UNHA 1347

Bell Laboratories

Cover Sheet for Technical Memorandum

The information contained herein is for the use of employees of Bell Laboratories and is not for publication (see GEI 13.9-3)

Title- Evaluation of Microprocessors Supporting the C Language: LSI-11, MAC-8, Z80*

Date- June 5, 1978

TM- 78-3323-3

Other Keywords-

Author(s)	Location and Room	Extension
M. H. Bradley	HO 1G431	3689
W. E. Sherbon	HO 1G433	6876
J. S. Waldhuter	HO 1G425	4521

Charging Case- 38763-063

Filing Case- 40033-010

ABSTRACT

A study was done comparing three microprocessors which support the C language: LSI-11, MAC-8, and Z80. The purpose of the study was to evaluate the performance of each of the processors for our exploratory application; ie, a word processing terminal with a UNIX**-like operating system. Four benchmark programs written in C language were used to evaluate the processors. Each program was run on all of the processors and the execution times were compared. Also, the C compiler for each machine was evaluated by comparing the amount of code generated for each program. The compiled code was loaded and executed on each target machine. No simulation was involved. For external calibration points, we also ran the benchmark programs on a dedicated DEC*** PDP 11/40 under a single user version of UNIX, and on the DEC PDP 11/70 under the UNIX time-sharing system. The compilers for the processors were further evaluated by compiling several UNIX user-level commands (cat, cp, ls, etc.) for each of the processors and comparing the amount of code generated for each machine. The results show that the MAC-8 does an admirable job for programs typical of our application; that is, requiring data manipulation with a minimum of numerical calculations. The results are clearly tabulated.

---

*Z80 is a Trademark of Zilog Inc.
**UNIX is a Trademark of Bell Laboratories.
***DEC Digital Equipment Corporation.

Pages Text 5 Other 13 Total 18

No. Figures 0 No. Tables 4 No. Refs. 12

Address Label

DISTRIBUTION
(BEPER GEI 13.9-3)

COMPLETE MEMORANDUM TO

CORRESPONDENCE FILES

OFFICIAL FILE COPY
PLUS ONE COPY FOR
EACH ADDITIONAL FILING
CASE REFERENCED

DATE FILE COPY
(FORM E-1328)

TO REFERENCE COPIES

BLAHUT,D E
-BRADLEY,M HELEN
-BRAINARD,R C
>BREEN,ROBERT N
+CABLE,W B
<COURTNEY-PRATT,J S
+ENDOE,L H
>FARRELL,JAMES W
-KERNIGHAN,BRIAN W
>KIRBY,DOROTHY B
+MOLINELLI,JOHN J
+NEWELL,JOHN A
+OETEL,WILLIAM C G
+RABINOWITZ,P
>RITCHIE,D M
>ROVEGNO,HELEN D
+SHERMAN,W E
+THOMAS,L C
>VOSS,GERALD C
+WALLHUTER,JEFFREY S
20 NAMES

COVER SHEET ONLY TO

CORRESPONDENCE FILES

3 COPIES PLUS ONE
COPY FOR EACH FILING
CASE

AAGESEN,JOHN
+AARON,M R
ABBRUSCATO,C R
ACQUAVIVA,V
-ADAMS,ROBERT L,3RD
AHO,ALFRED V
ALDENS,RAINER E
-AITKEN,GEORGE S
-ALAIMO,CHARLIE
ALCALAY,D
ALDERMESHIAN,HRAIR
ALLEN,R H
+ALLES,DAVID S
+ALQUIST,R P
-ALONSO,RAFAEL
-ALSBERG,DIETRICH A
<ALTERMAN,MICHAEL E
AMRON,IRVING

COVER SHEET ONLY TO

ANDERSON,FREDERICK L
ANDERSON,J C
+ANDERSON,KATHRYN J
ANDERSON,MILTON M
+ANDERSON,PETER H
+ANDERSON,T C
+ANDRADE,RAMON J
+ANGELO,E J,JR
ANGLIKOWSKI,R E
+ANTOLICK,DAVID B
+ARANGUREN,W L
+ARDIS,R B
+ARMSTRONG,D B
ARNOLD,S L
+ARNOLD,THOMAS F
+ASHBY,D
+ASHCRAFT,STEVEN E
+ASHENFELTER,ROBERT C
+ASHLEY,F B
ASTHANA,ABHAYA
ASTLE,H W
ATAL,BISHNU S
+AULL,DENIS W
AULT,C F
+AWIPI,M
BAGGA,YUDHVEER S
BAKER,W E
+BALABAN,PHILIP
+BALDINI,J J
+BALDWIN,GARY L
BALDWIN,GEORGE L
BALL,MARSHALL
+BALZER,G C
+BARNA,G J
+BARNETT,W T
BARNS,R L
BARON,ROBERT V
BARRALL,B R
+BARTHOLOMEW,W F,3RD
+BASFIELD,RICHARD J
+BASIAN,J P
+BATEMAN,R B
+BAUER,H C
+BAUER,STEPHEN M
+BAUGH,C R
BAUMAN,LEE M
+BAUMWOLSPINER,M
BAXTER,LESLIE A
+BAYER,D L
+BEAUCHAMP,HARRY L
BECKER,RICHARD A
+BECKETT,J T
BECKNER,MARK W
+BECK,JOHN H
+BECK,ROBERT J
+BELCHER,G G A
BELL,STEVE W
+BENOWITZ,P
+BERGERON,R F,JR
BERGH,A A
BERGLAND,G D
+BERNARDINI,J

COVER SHEET ONLY TO

BERNSTEIN,L
BERRYMAN,R D
+BERTRAM,WALTER J,JR
BETZEL,HENRY J
+BEUSCHER,HUGH J
+BHATIA,RAJIV
+BIDLACK,R H
BILOWOS,R M
+BIORN,PER
+BIREN,IRMA B
BITTNER,BARBARA B
+BLAHA,A L
BLANCHARD,JOETTE D
BLAZIER,S D
+BLENDINGER,L H
+BLINN,J C
BLOSSER,PATRICK A
+BLUM,MARION
+BOBILIN,R T
+BOCOCK,JON M
BODDIE,JAMES R
BODEN,F J
BOIVIE,RICHARD H
BONANNI,L E
+BOND,HOLTON C,JR
+BONOMI,M J
BORG,KEVIN E
+BOROS,VICTOR B
+BORUM,J C
+BOSCO,HARRY L
BOSE,CHINMOY B
BOSE,DEBASISH
+BOSTON,RONALD E
+BOTSFORD,N,JR
+BOURNE,STEPHEN R
+BOWMAN,H C
BOYCE,C D
BOYCE,W M
+BOYD,GARY D
+BOYER,WAYNE F
+BRADFORD,C E
+BRADLEY,J F
+BRAHM,D
+BRANTLEY,WILLIAM C
BREEDEN,ROBERT LOUIS
+BREEN,R S,JR
+BREILAND,JOHN E
+BRILEY,B E
+BROLIN,S J
+BROOKS,R D
+BROPHY,F J
+BROSSONI,GUIDO J
BROSS,JEFFREY D
+BROWNING,P S,JR
+BROWN,DAVID B
BROWN,ELLINGTON L
+BROWN,J L
+BROWN,JAMES M
BROWN,ROBIN L
+BRUNING,J H
+BUCHSBAUM,S J
+BUCK,I D

COVER SHEET ONLY TO

+BURGESS,JOHN T,JR
+BURGIEL,JOSEPH C
BURNETTE,W A
+BURNETT,E
+BURNS,HARRY S
+BURTNESS,R D
BUSALA,A
+BUTFILOSKI,H R
+BUTLER,E M
BUTLER,THOMAS T
+BUZZARD,C ALAN
+BYE,PETER L
+BYRNE,EDWARD R
+CALLAWAY,W B
+CAMLET,J V,JR
CAMPBELL,JERRY H
+CAMPBELL,MICHAEL R
+CANADAY,BUDD H
+CANDELIERI,ANTHONY V
+CANDREA,RONALD D
+CANTWELL,RICHARD F
+CARBONE,J T
+CARDWELL,ROBERT E
CAREY,J E
+CAREY,ROBERT E
+CARLSON,H N
+CARON,C S
+CARRIERE,JOSEPH F
+CATALANO,J F
CAVINESS,JOHN D
CELLI,JOHN R
+CERMAK,I A
CHAI,D T
CHAI,SOO YOUNG
CHAMBERS,B C
CHAMBERS,J M
+CHAMBERS,R P
+CHANDLER,N H
+CHANDLER,RICHARD A
CHANDRA,SATISH
CHANG,G K
+CHAPLIN,N J
+CHAPPELL,S G
CHAYUT,IRA G
+CHENG-QUISPE,ENRIQUE
CHEN,ROBERT
+CHEN,YU-SSU
CHERRY,LORINDA L
CHIN,J S
CHI,SOU-CHUNG
CHODROW,M M
CHOU,PAO-LO P
+CHOW,MING-CHWAN
+CHOW,W F
+CHRISTMAS,W P
CHRIST,C W,JR
CIARDI,JOHN J
+CIRILLO,A J
+CLARKE,P W
+CLAYTON,D P
+CLINE,W K
+COBEN,ROBERT M

COVER SHEET ONLY TO

+COCHRAN,W T
COHEN,DAVID
+COLBRY,BRIAN W
<COLE,LOUIS M
COLLICOTT,R B
+COLSON,J S,JR
+COLTON,JOHN R
+COMBS,D E
+CONFALONE,D E
+CONNERS,RONALD R
+COOK,RICHARD F
COOMER,STEPHEN D
+COOPER,J A,JR
+COPELAND,JOHN A
+COPP,DAVID H
+CORDELL,R R
+CORNELL,B G
+CORWIN,W
+COSTELLO,JOHN J
+COSTELLO,PETER E
+CRAGUN,DONALD W
+CRANDALL,J H
CRISTOFOR,EUGENE
+CRONIN,P M
+CROSWELL,JEROME O
+CROWE,MARGARET M
+CRUME,L L
+CSENCSITS,BRENDA M
+CUMMISKEY,P
+DAIL,JAMES E
+DANGELO,R S
DANIELSEN,O D
+DANIELS,L A
+DARNALL,BRUCE R
+DAUGHERTY,T H
+DAVIDSON,CHARLES LEWIS
+DAVIDSON,ROBERT P
+DAVIS,BERNARD P
DAVIS,R DREW
DE COSTA,J
DE FAZIO,M J
+DE GRAAF,D A
DEEM,G S
+DEGAN,JOHN J,JR
DELAPENA,ERIC J
+DELLECKER,RAYMOND W
+DEMERS,R A
+DENNIS,T M
DENNY,MICHAEL S
+DENSMORE,WAYNE
DEUTSCH,J A
DI PIETRO,R S
+DI SABATO,JOSEPH A
+DIETZEL,K ROBERT
+DIMARCO,ROBERT T
+DIMMICK,JAMES O
+DINN,NEIL F
+DISHMAN,JOHN M
+DITTY,L E
DIXON,DAVID A
DOLOTTA,T A
DONDERO,J J
...
1240 TOTAL

+ NAMED BY AUTHOR > CITED AS REFERENCE < REQUESTED BY READER (NAMES WITHOUT PREFIX WERE SELECTED USING THE AUTHOR'S SUBJECT OR ORGANIZATIONAL SPECIFICATION AS GIVEN BELOW)

MERCURY SPECIFICATION.....

COMPLETE MEMO TO:

COVER SHEET TO:
332

COHAMP = COMPUTER MICROPROCESSOR HARDWARE

HO CORRESPONDENCE FILES
HO 5C101

TM-78-3323-3
TOTAL PAGES 16

TO GET A COMPLETE COPY:

1. BE SURE YOUR CORRECT ADDRESS IS GIVEN ON THE OTHER SIDE.
2. FOLD THIS SHEET IN HALF WITH THIS SIDE OUT AND STAPLE.
3. CIRCLE THE ADDRESS AT RIGHT. USE NO ENVELOPE.
4. INDICATE WHETHER MICROFICHE OR PAPER IS DESIRED.

PLEASE SEND A COMPLETE

( ) MICROFICHE COPY ( ) PAPER COPY

TO THE ADDRESS SHOWN ON THE OTHER SIDE.

Bell Laboratories

subject: Evaluation of Microprocessors Supporting the C Language: LSI-11, MAC-8, Z80*
CC 38763-063, FC 40033-010

date: June 5, 1978

from: M. H. Bradley
W. E. Sherbon
J. S. Waldhuter

TM: 78-3323-3

MEMORANDUM FOR FILE

INTRODUCTION

Department 3323 is presently exploring the hardware and software requirements for intelligent data terminals. An important decision in the design of an intelligent terminal is the selection of a microprocessor. We are presently considering terminals with a local CPU which will run a general purpose operating system (similar to UNIX**), and several application programs for data entry and word processing. For this size software effort, it is necessary to make use of a high-level language such as C (reference 1). The purpose of this memorandum is to present a quantitative evaluation of three existing microprocessors which have an available C compiler. However, the reader is cautioned against making any global conclusions on relative processor performance based on this evaluation. We have focused on performance parameters relating specifically to our needs and have made no attempt to characterize these processors across a broad range of possible applications.

There are presently three microprocessors available which support the C language: DEC*** LSI-11 (reference 2), Bell Laboratories MAC-8 (references 3 and 4), and Zilog Z80 (reference 5). The MAC-8 and Z80 are 8 bit processors, whereas the LSI-11 is a 16 bit processor. In our evaluation, we were concerned with the speed of the processor and the amount of code generated by the C compiler. In order to compare speeds, a number of benchmark programs, similar to those we expect to be in final terminal software, were executed. The benchmark programs for the individual machines were written in C language and compiled for each target machine. Programs were then loaded and

---

*Z80 is a Trademark of Zilog Inc.
**UNIX is a Trademark of Bell Laboratories.
***DEC Digital Equipment Corporation.

executed on each machine, no simulation was involved. For external calibration points, we also ran the benchmark programs on a dedicated PDP 11/40 under a single user version of UNIX, and on the PDP 11/70 under the UNIX time-sharing system. In addition, we looked at the code generated for larger programs by compiling several UNIX user-level commands (cat, cp, ls, cal, etc.) for all of the machines above.

DESCRIPTION OF PROGRAMS

Four benchmark programs were run on each of the target machines. The programs were designed to be totally CPU bound (ie, the execution time was not dependent on non-CPU limitations such as disk access time). The complexity of the tests varied with No. 1 being a simple task and No. 4 the most complex. The following is a general description of these programs.

No. 1, Accumulator Increment
The code being timed consisted of two "for" statements. Several loop operations were performed by continuously incrementing and testing the accumulator. This gave an indication of the speed of the processor doing the simplest possible task.

No. 2, Character Exchange
Typical programs that we write for our applications will involve the manipulation of text in memory. We were therefore interested in the processor's ability to move bytes of memory. This program exchanged the data in two bytes of memory several times within a loop.

No. 3, Array Exchange
Programs that manipulate text are not always concerned with moving single bytes in memory. It is common to move large blocks of data such as arrays. Data in two 512-character arrays were swapped a number of times. This exercised the processor's ability to manipulate characters, as well as the pointers to characters.

No. 4, Seven Factorial
The final test was designed to evaluate the processor's ability to perform numerical computations. Seven factorial was computed repeatedly.

For a more detailed description of these benchmark programs refer to the C source code in Appendix A and the flow charts

for each program in Appendix B.

PROCEDURES AND CONDITIONS OF TESTS

Benchmark programs for the MAC-8 and Z80 were compiled on the UNIX time-sharing system. The MAC-8 C compiler was written at Bell Laboratories by H. D. Rovegno and is available on PWB/UNIX (reference 6). The Z80 C compiler which is also available on PWB/UNIX, was written by Interative Systems Corporation (ISC) for Teletype Corporation. After compilation, the object code for the MAC-8 and Z80 was downloaded into the target machine for execution. The LSI-11 supports a single-user version of UNIX, LSX (references 7 and 8). The benchmark programs for the LSI-11 were compiled under LSX using the C compiler for the PDP-11 family of processors. Object code was stored on the system for later execution. The benchmark programs were set up such that execution time could be measured in order of seconds. Timing of program execution began with the display of an "S" on the CRT screen and ended when the character "F" appeared. The time for each program was recorded with a stopwatch. Since calculation of processor speed depended on a portion of the program, we only recorded the size of code generated for that portion. Hence, the number of bytes generated for subroutines such as "printf" and "csv" are not included in the data.

The MAC-8 operating at 1 MHz, which is half the normal speed, was run on the Intel MDS*. The particular system we were using also required additional wait states on memory read and write to handle idiosyncrasies of the prototype MAC-8 chip available to us. On the average, 40 percent of instruction cycles were wait states. The results were adjusted to indicate the performance of a production-version MAC-8 (2 MHz).

There are two versions of the Z80 C compiler, which we will refer to as "ZILOG Z80" and "ZILOG Z80X". The "ZILOG Z80" C compiler is the most efficient in terms of program object code generated, but results in relatively slow execution times. The other version of the C compiler, "ZILOG Z80X", generates more code but the execution time is faster. The Z80 user has a choice of C compilers depending upon his needs. He can either generate less code or decrease execution time. The benchmark programs for the Z80 were run on a Cromemco microcomputer system at 2 MHz. Note, the Cromemco system is capable of operating at 4 MHZ, which would cut the execution times for the Z80 in half.

---

*MDS Microprocessor Development System.

The LSI-11 was capable of performing computational tasks faster than the MAC-8 and Z80. This is accomplished by the extended instruction set (EIS) chip that handles multiply and divide operations.

RESULTS

The results of these benchmark programs are shown in Tables 1 and 2. Table 1 indicates the execution time for the four benchmark programs running on each processor. The size of code generated for these programs is shown in Table 2. The tables also include the results for the PDP 11/40 and UNIX time-sharing system, to show the relationship between microprocessors and larger machines.

For non-computational tasks (benchmark programs 1-3), the MAC-8 appears to perform better than the other microprocessors, but only slightly better than the LSI-11. The LSI-11 shows its strength in execution of benchmark No. 4. This processor definitely has a hardware advantage over both the MAC-8 and Z80 for operations involving multiplication as well as division. The Z80 is the slowest processor for all operations and the compiler for the Z80 generates the most code. Some of the degradation in speed can be attributed to the additional instructions that the Z80 must handle per operation. The architecture of the Z80 is not well suited for supporting a high-level language like C; ie, the registers cannot be used as general purpose registers. This results in considerable swapping between memory and registers to handle C language statements. The MAC-8 is very efficient in code generation since the processor was designed specifically to run C programs.

In addition to looking at the code generated for the benchmark programs, it was possible to obtain more data regarding object code generation by looking at larger C programs. Source code for several UNIX user-level commands was compiled for the three target machines. Table 3 gives the results of this compilation. Using the LSI-11 as a reference, Table 4 shows the ratio between the size of the LSI-11 code and the size of the code for the MAC-8 and Z80. Code generated for the MAC-8 is close in size to the code for PDP-11 machines. The Z80 compiler performs better on larger programs than it does on small ones. Subroutines are generated by the Z80 compiler to handle many of the C constructs. This additional overhead becomes insignificant as the size of the program and the number of times these constructs are used increases.

CONCLUSION

If it is necessary to use a high level language such as C on a microprocessor, one should evaluate not only the speed of the machine but also its internal architecture. The Z80 running in this environment suffers a performance penalty because of its architecture. In our benchmark programs, the Z80 must run at 4 MHz to perform comparable to the MAC-8 at 2 MHz. It was expected that a 16 bit processor would perform better than an 8 bit machine. However, the MAC-8 appears to be comparable to the LSI-11 for applications with minimal numerical calculations. In the future, we plan to evaluate other microprocessors that support the C language. Evaluation of the new 16 bit machines, such as the Intel 8086* and Zilog Z8000**, will be possible when the necessary hardware and compilers are available.

ACKNOWLEDGEMENT

The authors would like to acknowledge A. Reszka of Teletype Corporation for his assistance with the Z80 C compilers. We would also like to thank H. B. Greene of Bell Laboratories for providing us with information about the operation of the MAC-8 in the Intel MDS.

M. H. Bradley

W. E. Sherbon

J. S. Waldhuter

MHB
HO-3323-WES-NROFF
JSW

Att.
References 12
Tables 4
Appendixes 2

---

*8086 is a Trademark of Intel Inc.
**Z8000 is a Trademark of Zilog Inc.

REFERENCES

1. "The C Programming Language", Brian W. Kernighan, Dennis M. Ritchie, 1978.

2. "LSI-11, PDP 11/03 Processor Handbook", Digital Equipment Corporation, 1975.

3. S. T. Campbell, "MAC-8 Microprocessor Summary", CC 38565-2, March 8, 1976.

4. D. E. Blahut, R. C. Brainard, "MAC-8 Microprocessor Hardware Manual", TM 76-4391-3, TM76-1353-7, June 17, 1976.

5. "Z80-CPU, Z80A-CPU Technical Manual", Zilog Corporation, 1977.

6. H. D. Rovegno, "MAC-8 C for Beginners", CC 49243, FC 39898-14, April 1, 1977.

7. H. Lycklama, "LSI-UNIX System", TM76-1352-4, October 6, 1976.

8. M. H. Bradley and R. N. Breen, "An Intelligent Terminal Using the LSI-11 and UNIX", CC 39065-47, FC 40033-12, April 15, 1977.

9. D. B. Kirby, "Assembler User's Manual", Tm 77-2265-2. May 20, 1977.

10. G. C. Vogel, "8080/Z80 Assembler Reference Manual", CC 28655-100, FC 40952-001, March 21, 1978.

11. J. W. Farrell, "Users's Guide to the SMAL2 Language for the Zilog Z80 Microprocessor", CC 28655-100 FC 40952-001, March 21, 1978.

12. "Z80-Assembly Language Programming Manual". Zilog Corporation, 1977.

TABLE 1

SPEED COMPARISON

BENCHMARK PROGRAM	DEC LSI-11	BTL MAC-8 (2MHz)	ZILOG Z80 (2MHz)	ZILOG Z80X (2MHz)	DEC PDP 11/40	UNIX TIMESHARE
PROGRAM NO. 1, ACCUMULATOR INCREMENT	34	25	148	97	10	~3
PROGRAM NO. 2, CHARACTER EXCHANGE	89	56	190	140	27	~10
PROGRAM NO. 3, 512 CHARACTER ARRAY EXCHANGE	79	78	191	166	24	~9
PROGRAM NO. 4, SEVEN FACTORIAL (7!)	86	576	768	640	25	~12

(SECONDS)

TABLE 2

CODE GENERATION COMPARISON

BENCHMARK PROGRAM	DEC LSI-11	BTL MAC-8	ZILOG Z80	ZILOG Z80X	DEC PDP 11/40	UNIX TIMESHARE
PROGRAM NO. 1, ACCUMULATOR INCREMENT	38	36	96	104	38	38
PROGRAM NO. 2, CHARACTER EXCHANGE	50	48	108	116	50	50
PROGRAM NO. 3, 512 CHARACTER ARRAY EXCHANGE	60	79	186	198	60	60
PROGRAM NO. 4, SEVEN FACTORIAL (7!)	66	145	172	192	66	66

(BYTES)

TABLE 3

CODE GENERATION (BYTES) OF UNIX

USER LEVEL COMMANDS

PROGRAM	DEC LSI-11	BTL MAC-8	ZILOG Z80	ZILOG Z80X
CAT	540	636	871	975
CP	1072	1129	1274	1402
LS	2526	2865	3599	4421
CAL	1194	1414	1710	2006
RM	702	769	846	998
NICE	220	230	290	322
OD	3616	3745	4421	4829
PR	2162	2606	3224	3604
WHO	450	497	610	730
SPLIT	570	681	821	885

TABLE 4

CODE GENERATION COMPARISON

OF UNIX USER LEVEL COMMANDS

(DEC LSI-11 USED AS REFERENCE)

PROGRAM	BTL MAC-8	ZILOG Z80	ZILOG Z80X
CAT	1.18	1.61	1.81
CP	1.05	1.19	1.31
LS	1.13	1.42	1.75
CAL	1.18	1.43	1.68
RM	1.10	1.21	1.42
NICE	1.05	1.32	1.46
OD	1.04	1.22	1.34
PR	1.21	1.49	1.67
WHO	1.10	1.36	1.62
SPLIT	1.19	1.44	1.55
AVERAGE	1.12	1.37	1.57
MINIMUM	1.04	1.19	1.31
MAXIMUM	1.21	1.61	1.81

APPENDIX A

```
/*
 * No.1, Accumulator Increment
 * An integer variable is incremented 32000 times
 * within a loop of 50
 */

extern printf();


main()
{
register int loop;
register int count;

        printf("S");
        for(count = 0; count < 50; count++)
                for(loop = 0; loop < 32000; loop++);
        printf("F");
}
```

```
/*
 * No. 2, Character Exchange
 * Two bytes of memory are exchanged 32000 times
 * within a loop of 50
 */


extern printf();


char loc1;
char loc2;

main()
{
register int loop;
register int count;


        printf("S");
        for(count = 0; count < 50; count++)
                {for(loop = 0; loop < 32000; loop++)
                        {loc1 = loc2;
                         loc2 = loc1;
                        }
                }
        printf("F");
}
```

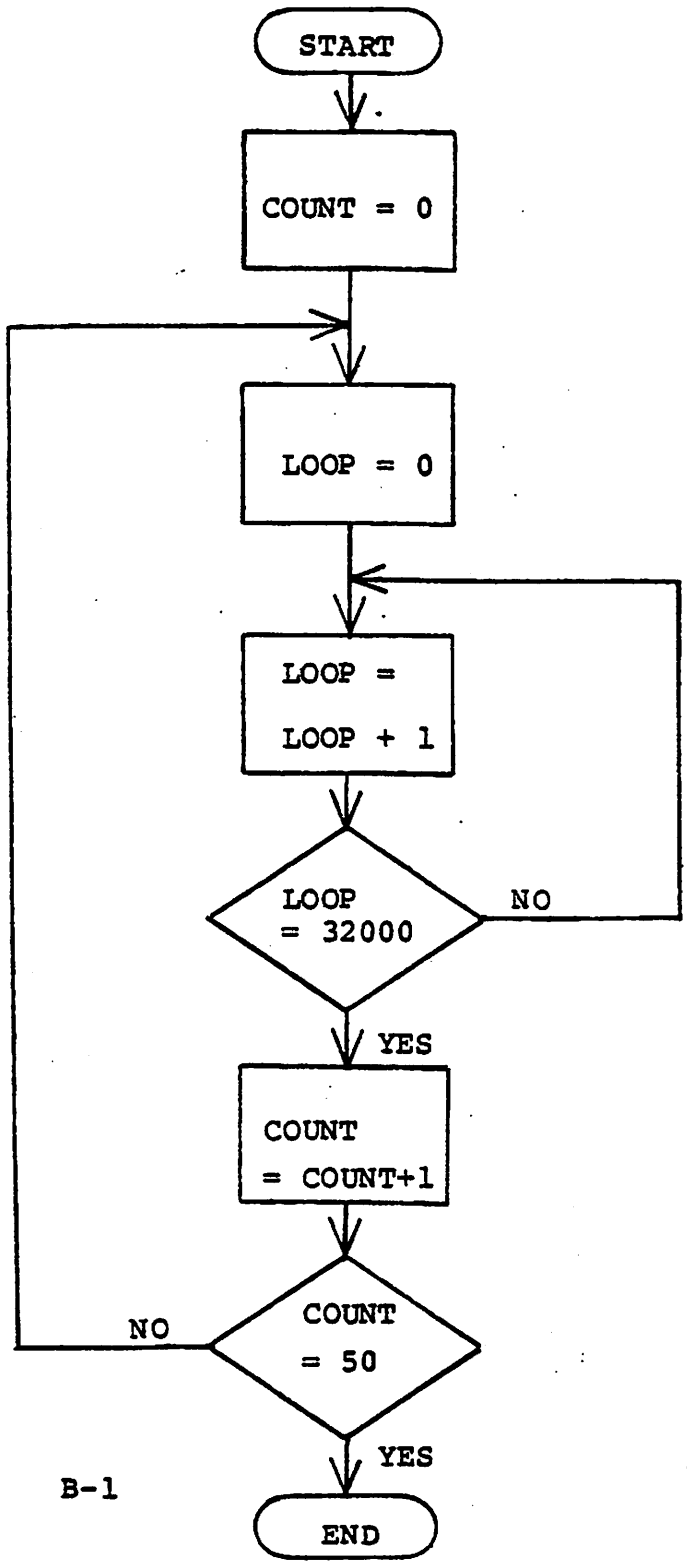
```
/*
 * No. 3, Array Exchange
 * Data in two arrays are swapped 320 times
 * within a loop of 5
 */


extern printf();


char a[512];
char b[512];

main()
{
register int loop;
register int count;
register int i;

        printf("S");
        for(count = 0; count < 5; count++)
                {for(loop = 0; loop < 320; loop++)
                        {for(i = 0; i < 512; i++)
                                {a[i] = b[i];
                                 b[i] = a[i];
                                }
                        }
                }
        printf("F");
}
```

```
/*
 * No. 4, Seven Factorial
 * Seven factorial is computed 32000 times
 * within a loop of 5
 */


extern printf();



main()
{
register int i;
register int j;
register int loop;
register int count;


        printf("S");
        for(count = 0; count < 5; count++)
                {for(loop = 0; loop < 32000; loop++)
                        {j = 1;
                         for(i = 1; i <= 7; i++) j = i*j;
                        }
                }
        printf("F");
}
```

APPENDIX B

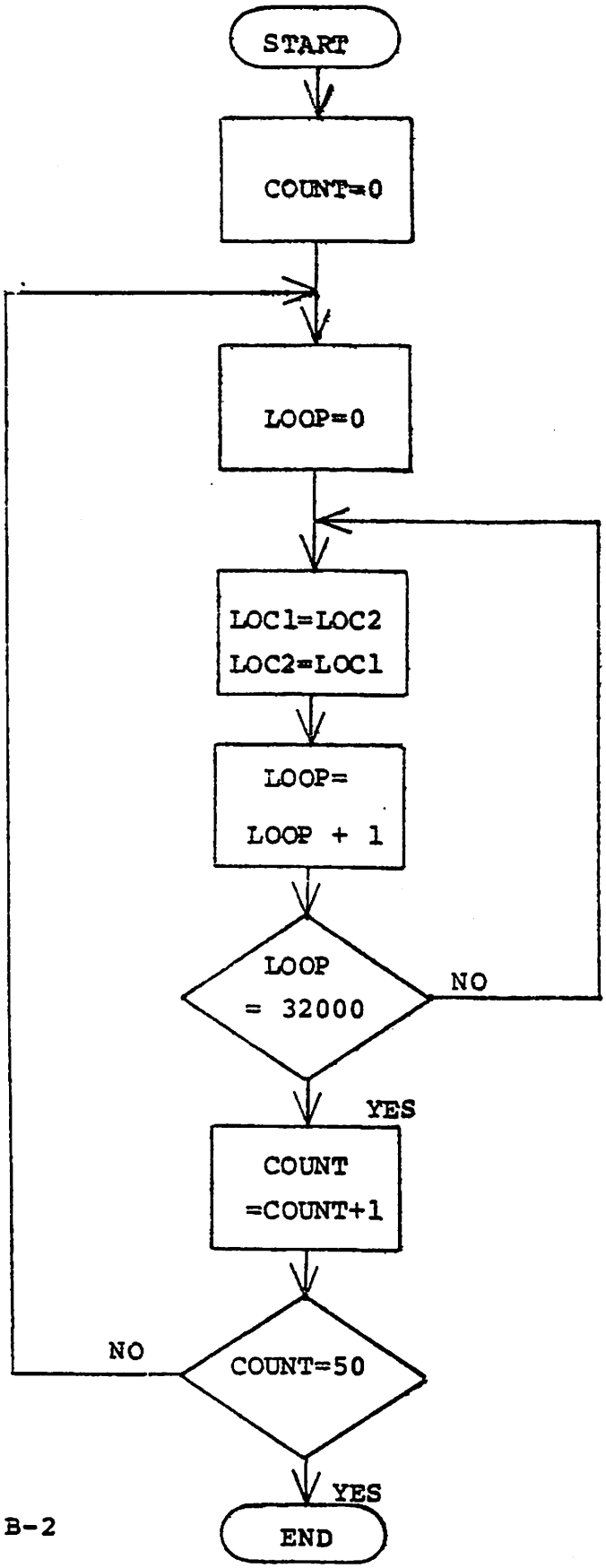
PROGRAM NO. 1, ACCUMULATOR INCREMENT

FLOW CHART

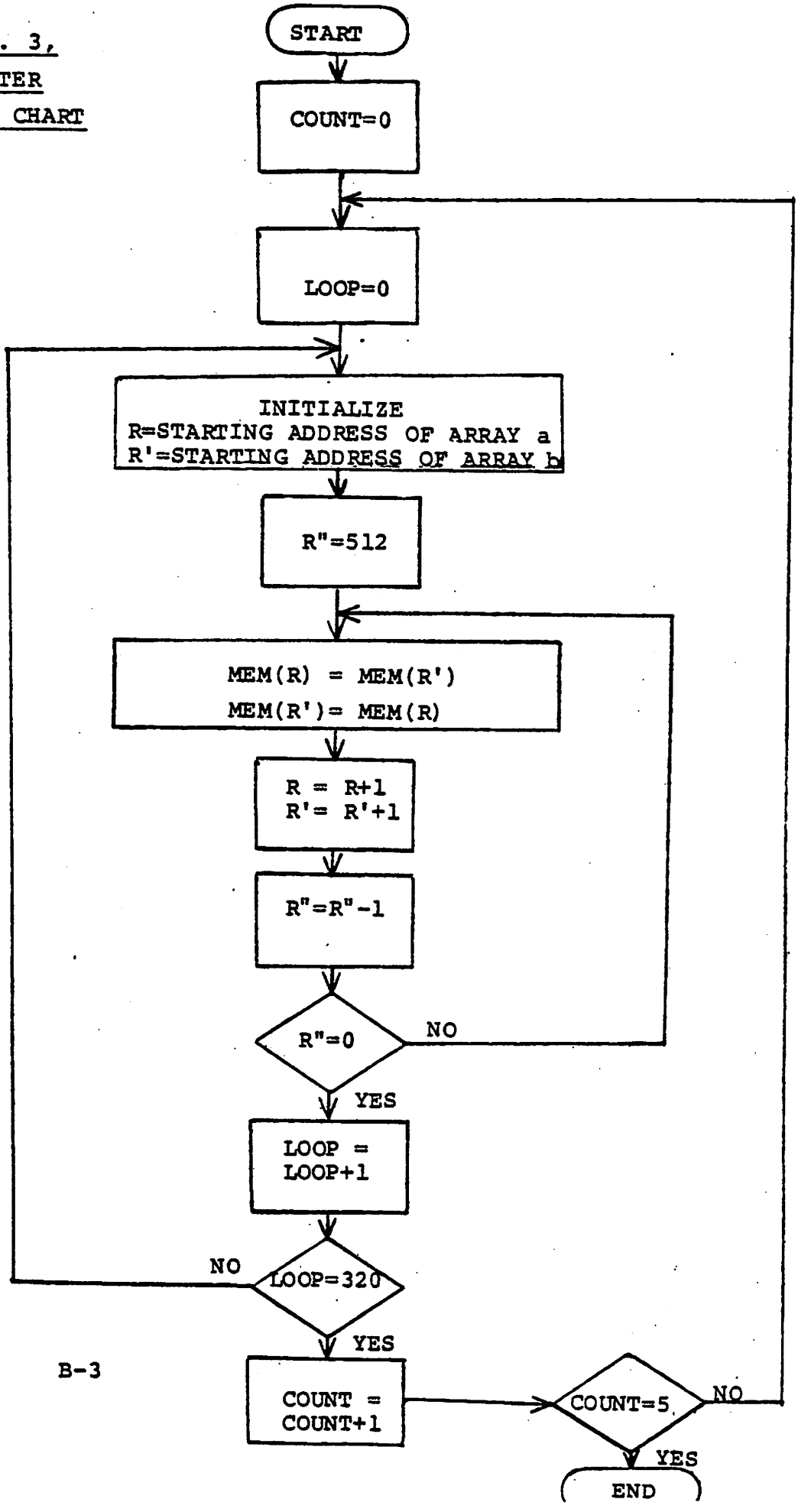


PROGRAM NO. 2, SINGLE MEMORY

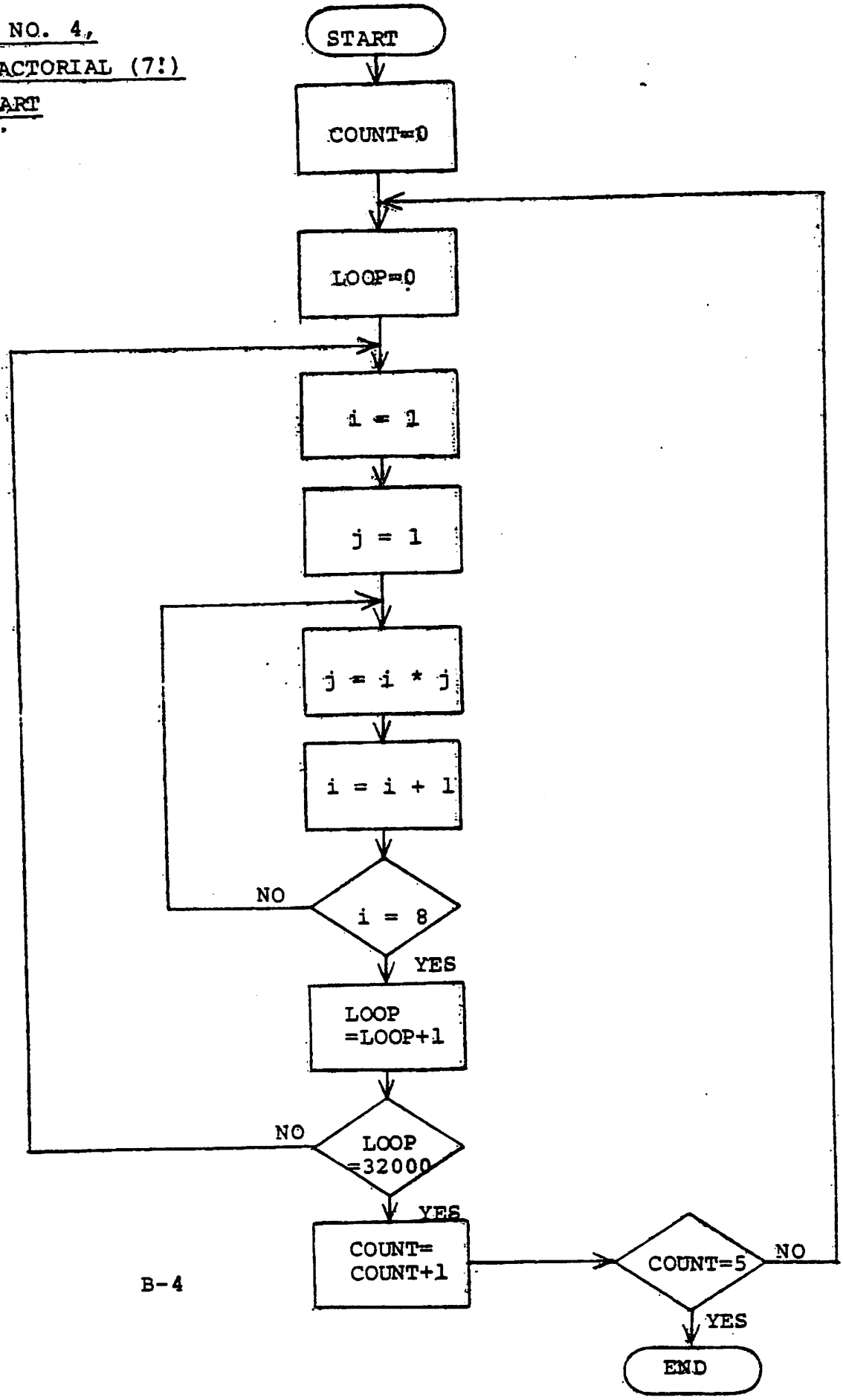
EXCHANGE FLOW CHART



PROGRAM NO. 3,
512 CHARACTER
ARRAY FLOW CHART



PROGRAM NO. 4,

SEVEN FACTORIAL (7!)

FLOW CHART