET=((IP-1)*780)        * SECTOR ADR OF THIS IP
     58              READ (2)IANS        * READ IN THE ANSWER ARRAY
     59              DO 5 IC=1,ICPY        * NBR OF COPIES LOOP
     60
     61              ISET=((IP-1)*780)        * SECTOR ADR OF THIS IP
     62              CALL SETADR(1,ISET)          * BEGINING OF CURRENT TEST TO BE PRINTED
```

```
      WRITE(6,200)
200     FORMAT(1HI,T51,'NAVAL BIODYNAMICS LABORATORY'/,T60,'BOX 29407',
     1/T57,'MICHOUD STATION',/,T50,'NEW ORLEANS LOUISIANA 70189',
     2//,T57,'BY QEI INC.',
     3/,T60,'FOR',
     4/,T53,'HUMAN FACTORS DIVISION'///,T2,'NAME:',T39,'SSN: ',T60,'DATE
     5/TIME: ',
     6/,T17,'THIS IS A TEST TO SEE HOW QUICKLY AND',
     $' ACCURATELY YOU CAN RECOGNIZE PATTERNS.'/,T17,'IT IS NOT EXPECTED
     $ TH'                                          '
     $AT YOU WILL FINISH ALL THE PROBLEMS IN THE TIME ALLOWED.'//,T17,'
     $'YO'                                          '
     $U ARE TO NOTE THE PATTERN AT THE LEFT OF EACH ROW OF PATTERNS.',
     $/,T17,
     $'THEN UNDERLINE THE PATTERNS IN EACH ROW WHICH MATCH THE ONE ON TH
     $E LEFT.',/,T17,'SOME PRACTICE EXAMPLES ARE GIVEN BELOW.',
     $' AND THE FIRS',
     $'T ONE IS CORRECTLY SOLVED.',/,T17,'PRACTICE FOR SPEED ON THE OTHERS.',/,
     $RS.',/,T17,
     $'THIS PRACTICE MAY HELP YOUR SCORE.',/,/)
       CALL FRINT      • PRINT OUT 1 ROW OF PATTERNS
      WRITE(6,201)(IANS(1,1),I=1,10)
201     FORMAT(1X,A6,6X,9(6X,A6)/)
       CALL PRINT      • PRINT OUT 1 ROW OF PATTERNS
       WRITE(6,202)IDATE,ITIME,IP,IC
202   FORMAT(1)17,'YOUR SCORE ON THIS TEST WILL BE THE NUMBER OF PROBLEMS
     $ THAT ARE CORRECTLY SOLVED.'/,T17,'WORK AS RAPIDLY AS YOU CAN WITHOU
     $OU',
     $'T SACRIFICING ACCURACY.',/,T17,'YOU WILL HAVE TWO MINUTES FOR EACH OF
     $H OF THE',
     $' TWO PARTS OF THIS TEST.',/,T17,'EACH PART COVERS SEVERAL PAGES.
     $KEEP GOING UNTIL ASKED TO STOP OR UNTIL YOU SEE',/,/,15('    STOP'
     $),
     $/,/,T17,'  DO NOT TURN THIS PAGE UNTIL ASKED TO DO SO.',T96,
     $'TEST GENERATION ID ',/,T96,'DATE  ',A6,' QEI# ',A6,/,T96,'TEST
     $',I2,' COPY 'I2,/)
       CALL SKIP
      DO 107 III=1,2
      CALL SKIP
      DO 106 II=1,5
      CALL SKIP
      CALL SKIP
      WRITE(6,300)IP,IC,III,II,IDATE,ITIME
300   FORMAT(1X,'TEST',I3,T15,'COPY ',I3,T25,'PART ',I3,T35,'PAGE ',I3,
     $' DATE ',A6,' ID QEI# ',A6)
      DO 106 I=1,5
       CALL PRINT
106   CONTINUE
      WRITE(6,204)
204   FORMAT(1H ,15('    STOP'))
      CALL SKIP
107   CONTINUE
        CALL SKIP
5      CONTINUE
C       PRINT OUT ANSWER SHEET
       LC=2                  • INIT TOTAL LINES COUNTER
       DO 600 IA=1,2             • PART LOOP
       CALL SKIP
       DO 501 IB=1,5        • PAGE LOOP
       CALL SKIP
      WRITE(6,207) IP,IA,IB,IDATE,ITIME
207   FORMAT(1X,'TEST',I3,T25,'PART ',I3,T35,'PAGE ',I3,
     $' DATE ',A6,' ID QEI# ',A6,//////)
```

```
126             DO 501 IC=1,5          • LINE LOOP
127             LC=LC+1
128            WRITE(6,402)(IANS(LC,I),I=1,10)
129      501    CONTINUE
130             WRITE(6,204)
131             CALL SKIP
132      600     CONTINUE
133      4      CONTINUE               • GO DO ANOTHER TESTIF SUCH IS THE CASE
134      402     FORMAT(1X,A6,6X,9(6X,A6)/////////////)
135            STOP
136             END
137      • .
138      • .
139      • .
140      • .
141      •FOR,SBISA   PAGE,PAGE
142            SUBROUTINE SKIP
143            WRITE(6,203)
144      203   FORMAT(1H1 )
145            RETURN
146                                                                    END
147      •FOR,SBISA RANDOM,RANDOM
148            SUBROUTINE   RANDOM(Z,ICHK,NBR)
149            INTEGER X,Z(10),Y
150            DATA ICON/0777777777777/
151             COMMON IPAT(6,6,10), IFLAG,IDATE,ITIME
152            IF(IFLAG.NE.0)GOTO5
153            CALL ERTRAN(9,IDATE,ITIME)  • USE TIME OF DAY AS INITIAL VALUE
154            Y=FLD(26,NBR,ITIME)
155            IFLAG=1
156            WRITE(6,100) IDATE,ITIME
157      100   FORMAT(T100,'DATE= ',A6,' TIME= ',A6)
158      5     IA=0
159             J=0
160             I=1
161            K1=5**4+1
162            KMOD=5**10
163      20    Y=AMOD(K1*Y+3,KMOD)
164            X=AMOD(Y,NBR )+1
165            L=X-1
166            I=I+1
167            IF(I.GT.1000)GOTO50  • IF UNABLE TO GEN A NEW NR IN 1K TRIES, STOP
168      C     !F(FLD(L,1,IA).EQ.FLD(1,1,ICON))GOTO20   • IF THIS NO PREV GEN'D
169            J=J+1                                    • THEN GENERATE ANOTHER!
170             IF(J.GT.NBR)GOTO30
171            Z(J)=BOOL(X)       • A NEW NBR SO ADD TO THE ARRAY !
172            FLD(L,1,IA)= FLD(1,1,ICON)
173            IF(IA.EQ.ICHK)GOTO  30
174      C      WRITE(6,39)J,L,X
175      39      FORMAT(T5,' J= ',I3,' L = ',I3,' X = 'I3)
176            GOTO 20
177      30    CONTINUE   •  PRINT  40,Z
178      C     PRINT 40,Z
179      C     PRINT 55,IA,ICHK
180      40    FORMAT(T50,1014)
181            RETURN
182      50    PRINT 55,IA,ICHK
183      55    FORMAT(1X,012,5X,012)
184            RETURN 0
185            END
186      • .
187      • .
188      • .
```

```
189       • .
190       •FOR,SBISA    LOHAGO,
191             SUBROUTINE PATMAK(NANS,IANS,IP)
192             DIMENSION X(10)
193             INTEGER Z(10)
194             COMMON IPAT(6,6,10),IFLAG,IDATE,ITIME
195             DIMENSION  ILINE(6),IANS(52,10)
196              DATA I6/0770000000000/
197             DATA I10/0777400000000/
198             DO 102 III=1,10  • CREATE TARGET PATTERN •
199              CALL RANDOM(ILINE,I6,6)
200             DO 101 II=1,6
201             DO 100 I=1,6
202             IPAT(I,II,III)=' '
203             IF(I.LE.ILINE(II)) IPAT(I,II,III)='X'
204       100   CONTINUE
205       101   CONTINUE
206       102   CONTINUE
207                     X(1)  =  2.74*ILINE(1)
208              CALL RANDU(X,1)     • PLANT TARGET 1 IN LOC 2 TO 10
209              N=X(1)*8.0+2.0
210              IF(N.EQ.1)N=10
211              IF(N.GT.10)N=10
212             CALL RANDOM (Z,I10,10)
213              M=Z(N)
214              IF(M.EQ.1)M=10
215             IANS(NANS,M)='------'
216              IF(NANS.EQ.1)GO TO 103
217              ENCODE(6,69,TMP,ICHAR)M
218       69     FORMAT(I6)
219              FLD(12,12,IANS(NANS,M))=FLD(24,12,TMP)
220       103    CONTINUE
221             DO 105 II=1,6
222             DO 104 I=1,6
223             IPAT(I,II,M)=IPAT(I,II,1)
224       104   CONTINUE
225       105   CONTINUE
226             NANS=NANS+1  • INCREMENT ROW COUNTER ON THE ANSWER ARRAY
227       C     WRITE(6,69)NANS
228              WRITE(1)IPAT,IDATE,ITIME
229              RETURN
230              END
231       • .
232       • .
233       • .
234       • .
235       •FOR,SBISA     PRINT,
236              SUBROUTINE PRINT
237              COMMON IPAT(6,6,10),IFLAG,IDATE,ITIME
238              READ(1)IPAT,IDATE,ITIME
239       C     TRANSPOSE AND PRINT PATTERNS
240             WRITE(6,210)(((IPAT(I,II,III),II=1,6), III=1,10),I=6,1)
241       210   FORMAT(/,6(1X,6A1,6X,9(6X,6A1),/,),/,/,/,/)
242             RETURN
243             ENTRY FRINT
244              READ(1)IPAT,IDATE,ITIME
245             WRITE(6,211)(((IPAT(I,II,III),II=1,6), III=1,10),I=6,1)
246       211   FORMAT (/,6(1X,6A1,6X,9(6X,6A1),/,))
247                                                                    END
248       • .
249       • .
250       • .
251       • .
```

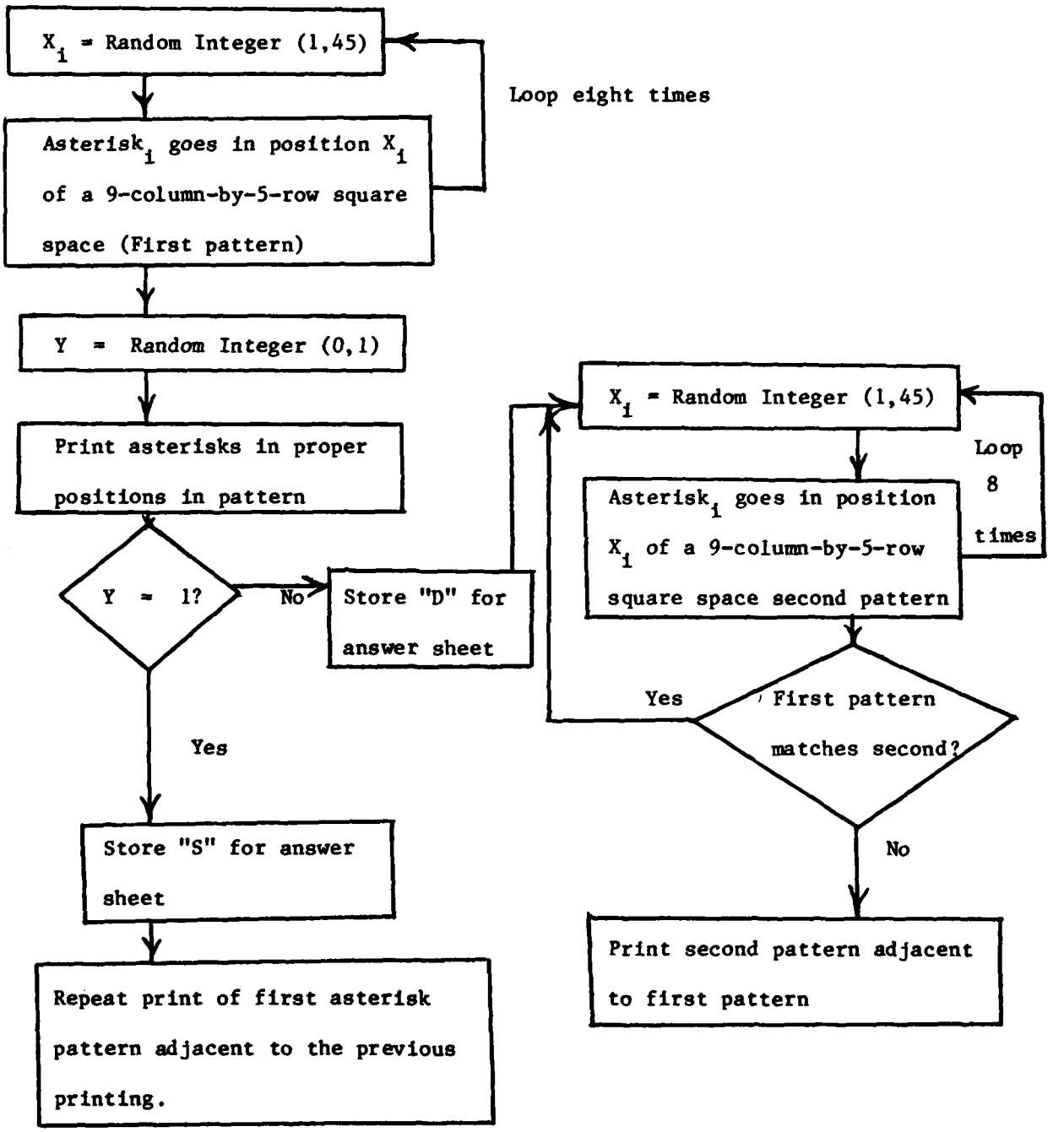
```
252       •MAP,IS     SYMB,MP0060/PATMACHER
253            LIB SCCLIB•MATHSTAT
254       •XQT      MP0060/PATMACHER
255       02
256       01


•XQT QEIDOC.MP0060/PATMACHER
        HOW MANY UNIQUE PATTERN RECOGNITION TESTS SHALL I GENERATE?
        PLEASE TYPE IN A 2 DIGIT NUMBER FROM 01 TO 99.

        HOW MANY COPIES OF THE  2 UNIQUE TESTS SHALL I    GENERATE?
        PLEASE TYPE IN A 2 DIGIT NUMBER FROM 01 TO 99
```

DATE= 081580 TIME= 162037
DATE= 081580 TIME= 162105

APPENDIX F

PATTERN COMPARISON (NUMSER)

Pattern Comparison Test Block Diagram for a Single Item



```
*FOR,IS NUMSER,NUMSER
FOR E3A8-01/11/82-14:15:05 (.0)

   MAIN PROGRAM

   STORAGE USED: CODE(1) 000371; DATA(0) 032011; BLANK COMMON(2) 000000


   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   RANINT
    0004   NINTR$
    0005   NWDU$
    0006   NI02$
    0007   NI01$
    0010   NSTOP$


   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000127 101L      0001   000117 103L      0001   000017 113G      0001   000024 116G      0001   000031 121G
    0001   000036 127G      0001   000112 134G      0001   000163 150G      0001   000213 164G      0001   000231 174G
    0000   000234 200F      0000   000251 201F      0000   000253 202F      0000   000262 203F      0001   000324 203G
    0000   000404 205F      0001   000325 205G      0001   000326 207G      0001   000352 223G      0001   000353 225G
    0000 R 000000 ANS       0000 I 000232 I         0000 I 000225 IANS      0000 I 000233 ICOPY     0000 I 000223 IFORM
    0000 I 000224 ILINE     0000 I 000231 J         0000 I 000226 K         0000 I 000227 KK        0000 I 000230 KKK
    0000 I 000526 MATRIX    0000 I 000526 NATRIX    0000 I 000221 NCOPY     0000 I 000220 NFORM     0000 I 000222 NL
    0003 I 000000 RANINT
```

```
00101      1•          DIMENSION MATRIX(6,9,5,48)                                          000000
00103      2•          DIMENSION ANS(48,3)                                                 000001
00104      3•          DIMENSION NATRIX(6,45,48)                                           000001
00105      4•          EQUIVALENCE(MATRIX(6,9,5,48),NATRIX(6,45,48))                       000001
00106      5•          INTEGER RANINT                                                      000001
00107      6•          NFORM=5                                                             000001
00110      7•          NCOPY=1                                                             000003
00111      8•          NL=48                                                               000005
00112      9•          DO 106 IFORM=1,NFORM                                                000011
00115     10•          DO 104 ILINE=1,NL                                                   000024
00120     11•          DO 107 IANS=1,3                                                     000031
00123     12•          ANS(ILINE,IANS)='D'                                                 000031
00124     13•    107   CONTINUE                                                            000036
00126     14•          DO 102 K=1,6                                                        000036
00131     15•          KK=RANINT(3,1)                                                      000044
00132     16•          KKK=K-1                                                             000051
00133     17•          DO 101 J=1,45                                                       000102
00136     18•          IF((K.EQ.2.OR.K.EQ.4.OR.K.EQ.6).AND.KK.EQ.2) GO TO 103              000112
00140     19•          NATRIX(K,J,ILINE)=' '                                               000113
00141     20•          GO TO 101                                                           000115
00142     21•    103   NATRIX(K,J,ILINE)=NATRIX(KKK,J,ILINE)                               000117
00143     22•          IANS=K/2                                                            000120
00144     23•          ANS(ILINE,IANS)='S'                                                 000122
00145     24•    101   CONTINUE                                                            000131
00147     25•          DO 100 I=1,8                                                        000131
```

```
00152     26*           J=RANINT(45,1)                                                    000163
00153     27*           IF(KK.NE.2.OR.K.EQ.1.OR.K.EQ.3.OR.K.EQ.5) MATRIX(K,J,ILINE)='*'   000167
00155     28*      100  CONTINUE                                                          000213
00157     29*      102  CONTINUE                                                          000213
00161     30*      104  CONTINUE                                                          000213
00163     31*           DO 105 ICOPY=1,NCOPY                                              000213
00166     32*           WRITE(6,203) IFORM                                                000213
00166     33*      C         INSTRUCTIONS                                                 000213
00171     34*           WRITE(6,205)                                                      000220
00173     35*           DO 105 ILINE=1,NL                                                 000231
00176     36*           IF(ILINE.EQ.1.OR.ILINE.EQ.7.OR.ILINE.EQ.13.OR.ILINE.EQ.19.OR.ILINE 000231
00176     37*          *.EQ.25.OR.ILINE.EQ.31.OR.ILINE.EQ.37.OR.ILINE.EQ.43)              000231
00176     38*          *WRITE(6,201)                                                      000231
00201     39*           WRITE(6,200)(((MATRIX(K,I,J,ILINE),I=1,9),K=1,6),J=1,5)           000316
00215     40*      105  CONTINUE                                                          000342
00220     41*           WRITE(6,202)IFORM,((ANS(ILINE,IANS),IANS=1,3),ILINE=1,NL)         000342
00232     42*      106  CONTINUE                                                          000364
00234     43*      200  FORMAT(4(2(5X,9A1),15X,2(5X,9A1),15X,2(5X,9A1)/)                 000364
00234     44*          $,3(5X,9A1,' --- ',9A1,15X)////)                                  000364
00235     45*      201  FORMAT(1H1,/,1H1)                                                000364
00236     46*      202  FORMAT(1H1,' ANSWERS,FORM '13,/,48(3(5X,A1)/))                   000364
00237  *DIAGNOSTIC*  THIS STATEMENT HAS TOO FEW RIGHT PARENTHESES.
00237     47*      203  FORMAT(1H1,/,1H1,' NAME:',25X,'SUBJECT NUMBER:',15X,'DAY/DATE/TIME 000364
00237     48*          *:',/,/,                                                          000364
00237     49*          *' PATTERN  COMPARISON TEST,FORM',I3,/,/,20X,'INSTRUCTIONS',/,/,/  000364
00237     50*          $                              ,5X,'THIS IS A TEST OF HOW QUICKLY YOU C 000364
00237     51*          *AN COMPARE TWO PATTERNS',/,' AND DECIDE WHETHER THEY ARE THE SAME. 000364
00237     52*          *IF THE PATTERNS ARE THE SAME,PUT  S   ON THE LINE BETWEEN THEM.',  000364
00237     53*          $/,' IF THE PATTERNS ARE DIFFERENT,PUT  D  ON THE LINE BETWEEN',    ' 000364
00237     54*          * THEM.',/,' WORK FROM LEFT TO RIGHT, ROW BY ROW.  DO AS MANY ITEMS 000364
00237     55*          * AS YOU CAN WITHOUT MAKING MISTAKES. '                           000364
00240     56*      205  FORMAT(1H0,5X,'YOUR SCORE WILL BE THE NUMBER OF PAIRS MARKED CORRE 000364
00240     57*          *CTLY MINUS A FRACTION OF THE NUMBER MARKED',/,' INCORRECTLY.  THER 000364
00240     58*          *EFORE IT WILL NOT BE TO YOUR ADVANTAGE TO GUESS UNLESS YOU HAVE SO 000364
00240     59*          *M',                                                              ' 000364
00240     60*          *E',/,' IDEA OF WHETHER OR NOT THE PATTERNS ARE THE SAME.',/,' Y',  000364
00240     61*          *'OU WILL HA',                                                    ' 000364
00240     62*          *VE TWO  (2) MINUTES FOR THIS TEST.  IT HAS SEVERAL PAGES,',/,' SO 000364
00240     63*          * WHEN YOU FINISH ONE PAGE PLEASE GO ON TO THE NEXT UNTIL YOU ARE A 000364
00240     64*          *SKED TO STOP.',//,' DO NOT TURN THIS PAGE UNTIL ASKED TO BEGIN.')  000364
00241     65*           STOP                                                              000364
00242     66*           END                                                               000370

      END OF COMPILATION:          1 DIAGNOSTICS.
```

```
@FOR,1S RANINT,RANINT
FOR E3AB-01/11/82-14:15:15 ',0)

   FUNCTION RANINT      ENTRY POINT 000052

   STORAGE USED: CODE(1) 000061; DATA(0) 000016; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   IRAND
    0004   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000003 100L       0001   000037 101L       0001   000041 102L       0000   000006 INJP$      0003 I 000000 IRAND
    0000 I 000001 J          0000 I 000004 K          0000 I 000003 L          0000 I 000000 RANINT    0000 R 000002 X


00101      1•           INTEGER FUNCTION RANINT(I,M)                                                    000000
00103      2•           IF(I.EQ.1)  GO TO 101                                                           000000
00105      3•     100   J=(RAND/X)                                                                      000003
00106      4•           L=MOD(34359738367,I)                                                            000006
00107      5•           K=34359738367-L                                                                 000012
00110      6•           IF(J.GT.K) GO TO 100                                                            000014
00112      7•           RANINT=MOD(J,I)                                                                 000017
00113      8•           IF(RANINT.EQ.0.AND.M.EQ.1) GO TO 100                                            000023
00115      9•           GO TO 102                                                                       000035
00116     10•     101   RANINT=1                                                                        000037
00117     11•     102   RETURN                                                                          000041
00120     12•           END                                                                             000060

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
*FOR,IS IRAND,IRAND
FOR E3AB-01/11/82-14:15:20 (,0)

   FUNCTION IRAND       ENTRY POINT 000100

   STORAGE USED: CODE(1) 000104; DATA(0) 000165; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   XPII
    0004   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0000 I 000151 A        0000 L 000151 AA       0000 I 000152 B        0000 L 000152 BB       0000   000156 INJP$
    0000 I 000003 INTSIZ   0000 I 000000 IRAND    0000 I 000146 J        0000 I 000150 K        0000 L 000153 LCOMPJ
    0000 L 000154 LCOMPK   0000 I 000004 M        0000 I 000153 MCOMPJ   0000 I 000154 MCOMPK   0000 I 000147 N
    0000 I 000001 P        0000 I 000002 Q


00100      1*     C FUNCTION IRAND.....................................................          000000
00100      2*     C                                                                              000000
00100      3*     C OCTOBER 17, 1977                                                             000000
00100      4*     C                                                                              000000
00100      5*     C GENERATE PSEUDO-RANDOM POSITIVE INTEGER WITH RECTANGULAR DISTRIBUTION         000000
00100      6*     C IN FULL RANGE OF POSITIVE INTEGERS.                                          000000
00100      7*     C                                                                              000000
00100      8*     C.....................................................................         000000
00100      9*     C                                                                              000000
00100      10*    C THE ONE ARGUMENT IS DUMMY, AND HAS NO EFFECT.                                000000
00100      11*    C                                                                              000000
00101      12*          FUNCTION IRAND(X)                                                        000000
00103      13*          LOGICAL AA,BB,LCOMPJ,LCOMPK                                 00000200     000000
00104      14*          INTEGER A,B,P,Q,INTSIZ,M(98)                                00000300     000000
00105      15*          DATA J/0/,P/98/,Q/27/                                       00000400     000000
00111      16*          DATA INTSIZ/35/                                             00000500     000000
00113      17*          DATA M/ 5540107616, 9465596368,31090102751,18771747577,     00000600     000000
00113      18*         A 28429589395,17910665383, 2755776712,15772108374,24858503636,  00000700  000000
00113      19*         B 24050196658,10406361912,25894330488,10239025078,23172152762,  00000800  000000
00113      20*         C 16749650049,10546720580,16558947557, 4477013940, 5230881612,  00000900  000000
00113      21*         D  5972042252,10375692927,30555741521, 9897021507,13528500273,  00001000  000000
00113      22*         E  4676199511, 4790613885,34237141705,16092853401, 6242230728,  00001100  000000
00113      23*         F 18576454781,32549771790, 5345127795, 9050133044, 1996740055,  00001200  000000
00113      24*         G 33666709376,32156029504,31994998514, 1331568448,25461263460,  00001300  000000
00113      25*         H 10347229411,30965797332,12720210185,10085710696,28347389480,  00001400  000000
00113      26*         I 21198968033,29917444937, 5707930159,16990730598, 2839223953,  00001500  000000
00113      27*         J  4190023379,11263804815,12759423169, 2851834982,26267332518,  00001600  000000
00113      28*         K 24635128900,21358328836,15048767990,31322345056,30055861835,  00001700  000000
00113      29*         L 32542479077,27377696085,34215012208,10363747822,24895847765,  00001800  000000
00113      30*         M 19361735837,28691085376,29117273251,12029489027, 5529942408,  00001900  000000
00113      31*         N  9207135596,26254700186,22067063391,25885314502, 8687457660,  00002000  000000
```

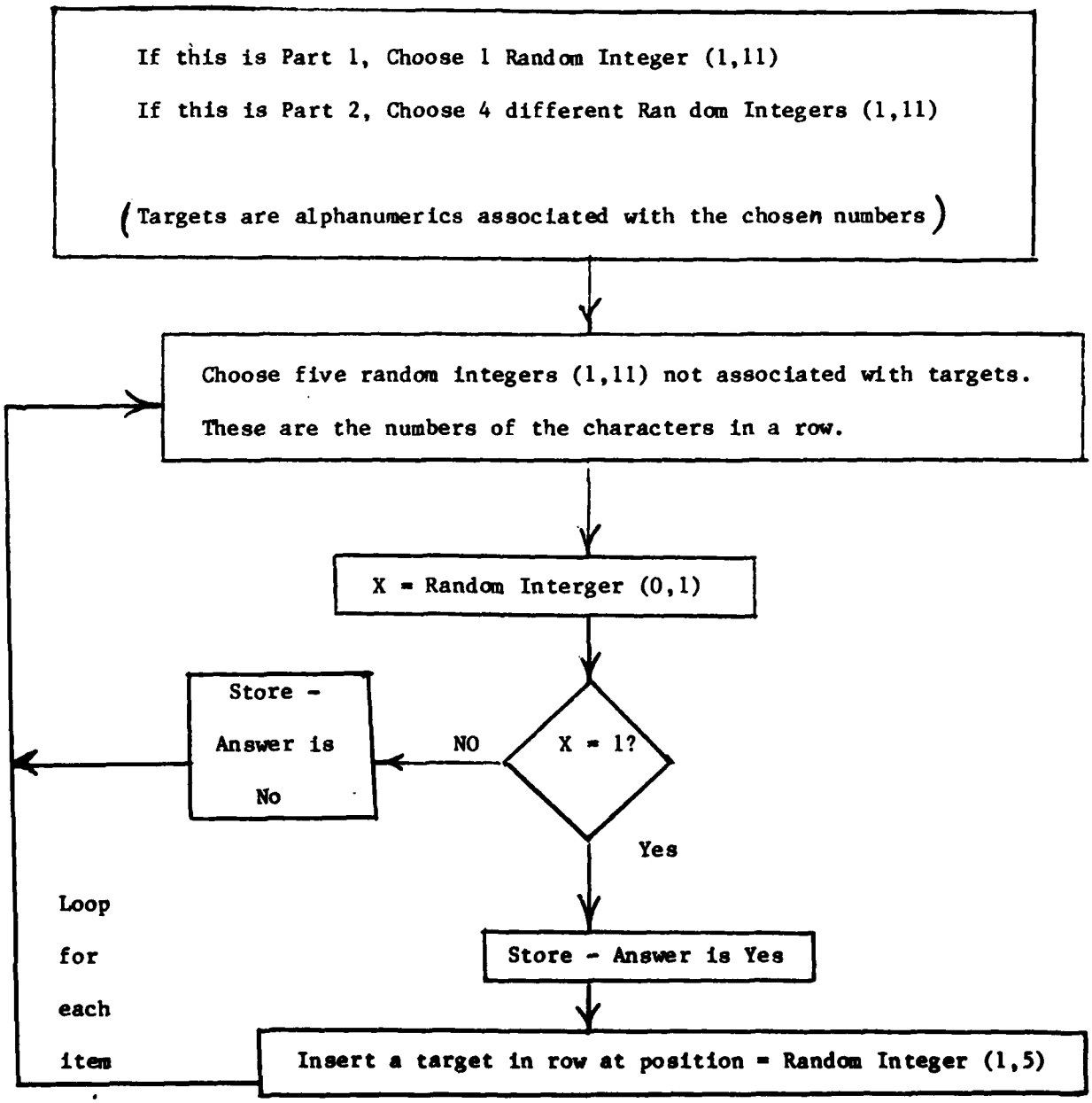
```
00113     32*          O 20893679244.23627546845.23468987177.29476161617.26666237334.      00002100    000000
00113     33*          P  3472118440.10963653676.14433404634.  5189158B5.12635407907.      00002200    000000
00113     34*          Q 11244663113.17297788094.25153870389.  5144201994.10310609825.      00002300    000000
00113     35*          R 13823834309.21578907839.  1350079164.31293115939.15063593696.      00002400    000000
00113     36*          S 15755048953.28707348220.27273213702.20062610167/                  00002500    000000
00115     37*           EQUIVALENCE (AA,A),(BB,B),(MCOMPJ,LCOMPJ),(MCOMPK,LCOMPK)          00002600    000000
00116     38*           N=(2**(INTSIZ-1)-1)*2+1                                            00002700    000000
00117     39*           J=J+1                                                              00002800    000012
00120     40*           IF(J.GT.P) J=1                                                     00002900    000015
00122     41*           K=J+Q                                                              00003000    000023
00123     42*           IF(K.GT.P)K=K-P                                                    00003100    000026
00125     43*           MCOMPJ=N-M(J)                                                      00003200    000037
00126     44*           MCOMPK=N-M(K)                                                      00003300    000043
00127     45*           A=M(K)                                                             00003400    000047
00130     46*           B=M(J)                                                             00003500    000051
00131     47*           BB=LCOMPJ.AND.AA.OR.LCOMPK.AND.BB                                  00003600    000053
00132     48*           M(J)=B                                                             00003700    000062
00133     49*           IRAND=M(J)                                                         00003800    000064
00134     50*           RETURN                                                             00003900    000065
00135     51*           END                                                                00004000    000103

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

APPENDIX G
LETTER SEARCH

Letter Search Test Block Diagram for a Single Item

Lookup Table

Alphanumeric Character	F	G	H	K	L	M	N	R	T	X	7
index number	1	2	3	4	5	6	7	8	9	10	11



```
*FOR,IS MAIN,MAIN
FOR E3AB-01/12/82-09:36:07 (,0)

   MAIN PROGRAM

   STORAGE USED: CODE(1) 000420; DATA(0) 011642; BLANK COMMON(2) 000000


   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   RANINT
    0004   NINTR$
    0005   NWDU$
    0006   NIO1$
    0007   NIO2$
    0010   NSTOP$


   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000130 105L      0001   000072 106L      0001   000171 108L      0001   000014 111G      0001   000021 115G
    0001   000062 122G      0001   000071 130G      0001   000121 140G      0001   000137 152G      0001   000152 155G
    0001   000170 160G      0001   000202 164G      0000   011570 200F      0000   011605 201F      0000   011551 205F
    0001   000306 216G      0001   000311 221G      0001   000316 225G      0001   000325 234G      0001   000332 240G
    0001   000333 242G      0001   000356 256G      0001   000366 266G      0001   000373 272G      0000   011574 299F
    0000 I 000000 ARRAY     0000 I 010624 IANS      0000 I 011546 ICOPY     0000 I 011526 IFORM     0000 I 011532 II
    0000 I 011535 III       0000 I 011547 IL        0000 I 011542 IN        0000 I 011534 IQ        0000 I 011530 ITARG
    0000 I 011544 ITOS      0000 I 011545 I1        0000 I 011541 I2        0000 I 011533 I3        0000 I 011537 J
    0000 I 011531 JI        0000 I 011550 JJ        0000 I 011540 K         0000 I 011543 KM        0000 I 011536 L
    0000 I 010610 LETTER    0000 I 011527 M         0000 I 011525 NCOPY     0000 I 011524 NFORM     0003 I 000000 RANINT
    0000 I 010600 TARGET


00101      1*          DIMENSION ARRAY(28,10,8,2),TARGET(4,2),LETTER(12),IANS(28,  8,2)                000000
00103      2*          DATA LETTER /'F','G','H','K','L','M','N','R','T','X','7',' '/                   000001
00105      3*          INTEGER RANINT,ARRAY,TARGET                                                     000001
00106      4*          NFORM=75                                                                        000001
00107      5*          NCOPY=1                                                                         000003
00110      6*          DO 110 IFORM=1,NFORM                                                            000006
00113      7*          M=1                                                                             000014
00114      8*          DO 103 ITARG=1,4,3                                                              000021
00117      9*          IF(ITARG.EQ.4) M=2                                                              000037
00121     10*          DO 130 JI=1,4                                                                   000043
00124     11*          TARGET(JI,M)=LETTER(12)                                                         000062
00125     12*    130   CONTINUE                                                                        000072
00127     13*          DO 102 II=1,ITARG                                                               000072
00132     14*    106   I3=RANINT(12,1)                                                                 000072
00133     15*          TARGET(II,M)=LETTER(I3)                                                         000076
00134     16*          IF(ITARG.EQ.1.OR.II.EQ.1) GO TO 105                                             000101
00136     17*          IQ=II-1                                                                         000111
00137     18*          DO 104 III=1,IQ                                                                 000114
00142     19*          IF(TARGET(III,M).EQ.TARGET(II,M))GO TO 106                                      000121
00144     20*    104   CONTINUE                                                                        000137
00146     21*    105   CONTINUE                                                                        000137
00147     22*    102   CONTINUE                                                                        000137
```

```
00151     23•           DO 100 L=1,8                                                    000137
00154     24•           DO 100 J=1,28                                                   000152
00157     25•           DO 101 K=1,5                                                    000171
00162     26•     108   I2=RANINT(I2,1)                                                000171
00163     27•           DO 109 IN=1,ITARG                                               000175
00166     28•           IF(LETTER(I2).EQ.TARGET(IN,M)) GO TO 108                        000202
00170     29•           ARRAY(J,K,L,M)=LETTER(I2)                                       000205
00171     30•     109   CONTINUE                                                        000212
00173     31•           KM=K-1                                                          000212
00174     32•           IF(ARRAY(J,K,L,M).EQ.ARRAY(J,KM,L,M)) GO TO 108                 000215
00176     33•     101   CONTINUE                                                        000227
00200     34•           ITOS = RANINT (11,1)                                            000227
00201     35•           IANS(J ,  L,M)=LETTER(7)                                        000234
00202     36•           IF(ITOS.LE.5) IANS(J ,  L,M)=LETTER(10)                         000236
00204     37•           I1=1                                                            000244
00205     38•           IF(ITARG.EQ.4) I1=RANINT(5,1)                                   000246
00207     39•           ARRAY(J,ITOS,L,M)=TARGET(I1,M)                                  000264
00210     40•     100   CONTINUE                                                        000311
00213     41•     103   CONTINUE                                                        000311
00215     42•           DO 111 ICOPY=1,NCOPY                                            000311
00220     43•           DO 107 M=1,2                                                    000311
00223     44•           WRITE(6,205) (TARGET(IL,M),IL=1,4 ),IFORM                       000311
00232     45•     205   FORMAT(1H1,'TARGET:  ',4A3,10X,'SUBJECT NAME/NUMBER:',25X,'DATE/TIM 000325
00232     46•          *E:',15X,'FORM ',I2)                                             000325
00233     47•           DO 107 JJ=1,28                                                  000325
00236     48•           WRITE(6,200)((ARRAY(JJ,K,L,M),K=1,5),L=1,8)                     000325
00247     49•     200   FORMAT(1H0,7(5A2,7X ),5A2)                                      000356
00250     50•     107   CONTINUE                                                        000356
00253     51•     111   CONTINUE                                                        000356
00255     52•           DO 121 M=1,2                                                    000356
00260     53•           WRITE(6,299) IFORM,M                                            000356
00264     54•     299   FORMAT(1H1,'ANSWERS,LETTER SEARCH FORM',I2,' PART ',I1)         000366
00265     55•           DO 121 JJ=1,28                                                  000366
00270     56•           WRITE(6,201)  (IANS(JJ,  L,M)       ,L=1,8)                     000366
00276     57•     201   FORMAT(1H0,8(A1,15X))                                           000413
00277     58•     121   CONTINUE                                                        000413
00302     59•     110   CONTINUE                                                        000413
00304     60•           STOP                                                            000413
00305     61•           END                                                             000417

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
*FOR,IS RANINT,RANINT
FOR E3AB-01/12/82-09:36:12 (,0)

   FUNCTION RANINT      ENTRY POINT 000052

   STORAGE USED: CODE(1) 000061; DATA(0) 000016; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   IRAND
    0004   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0001   000003 100L       0001   000037 101L       0001   000041 102L       0000   000006 INJP$      0003 I 000000 IRAND
    0000 I 000001 J          0000 I 000004 K          0000 I 000003 L          0000 I 000000 RANINT    0000 R 000000 X


                                                                                                                  000000
00101      1*           INTEGER FUNCTION RANINT(I,M)                                                              000000
00103      2*           IF(I.EQ.1)  GO TO 101                                                                     000003
00105      3*     100   J=IRAND(X)                                                                                000006
00106      4*           L=MOD(34359738367,I)                                                                      000012
00107      5*           K=34359738367-L                                                                           000014
00110      6*           IF(J.GT.K) GO TO 100                                                                      000017
00112      7*           RANINT=MOD(J,I)                                                                           000023
00113      8*           IF(RANINT.EQ.0.AND.M.EQ.1) GO TO 100                                                      000035
00115      9*           GO TO 102                                                                                 000037
00116     10*     101   RANINT=1                                                                                  000041
00117     11*     102   RETURN                                                                                    000060
00120     12*           END

       END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR,IS IRAND,IRAND
FOR E3AB-01/12/82-09:36:16 (,0)

   FUNCTION IRAND       ENTRY POINT 000100

   STORAGE USED: CODE(1) 000104; DATA(0) 000165; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   XPII
    0004   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0000 I 000151 A         0000 L 000151 AA        0000 I 000152 B         0000 L 000152 BB        0000   000156 INJP$
    0000 I 000003 INTSIZ    0000 I 000000 IRAND     0000 J 000146 J         0000 J 000150 K         0000 L 000153 LCOMPJ
    0000 L 000154 LCOMPK    0000 I 000004 M         0000 I 000153 MCOMPJ    0000 I 000154 MCOMPK    0000 I 000147 N
    0000 I 000001 P         0000 I 000002 Q


00100      1*     C FUNCTION IRAND.....................................................                000000
00100      2*     C                                                                                    000000
00100      3*     C OCTOBER 17, 1977                                                                   000000
00100      4*     C                                                                                    000000
00100      5*     C GENERATE PSEUDO-RANDOM POSITIVE INTEGER WITH RECTANGULAR DISTRIBUTION              000000
00100      6*     C IN FULL RANGE OF POSITIVE INTEGERS.                                                000000
00100      7*     C THE G'S ARGUMENT IS DUMMY, AND HAS NO EFFECT.                                      000000
00100      8*     C                                                                                    000000
00100      9*     C.....................................................................               000000
00100     10*     C                                                                                    000000
00100     11*     C                                                                                    000000
00101     12*           FUNCTION IRAND(X)                                                              000000
00103     13*           LOGICAL AA,BB,LCOMPJ,LCOMPK                                    00000200        000000
00104     14*           INTEGER A,B,P,Q,INTSIZ,M(98)                                   00000300        000000
00105     15*           DATA J/0/,P/98/,Q/27/                                          00000400        000000
00111     16*           DATA INTSIZ/35/                                                00000500        000000
00113     17*           DATA M/ 5540107616, 9465596368,31090102751,18771747577,        00000600        000000
00113     18*          A 28429589385,17910665383, 2755776712,15772108374,24868503636,  00000700        000000
00113     19*          B 24050196658,10406361912,25894330488,10239025078,23172152762,  00000800        000000
00113     20*          C 16749650049,10546720580,16558947567, 4477013940, 5230881612,  00000900        000000
00113     21*          D  5872042252,10375682927,30555741521, 9897021507,13528500273,  00001000        000000
00113     22*          E  4676199511, 4790613885,34237141705,16092853401, 6242230728,  00001100        000000
00113     23*          F 18576454781,32549771790, 5345127795, 9050133044, 1996740055,  00001200        000000
00113     24*          G 33666709376,32156029504,31934998514, 1331568448,25461263460,  00001300        000000
00113     25*          H 10347229411,30965797332,12720210185,10895710696,28347389480,  00001400        000000
00113     26*          I 21198968033,29917444937, 5707930159,16990730598, 9839223853,  00001500        000000
00113     27*          J  4190023379,11263804815,12759423169, 2851934982,26267332518,  00001600        000000
00113     28*          K 24535128900,21358828836,15048763990,31322345056,30055061835,  00001700        000000
00113     29*          L 32542479077,27377696085,34215017208,10363747822,24895847765,  00001800        000000
00113     30*          M 19361735837,28691085376,29117273251,12029489027, 5529942408,  00001900        000000
00113     31*          N  9207135596,26254700186,22067063391,25885314502, 8687457660,  00002000        000000
```

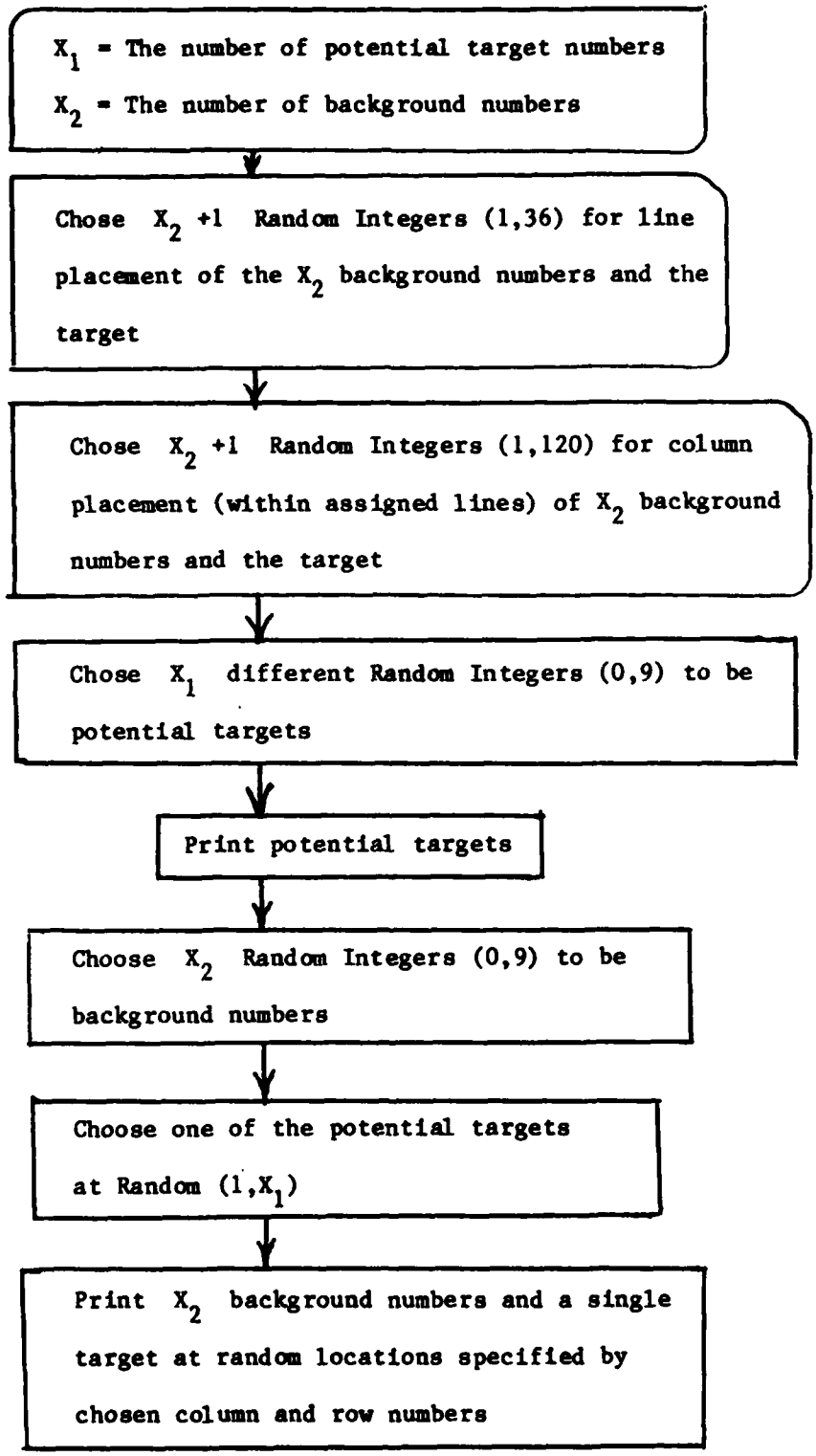
```
00113     32*          Q 2089367924 4,23627546845,23468987177,29476161617,26666237334,     00002100    000000
00113     33*          P  3472118440,10963653676,14433404634,  518915885,12635407907,     00002200    000000
00113     34*          Q 11244663113,17297788094,25153870389, 5144201994,10310609825,     00002300    000000
00113     35*          R 13823834309,21578907839, 1350079164,31293115939,15063593696,     00002400    000000
00113     36*          S 15755048953,28707348220,27273213702,20062610167/                 00002500    000000
00115     37*           EQUIVALENCE (AA,A),(BB,B),(MCOMPJ,LCOMPJ),(MCOMPK,LCOMPK)         00002600    000000
00116     38*           N=(2**(INTSIZ-1)-1)*2+1                                           00002700    000000
00117     39*           J=J+1                                                             00002800    000012
00120     40*           IF(J.GT.P) J=1                                                    00002900    000015
00122     41*           K=J+Q                                                             00003000    000023
00123     42*           IF(K.GT.P)K=K-P                                                   00003100    000026
00125     43*           MCOMPJ=N-M(J)                                                     00003200    000037
00126     44*           MCOMPK=N-M(K)                                                     00003300    000043
00127     45*           A=M(K)                                                            00003400    000047
00130     46*           B=M(J)                                                            00003500    000051
00131     47*           BB=LCOMPJ.AND.AA.OR.LCOMPK.AND.BB                                 00003600    000053
00132     48*           M(J)=B                                                            00003700    000062
00133     49*           IRAND=M(J)                                                        00003800    000064
00134     50*           RETURN                                                            00003900    000065
00135     51*           END                                                               00004000    000103

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

```
@FOR.IS SKIP.SKIP
FOR E3AB-01/12/82-09:36:20 (.0)

   SUBROUTINE SKIP      ENTRY POINT 000014

   STORAGE USED: CODE(1) 000016; DATA(0) 000006; BLANK COMMON(2) 000000

   EXTERNAL REFERENCES (BLOCK, NAME)

    0003   NWDU$
    0004   NIO2$
    0005   NERR3$

   STORAGE ASSIGNMENT  (BLOCK, TYPE, RELATIVE LOCATION, NAME)

    0000   000000 203F      0000   000002 INJP$


00101      1•           SUBROUTINE SKIP                          000000
00103      2•           WRITE(6,203)                             000000
00105      3•     203   FORMAT(1H1)                              000004
00106      4•           RETURN                                   000004
00107      5•           END                                      000015

      END OF COMPILATION:        NO  DIAGNOSTICS.
```

APPENDIX H

NUMBER SEARCH

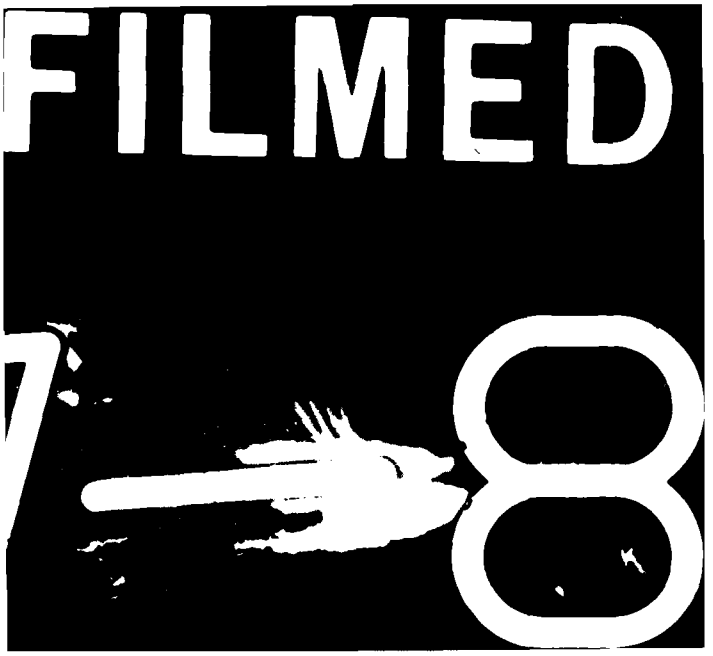
Number Search Test Block Diagram for a Single Item



```
@ELT,SLID   .NUMBSER
ELT BRI S74QIC 01/15/82 16:28:36 (->0)
     1.            00      @FOR,IS MAIN,MAIN
     2.            00            DIMENSION JTARG(5),JITEM(60),ISPOT(60),ILINE(60),FORM(4)
     3.            00            DATA FORM(1),FORM(3),FORM(4)/'(','X,I','')'/
     4.            00      0     INTEGER RANINT
     5.            00            NCOPY=1
     6.            00            NFORM=1
     7.            00            DO 112 IFORM=1,NFORM
     8.            00            DO 113 ICOPY=1,NCOPY
     9.            00            WRITE(6,203)
    10.            00      203   FORMAT(1H1,/,1H1,' NAME:',25X,'SUBJECT NUMBER:',15X,'DAY/DATE/TIME
    11.            00           $:',/,/1X,'NUMBER SEARCH TEST',/,/,5X,'THIS IS A TEST OF HOW QUICKL
    12.            00           *Y YOU CAN FIND TARGETS.',/,1X,'SOMETIMES YOU WILL BE LOOKING FOR A
    13.            00           * SINGLE TARGET NUMBER,',/,1X,'WHICH WILL BE ON THE PAGE WITH OTHER
    14.            00           * NUMBERS.  OTHER TIMES THERE WILL BE SEVERAL TARGET NUMBERS,  ',/,
    15.            00           *1X,'BUT ONLY  ONE OF THEM WILL APPEAR IN THE SEARCH AREA.  WHEN TO
    16.            00           *LD TO BEGIN,',/,1X,'READ THE TARGET(S) ON THE UPPER PAGE, THEN PUT
    17.            00           * A SLASH THROUGH THAT TARGET ON THE LOWER PAGE.'
    18.            00           *                                                ',/,1X,'CONTINUE UNTIL
    19.            00           *TOLD TO STOP.  THEN LOOK UP AT THE CLOCK,NOTE YOUR COMPLETION TIME
    20.            00           *,AND WRITE IT IN THE SPACE PROVIDED.',/,1X' IF YOU HAVE ANY QUESTI
    21.            00           *ONS, PLEASE ASK THEM NOW. DO NOT BEGIN UNTIL TOLD TO DO SO.')
    22.            00            DO 100 ITARG=1,4,3
    23.            00            DO 100 ITEM=10,40,30
    24.            00            ISAMPL =2*SQRT(ITEM*(1+ITARG/3))
    25.            00            DO 109 L=1,ISAMPL
    26.            00            DO 101 I=1,ITARG
    27.            00      102   JTARG(I)=RANINT(9,0)
    28.            00            K=I-1
    29.            00            DO 101 J=1,K
    30.            00            IF(JTARG(I).EQ.JTARG(J).AND.I.NE.1) GO TO  102
    31.            00      101   CONTINUE
    32.            00            DO 103 I=1,ITEM
    33.            00      105   JITEM(I)=RANINT(9,0)
    34.            00            DO 104 J=1,ITARG
    35.            00            IF(JITEM(I).EQ.JTARG(J))            GO TO  105
    36.            00      104   CONTINUE
    37.            00            ISPOT(I)=RANINT(127,1)
    38.            00      108   ILINE(I)=RANINT(55,1)
    39.            00            K=I-1
    40.            00            DO 107 J=1,K
    41.            00            IF(ILINE(J).EQ.ILINE(I).AND.I.NE.1) GO TO 108
    42.            00      107   CONTINUE
    43.            00      103   CONTINUE
    44.            00            KTARG=RANINT(ITARG,1)
    45.            00            KITEM=RANINT(ITEM,1)
    46.            00            JITEM(KITEM)=JTARG(KTARG)
    47.            00            WRITE(6,201) (JTARG(I),I=1,ITARG)
    48.            00      201   FORMAT(1H1,'TARGET(S):',5I2)
    49.            00            WRITE(6,202)
    50.            00      202   FORMAT(1H1)
    51.            00            DO 106 IILINE=1,55
    52.            00            DO 110 I=1,ITEM
    53.            00            ENCODE(3,10,TEMP,IERR)  ISPOT(I)
    54.            00            FORM(2)='    '
    55.            00            FLD(18,18,FORM(2))=FLD(0,18,TEMP)
    56.            00      10    FORMAT(I3)
    57.            00            IF(ILINE(I).EQ.IILINE) GO TO 111
    58.            00      110   CONTINUE
    59.            00            WRITE(6,200)
    60.            00      200   FORMAT(1X)
    61.            00            GO TO 106
```

```
      WRITE(6,FORM) JITEM(I)
  106 CONTINUE
  109 CONTINUE
      WRITE(6,204)
  204 FORMAT('1H1,5(5X,'STOP CHECK TIME'),/,1X,'COMPLETION TIME:',/,1X,
     *'DO NOT TURN PAGE UNTIL TOLD TO DO SO.',/,1H1)
  100 CONTINUE
  113 CONTINUE
  112 CONTINUE
      STOP
      END
*FOR,IS RANINT,RANINT
      INTEGER FUNCTION RANINT(I,M)
      IF(I.EQ.1)  GO TO 101
  100 J=IRAND(X)
      L=MOD(34359738367,I)
      K=34359738367-L
      IF(J.GT.K) GO TO 100
      RANINT=MOD(J,I)
      IF(RANINT.EQ.0.AND.M.EQ.1) GO TO 100
      GO TO 102
  101 RANINT=1
  102 RETURN
      END
*FOR,IS IRAND,IRAND
C FUNCTION IRAND.....................................................
C
C OCTOBER 17, 1977
C
C GENERATE PSEUDO-RANDOM POSITIVE INTEGER WITH RECTANGULAR DISTRIBUTION
C IN FULL RANGE OF POSITIVE INTEGERS.
C THE ONE ARGUMENT IS DUMMY, AND HAS NO EFFECT.
C
C.....................................................................
C
C
      FUNCTION IRAND(X)
      LOGICAL AA,BB,LCOMPJ,LCOMPK                                          00000200
      INTEGER A,B,P,Q,INTSIZ,M(98)                                         00000300
      DATA J/0/,P/98/,Q/27/                                                00000400
      DATA INTSIZ/35/                                                      00000500
      DATA M/ 5540107616, 9465596368,31090102751,18771747577,              00000600
     A 28429589985,17910665383, 2755776712,15772108374,24868503636,       00000700
     B 24050196658,10406361912,25894330488,10239025078,23172152762,       00000800
     C 16749650049,10596720590,16558947567, 4477013940, 5230881612,       00000900
     D  5872042252,10375682927,30555741521, 9897021507,13528500273,       00001000
     E  4676199511, 4790613885,34237141705,16092853401, 6242230728,       00001100
     F 18576454781,32549771790, 5345127795, 9050133044, 1996740055,       00001200
     G 33666709376,32156029504,31994998514, 1331560448,25461263460,       00001300
     H 10347229411,30965797332,12720210785,10885710696,28347389480,       00001400
     I 21198968033,29917444937, 5707930159,16990730598, 9839223853,       00001500
     J  4190023379,11263804815,12759423169, 2851834982,26267332518,       00001600
     K 24635128900,21358828836,15048763990,31322345053,30055861835,       00001700
     L 32542479077,27377696085,34215017208,10363747822,24895847765,       00001800
     M 19361735837,28691085376,29117273251,12029489027, 5529942408,       00001900
     N  9207135596,26254700186,22067063391,25385314502, 8687457660,       00002000
     O 20993679244,23627546845,23468987177,29476161617,26666237334,       00002100
     P  3472118440,10963653676,14433404634,  518915885,12635407907,       00002200
     Q 11244663113,17297788094,25153870389, 5144201994,10310609825,       00002300
     R 13823834309,21578907839, 1350079164,31293115939,15063593696,       00002400
     S 15755048953,28707348220,27273213702,20062610167/                   00002500
      EQUIVALENCE (AA,A),(BB,B),(MCOMPJ,LCOMPJ),(MCOMPK,LCOMPK)            00002600
      N=(2**(INTSIZ-1))*2+1                                                00002700
```

```
  125.          00            J=J+1                                                    00002800
  126.          00            IF(J.GT.P) J=1                                           00002900
  127.          00            K=J+Q                                                    00003000
  128.          00            IF(K.GT.P)K=K-P                                          00003100
  129.          00            MCOMPJ=N-M(J)                                            00003200
  130.          00            MCOMPK=N-M(K)                                            00003300
  131.          00            A=M(K)                                                   00003400
  132.          00            B=M(J)                                                   00003500
  133.          00            BB=LCOMPJ.A. .AA.OR.LCOMPK.AND.BB                        00003600
  134.          00            M(J)=B                                                   00003700
  135.          00            IRAND=M(J)                                               00003800
  136.          00            RETURN                                                   00003900
  137.          00            END
  138.          00       *MAP,IS  THIS,MP0060/NUMSER
  139.          00        LIB SCC$*RLIB.

END ELT.  ERRORS: NONE. TIME:  0.655 SEC. IMAGE COUNT: 139

*BRKPT  PRINT$/NS
```


FILMED
7
8