6400

6600

6400

SYSTEMS BULLETIN 2

6600

6400

CONTROL DATA
CORPORATION

6600

6400/6600

SYSTEMS BULLETIN

2

ERRATA

Page 5, 9 - Insert Title at head of Table.

Page 12 - Delete entire page.

Page 58 - Move entire page to bottom of Page 64.

Page 65 - Third sentence from end, read:
"PERT TIME is now ready for job execution."

Page 73 - Following "10-1 10.1" insert all of Page 74.

29 June 1966

CONTENTS

INTRODUCTION

6400/6600 Systems Availability Report

The systems listed below are currently available from Program Distribution in Palo Alto. Please submit requests along with the correct number of magnetic tapes to your local CONTROL DATA representative.

System	Version	*Tapes Required		Maint. Doc. Available	Price Per Copy
		Standard Package	Assembly Listing		
Chippewa Operating System** (Specify 3000 or 6000 controllers)	1.1	1	1		
FORTRAN**	1.1	0	0		
ASCENT**	1.1	0	0	X	5.20
CDCKWIC	1.1	1		X	.75
MATRIX ALGEBRA SUBROUTINES	1.0	0			
PERT TIME	1.0	1		X	1.50

*Tapes required. The column headed Standard Package shows the number of tapes required by Program Distribution before requests for a system can be honored. If this column is 0, the system is normally released as a card deck. The column headed Assembly Listing indicates the number of tapes required if the requester desires a list tape. Listings are not sent unless they are specifically requested and a tape is furnished. The tape requirements do not apply to requesters outside the domestic marketing area. All tapes are one-half inch wide.

**The system library and source are distributed on one reel of tape which contains seven files. File one is the 3000 or 6000 (tape controllers) binary system library; file two contains the ASCENT COSY source; file three contains the FORTRAN COSY source; file four contains the FORTRAN object library COSY source; file five contains the UTILITY programs COSY source; file six contains the Central Memory Resident COSY source; and file seven contains the PPU programs COSY source.

6000 →

LIBRARY PREPARATION

The master tape for this release contains seven binary files.

FILE 1.	3000 system binary or 6000 system binary
FILE 2.	ASCENT COSY
FILE 3.	FORTRAN COSY
FILE 4.	FORTRAN object library COSY
FILE 5.	Utility Programs COSY
FILE 6.	Central Memory Resident COSY
FILE 7.	PPU programs COSY

CENTRAL MEMORY RESIDENT (CMR)

A COSY deck of CMR has been included in this release (File 6). CMR is now composed of sixteen elements contained in one COSY program.

		Location (octal)	
1.	POINTERS	0-30	
2.	DATE LINE	31-36	Preset in CMR
3.	START	37-57	
4.	PPCOM	60-177	
5.	CPAREA	200-1777	
6.	CPRES	2000-2077	
7.	EST	2100-2177	
8.	CLD	2200-2377	
9.	PLD	2400-2477	
10.	TRT0	2500-2577	
11.	TRT1	2600-2677	
12.	TRT2	2700-2777	
13.	TRT3	3000-3077	
14.	TRT4	3100-3177	
15.	FNT/FST	aaaa-3777	
16.	DFB	4000-4777	

aaaa may vary from 2600 to 3200 depending on the number of disks in the system. Each disk requires a 100_8 word TRT (TRT0, TRT1, TRT2, TRT3, TRT4). When the system contains fewer than 5 disks, the origin of FNT/FST may be moved back into the space reserved for the unused TRT tables.

EQUIPMENT STATUS TABLE (EST)

The format of EST for 6000 equipment is as follows:

z	00cc	e0uu	o	h	x000
48	36	24	23	12	0

z=2000 Signifies an empty EST entry. The remaining bytes are zero.

=0000 Signifies the entry defines a piece of equipment in the system. The remaining bytes are significant.

cc Channel on which the equipment is attached.

e 6000 synchronizer number.

uu Unit number

o On/off bit; 0 indicates off, 1 indicates on. This bit can be changed with the ONnn/OFFnn statements from the console.

h Equipment type in display code:

DA	Channel 0 disk unit
DB	Channel 1 disk unit
DC	Channel 2 disk unit
DD	Channel 3 disk unit
DE	Channel 4 disk unit
CR	Card reader
CP	Card punch
DS	Display console
LP	Line printer
MT	607 magnetic tape
WT	626 magnetic tape

x Zero indicates 6000 equipment.

The format of EST for 3000 equipment has changed to allow for multi-channel controllers. The only equipment drivers presently taking advantage of more than one channel are the tape drivers. All 3000 drivers are changed to expect the new format:

Z	BB AA	DD CC	O HH	SEUU

where Z = 2000 Signifies an empty EST entry. The remaining bytes are zero. (12 bits)

= 0000 Signifies the entry defines a piece of equipment.

AA, BB, CC, DD are channels connected. (6 bits each)

O is the on/off bit. (1 bit)

HH is the equipment type (11 bits) in display code as listed above.

S is the 6681 number. (3 bits)

E is the equipment number. (3 bits)

UU is the unit number. (6 bits)

SYSTEM FILE

The system file includes all parts of the release. Two types of system files are being released; one type for installations with 3000 series tape controllers, the other for installations with 6000 series controllers. 3000 series versions of BOOT and LOADER replace LOADER-6000 for installations with 3000 series tapes.

The only modifications which may be necessary to the system file are changes to the Equipment Status Table (EST). Memory size is automatically assigned by MTR at dead start time. The EST modifications may be made in a number of ways.

1. Changing EST (locations 2100-2200) from the console after dead start.
2. Using Octal Correction Cards to modify EST with a deck similar to the following:

```
REQUEST TAPE1.                  (assign system tape)
REQUEST TAPE2.                  (assign unit for new tape)
REWIND (TAPE1)
REWIND (TAPE2)
COPYBR (TAPE1,TAPE2,4)          (6000 installations - copy 3 records)
RBR,1.
LOC.
WBR,2.
COPYBF (TAPE1,TAPE2)
REWIND (TAPE2)
CATALOG (TAPE2)
record separator card (7, 8, 9 punch in column 1)
(octal corrections for EST)
file separator card (6, 7, 8, 9 punch in column 1)
```

3. Use CMR from COSY file with appropriate modification cards.
4. Two versions of ASCENT have been provided on the system tape. One has a symbol table of length 6000_{10} to provide a means of assembling large programs such as RUN. This version is called by the name ASCENTR, and a field length of at least 55000_8 must be specified on the JOB card. The other version has a symbol table of length 1200_{10}, is called by the name ASCENT, and requires a field length of only 37000_8. See the ASCENT description for assembly instructions to produce ASCENTR.

```
RECORD     LENGTH    PACKAGE      CKSUM      LENGTH
     1        243     LOADER       1571         243
     2        274          ▪
     3        734        DSD       3136         734
     4        750        MTR       2507         750
     5       5000        CMR       7437        5000
     6       1614       ACOS       1433         105
                        ALOG       4270          42
                      ALOG10       0473          44
                        ASIN       1274          77
                        ATAN       3311          64
                         COS       2524          56
                       DVCHK       0744          10
                         END       2233          62
                        EXIT       6321          61
                         EXP       7701          42
                      IBAIEX       0206          22
                      OVERFL       1221           7
                       PAUSE       6110          23
                      RBAIEX       1022          14
                      RBAREX       7635         135
                      REMARK       5537          14
                         SIN       6726          56
                       SLITE       5420          17
                      SLITET       5402          22
                        SQRT       6262          32
                      SSWTCH       7246          22
                       START       4272          10
                        STOP       4730          62
                         TAN       3274          75
                        TANH       4171          51
                        TIME       4612          26
                           ▾
     7       4543        1AJ       4204         106
                         1BJ       3644         131
                         1LJ       2006         207
                         1OT       1613         253
                         1RO       5301         126
                         2BD       1432         121
                         2BP       2632         114
                         2CF       0555          54
                         2DF       0274          70
                         2DT       6025          46
                         2RC       2702         204
                         2RD       0161         117
                         2SD       5224          71
                         2TB       2503         156
                         2TJ       1326         122
                         2TR       1206         230
                         2TS       4366         372
                         2TW       4141         227
                         2WD       0757         125
                         7DP       1073          34
                         7TP       5157          40
                         CHK       4643          27
                         CIO       7263          74
                         CLL       6154         172
                         CLS       1614         135
                         EXU       0721         116
                         MSG       2222          26
                           ▪
     8       1240        007       6117        1240
```

CATALOG OF 3000 SERIES CONTROLLERS BINARY SYSTEM

```
   9        45       1CO     3565        45
  10       122       1DF     5415       122
  11        22       1DS     6234        22
  12        16       1FM     2261        16
  13       114       1LT     2603       114
  14        16       1PL     2173        16
  15       242       1PO     6600       242
  16       114       1RF     7006       114
  17       152       1RI     6205       152
  18       146       1TD     2725       146
  19       134       2BT     6543       134
  20        51       2EF     5240        51
  21       211       2LP     1724       211
  22       223       2PC     3652       223
  23       203       2RT     5306       203
  24       200       2WT     1621       200
  25       236       3OT     7007       236
  26        34       CPL     2214        34
  27      1205       DIS     7354      1205
  28       356       DMP     0365       356
  29        26       HLP     3751        26
  30       102       LBC     6135       102
  31       157       LOC     7374       157
  32        61       PBC     2402        61
  33       154       PBS     7765       154
  34        33       RBR     1705        33
  35        22       RFL     5523        22
  36        34       SOS     3363        34
  37        16       TIM     6361        16
  38        34       WBR     7245        34
  39         0
  40        62    ACGOER     1527        62
  41       105     ATAN2     5456       105
  42       140    BACKSP     6433       140
  43       224    BUFFEI     4137       224
  44       213    BUFFEO     4657       213
  45        30      CABS     1332        30
  46        34    CBAIEX     1405        34
  47       124      CCOS     6437       124
  48       116      CEXP     7213       116
  49        51     CHAIN     4620        51
  50       134      CLOG     7673       134
  51       124      CSIN     4642       124
  52        40     CSQRT     1112        40
  53        14      DABS     4655        14
  54       155     DATAN     4140       155
  55       156    DATAN2     0404       156
  56       225    DBADEX     2632       225
  57        26    DBAIEX     6046        26
  58       224    DBAREX     6646       224
  59        11      DBLE     1162        11
  60       154      DCOS     3703       154
  61       121      DEXP     4056       121
  62       252    DISPLA     0073       252
  63       147      DLOG     4353       147
  64       154    DLOG10     0436       154
  65        37      DMOD     5267        37
  66        17     DSIGN     5300        17
  67       154      DSIN     0217       154
  68        43     DSQRT     5420        43
  69       770      DUMP     0765       770
  70       125    ENDFIL     7540       125
```

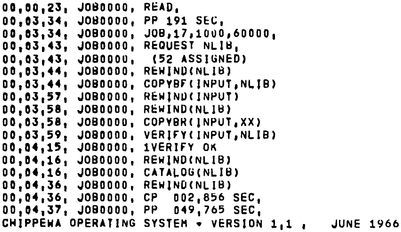
```
          71        26     IDINT      5630        26
          72       117    IFENDF      0765       117
          73       211    INPUTB      4361       211
          74      1357    INPUTC      3345      1357
          75      1243    INPUTS      6335      1243
          76         4    IOCHEC      1110         4
          77       170    IOCHEK      6751       170
          78       114    LENGTH      3764       114
          79         5      LOCF      5124         5
          80       164    OUTPTB      5455       164
          81      1407    OUTPTC      2404      1407
          82      1303    OUTPTS      4634      1303
          83       735     PDUMP      0016       735
          84        14      RANF      7006        14
          85       221    RBADEX      4323       221
          86       135    REWINM      3477       135
          87        23    SECOND      5277        23
          88        15      SNGL      7546        15
          89         4    SYSTEM      4255         4
          90         5     XLOCF      4571         5
          91        35      BKSP      2446        35
          92       267   CATALOG      0705       267
          93       100      COPY      4306       100
          94        47    COPYBF      7004        47
          95        47    COPYBR      6020        47
          96        47    COPYCF      6704        47
          97        47    COPYCR      5720        47
          98       120   COPYSBF      2335       120
          99       255     COPYX      0677       255
         100        21    REWIND      2677        21
         101        21    UNLOAD      1216        21
         102       502    VERIFY      5643       502
         103     10337    ASCENT      6251     10337
         104     10337   ASCENTR      0147     10337
         105     26461       RUN      7433     26461
         106         0
END OF FILE
```

```
RECORD     LENGTH    PACKAGE      CKSUM     LENGTH
     1        313          *
     2        734        DSD       3136        734
     3        750        MTR       2507        750
     4       5000        CMR       7437       5000
     5       1614       ACOS       1433        105
                        ALOG       4270         42
                      ALOG10       0473         44
                        ASIN       1274         77
                        ATAN       3311         64
                         COS       2524         56
                       DVCHK       0744         10
                         END       2233         62
                        EXIT       6321         61
                         EXP       7701         42
                      IBAIEX       0206         22
                      OVERFL       1221          7
                       PAUSE       6110         23
                      RBAIEX       1022         14
                      RBAREX       7635        135
                      REMARK       5537         14
                         SIN       6726         56
                       SLITE       5420         17
                      SLITET       5402         22
                        SQRT       6262         32
                      SSWTCH       7246         22
                       START       4272         10
                        STOP       4730         62
                         TAN       3274         75
                        TANH       4171         51
                        TIME       4612         26
                           *
     6       4543        1AJ       4204        106
                         1BJ       3644        131
                         1LJ       2006        207
                         1OT       1613        253
                         1RO       5301        126
                         2BD       1432        121
                         2BP       2632        114
                         2CF       0555         54
                         2DF       0274         70
                         2DT       6025         46
                         2RC       2702        204
                         2RD       0161        117
                         2SD       5224         71
                         2TB       2503        156
                         2TJ       1326        122
                         2TR       1206        230
                         2TS       4366        372
                         2TW       4141        227
                         2WD       0757        125
                         7DP       1073         34
                         7TP       5157         40
                         CHK       4643         27
                         CIO       7263         74
                         CLL       6154        172
                         CLS       1614        135
                         EXU       0721        116
                         MSG       2222         26
                           *
     7       1240        007       6117       1240
     8         45        1CO       3565         45
```

CATALOG OF 6000 SERIES CONTROLLERS BINARY SYSTEM

```
  9      122      1DF     5415      122
 10       22      1DS     6234       22
 11       16      1FM     2261       16
 12      114      1LT     2603      114
 13       16      1PL     2173       16
 14      242      1PO     6600      242
 15      114      1RF     7006      114
 16      152      1RI     6205      152
 17      146      1TD     2725      146
 18      134      2BT     6543      134
 19       51      2EF     5240       51
 20      211      2LP     1724      211
 21      223      2PC     3652      223
 22      203      2RT     5306      203
 23      200      2WT     1621      200
 24      236      3OT     7007      236
 25       34      CPL     2214       34
 26     1205      DIS     7354     1205
 27      356      DMP     0365      356
 28       26      HLP     3751       26
 29      102      LBC     6135      102
 30      157      LOC     7374      157
 31       61      PBC     2402       61
 32      154      PBS     7765      154
 33       33      RBR     1705       33
 34       22      RFL     5523       22
 35       34      SOS     3363       34
 36       16      TIM     6361       16
 37       34      WBR     7245       34
 38        0
 39       62   ACGOER     1527       62
 40      105    ATAN2     5456      105
 41      140   BACKSP     6433      140
 42      224   BUFFEI     4137      224
 43      213   BUFFEO     4657      213
 44       30     CABS     1332       30
 45       34   CBAIEX     1405       34
 46      124     CCOS     6437      124
 47      116     CEXP     7213      116
 48       51    CHAIN     4620       51
 49      134     CLOG     7673      134
 50      124     CSIN     4642      124
 51       40    CSQRT     1112       40
 52       14     DABS     4655       14
 53      155    DATAN     4140      155
 54      156   DATAN2     0404      156
 55      225   DBADEX     2632      225
 56       26   DBAIEX     6046       26
 57      224   DBAREX     6646      224
 58       11     DBLE     1162       11
 59      154     DCOS     3703      154
 60      121     DEXP     4056      121
 61      252   DISPLA     0073      252
 62      147     DLOG     4353      147
 63      154   DLOG10     0436      154
 64       37     DMOD     5267       37
 65       17    DSIGN     5300       17
 66      154     DSIN     0217      154
 67       43    DSQRT     5420       43
 68      770     DUMP     0765      770
 69      125   ENDFIL     7540      125
 70       26    IDINT     5630       26
```

```
       71       117    IFENDF      0765       117
       72       211    INPUTB      4361       211
       73      1357    INPUTC      3345      1357
       74      1243    INPUTS      6335      1243
       75         4    IOCHEC      1110         4
       76       170    IOCHEK      6751       170
       77       114    LENGTH      3764       114
       78         5      LOCF      5124         5
       79       164    OUTPTB      5455       164
       80      1407    OUTPTC      2404      1407
       81      1303    OUTPTS      4634      1303
       82       735     PDUMP      0016       735
       83        14      RANF      7006        14
       84       221    RBADEX      4323       221
       85       135    REWINM      3477       135
       86        23    SECOND      5277        23
       87        15      SNGL      7546        15
       88         4    SYSTEM      4255         4
       89         5     XLOCF      4571         5
       90        35      BKSP      2446        35
       91       267   CATALOG      0705       267
       92       100      COPY      4306       100
       93        47    COPYBF      7004        47
       94        47    COPYBR      6020        47
       95        47    COPYCF      6704        47
       96        47    COPYCR      5720        47
       97       120   COPYSBF      2335       120
       98       255     COPYX      0677       255
       99        21    REWIND      2677        21
      100        21    UNLOAD      1216        21
      101       502    VERIFY      5643       502
      102     10337    ASCENT      6251     10337
      103     10337   ASCENTR      0147     10337
      104     26461       RUN      7433     26461
      105         0
END OF FILE
```



```
00,00,23, JOB0000, READ,
00,03,34, JOB0000, PP 191 SEC,
00,03,34, JOB0000, JOB,17,1000,60000,
00,03,43, JOB0000, REQUEST NLIB,
00,03,43, JOB0000,  (52 ASSIGNED)
00,03,44, JOB0000, REWIND(NLIB)
00,03,44, JOB0000, COPYBF(INPUT,NLIB)
00,03,57, JOB0000, REWIND(INPUT)
00,03,58, JOB0000, REWIND(NLIB)
00,03,58, JOB0000, COPYBR(INPUT,XX)
00,03,59, JOB0000, VERIFY(INPUT,NLIB)
00,04,15, JOB0000, 1VERIFY OK
00,04,16, JOB0000, REWIND(NLIB)
00,04,16, JOB0000, CATALOG(NLIB)
00,04,36, JOB0000, REWIND(NLIB)
00,04,36, JOB0000, CP  002,856 SEC,
00,04,37, JOB0000, PP  049,765 SEC,
CHIPPEWA OPERATING SYSTEM - VERSION 1,1 ,   JUNE 1966
```

DEAD START PANEL

The dead start panel is set as follows:

	3000 Tape Controller Version	6000 Tape Controller Version
0001	75xx	1410
0002	77xx	73xx
0003	e00u	0006
0004	77xx	75xx
0005	0010	7113
0006	77xx	0000
0007	1400	77xx
0010	74xx	206u
0011	2001	77xx
0012	0000	202u
0013	71xx	74xx
0014	0000	71xx

where e= controller number, u= unit number, and xx= channel number on which the system tape is mounted. (For 3000 systems, xx may only be channels 12 or 13.)

COSY FILES

The COSY files contain new binary COSY decks for all programs in the release. The COSY decks contain new sequence numbers.

Each COSY file is recorded in odd (binary) parity. Each program comprises one logical record. ASCENT is used to assemble the programs. COPYBR and COBYBF may be used to position the tape for assembling selected programs from the COSY tape.

Examples:

1. To assemble the 6th COSY file from the master tape, generating listings and binary deck:

```
JOB1, 5, 500, 40000.
REQUEST COSY.
REWIND (COSY)
COPYBF (COSY, XX, 5)              skip 1st 5 files
ASCENT (L, 0, PA, 0, 0, COSY)
```

```
record separator card           (7, 8, 9 punch in column 1)
        COSY
record separator card           As many COSY and record separator cards as
                                routines to be assembled.
        COSY
          .
          .
          .
record separator card
        COSY
record separator card
        COSY
file separator card             (6, 7, 8, 9 punch in column 1)
```

2. To assemble and list the second and fifth programs from the 6th file of the master tape.

```
JOB2,5,500,40000.
REQUEST COSY.
REWIND (COSY)
COPYBF (COSY,XX,5)
COPYBR (COSY,XX)                (skip over first program)
ASCENT (L,0,0,0,0,COSY)         (assemble second program)
record separator card
        COSY
file separator card
JOB2,5,500,40000.
REQUEST COSY.
COPYBR (COSY,XX,2)              (skip over third and fourth programs)
ASCENT (L,0,0,0,0,COSY)         (assemble fifth program)
record separator card
        COSY
file separator card
```

Note that these jobs must be run serially.

3. To copy and verify the release tape. The new system binary file will be cataloged after the verification.

```
JOB3,5,500,40000.
REQUEST OLD.
REQUEST NEW.
REWIND (OLD)
REWIND (NEW)
COPYBF (OLD,NEW,7)
REWIND (OLD)
REWIND (NEW)
VERIFY (OLD,NEW)
VERIFY (OLD,NEW)
VERIFY (OLD,NEW)
VERIFY (OLD,NEW)
VERIFY (OLD,NEW)
VERIFY (OLD,NEW)
VERIFY (OLD,NEW)
REWIND (NEW)
CATALOG (NEW)
record separator card
file separator card
```

The records contained on each of the COSY files are listed below.

File 2: ASCENT

1. ASCENT

File 3: FORTRAN

1. RUN

File 4: FORTRAN Object Library

1. ACGOER
2. ACOS
3. ALOG
4. ALOG10
5. ASIN
6. ATAN
7. ATAN2
8. BACKSP
9. BUFFEI
10. BUFFEO
11. CABS
12. CBAIEX
13. CCOS
14. CEXP
15. CHAIN
16. CLOG
17. COS
18. CSIN
19. CSQRT
20. DABS
21. DATAN
22. DATAN2
23. DBADEX
24. DBAIEX
25. DBAREX
26. DBLE
27. DCOS
28. DEXP
29. DISPLA
30. DLOG
31. DLOG10
32. DMOD
33. DSIGN
34. DSIN
35. DSQRT
36. DUMP

File 4: FORTRAN Object Library (Continued)

37. DVCHK
38. END
39. ENDFIL
40. EXIT
41. EXP
42. IBAIEX
43. IDINT
44. IFENDF
45. INPUTB
46. INPUTC
47. INPUTS
48. IOCHEC
49. IOCHEK
50. LENGTH
51. LOCF
52. OUTPTB
53. OUTPTC
54. OUTPTS
55. OVERFL
56. PAUSE
57. PDUMP
58. RANF
59. RBADEX
60. RBAIEX
61. RBAREX
62. REMARK
63. REWINM
64. SECOND
65. SIN
66. SLITE
67. SLITET
68. SNGL
69. SQRT
70. SSWTCH
71. START
72. STOP
73. SYSTEM
74. TAN
75. TANH
76. TIME
77. XLOCF

Note: There is an erroneous zero length record between DSIGN and DSIN in File 4.

File 5: Utility Programs

1. BKSP
2. CATALOG
3. COPY
4. COPYBF
5. COPYBR
6. COPYCF
7. COPYCR
8. COPYSBF
9. COPYX
10. REWIND
11. UNLOAD
12. VERIFY

File 6: Central Memory Resident

1. A listing of CMR follows.

```
                                                  ASCENT CMR          ,              11 JUNE 66,  RON ZEILINGER    00001
                                                  ,***********************************************************     00002
                                                  .                                                                00003
                                                  .          CMR -- CENTRAL MEMORY RESIDENT                        00004
                                                  .                                                                00005
                                                  ,***********************************************************     00006
                                                  .                                                                00007
                                                  .          ALL CENTRAL MEMORY TABLES ARE POSITIONED              00008
                                                  .           AND POINTERS ARE SET ACCORDING TO                    00009
                                                  .           THE STARTING ADDRESSES SET BELOW --                  00010
                                                  .                                                                00011
     000200                              CPAREA   EQU     200B        ,CONTROL POINT AREAS                         00012
     002000                              CPRES    EQU    2000B        ,CP RESIDENTS                                00013
     002100                              EST      EQU    2100B        ,EQIIPMENT STATUS TABLE                      00014
     002200                              CLD      EQU    2200B        ,CENTRAL LIBRARY DIRECTORY                   00015
     002400                              PLD      EQU    2400B        ,PERIPHERAL LIBRARY DIRECTORY                00016
     002500                              TRT0     EQU    2500B        ,TRACK RESERVATION TABLE -- DISK 0           00017
     003200                              FNT      EQU    3200B        ,FILE NAME + STATUS TABLE                    00018
     004000                              DFB      EQU    4000B        ,DAYFILE BUFFER                              00019
     005000                              RSL      EQU    5000B        ,RESIDENT CP SUBROUTINE LIBRARY              00020
     007000                              RPL      EQU    7000B        ,RESIDENT PERIPHERAL LIBRARY                 00021
                                                  .                                                                00022
                                                  .                                                                00023
     004003                              DFBIN    EQU    DFB+3        ,INPUT POINTER FOR DFB                       00024
     002600                              TRT1     EQU    TRT0+100B    ,TRACK RESERVATION TABLE -- DISK 1           00025
     002700                              TRT2     EQU    TRT1+100B    ,TRACK RESERVATION TABLE -- DISK 2           00026
     003000                              TRT3     EQU    TRT2+100B    ,TRACK RESERVATION TABLE -- DISK 3           00027
     003100                              TRT4     EQU    TRT3+100B    ,TRACK RESERVATION TABLE -- DISK 4           00028
     007777                              LTRK     EQU    7777B        ,LAST TRACK NO,  (DISK POSITION)             00029
     000100                              SLOZ     EQU    0100B        ,SECTOR LIMIT FOR OUTER ZONE HALF-TRACKS     00030
     000062                              SLIZ     EQU    0062B        ,SECTOR LIMIT FOR INNER ZONE HALF-TRACKS     00031
                                                  .                                                                00032
                                                  .          POINTERS TO CM TABLES                                 00033
                                                  .                                                                00034
000000  03152200000000005000                      VFD    D18/CMR,N42/5000B     ,SYSTEM LABEL -- CMR                00035
000001  70000000000000000000                      VFD    A12/RPL               ,      LOC, 1 = RPL POINTER         00036
000002  24002500000000000000                      VFD    A12/PLD,A12/TRT0      ,      LOC, 2 = PLD POINTER         00037
000003  40004003400050000000                      VFD    A12/DFB,A12/DFBIN,A12/DFB,A12/RSL, = DFB POINTER          00038
000004  32004000000000000000                      VFD    A12/FNT,A12/DFB       ,      LOC, 4 = FNT POINTER         00039
000005  21002200000000000000                      VFD    A12/EST,A12/CLD       ,      LOC, 5 = EST POINTER         00040
000006  50007000000000000000                      VFD    A12/RSL,A12/RPL       ,      LOC, 6 = RSL POINTER         00041
000007  22002400000000000000                      VFD    A12/CLD,A12/PLD       ,      LOC, 7 = CLD POINTER         00042
000010  25007777000001000062                      VFD    A12/TRT0,A12/LTRK,A24/SLOZ,A12/SLIZ, TRT DISK 0           00043
000011  26007777000001000062                      VFD    A12/TRT1,A12/LTRK,A24/SLOZ,A12/SLIZ, TRT DISK 1           00044
000012  27007777000001000062                      VFD    A12/TRT2,A12/LTRK,A24/SLOZ,A12/SLIZ, TRT DISK 2           00045
000013  30007777000001000062                      VFD    A12/TRT3,A12/LTRK,A24/SLOZ,A12/SLIZ, TRT DISK 3           00046
000014  31007777000001000062                      VFD    A12/TRT4,A12/LTRK,A24/SLOZ,A12/SLIZ, TRT DISK 4           00047
000015  00000000000000000000                      BSSZ   3            ,CHANNEL STATUS TABLE (CST)                  00048
000020  00030000000000000000                      VFD    N12/3        ,STATUS FOR PSEUDO-CONTROL POINT 0           00049
000021  15171611241722000000                      VFD    D42/MONITOR ,JOB NAME FOR CONTROL PT, 0                   00050
000022  00000000000000000000                      BSSZ   5            ,IDLE TIMES                                  00051
000027  00010000000000000000                      VFD    N12/1        ,INITIAL P ADDR, FOR SIMULATOR               00052
000030  55333357333357333357                      DPC    * 00,00,00,*            ,TIME (HR,MIN,SEC,)               00053
000031  55031011202005270155                      DPC    * CHIPPEWA OPERATING SYSTEM - VERSION 1,1 ,   JUNE *      00054
000032  17200522012411160755
000033  23312324051555465526
```

```
000034  05222311171655345734
000035  55565555551225160555
000036  34444141000000000000          VFD    N24/34444141B   ,1966
                                      ,
                                      ORG    37B            ,
000037  00000000000000000000          BSSZ   21B            ,STARTING TIMES
                                      ORG    60B            ,
000060  00000000000000000000          BSSZ   120B           ,PP COMUNICATION AREAS
                                      ,
                                      ORG    CPAREA         ,CONTROL POINT AREAS
000200  00000000000000000000          BSSZ   1              ,
000201  00014000000000000000          VFD    N15/14B        ,RA = CONTROL PT, 1
000202  00000000000000000000          BSSZ   16B            ,EXCHANGE JUMP PKG
000220  00000000000001400000          CON    1400000B       ,
000221  00000000000000000000          BSSZ   160B           ,
000401  00014000000000000000          VFD    N15/14B        ,RA = CONTROL PT, 2
000402  00000000000000000000          BSSZ   16B            ,EXCHANGE JUMP PKG
000420  00000000000001400000          CON    1400000B       ,
000421  00000000000000000000          BSSZ   160B           ,
000601  00014000000000000000          VFD    N15/14B        ,RA = CONTROL PT, 3
000602  00000000000000000000          BSSZ   16B            ,EXCHANGE JUMP PKG
000620  00000000000001400000          CON    1400000B       ,
000621  00000000000000000000          BSSZ   160B           ,
001001  00014000000000000000          VFD    N15/14B        ,RA = CONTROL PT, 4
001002  00000000000000000000          BSSZ   16B            ,EXCHANGE JUMP PKG
001020  00000000000001400000          CON    1400000B       ,
001021  00000000000000000000          BSSZ   160B           ,
001201  00014000000000000000          VFD    N15/14B        ,RA = CONTROL PT, 5
001202  00000000000000000000          BSSZ   16B            ,EXCHANGE JUMP PKG
001220  00000000000001400000          CON    1400000B       ,
001221  00000000000000000000          BSSZ   160B           ,
001401  00014000000000000000          VFD    N15/14B        ,RA = CONTROL PT, 6
001402  00000000000000000000          BSSZ   16B            ,EXCHANGE JUMP PKG
001420  00000000000001400000          CON    1400000B       ,
001421  00000000000000000000          BSSZ   160B           ,
001601  00014000000000000000          VFD    N15/14B        ,RA = CONTROL PT, 7
001602  00000000000000000000          BSSZ   16B            ,EXCHANGE JUMP PKG
001620  00000000000001400000          CON    1400000B       ,
001621  00000000000000000000          BSSZ   CPRES-*        ,
                                      ,
                                      ,      CENTRAL PROCESSOR RESIDENT ROUTINES
                                      ,
                                      ORG    CPRES          ,
    002022                    MOVE    EQU    CPRES+22B      ,
002000  00002022000000000000          VFD    N6/0,A18/MOVE      ,SET P = START OF STORAGE MOVE
002001  00000000000000000000          CON    0              ,
002002  00400000000000000000          VFD    N9/4           ,SET EXIT MODE
002003  00000000000000000000          BSSZ   17B            ,EXCHANGE JUMP PKG
    000000                    EXIT    EQU    0              ,
                                      ,
                                      ,      STORAGE MOVE ROUTINE FOR RELOCATING CENTRAL MEMORY
                                      ,
                                      ,ON ENTRY MONITOR HAS PRESET --
                                      , B1 = BLOCK STARTING ADDRESS OF STORAGE TO BE MOVED
                                      , B2 = BLOCK ENDING ADDRESS
                                      , B3 = AMOUNT AND DIRECTION OF RELOCATION
```

```
                                                     .
002022  0412000000               ENTRY    EQ     B1=B2,EXIT   ,IF BSA = BEA, THERE IS NOTHING TO MOVE
                  6170000001              SB7  = 1            ,
002023  0730002025                        NG     B3,DOWN      ,JP IF DIRECTION OF MOVE IS DOWNWARD
                                          ,
                  6150777775              SB5  = -2           ,WHEN THE MOVE IS UPWARD,
002024  57127                             SA1    B2-B7        , START MOVING WORDS (4 AT A TIME)
             56225                        SA2    B2+B5        ,                  FROM THE END OF THE BLOCK
                  0200002026              JP     LOOP         ,
002025  66577                    DOWN     SB5    B7+B7        ,WHEN THE MOVE IS DOWNWARD,
             56110                        SA1    B1           , START MOVING WORDS (4 AT A TIME)
                  56217                   SA2    B1+B7        ,                FROM THE START OF THE BLOCK
                                          ,
002026  54315                    LOOP     SA3    A1+B5        ,READ 4 WORDS AT A TIME
             54425                        SA4    A2+B5        ,
                  10610                   BX6    X1           ,
                       22702              LX7    X2           ,
002027  54613                             SA6    A1+B3        ,STORE THE 4 WORDS
             54723                        SA7    A2+B3        , RELOCATED BY THE AMOUNT IN B3
                  10630                   BX6    X3           ,
                       22704              LX7    X4           ,
002030  54633                             SA6    A3+B3        ,
             54743                        SA7    A4+B3        ,
                  6111000004              SB1    B1+4         ,DECREMENT BLOCK SIZE BY 4 WORDS
002031  54135                             SA1    A3+B5        ,
             54245                        SA2    A4+B5        ,
                  0712002026              LT     B1,B2 LOOP   ,UNTIL STARTING ADDR = ENDING ADDR
002032  0200000000                        JP     EXIT         ,
                                          ,
002033  00000000000000000000              BSSZ   CPRES+40B-*  ,
002040  00000002000000000000              VFD    N24/2        ,
002041  00002060000000000000              VFD    N24/2060B    ,
002042  00000020000000000000              VFD    N24/20B      ,
002043  00000000000000000000              BSSZ   15B          ,EXCHANGE JUMP PKG
002060  33333300000000000000              VFD    N18/333333B  ,
002061  00000000000000000000              CON    0            ,
002062  04000000020000000000              VFD    N6/4,N24/2   ,
002063  00000000000000000000              BSSZ   EST-*        ,
                                                  ,************************************************************
                                                  ,
                                                  ,       EST -- EQUIPMENT STATUS TABLE
                                                  ,
                                                  ,************************************************************
                                                  ,
                                                  ,       FORMAT FOR 3000-SERIES CONTROLLERS
                                                  ,
                                                  ,  1ST BYTE = CONTROL PT, ADDRESS (TO WHICH EQUIP IS ASSIGNED)
                                                  ,  2ND BYTE = A + B CHANNELS CONNECTED INTO CONTROLLER
                                                  ,  3RD BYTE = C + D CHANNELS CONNECTED INTO CONTROLLER
                                                  ,  4TH BYTE = EQUIP, TYPE (2 LETTERS IN DISPLAY CODE)
                                                  ,  4TH BYTE, LEFTMOST BIT = INTERLOCK (0 = ON, 1 = OFF)
                                                  ,  5TH BYTE = 6681 NO, + 3000 EQUIP, NO, + UNIT NO,
                                                  ,
                                                  ,************************************************************
                                                  ,
                                                  ,       FORMAT FOR 6000 SYNCHRONIZERS
                                                  ,
                                                  ,  1ST BYTE = CONTROL POINT ADDRESS,  OR 2000 IF EMPTY ENTRY
                                                  ,  2ND BYTE = CHANNEL NO,
                                                  ,  3RD BYTE = SYNCHRONIZER + UNIT NO,
                                                  ,  4TH BYTE = INTERLOCK + EQUIP, TYPE IN DISPLAY CODE
                                                  ,  5TH BYTE = NOT USED
                                                  ,
                                                  ,************************************************************
   000000                                 IN      EQU    0             ,EQUIPMENT IS IN THIS CONFIGURATION
   002000                                 OUT     EQU    2000B         ,EMPTY EST ENTRY -- NO EQUIP, ATTACHED
   000000                                 DACH    EQU    0             ,DISK 0 (DA) CHANNEL = 0
   001000                                 SYNC    EQU    1000B         ,DISK SYNCHRONIZER = 1, UNIT NO, = 0
   000012                                 CR1C    EQU    12B           ,CARD READER 1 ON CHAN 12B
   000400                                 CR1E    EQU    400B          ,CARD READER 1 = EQUIP, 4
   000013                                 CPCH    EQU    13B           ,CARD PUNCH ON CHAN, 13
   000700                                 CPEQ    EQU    700B          ,CARD PUNCH = EQUIP, 7
   000010                                 DS1C    EQU    10B           ,DISPLAY SCOPE 1 ON CHAN, 10
   007000                                 DSYN    EQU    7000B         ,DISPLAY SCOPE SYNCHRONIZER = 7
   000011                                 LP1C    EQU    11B           ,LINE PRINTER 1 ON CHAN, 11
   000600                                 LP1E    EQU    600B          ,LINE PRINTER 1 = EQUIP, 6
   000011                                 LP2C    EQU    11B           ,LINE PRINTER 2 ON CHAN, 11
   000700                                 LP2E    EQU    700B          ,LINE PRINTER 2 = EQUIP, 7
   000012                                 MTCH    EQU    12B           ,MAG, TAPE CONTROLLER ON CHANNEL 12
   000500                                 MT1E    EQU    500B          ,MT1 = EQUIP, 5, UNIT 0
   000501                                 MT2E    EQU    501B          ,MT2 = EQUIP, 5, UNIT 1
   000502                                 MT3E    EQU    502B          ,MT3 = EQUIP, 5, UNIT 2
   000503                                 MT4E    EQU    503B          ,MT4 = EQUIP, 5, UNIT 3
                                                  ,
                                                  ORG    EST
                                                  ,
                                                  ,       THIS EQUIP, TABLE FOR 6400 AT PALO ALTO
                                                  ,
002100  00000000100004010000                      VFD    A12/IN,A12/DACH,A12/SYNC,D12/DA   ,DISK 0 (DA) ON CHAN 0
002101  20000000000000000000                      VFD    A12/OUT       ,
002102  20000000000000000000                      VFD    A12/OUT       ,
002103  20000000000000000000                      VFD    A12/OUT       ,
002104  20000000000000000000                      VFD    A12/OUT       ,
002105  00000012000003220400                      VFD    A12/IN,A12/CR1C,N12/0,D12/CR,A12/CR1E   ,CARD READER 1
```

```
002106  20000000000000000000        VFD     A12/OUT      ,                                            00204
002107  00000013000003200700        VFD     A12/IN,A12/CPCH,N12/0,D12/CP,A12/CPEQ      ,CARD PUNCH      00205
002110. 00000010700004230000        VFD     A12/IN,A12/DS1C,A12/DSYN,D12/DS            ,DISPLAY SCOPE   00206
002111  20000000000000000000        VFD     A12/OUT      ,                                            00207
002112  20000000000000000000        VFD     A12/OUT      ,                                            00208
002113  20000000000000000000        VFD     A12/OUT      ,                                            00209
002114  20000000000000000000        VFD     A12/OUT      ,                                            00210
002115  20000000000000000000        VFD     A12/OUT      ,                                            00211
002116  20000000000000000000        VFD     A12/OUT      ,                                            00212
002117  20000000000000000000        VFD     A12/OUT      ,                                            00213
002120  00000011000014200600        VFD     A12/IN,A12/LP1C,N12/0,D12/LP,A12/LP1E      ,LINE PRINTER 1  00214
002121  00000011000014200700        VFD     A12/IN,A12/LP2C,N12/0,D12/LP,A12/LP2E      ,LINE PRINTER 2  00215
002122  20000000000000000000        VFD     A12/OUT      ,                                            00216
002123  20000000000000000000        VFD     A12/OUT      ,                                            00217
002124  20000000000000000000        VFD     A12/OUT      ,                                            00218
002125  20000000000000000000        VFD     A12/OUT      ,                                            00219
002126  20000000000000000000        VFD     A12/OUT      ,                                            00220
002127  20000000000000000000        VFD     A12/OUT      ,                                            00221
002130  20000000000000000000        VFD     A12/OUT      ,                                            00222
002131  20000000000000000000        VFD     A12/OUT      ,                                            00223
002132  20000000000000000000        VFD     A12/OUT      ,                                            00224
002133  20000000000000000000        VFD     A12/OUT      ,                                            00225
002134  20000000000000000000        VFD     A12/OUT      ,                                            00226
002135  20000000000000000000        VFD     A12/OUT      ,                                            00227
002136  20000000000000000000        VFD     A12/OUT      ,                                            00228
002137  20000000000000000000        VFD     A12/OUT      ,                                            00229
002140  20000000000000000000        VFD     A12/OUT      ,                                            00230
002141  20000000000000000000        VFD     A12/OUT      ,                                            00231
002142  20000000000000000000        VFD     A12/OUT      ,                                            00232
002143  20000000000000000000        VFD     A12/OUT      ,                                            00233
002144  20000000000000000000        VFD     A12/OUT      ,                                            00234
002145  20000000000000000000        VFD     A12/OUT      ,                                            00235
002146  20000000000000000000        VFD     A12/OUT      ,                                            00236
002147  20000000000000000000        VFD     A12/OUT      ,                                            00237
002150  00000012000015240500        VFD     A12/IN,A12/MTCH,N12/0,D12/MT,A12/MT1E      ,MAG, TAPE 1     00238
002151  00000012000015240501        VFD     A12/IN,A12/MTCH,N12/0,D12/MT,A12/MT2E      ,MAG, TAPE 2     00239
002152  00000012000015240502        VFD     A12/IN,A12/MTCH,N12/0,D12/MT,A12/MT3E      ,MAG, TAPE 3     00240
002153  00000012000015240503        VFD     A12/IN,A12/MTCH,N12/0,D12/MT,A12/MT4E      ,MAG, TAPE 4     00241
002154  20000000000000000000        VFD     A12/OUT      ,                                            00242
002155  20000000000000000000        VFD     A12/OUT      ,                                            00243
002156  20000000000000000000        VFD     A12/OUT      ,                                            00244
002157  20000000000000000000        VFD     A12/OUT      ,                                            00245
002160  20000000000000000000        VFD     A12/OUT      ,                                            00246
002161  20000000000000000000        VFD     A12/OUT      ,                                            00247
002162  20000000000000000000        VFD     A12/OUT      ,                                            00248
002163  20000000000000000000        VFD     A12/OUT      ,                                            00249
002164  20000000000000000000        VFD     A12/OUT      ,                                            00250
002165  20000000000000000000        VFD     A12/OUT      ,                                            00251
002166  20000000000000000000        VFD     A12/OUT      ,                                            00252
002167  20000000000000000000        VFD     A12/OUT      ,                                            00253
002170  20000000000000000000        VFD     A12/OUT      ,                                            00254
002171  20000000000000000000        VFD     A12/OUT      ,                                            00255
002172  20000000000000000000        VFD     A12/OUT      ,                                            00256
002173  20000000000000000000        VFD     A12/OUT      ,                                            00257
002174  20000000000000000000        VFD     A12/OUT      ,                                            00258
002175  20000000000000000000        VFD     A12/OUT      ,                                            00259
```

```
ED  0     ASCENT CMR         ,                  ASCENT - VERSION 1,1                 PAGE NO,       7

002176  20000000000000000000              VFD    A12/OUT      ,
002177  20000000000000000000              VFD    A12/OUT      ,
                                          .
                                          .        CLD -- CENTRAL LIBRARY DIRECTORY
                                          .
                                          ORG    CLD          ,
002200  00000000000000000000              BSSZ   PLD-*        ,
                                          .
                                          .        PLD -- PERIPHERAL LIBRARY DIRECTORY
                                          .
                                          ORG    PLD          ,
002400  00000000000000000000              BSSZ   TRT0-*       ,
                                          .
                                          .        TRT0 -- TRACK RESERVATION TABLE FOR DISK 0
                                          .                FOLLOWED BY TRT1, TRT2, TRT3, TRT4
                                          .                ONE TABLE FOR EACH DISK REQUIRES 100B WORDS
                                          .
                                          ORG    TRT0         ,
002500  00000000000000000000              BSSZ   FNT-*        ,
                                          .
                                          .        FNT/FST -- FILE NAME + STATUS TABLES
                                          .
                                          ORG    FNT          ,
003200  04013106111405000020              VFD    D42/DAYFILE,N18/20B      ,1ST ENTRY = DAYFILE
003201  00000000000000000000              BSSZ   DFB-*        ,
                                          .
                                          .        DFB -- DAYFILE BUFFER
                                          .
                                          ORG    DFB          ,
004000  55333357333357333357              DPC    * 00,00,00, DEAD-STAR*     ,INITIAL DAYFILE ENTRY
004001  55040501044623240122
004002  24550000000000000000              VFD    D12/T,,N48/0,
004003  00000000000000000000              BSSZ   RSL-*        ,
                                          END    CMR          ,

NUMBER OF LINES WITH DIAGNOSTICS  ---          0
```

File 7: Peripheral Processor (PPU)

1. BOOT - 3X2X
2. CHK
3. CIO
4. CLL
5. CLS
6. CPL
7. DIS
8. DMP
9. DSD
10. EXU
11. HLP
12. LBC
13. LDR - 3X2X
14. LDR - 6000
15. LOC
16. MSG
17. MTR
18. PBC - 3446
19. PBS - 3446
20. RBR
21. RFL
22. SOS
23. TIM
24. WBR
25. 007
26. 1AJ
27. 1BJ
28. 1CO - 3256
29. 1DF
30. 1DS
31. 1FM
32. 1LJ - 3447
33. 1LT
34. 1OT
35. 1PL
36. 1PO - 3446
37. 1RF
38. 1RI
39. 1RO
40. 1TD
41. 2BD
42. 2BP
43. 2BT - 6000
44. 2CF
45. 2DF
46. 2DT
47. 2EF
48. 2LP - 3256
49. 2PC -3446
50. 2RC - 3447
51. 2RD
52. 2RT - 6000
53. 2SD
54. 2TB - 3X2X
55. 2TJ
56. 2TR - 3X2X
57. 2TS
58. 2TW - 3X2X
59. 2WD
60. 2WT - 6000
61. 3OT - 3256
62. 7DP
63. 7TP - 3X2X

Note: There is an erroneous zero length record between 7DP and 7TP - 3X2X in File 7.

SYSTEM DECK/TAPE

The following describes a system deck/tape prepared for a 6400/6600 computer using 3000 series tape controllers. Record numbers in this discussion are valid for a system deck/tape with 6000 series tape controllers if all such record numbers are decreased by one. This is necessary since the 6000 controller version does not need the first logical record.

A system deck/tape consists of six sections: LDR/PPR, DSD, and MTR; CMR; RSL; RPL; PLD; CLD. A brief description of each section follows:

LDR/PPR, DSD, MTR

The first four logical records on the system deck/tape, in order, must be the system tape loader occupying logical records 1 and 2 (LDR/PPR), the system display routine (DSD), and the monitor routine (MTR).

CMR

Logical record 5, the central memory tables, is transferred to the first 5000_8 words of central memory. CMR contains system tables, pointers to system tables, flags, the Dayfile buffer, system label information, and miscellaneous other items.

RSL and RPL

Logical record 6, the Resident Subroutine Library, and logical record 7, the Resident Peripheral Library, contain the central processor routines (RSL) and the peripheral processor routines (RPL) to be stored in central memory. The length of each of these records is the sum of the field lengths of the individual routines contained in it plus one. All the routines in each logical record are filed alphabetically. Each of the two logical records is terminated by a zero word (card with 3,7,9 punch in column 1). This card is then followed by the standard record separator card (7,8,9 punch in column 1).

PLD

Logical records 8 through 8 + n-1 (where n is the number of routines) contain the Peripheral Library Directory routines which form PLD, one routine per logical record. Logical record 8 + n must be a zero length record to flag the end of the PLD. A zero length record is produced by two consecutive record separator cards.

CLD

Logical records 8 + n + 1 through 8 + n + m (where m is the number of routines in CLD and n is as defined above) contain the Central Library Directory routines which form CLD, one routine per logical record. Logical record 8 + n + m + 1 must be a zero length record to flag the end of the CLD.

A file separator card (6,7,8,9 punch in column 1) must follow the last record separator card.

BINARY DECK PREPARATION

Assuming binary decks for all programs have been generated, separated, and labelled, the following is the deck structure required in order to generate, verify and catalog the new system tape. If the system does not verify, re-run the job. The catalog should agree with the attached catalog unless the user wishes to alter the contents of RPL, PLD, RSL, or CLD.

```
JOB,17,1000, 60000.
REQUEST NLIB.   (assign tape to write)
REWIND (NLIB)
COPYBF (INPUT,NLIB)   (copy complete system)
REWIND (INPUT)
REWIND (NLIB)
COPYBR (INPUT, XX)   (space over control cards)
VERIFY (INPUT,NLIB)   (verify system)
REWIND (NLIB)
CATALOG (NLIB)   (itemize contents)
REWIND (NLIB)
record separator card (7,8,9 punch in column 1)
LOADER-3000             \
record separator card    >    or LOADER-6000
BOOT-3000               /
record separator card
DSD
record separator card
MTR
record separator card
CMR
record separator card
ACOS                 \
ALOG                  |
   .                  |
   .                   >  RSL
   .                  |
TANH                  |
TIME                 /
```

zero word terminator card to terminate RSL (3,7,9 punch in column 1)

record separator card

```
1AJ                       ⎫
1BJ                       ⎪
.                         ⎪
.                         ⎬ RPL
.                         ⎪
EXU                       ⎪
MSG                       ⎭
```

zero word terminator card to terminate RPL(3,7,9 punch in column 1)

```
record separator card     ⎫
007                       ⎪
record separator card     ⎪
1CO                       ⎪
record separator card     ⎪
.                         ⎬ PLD
.                         ⎪
.                         ⎪
TIM                       ⎪
record separator card     ⎪
WBR                       ⎪
record separator card     ⎭
```

zero length record to terminate PLD

record separator card

```
ACGOER                    ⎫
record separator card     ⎪
ATAN2                     ⎪
record separator card     ⎪
BACKSP                    ⎪
record separator card     ⎬ CLD
.                         ⎪
.                         ⎪
.                         ⎪
record separator card     ⎪
SYSTEM                    ⎪
record separator card     ⎪
XLOCF                     ⎭
```

```
record separator card
BKSP
record separator card
CATALOG
record separator card
COPY
record separator card
COPYBF
record separator card
COPYBR
record separator card
COPYCF
record separator card
COPYCR
record separator card
COPYSBF                      CLD (con't)
record separator card
COPYX
record separator card
COSY
record separator card
REWIND
record separator card
UNLOAD
record separator card
VERFIY
record separator card
ASCENT
record separator card
ASCENTR
record separator card
RUN
record separator card
zero length record to terminate CLD
record separator card
file separator card
```

See ASCENT section for preparation of ASCENTR, CMR section for modifying EST.

LIST TAPE

The composite list tape contains the following routines and is written in packed display code:

File 1	ASCENT
File 2	FORTRAN (RUN)
File 3	FORTRAN Object Routines
File 4	Utility
File 5	Central Memory Resident (CMR)
File 6	Peripheral Processor (PPU) Routines
File 7	ASCENTR

6000 → **RELEASE SUMMARIES**

CHIPPEWA OPERATING SYSTEM VERSION 1.1

NEW FEATURES

Version 1.1 of the Chippewa Operating System contains new features and improvements to old features in the areas of 3000 series drivers, OUTPUT control point, roll-out, roll-in, priority assignment, Dayfile dump, deadstart-restart, dump and others.

1. 3000 Series Drivers

The 3000-series drivers allow the use of features of standard 3000 series controllers which are not available with the 6000 series controllers. These drivers also include improved error handling procedures and more easily handled operator intervention.

1.1 New features—All Operations

- Removal of system hang-ups due to equipment malfunctions. See Section 1.6.
- Provision for peripheral controllers to access from one to four channels.
- Improved message handling. All equipment messages are displayed at the fourth line of the control point. Third line of the control point display contains control cards and job messages and diagnostics as before.
- The EST format for 3000 series peripheral equipment is:

CP	BB AA	DD CC	O HH	SEUU

where CP is the control point assignment (12 bits)
AA, BB, CC, DD are channels connected (6 bits each)
O is the on/off bit (1 bit)
HH is the hardware type (11 bits)
S is the 6681 number (3 bits)
E is the 3000 equipment number (3 bits)
UU is the unit number (6 bits)

1.2 New Features—Card Operations

- Console message to indicate that a card which has caused a compare error should be re-read. See 1.5.
- Offsetting of a card (blank) at the beginning of the punching of a logical record.
- Offsetting and repunching of a card on which a compare error has been detected.
- Ability to punch (on one or more cards) any length line of packed display code. A buffer with more than 80 characters of output is punched on successive cards.
- Ability to punch full 80 column binary cards by using the file name PAB for the card punch.
- Trailing blank bytes are deleted when a Hollerith card is read instead of trailing blank characters.
- In order to check card punch compare errors, the OUT pointer lags one card behind the punch.

1.3 New Features—Tape Operations

- Ability to search file mark forward and reverse. The tape is positioned to the other side of the closest end of file mark in the direction specified by the CIO call code.

  CIO code - Forward 030_8
  Reverse 130_8

- Better control over write parity errors. Bad area of tape erased. See 1.5.
- Termination with a file mark of all tapes written in BCD. Any operation requesting backward motion after a write operation causes a file mark to be written prior to the backward operation.
- Provision for unloading at end of tape and continuing the operation on another tape.

  When an end of tape signal is detected after writing a physical record the tape is rewound and unloaded.

  The message MT XX END OF TAPE - FILE NAME is displayed for the operator. If another record is to be written, it will be written on unit xx when that unit again becomes ready. Read and write operations are compatible so that the programmer need not be concerned with end of tape.

- Optional density selection by means of the REQUEST or ASSIGN cards.

  REQUEST filename, XX.

  ASSIGN unit number, filename, XX.

  where XX = LO, HI, HY indicating the density of 200, 556 or 800 bpi. If no density is specified when units are assigned or requested (old control cards are valid), the density set by the operator is used.

- BCD read. Physical records are read until buffer is full or end of file sensed. End of file responds with end of record status. The next BCD read responds with end of file status and tape is not moved. This allows BCD tapes to be read at tape speed.

1.4 Error Procedures Requiring No Operator Intervention

Certain errors are processed by the drivers and require no operator intervention. These are:

- Compare error during punch operation. When a compare error occurs the bad card and the card following it are offset. These two cards should be removed from the deck. The proper cards are punched following the offset cards. The message CP XX COMPARE ERROR is entered in the Dayfile.
- Recoverable write parity error on tape. The message MT XX WPE RECOVERED is entered in the Dayfile.
- If, during a backspace operation, load point is encountered unexpectedly, the block count in FST is reset to zero. This condition occurs when a tape has been manually repositioned.

1.5 Error Procedures Requiring Operator Intervention

The following data transmission errors cause a message to be entered in the Dayfile and a message for the operator which requires action on his part:

- Compare error during card read. Message to the Dayfile: CR XX COMPARE ERROR. Message to the operator: CR XX RE-READ LAST 2 CARDS. The operator should reload the last two cards in the output tray of the reader and type X.GO. where X is the control point.
- Binary Card Error

  Message to the Dayfile: CR XX BINARY CARD ERROR.

  Message to the operator: CR XX RE-READ LAST 2 CARDS.

  The operator should reload the last two cards in the output tray and type X.GO. where X is the control point.
- Card jam during card read. Message to the operator: CR XX REREAD LAST CARD.

  The operator should reload the last card in the output tray and ready the reader. The reading will resume without further action.
- Tape Read Parity Errors

  Message to the Dayfile and operator: MT XX PARITY ERROR. Typing X.GO. (where X is the control point) will cause the read data to be accepted as is and the read operation will proceed. (The block on tape was re-read 10 times before the message was displayed.)
- Unrecoverable write parity errors. Message to Dayfile and operator: MT XX WPE UNRECOVERED. Typing X.GO. will allow the operation to proceed ignoring the error on the tape. (It is possible to change the tape at this point.) The following operations are performed to detect unrecoverable write parity errors:

  a. Backspace record

  b. Rewrite record

  c. If no parity error, exit

  d. Backspace record

e. Rewrite record

f. If no parity error, exit

g. Backspace record

h. Erase bad spot

i. Write record

j. Backspace 2 records

k. Read 1 record

l. If parity error on read, an unrecoverable write parity error is declared.

m. Read 1 record

n. If no parity error, exit

o. If parity error, repeat steps a. through n. once more. Second time through this step causes an unrecoverable parity error to be declared.

1.6 Equipment Malfunction Errors

Equipment malfunction errors may be responded to by dropping the job, changing the equipment code on the controller and/or unit or correcting the condition.

- Equipment not ready. The message EQ XX NOT READY is displayed.
- Equipment cannot be connected. The message EQ XX REJECT is displayed. This error may be caused by (a) controller and/or unit numbers which do not agree with the EST entry (b) equipment malfunction or (c) transmission parity error. If a transmission parity error is indicated at the appropriate controller, the channel should be master cleared to clear the parity error indication by typing MCH XX (where XX is channel number). See section 8.5. After the channel is cleared the operation continues with a reissue of the rejected function.
- No write enable on tape unit. The message MT XX NO WRITE ENABLE is displayed. The appropriate tape with the write ring in may be mounted and the operation will continue.

2. OUTPUT Control Point

The OUTPUT control point provides offline processing for print, punch and other (defined by user) output. Up to six files can be processed simultaneously at the one control point. Storage is assigned based on output devices defined in the Equipment Status Table and equipments are assigned as needed.

2.1 New Features

- Operator control over print and punch.

  a) Reprint file.

  b) Stop print or punch operation.

  c) Suppression of program page control during printing.

- Each punch job is identified by a readable punched card at the beginning of the job.
- Additional file types are defined in FNT.

  The format of the low order byte of FNT is:

I G		E X	F T	C P
1	4	1	3	3

CP control point

FT file type - new types 4=PUNCH, 5=FILM, 7=PLOT.

EX express control point indicator

IG output ignore bit. This bit is not currently being used in this system. It has been specified in anticipation of software to handle remotes.

- Special file names are processed automatically by OUTPUT and must not be assigned to equipment by control card.

OUTPUT directed to the printer

PUNCH directed to the punch - Hollerith format

PUNCHB directed to the punch - Chippewa Binary format

P80C directed to the punch - ASCENT COSY and binary output

2.2 Operator Notes

- The control point display for OUTPUT is as follows:

```
2.   OUTPUT   ,  ,  ,  ,  -2---67-
15100,      6100, 20. 21. 22. 23. 24.
1.   PRINT 2.IDLE 3.PUNCH 4.PRINT 5.PRINT 6.PRINT
JOBNAME    JOB0001    JOB0002    JOB0003    JOB0004
```

Each of the numbers on the third line refer to buffer points. These buffer points are assigned as they are needed. The number of output devices in the Equipment Status Table determines the number of buffer points set up.

Because the assignment is dynamic, each is labeled with the operation is process.

- Control entries. The following entries are used for controlling the OUTPUT operations:

X = CONTROL POINT NUMBER

Y = BUFFER POINT NUMBER

a) X.ENDY. Terminate operation at Y.

b) X.REPEATY. Repeat current operation at Y when complete.

c) X.SUPPRESSY. Suppress page control of printer at Y. If Y = 7 for a and b above, all buffer points will be affected. If Y = 7 for c above all buffer points concerned with printers will be affected. An operation is restarted after an equipment malfunction by typing:

```
X.REPEATY.

X.ENDY.
```

- Punch card operations. At the beginning of the punch-out of each card is punched with a visual image of the job name for ease of identification.
- Printer operation. The print file may cause the printer to stop and a message to appear at the fourth line of the control point. The message appears in the form X. LPnn message, where X is the buffer point number and nn is the printer number. After the action specified by the message is taken, the print operation may be resumed by typing X.GO.

2.3 Programmer Notes

For repeat listings or changing of printer forms, the special pseudo format characters PM will cause the printer to stop and the transmission of up to 30 characters following the format characters to the console message area for operator action.

FORTRAN statements to do this would be as follows:

```
   PRINT 10
10 FORMAT (21HPMREPEAT THIS LISTING)
   PRINT 11
11 FORMAT (29HPMCHANGE FORMS TO CORP. 345-A)
```

3. Roll-Out, Roll-In

This feature permits the operator to request that a job which has control of the central processor at a specified control point be "rolled out" so that all of the memory assigned to the job except its control point area is released. Neither the control point nor the equipment assigned to the job is released. Later, the operator can request that the rolled-out job be restored and its execution resumed.

This action is caused by operator messages

```
X.ROLLOUT.

X.ROLLIN.
```

where X is the control point number.

4. Priority Assignment

The operator can change the priority of any file or control point from the console.

To change the priority of INPUT/OUTPUT files listed in the H-display the operator types the following:

```
ENPR,xx,f,y.
```

where xx is the new priority, and f is the name of the file to be changed. y is I, O, P, indicating input, output, punch file priority is to be changed. If y is not specified only input file priorities may be changed. The file name may be any file in the system (FNT).

The priority of a job at a control point may be changed without using DIS. In this case, the operator types the following:

```
x.ENPR,yy.
```

where x is the control point number, and yy is the new priority.

The time limit of a job at a control point may be changed without using DIS. In this case, the operator types the following:

```
x.ENTL,yyyy.
```

where x is the control point number, and yyyy is the new time limit in octal number of seconds. The time limit in the control point area is reset.

5. Dayfile Dump

This feature permits the operator to request that the Dayfile be dumped and the disk space released, thus shortening search time through the Dayfile. The operator initiates this action by freeing a control point and typing X.DAYFILE. where X is the control point number. The operator will then be requested to assign an equipment on which to dump the Dayfile. Acceptable equipments are tape, printer, punch. The Dayfile will be copied to the equipment in coded format, the disk space will be released, and the pointers reset to the beginning of the new Dayfile. If the dump is to tape, an end of file mark is written on the tape after the dump and the tape is left positioned after the file mark. Jobs currently in the system will not receive a printout of any messages put in the Dayfile prior to the dumping action. This loss of messages may be avoided by allowing the system to idle down before initiating the dump.

6. Deadstart-Restart

The Deadstart-Restart package provides for restarting the system after the occurrence of non-catastrophic hang-ups.

All files except those of type local are rewound and preserved and all jobs at control points are restarted from their beginnings. The operator procedure is:

a) A normal deadstart which loads the system in from tape.

b) When the scope comes up the message RECOVER. is keyed in instead of AUTO.

c) When CP1 blanks type AUTO.

7. Dump

The DMP. option is changed to dump the control point area in addition to the exchange area.

8. General Improvements

8.1 Use of All Available Disks.

The system uses all available disks. The number of disks available to the system is specified in EST by disks being turned ON. Disks which are turned OFF are not used by the system. Thus, if a user requires the use of an entire disk it should remain OFF. Note that this usage of the ON/OFF flag is different between disk and other equipments. OFF for disk restricts the system from using it. In a multiple disk configuration, the system assigns disks in a cyclic fashion to successive files. This scheme greatly reduces the amount of CPU idle time waiting for disk positioning since all disks may be used simultaneously. Although the system will use all disks, which are turned ON, control cards assigning extra disks such as ASSIGN DB,FILE. may still be used. However, the capacity of DB is no longer the full disk because the system may have used parts of it.

8.2 Storage Move

The storage move program in CPRES located in CMR, locations 2000_8 - 2077_8, is improved to reduce the number of hardware conflicts. The new program averages .92 micro-seconds/word versus 1.4 micro-seconds/word used by the old version. This change provides faster CPU throughput since at least two storage moves are required for every job.

8.3 Peripheral Library Loader (PLL)

All PP routines are loaded with PLL. PLL is the general overlay call subroutine in PP resident (PPR) which loads a routine from either RPL or PLD. However, 7DP must stay in RPL in order to handle disk parity errors if they occur without referencing the disk. Therefore, the length of RPL may be reduced to include only 7DP. It should be noted that the pointers at CM locations 000001-000002 must be adjusted when decreasing the size of RPL.

8.4 JOB Card Error

If a job contains a JOB card error or too many control cards, a diagnostic is entered in the Dayfile and reading proceeds to the next job. These errors are diagnosed during the reading of cards; therefore, the only programmer output will be the Dayfile.

8.5 Master Clear

The operator may perform a master clear on a channel.

```
MCHxx.
```

where xx is the channel.

8.6 TIME and DATE Statements

The operator statement TIME only resets the current time.

```
TIME.hh.mm.ss.
```

where hh.mm.ss are the hours, minutes, seconds for 24 hour clock. Only eight characters may be entered.

The operator may set the date by typing up to 10 characters;

```
DATE.cccccccccc.
```

where mm/dd/yy is the usual format for giving month, day, and year.

The date is displayed immediately following the time at the top of the left screen.

The operator cannot change the remaining characters of the date line; these are installation parameters for identifying the system.

The complete line is printed as the last line after the Dayfile on all print output.

8.7 DSD Diagnostics

Three messages or diagnostics can appear above the line of keyboard input.

FORMAT ERROR - the keyboard type-in contains a formatting error.

ILLEGAL ENTRY - the keyboard input is totally unrecognizable to the system.

WAIT MTR FCN XX - the system is in step mode (or MTR is busy) and DSD is waiting for MTR when processing the keyboard input. XX is the number of the monitor function.

8.8 H-Display

Punch and common files, in addition to input and output files are displayed by the H-display in DSD.

8.9 Read Buffer Parameter

The Read Buffer Parameter routine (2BP) has been modified to produce the messages:

```
FILE NAME ERROR xxxxxx
BUFFER ARG ERROR xxxxxx
```

where xxxxxx is the location at which the error occurred.

It has also been modified so that if space is not available in FNT it waits for space instead of aborting the job.

8.10 Monitor (MTR)

Monitor (MTR) has been modified to add the new monitor function 26-reset dayfile pointers.

8.11 2EF

2EF has been modified to give an exchange package dump when an arithmetic error occurs. The messages "ARITH ERROR" and "MODE = x. ADDRESS = yyyyyy" apear in the Dayfile, where x is the exit mode and yyyyyy is the address at which the error occurred.

9. Operating Aids and Restrictions

9.1 Note on Fitting Large Programs into Memory

If a program is too large to fit into memory during normal system operation the following procedure will maximize memory available for loading:

1. Empty all control points by DROPping everything.
2. 1.READ. get all control cards to the disk
3. 1.DROP.
4. 1.NEXT. initiate execution
5. 1.DROP. after execution terminates
6. AUTO. continue normal operations.

9.2 Note on Assignment of P80C

The I/O within ASCENT conflicts with the punch driver. Hence, P80C must not be assigned to the punch. Instead it should go to the disk where OUTPUT will pick it up and punch it.

9.3 Rollout Restriction

An attempt to roll out a program that has not yet executed in central memory will hang up the control point, and eventually, the system. Once a program has had control in central memory it may be rolled out regardless of its status. The system I/O packages such as PRINT and READ cannot be rolled out.

VERIFY ROUTINE

The Verify Routine has been rewritten. This CP routine is used to compare the contents of one binary file against the contents of another binary file. Appropriate comments regarding a successful or unsuccessful comparison are provided.

VERIFY is called by a control card with the following format:

VERIFY (P_1, P_2, P_3, P_4)

where P_1 and P_2 are the two files to be compared.

P_3 is the number of comment lines of type E to be printed in event of an unsuccessful comparison.

P_4 is the destination file for the comments (normally assigned to OUTPUT)

The parameters from right to left may be truncated at any point and the parameters which are assembled in the routine will be used. The values are

P_1 = FILE1

P_2 = FILE2

P_3 = 1

P_4 = OUTPUT

The following diagnostics are obtainable from VERIFY:

1. "VERIFY OK"

   This comment indicates a successful comparison of the two files. This message is also placed in Dayfile.

2. "VERIFY FAILURE"

   This comment which is also placed in Dayfile indicates a failure to compare for one of the following reasons.

   - The number of logical records in one file does not agree with the number of logical records in the other.
   - The number of words in a record of one file does not agree with the number of words in the corresponding record of the other file.
   - A word in a record does not compare with the corresponding word in the same record of the other file.

3. "n(10) m(8) EXCESS RECORDS IN FILE (f)"

   This comment indicates that there were n more records in the named file, f, than in the other file.

4. "n(10) m(8) EXCESS WORDS IN RECORD p(10) q(8) OF FILE (f)"

This comment indicates that record p of file f contains n words more than the corresponding record of the other file.

5. nnnnnn xxxxxxxxxxxxxxxxxxxx yyyyyyyyyyyyyyyyyyyy

This comment is printed for each pair of words which are not in agreement. The location n is octal. Each line is identified by a record number.

The field length required to verify two files is approximately 4000_8. Since the buffers are used in a circular fashion, no additional allowance is needed for large records.

PPU ROUTINES - INTERNAL DIFFERENCES

The following lists the differences between version 1.0 and version 1.1 PPU routines.

BOOT 3X2X

Unchanged from Version 1.0.

CHK CHECK

Unchanged from Version 1.0.

CIO INPUT/OUTPUT

Rewritten. Search file function for 3X2X tapes has been added.

CLL CENTRAL LIBRARY LOADER

Modified to incorporate handling of files on more than one disk.

CLS LOAD CENTRAL SEGMENT

Modified to incorporate handling of files on more than one disk.

DIS JOB DISPLAY

Allows an extra line in the control point display.

DMP DUMP

Modified to incorporate dump of control point area.

DSD SYSTEM DISPLAY

Modified to incorporate new console requests. Also modified so that the scope will not black out under Step Mode with the exception that AUTO, RECOVER, ENPR will still cause the blackout.

EXU EXECUTE

Modified to incorporate handling of files on more than one disk.

HLP HELP

Unchanged from Version 1.0.

LBC LOAD BINARY CORRECTIONS

Modified to incorporate handling of files on more than one disk.

LDR 3X2X LOADER/PP RESIDENT

Modified for deadstart recovery.

LDR 6000 LOADER/PP RESIDENT

Modified for deadstart recovery.

LOC LOAD OCTAL CORRECTIONS

Modified to incorporate handling of files on more than one disk.

MSG MESSAGE

Unchanged from Version 1.0.

MTR MONITOR

Modified to add the new monitor function 26 - reset dayfile pointers.

The format for the function 26 request is:

26	BT	CT	CS	ST
60				0

26	function
BT	beginning track
CT	current track
CS	current sector
ST	spare track

PAS 3000 PERIPHERAL ASSEMBLER

Not included in Version 1.1.

PAS 6000 PERIPHERAL ASSEMBLER

Not included in Version 1.1.

PBC 3446 PUNCH BINARY CARDS

Modified to use the overlay 2PC

PBC 6000

Not included in Version 1.1.

PBS 3446

Modified to recognize the new EST format.

PBS 6000

Not included in Version 1.1.

RBR	READ BINARY RECORD
	Unchanged from Version 1.0.
RFL	REQUEST FIELD LENGTH
	Unchanged from Version 1.0.
SOS	Unchanged from Version 1.0.
TIM	Unchanged from Version 1.0.
WBR	WRITE BINARY RECORD
	Unchanged from Version 1.0.
007	SIMULATOR
	Unchanged from Version 1.0.
1AJ	ADVANCE JOB
	Rewritten to use overlay 2CF.
1BJ	BEGIN JOB
	Unchanged from Version 1.0.
1DJ	3256 DUMP JOB
	Not included in Version 1.1. (See OUTPUT)
1DJ	6000 DUMP JOB
	Not included in Version 1.1. (See OUTPUT)
1DS	ASSIGN DIS
	Unchanged from Version 1.0.
1LJ	3447 LOAD JOB
	Modified for improved equipment handling.
1LJ	6000 LOAD JOB
	Not included in Version 1.1.
1LT	LOAD TAPE
	Modified to distinguish between 3X2X and 6000 tapes.
1PD	PRINT DAYFILE
	Not included in Version 1.1. (See OUTPUT)
1TD	TAPE DUMP
	Modified to distinguish between 3X2X and 6000 tapes.
2BD	BACKSPACE DISK
	Unchanged from Version 1.0.

2BP READ BUFFER PARAMETER

Rewritten to accommodate files on more than one disk.

Diagnostics added:

FILE NAME ERROR address.

BUFFER ARG ERROR address.

If space is not available in FNT the routine waits for space instead of aborting the job.

2BT BACKSPACE TAPE

Modified to handle 6000 tapes only. (See 2TB)

2DF DROP FILES

Unchanged from Version 1.0.

2DJ 3000

Not included in Version 1.1. (See OUTPUT)

2DJ 6000

Not included in Version 1.1.

2DT DROP TRACK

Unchanged from Version 1.0.

2EF ERROR FLAG

Rewritten. This routine gives an exchange package dump when an arithmetic error occurs. The messages "ARITH ERROR" and "MODE=X. ADDRESS=yyyyyy" appear in the Dayfile.

2LP 3256 PRINT

Rewritten as part of improved 3000 drivers.

2LP 6000 PRINT

Not included in Version 1.1.

2PC 3446 PUNCH CARDS

Rewritten as part of improved 3000 drivers.

2PC 6000 PUNCH CARDS

Not included in Version 1.1.

2RC 3447 READ CARDS

Rewritten as part of improved 3000 drivers.

2RC 6000 READ CARDS

Not included in Version 1.1.

2RD	READ DISK Unchanged from Version 1.0.
2RT	READ TAPE Modified to handle 6000 tapes only. (See 2TR)
2SD	SEARCH DAYFILE Unchanged from Version 1.0.
2TJ	TRANSLATE JOB CARD Rewritten so that JOB card must adhere to specifications in manual.
2TS	TRANSLATE CONTROL STATEMENT Changed to allow density on REQUEST and ASSIGN cards.
2WT	WRITE TAPE Modified to handle 6000 tapes only. (See 2TW)
2WD	WRITE DISK Unchanged from Version 1.0.
7DP	DISK PARITY Unchanged from Version 1.0.

The following PPU routines are new in Version 1.1.

Output Control Point

1OT	Output control
3OT	Printer control
1PO	Punch output control
2CF	Complete files
1FM	Dummy FILM output (aborts job if file of type FILM is encountered)
1PL	Dummy PLOT output (aborts job if file of type PLOT is encountered)
1CO	Prints DAYFILE, completes OUTPUT.

Tape Drivers

2TB	3x2x tape function driver
2TR	3x2x tape read driver
2TW	3x2x tape write driver
7TP	3x2x tape write parity error handler

Dayfile

1DF	Dayfile dump

Roll-In/Roll-Out

1RI	Roll-in
1RO	Roll-out

Deadstart Recovery

1RF	Deadstart recovery

FORTRAN Compilation

CPL	Load central program

KNOWN DEFICIENCIES

1. X.REPEATY. in OUTPUT is not working.

2. Deadstart recovery causes sequencing of jobs to start from zero again. Hence, if jobs are reloaded into the system after deadstart recovery in the same sequence as they are loaded prior to recovery, two Dayfiles will appear with each job.

3. Deadstart recovery is not reliable 100% of the time. It is advisable after recovery to allow the system to idle down and then deadstart before entering new jobs into the system.

4. Printer format control is occasionally suppressed for no apparent reason.

5. If an invalid TIME is entered the scope is disturbed.

6. Occasionally the Dayfile for a job does not come out properly.

7. Too many control cards produce the proper message but destroy the system.

8. Some job card errors cause the system to hang up.

9. If a unit referenced in a FORTRAN WRITE statement is assigned to the card punch no cards are punched.

10. Two FORTRAN BACKSPACE statements in a row apparently backspace only one record if the unit is 1/2" tape.

11. The decimal record number in the error printout from VERIFY is incorrect. The octal record number is correct.

12. PSR's through #33 have been processed and the following have not been resolved at the time of this release: 15,19,20,21,26,27,28,29.

13. ASSIGN 01 after a REQUEST TAPE 6 is ignored.

14. After recovery, punch files are lost.

15. Transmission parity errors on tape controllers are not always cleared out by typing "MCH12".

16. When messages on the "A" display are too long, the remainder of the message is overprinted on the following line.

17. OUTPUT should not be brought to a control point that is beyond one at which a job is executing.

Example:

Control point	Activity	
1.	READ	
2.	job executing	
3.	OUTPUT	OUTPUT will not execute properly at any of control points 3 through 7.

ASCENT
VERSION 1.1

NEW FEATURES

1. An ASPER or ASCENT pseudo-op card must be the first card (other than comments) in the program. In any other position, such a card is treated as containing an indecipherable operation code. If the first non-comment card of a program is neither ASCENT nor ASPER, an ASCENT program is assumed. (This does not represent a change.)

   Formerly, if the END card terminating a program was forgotten, the assembler might switch from ASCENT to ASPER or vice versa before completing the processing of a program, and produce chaos and/or a hangup. Now, incorrect assembly is produced up to the next END card, but correct assembly continues thereafter.

2. A dollar sign now terminates the scanning of a card (like a period). Formerly, two or more instructions could be punched on one card, with dollar signs separating them. This facility has been discontinued, and dollar signs should not appear in program cards except in comments, or in BCD or DPC fields.

3. At the end of each assembly listing, there is now a count of the number of lines that have error flags.

4. A Z flag now appears on every line whose address is not less than 4096 (10000B) in ASPER, or the equivalent of "AS11" for ASCENT. As11 is now equated to 65535 (177777B).

5. Assembly options may now be bypassed by using consecutive commas on the ASCENT control card. Formerly, it was necessary to place one and only one zero in the field for the option desired bypassed.

6. Formerly, if COSY input was to be read from a file other than the INPUT file, the file name (as designated in the sixth parameter of the ASCENT control card) had to be COSY. Now, any file name may be used except system file names or SCR (used by ASCENT for the scratch file).

7. The program name will be placed in the listing heading starting on the second page, providing the program begins with an ASCENT or ASPER pseudo-op containing the program name. In addition the header line on every page says ASCENT 1.1.

8. The previous version of ASCENT had two unused parameters: MOD (parameter 9) and OUT (parameter 13). These have been eliminated in Version 1.1. The only effect this will have on users is that the CLASS flag is now parameter 12, and the PAS flag is parameter 13. These flags now occupy core locations 15 and 16, respectively.

9. If the assembler requires a longer field length than that specified on the JOB card, the job will be aborted with the Dayfile message:

NOT ENOUGH FIELD LENGTH. .NEED NNNNNNB. (NNNNNNB is the required field length.) In previous versions, the result was an arithmetic error.

10. The COSY deck provided, when assembled with no modifications, produces the smaller version of the assembler, ASCENT. To produce the larger version, ASCENTR, the following modifications must be made:

```
            REPLACE     72
            VFD         D42/ASCENTR, A18/ENDEND
            REPLACE     91
    AS25    EQU         6000
```

When generating BCD list tapes from the COSY tapes, the assembly should be divided into two separate jobs, since the listing will not fit on one 2400 foot reel of one-half inch magnetic tape.

CORRECTIONS

1. In ASCENT, if an opcode is not a pseudo-op or defined macro, then if its second character is A, B, or X its third must be 0 - 7 and its fourth must be blank. Otherwise, an O flag occurs.

   Formerly, an opcode like BXI could slip through and be assembled as BX0 without an error flag.

2. In ASCENT, an instruction like

   SA1 A2 + B3

   will be assembled as SA1 A2+B3. Formerly, the blank after the operator terminated the scan and caused an error. It is still true that two consecutive blanks anywhere after the beginning of the operand field terminate the scan.

3. In ASCENT, if the operand to be inserted by the assembler into the K field of a 30-bit instruction has an absolute value $\geq 2^{18}$, an R flag is put in the listing.

4. If an EQU pseudo-op defines a symbol in terms of a symbol that has not been defined earlier in the program, there will be a U flag. However, the symbol in the label field of that EQU, though not defined, will not cause further U flags if it appears in the operand fields of other cards. Formerly, such an error was not flagged, and the symbol which EQU failed to define was treated as =0. Note that if an EQU pseudo-op is error-flagged for any other reason, it will get a U flag in addition to the proper flag.

5. The label field of every card is much more strictly examined.

   a) If column 1 contains a C, the card is a comment and is not further processed.

   b) If column 1 is neither C nor blank, the card is treated as if column 1 were blank, but an L flag appears in the listing.

   c) If the first non-blank beginning from column 2 is a period, the card is a comment and is not further processed.

d) If the card is not a comment, column 10 must be blank. If not, the card is treated as if column 10 were blank, and an L flag appears in the listing.

e) If the card is not a comment, columns 2-9 (the label field) must contain one of the following:

- All blanks
- A plus or minus sign and no other non-blanks (forcing or anti-forcing)
- A series of 1 to 7 characters, the first of which is a letter, and the rest of which are letters or digits. There may be blanks at either end, but no embedded blanks or special characters. In ASCENT the label may not be the name of a register like A1 or B3. In an ASPER program such labels are acceptable.

Otherwise, an L flag appears in the listing. If the only fault is that the label is too long (8 characters), it is put into the symbol table with its 8th character cut off. Whenever another instruction references such an 8-character symbol, an S flag appears in the listing, and the assembler ignores the 8th character. Any other fault in the label field causes the assembler to treat it as a blank label.

6. If an ASPER instruction attempts to address a literal, there is a U flag. Formerly, there was no error indication.

7. Formerly, negative and complex literals did not work at all, and it was probably impossible to get more than one positive literal into a program. Now all the allowed types of literals are assembled properly.

8. When two numbers separated by a comma comprise a literal, they must be single-precision floating-point numbers. Formerly, no explicit check was made for this.

   If a comma occurs within a literal, there is now an F flag if either of the numbers before and after the comma is not a single precision floating-point number. However, if the only error is that one or both of the numbers is double precision, the assembler will use it as if it were single precision, though putting the F flag in the listing.

9. Where parentheses enclose a pair of numbers on a CON card, showing that they are intended to form a complex constant, the same checks are now made as for a complex literal. Formerly this was not so.

10. An integer coded as octal (e.g., terminated by B) but containing 8 or 9 as a digit produces an F flag. Formerly, no error indication was given.

11. In a BCD or DPC pseudo-op card, the operand field must begin either with two decimal digits or with an asterisk. If neither of these conditions is satisfied, there is now an F flag, and the first ten characters of the operand are converted. Formerly, there was no error indication and an unpredictable number of characters was assembled.

12. In floating point constants and literals, an unsigned exponent is treated as positive. Formerly an unsigned exponent hung up the program at location 6.

13. The LIST pseudo-op now works properly. Formerly, it had unpredictable effects. Any line with an error flag is now listed even though it was intended to be suppressed by LIST.

14. The following errors in VFD processing have been corrected:

    a) VFD N60/7 gave an error flag instead of working properly.

    b) VFD D6/(,N42/0,D12/) hung up the program, which was trying to make a literal out of the material between parenthesis.

    This VFD is still an error, because a D-field must not contain any special characters. But it no longer hangs up the program.

    c) A field like N24/X did not produce an error flag, though only integers are allowed in N-fields. Now it gives a V flag.

15. A card with zero word count is no longer added at the end of a Chippewa binary deck containing an exact multiple of 15 words.

16. End-of-record cards are no longer punched except between decks.

6400/6600 FORTRAN VERSION 1.1

Version 1.1 of 64/6600 FORTRAN is an improvement of Version 1.0, correcting deficiencies in the former system and adding some new features. Object time input-output routines created at Kirtland Air Force Base and some object time mathematical function routines written at CERN replace the corresponding routines of Version 1.0.

NEW FEATURES

RUN Compiler

The following additions have been made to the compiler:

1. Expanded diagnostics—each diagnostic message now consists of a descriptive phrase or sentence in addition to the two-character identifier of version 1.0. The message appears on the listing following the statement in error. A list of the messages is given in the section on documentation changes.

2. Alphabetized variable map—the contents of the map have been reordered to present the information in alphabetical order by variable name. The position on the output listing remains the same as in version 1.0.

3. Ordered subroutine assignments—the format and position of the output remains the same as in version 1.0 but the names are reordered by entry point location.

4. Cross reference maps—under L or S mode, the following appears on the output whenever the ninth parameter on the RUN card is not blank or zero:

   a. Arithmetic statement function locations and references

   b. Statement number locations and references

   c. Variable locations and references. Title lines always appear even if there are no references.

5. Double precision constant conversion—the compiler converts double precision constants into proper double precision representation discarding any digits which do not contribute to significance. The constant need not contain a decimal point; the form nD±S is acceptable.

6. Real constant form—the compiler accepts and converts a real constant without a decimal point; the form nE±S is acceptable.

7. Multiple replacement statements—the compiler processes statements of the form V1 = V2 = . . .=VN = e where the V's are variable names and e is an expression.

8. An array reference in an EQUIVALENCE statement may contain up to 3 constant subscripts.

9. If the symbolic name of a built-in function is declared EXTERNAL, use of that name as a function reference generates coding to call the external function rather than to evaluate the expression in-line.

10. Working Space (temporary locations and locations of variables) is now allocated after all library routines have been loaded. The object code overlays a major portion of the compiler beginning at RA + 2000B. This results in a smaller FL for programs with large amounts of working storage. The minimum FL required for compilation is now 35000B.

Object Routines

Version 1.1 contains new input-output routines and some new single precision mathematical function routines. These are replacements for the corresponding routines of the same names in version 1.0. Write-ups for the routines are found in the section on documentation changes.

The routines are:

ALOG, ALOG 10

ATAN, ATAN2

BUFFEI

BUFFEO

COS,SIN

EXP

IOCHEK

INPUTB

INPUTC

INPUTS

LENGTH

OUTPUTB

OUTPUTC

OUTPUTS

SQRT

CORRECTIONS

Run Compiler

The compiler has been updated to correct all known deficiencies except those noted in this bulletin. Corrective code has been generated to fix the problems reported in PSR 11,45,48,54,62,69,73,74,76,77, 79,81,83,87,89,91,92,96,97,102,103, and 105.

The following incorrect code is diagnosed by the compiler:

1. READ INPUT TAPE 2,2, (XOPER (I), . =1,8)—the point is detected and flagged.

2. I = 1. (X+40.7)—the missing operation is detected and flagged.

3. Illegal transfers to DO terminators, doubly defined statement numbers, and improperly nested DO's are flagged.

The following correct code is handled properly by the compiler:

1. CALL X (. TRUE. , .FALSE.)—logical constants are allowed as parameters to subroutine calls or external function references.

2. DO101 = 40.02 is compiled into a replacement statement instead of being flagged as an improper DO.

3. 012967=20.7—recognizes the replacement variable name is not an octal constant.

4. END (20) = 40.7—END,PAUSE, STOP are allowed as array names.

5. CALL X (R (2), T) where T is EXTERNAL—an EXTERNAL name may follow an array element reference in a parameter list.

6. X = (G**(2*L+2)) / (H ** (2* L+2))—the common exponent is properly saved and restored.

7. C1 = CONJG (C2) where C1, C2, are complex—complex conjugates of complex numbers are now created properly.

8. Handling of Hollerith fields has been modified to handle special characters such as "$", "(", ")", ".", and "," where they occur in an IF, DATA, or replacement statement.

Coding is now generated to:

1. Cause a fatal error for a negative valued index in a computed GO TO.

2. Allow proper traceback from a subroutine whose name was passed as a parameter.

A check is made to insure that RFL has assigned specified object field length before execution is begun.

Object Routines

Corrections have been made to BACKSP, CEXP, CHAIN, DBAIEX, DISPLA, ENDFIL, IDINT, IFENDF, IOCHEC, RANF, RBAIEX, REWINM, END, EXIT, and STOP.

The new or corrected routines provide fixes for PSR 28,29,49,64,86,88,93,94,95,100,101,104,106,107, 108,111,115, and 121 on FORTRAN and for PSR 13,14, and 18 on the operating system.

KNOWN DEFICIENCIES

Run Compiler

1. An in-line function name may not appear in an EXTERNAL statement and be passed as a parameter. If a FORTRAN II library function name appears in an EXTERNAL statement, it cannot be used as a parameter although its corresponding FORTRAN IV name may be. If a FORTRAN IV library function name appears in an EXTERNAL statement the corresponding FORTRAN II name cannot be used as a parameter.

2. If a function name appears in a type statement, it must also appear in a preceding EXTERNAL statement, even though it is not passed as a parameter.

3. If a symbolic name appears in an EXTERNAL statement in some subprogram, it may not be used as a local variable in a subsequent subprogram if it is to be passed as a parameter. For example,

```
SUBROUTINE  S1
EXTERNAL    X
     .
     .
     .
END
SUBROUTINE  S2
CALL        S3    (X)
     .
     .
     .
END
```

   causes the X in the parameter list in S2 to be incorrectly identified with the EXTERNAL X in S1.

4. If a symbolic name appears as a function name in some subprogram, it may not be given type in a subsequent subprogram and used as a local variable. For example,

```
SUBROUTINE  A1
Y = FN    (X)
     .
     .
     .
END
```

```
SUBROUTINE   A2
INTEGER  FN
      .
      .
      .
END
```

causes the FN in A2 to be incorrectly identified with the function name FN in A1.

5. The real and double precision constant forms without a decimal point are not allowed in DATA statements. Hollerith constants of the forms (nHf$_1$. . .f$_n$), (+nHf$_1$ . . .f$_n$), and +nHf$_1$ . . .f$_n$ are not allowed in DATA statements.

6. If attempt is made in a DATA statement to store excess values in an array through implied DO-looping, a DATA RANGE diagnostic is given; if the short list notation, i.e., the array name only, is used, excess values are discarded and no diagnostic is given.

7. If the sum of the storage required for a COMMON declaration exceeds 2^{17}, the error goes unnoticed.

8. In-line multiplication of complex or double precision quantities may generate infinite partial results even though the final results would be in range.

9. If mixed double precision and integer arithmetic involving the loop variable occurs within a simple DO-loop, an infinite loop ensues. As an example,

```
       DO  1    K=1,5
   1   D(K)=   1.0 D*K
```

causes a loop. As a temporary expedient, changing the code to

```
       DO 1    K=1,5
       XKAY = K
   1   D(K) =    1.0 D*XKAY
```

removes the difficulty.

10. Attempt is made to avoid recalculation of common subexpressions in arithmetic replacement statements in simple DO-loops. This value is not always saved and restored properly. (PSR 118)

11. The logical constants, .TRUE. and .FALSE., cannot yet be used as parameters in an arithmetic statement function reference.

12. The EQUIVALENCE problem given in PSR 13 has not yet been fixed. The example was:

```
       EQUIVALENCE  (A,I)
       J=1  $  I=0
   5   A=OR  (I,AND  (IX,63) )
       IY  (J+1)=I
```

The compiler does not note the equivalence but does note that I is still in a register at the time the last statement is processed. This results in the old value being stored in the array. Giving a statement number to the last statement will result in correct execution.

13. Appearance of a non-dimensioned variable with subscript to the left of an equal sign causes it to become an arithmetic statement function definition. Appearance of the same name in the expression to the right of the equal sign will cause an infinite loop during execution if the name is ever "referenced". (PSR 116)

14. Within a FUNCTION subprogram, the function name may not be passed as a parameter in a function reference.

15. Addresses for variable or statement number references in the cross-reference maps for a subprogram are incorrect. For a given subprogram, all references are biased by the same amount. A fix for this will be given in an early PSR summary.

16. Some misformed statements are not diagnosed by the compiler and execution can be initiated without warning. Further diagnostic checking by the compiler is being implemented and corrective code will be distributed. Examples of statements for which no compiler diagnostic is given are:

ASF (X) = ASF (X) -1	(arithmetic statement function which calls itself)
READ, A	(no FORMAT statement number)
IF (SENSE SWITCH 48) 1,2	(only the first digit of the switch number is checked)
DECODE (0,10,V)L	(values of parameters are not checked to see that they are in a reasonable range)

17. Although an arithmetic IF statement used as a DO terminator is diagnosed as an error, the FORTRAN II machine condition tests, e.g., IF (SENSE SWITCH i), are not.

18. If the statement immediately following a PROGRAM statement is mispunched so that it has a continuation mark in column 6, it is ignored.

19. Use of a FORMAT statement number in a control statement, e.g., GO TO n, is not diagnosed.

20. The statements

    IF(IOCHEK,i)n_1,n_2

    IF(UNIT,i)n_1,n_2,n_3,n_4

    do not detect parity errors.

21. The statement

    BUFFER IN (u,p) (A,B)

    where u is a 1/2" tape and p=0 (even parity) causes an entire file to be read for each BUFFER IN.

22. BUFFER IN (u,p) (A,B)

    where u is a 1/2" tape and p=0 (even parity) causes no data to be transferred to the list.

Object Routines

1. CABS (Z) and CSQRT (Z), where $Z = X + iY$, compute $X^2 + Y^2$ during the evaluation of the functions. ARITH ERROR occurs when the sum of squares exceeds floating point capacity even though the result would be within range.

2. DMOD (D,D1), gives incorrect results when the arguments have unlike signs. For example, DMOD (-4050.D, 16.D) is computed as 14.D rather than -2.D.

3. CLOG gives incorrect real part for arguments whose real part is large.

4. Results of TANH are not always normalized.

5. EXP gives ARITH ERROR for large arguments even though the result should be within machine capacity.

6. Most of the double precision functions assume a properly aligned double precision argument is being passed and no check is made to ensure this. Incorrect results can ensue.

7. There is inadequate checking for infinite and indefinite arguments in many of the mathematical function routines.

8. If the data and format lists for an ENCODE or DECODE statement require more characters than specified by the first parameter, no execution time diagnostic is given. The specified number of characters is processed and a normal exit is taken.

9. A non-zero floating point number with zero exponent is still output as zero under the real or double precision conversions.

10. CBAIEX destroys the contents of B7. (PSR 127)

11. DMOD does a divide but contains no zero or overflow test on the arguments.

Problems reported on the following PSR's are not fixed: 13,43,85,109,110,113,116,118,119,122,123,125, 126,127,129.

Three magnetic tape units are required for the work files. Additional tape units are required if MASTER data tapes are being used or created.

The binary tape must be loaded in two parts or the diagnostic "TOO MANY CONTROL CARDS" will be displayed and the job aborted.

6400/6600 CDCKWIC VERSION 1.1

6400/6600 CDCKWIC Version 1.1 is now available. The release consists of:

A 1/2" magnetic tape containing the FORTRAN source program;

A card deck used to compile the program and execute a standard test case; and

An Applications Development Bulletin, Publication Number 60137700, describing CDCKWIC.

Installation Procedure

Mount the magnetic tape, and place the card deck in the card reader. Assign the tape to TAPE 7. CDCKWIC will be compiled, and will execute the enclosed test case. A copy of the binary program of CDCKWIC will be placed in the COMMON file on the disk.

Verification Description and Sample Output

Pages 2 and 3 of the output from the sample test case are listed on pages 13 and 14 of the Applications Development Bulletin.

NEW FEATURES

The size of T has been changed from 12000 to 4000 to ensure that CDCKWIC will fit into any size 6400/6600. The field length required is 57200_8 locations.

Subroutine SCAN has been rewritten, enabling Version 1.1 to run approximately twice as fast as Version 1.0.

CORRECTIONS

Several errors which occurred in Version 1.0 when large files were processed have been corrected in Version 1.1.

6400/6600 MATRIX ALGEBRA SUBROUTINES VERSION 1.0

6400/6600 Matrix Algebra Subroutines Version 1.0 is now available. The release consists of the following items:

A card deck containing the source program in FORTRAN and ASCENTF source language. This deck contains approximately 1000 cards;

6000 series Matrix Algebra Subroutines Reference Manual, Publication Number 60135200; and

A verification program card deck. This deck contains approximately 25 cards.

Verification Description and Sample Output

The verification program defines a matrix and calls the Matrix Algebra Subroutines to transpose it. The program then checks the transposed matrix for errors. If no errors are found, the message MATRIX ALGEBRA SUBROUTINES VALIDATION IS OK will be output.

The verification program card deck is one job, containing a FORTRAN compilation and execution. The Matrix Algebra Subroutines source deck should be inserted between the END and file separator cards.

PERT TIME
VERSION 1.0

64/6600 PERT TIME is available. The release consists of:

A tape containing the compiled binary and the source file of the program;

Program decks consisting of the control cards required to load the tape to disk; and

One sample test case program for execution and output listing of the test case.

The source is accessible by bypassing two (2) file separators beyond the binary records.

PERT TIME Version 1.0 requires either Version 1.0 or 1.1 of Chippewa Operating System.

The following documents are available for 6000 PERT TIME Version 1.0:

64/6600 PERT TIME Reference Manual, Pub. No. 60133600

64/6600 PERT TIME Internal Maintenance Specifications

Sales Technical Memorandum - 64/6600 PERT TIME

OPERATING INSTRUCTIONS

To load binary program from tape to disk common file:

```
PERT,1,500,120000.
REQUEST TAPE5.
COPYBR (TAPE5,PERT66)
COPYBR (TAPE5,SEG2)
COPYBR (TAPE5,SEG3A)
COPYBR (TAPE5,SEG3B)
COPYBR (TAPE5,SEG4A)
COPYBR (TAPE5,SEG4B)
COPYBR (TAPE5,SEG5)
COPYBR (TAPE5,SEG6)
COPYBR (TAPE5,SEG7)
COPYBR (TAPE5,SEG8)
COMMON PERT66.
COMMON SEG2.
COMMON SEG3A.
COMMON SEG3B.
COMMON SEG4A.
COMMON SEG4B.
COMMON SEG5.
COMMON SEG6.
COMMON SEG7.
COMMON SEG8.
REWIND (PERT66)
REWIND (SEG2)
REWIND (SEG3A)
REWIND (SEG3B)
REWIND (SEG4A)
REWIND (SEG4B)
REWIND (SEG5)
REWIND (SEG6)
REWIND (SEG7)
REWIND (SEG8)

R/S *
EOF *
```

R/S = RECORD SEPARATOR
EOF = END OF FILE

```
PERT2,1,500,120000.
REQUEST TAPE5.
COPYBR (TAPE5,SEG9)
COPYBR (TAPE5,SEG10)
COPYBR (TAPE5,SEG10B)
COPYBR (TAPE5,SEG11)
COPYBR (TAPE5,SEG12)
COPYBR (TAPE5,SEG13)
COPYBR (TAPE5,SEG14)
COPYBR (TAPE5,REPTGEN)
COMMON SEG9.
COMMON SEG10.
COMMON SEG10B.
COMMON SEG11.
COMMON SEG12.
COMMON SEG13.
COMMON SEG14.
COMMON REPTGEN.
REWIND (SEG9)
REWIND (SEG10)
REWIND (SEG10B)
REWIND (SEG11)
REWIND (SEG12)
REWIND (SEG13)
REWIND (SEG14)
REWIND (REPTGEN)
R/S
EOF
```

To list or punch source:

```
PERT,1,500,50000.
REQUEST TAPE5.
COPYBR (TAPE5,INPUT,18)
COPYBF (TAPE5,X)
COPYBF (TAPE5,X)
COPYBF (TAPE5, "operator")
R/S
EOF
```

The "operator" may be:

a) PUNCH to punch the source deck
b) OUTPUT to list the source file

To execute the program after it has been loaded and commoned, the following control cards are required:

```
PERT,1,5000,120000.
COMMON PERT66.
COMMON SEG2.
     .       .
     .       .
     .       .
COMMON SEG14.
COMMON REPTGEN.
REWIND (PERT66)
REWIND (SEG2)
     .       .
     .       .
     .       .
REWIND (SEG14)
REWIND (REPTGEN)
PERT66.
R/S

    PERT DATA DECK

R/S
EOF
```

To make and save MASTER file, the following change must be made in the control card configuration:

```
REWIND (REPTGEN)
REQUEST TAPE6.          added to deck
REWIND (TAPE6)          added to deck
PERT66.
R/S

    PERT DATA DECK

R/S
EOF
```

The MASTER tape will be created on TAPE6.

To update old MASTER and save a new MASTER the following change must be made in the control card configuration:

```
REWIND (REPTGEN)
REQUEST TAPE4.          add to deck
REQUEST TAPE6.          add to deck
REWIND (TAPE4)          add to deck
REWIND (TAPE6)          add to deck
PERT66.
R/S

    PERT DATA DECK

R/S
EOF
```

The old MASTER will be mounted on TAPE4 and the new MASTER created on TAPE6.

For larger test cases in the amount of 1000 activities or greater, CPU time may be reduced by assigning TAPE1, TAPE2, and TAPE3 to the 1/2" magnetic tape units by this method:

```
REWIND (REPTGEN)
REQUEST TAPE1.          add to deck
REQUEST TAPE2.          add to deck
REQUEST TAPE3.          add to deck
REWIND TAPE1.           add to deck
REWIND TAPE2.           add to deck
REWIND TAPE3.           add to deck
PERT66.
R/S

    PERT DATA DECK

R/S
EOF
```

The system is not intended to be compiled and executed directly from the source (second) tape file; however, this can be done by the following procedure.

1. Read in the first file off the tape with the deck, assigning the PERT TIME tape when requested:

```
ACOPY, 17,500,70000.
REQUEST TAPE5.
COPYBR (TAPE5,TAPE 1, 18)
COPYBF (TAPE5, TAPE 1)
COPYBF (TAPE5, TAPE 1)
R/S
EOF
```

2. When reading has stopped type in:

```
1.  DROP.
1.  LOAD.
1.  ASSIGN (PERT TIME tape).
```

3. When reading of the second file is completed and compilation has begun, one control point, e.g., 3, will read:

```
3.  RUN (B, , , , ,99999)
```

type in:

```
3.  DIS
```

4. While compilation continues type in the names of the eighteen segments:

```
ENS.COMMONPERT66.
RNS.
ENS.COMMONSEG 2.
RNS.
.              .
.              .
.              .
.              .
.              .
.              .
.              .
ENS.COMMONSEG 14.
RNS.
ENS.COMMONREPTGEN.
RNS.
```

Then:

```
DROP.
```

PERT TIME is not ready for job execution. The results of this compilation should not be expected to verify against the first file of the PERT TIME tape which was compiled under Chippewa 1.0.

Samples of the Activity and Milestone reports produced by the release validation deck follow:

```
                                         CLASSIFICATION
                                                                                            PAGE   1
                                           PERT/TIME
                                        ACTIVITY REPORT
                                REPORTING ORGN.         CONTRACT NO.
 PERT TIME TEST (PROBABILITY)N                                     333              TERM-
1ST SORT KEY   LATEST ALLOWED DATE                                                  REPORT DATE-  9/17/64
2ND SORT KEY   LEAST SLACK                                                          RELEASE DATE- 9/17/64
3RD SORT KEY   ORGANIZATION CODE
4TH SORT KEY   ACCOUNT NO.
   EVENT                                          ACTIV.        DATE            DATE            REMAINING
    PRED.  SUCC.    ACTIVITY DESCRIPTION   PROB.   TIME  EXPECTED  ALLOWED  COMP/SCHED SLACK   TIME   ORG.  ACCOUNT NO.
     00      01                             0,     -0,0            9/30/64  A 9/17/64    1.7    0,0  ORG1  012345678900
I    01      04                             ,99     5,3  10/26/64 11/19/64               3.7    5,3  ORG1       51
I    01      02                             ,99     7,6  11/10/64 11/23/64               1.7    7,6  ORG4       01
I    01      03                             ,99     3,6  10/13/64 11/24/64               5.9    3,6  ORG3  001253567890
     04      07                             ,99     2,3  11/10/64 12/ 8/64               3.7    7,6  ORG2       53
     02      05                             ,88     4,3  12/11/64 12/23/64               1.7   11,9  ORG4       02
     03      05                             ,99     4,1  11/11/64 12/23/64               5.9    7,7  ORG4       03
     03      06                             ,99     4,1  11/11/64 12/30/64               6.6    7,7  ORG1       01
     04      08                             ,99     3,8  11/20/64  1/18/65               7.7    9,1  ORG1       52
     07      11                             ,99     6,0  12/23/64  1/21/65               3.7   13,6  ORG2       52
     06      11                             ,99     3,0  12/ 3/64  1/21/65               6.6   10,7  ORG2       81
     05      10                             ,86     4,0   1/12/65  1/22/65               1.7   15,9  ORG5  023456718560
     06      10                             ,99     3,0  12/ 3/64  1/22/65               6.9   10,7  ORG1       03
     11      16                             ,99     3,5   1/20/65  2/15/65               3.7   17,1  ORG2       81
     08      12                             ,99     4,3  12/22/64  2/17/65               7.7   13,4  ORG1       54
     10      15                             ,85     5,0   2/16/65  2/26/65               1.7   20,9  ORG2       01
     12      14                             ,99     3,1   1/15/65  3/10/65               7.7   16,5  ORG5       51
     19      20                             ,84    -0,0   3/ 8/65  3/18/65               1.7   23,7  ORG2       82
     15      19                             ,84     2,8   3/ 8/65  3/18/65               1.7   23,7  ORG5       61
     16      20                             ,99     4,6   2/22/65  3/18/65               3.7   21,7  ORG2       82
     14      17                             ,99     4,6   2/17/65  4/12/65               7.7   21,1  ORG2       41
     09      13                             ,99    -0,0  12/11/64  4/28/65              19.2   11,9  ORG1       31
     02      09                             ,99     4,3  12/11/64  4/28/65              19.2   11,9  ORG1       31
     17      21                             ,99     2,5   3/ 5/65  4/29/65               7.7   23,6  ORG1       42
     13      18                             ,99     1,1  12/18/64  5/ 5/65              19,2   13,0  ORG4       71
     21      24                             ,99     3,0   3/26/65  5/20/65               7.7   26.6  ORG2       41
     22      23                             ,99    -0,0   4/29/65  6/17/65               7.0   31,3  ORG2       72
     20      22                             ,99     7,6   4/29/65  6/17/65               7.0   31,3  ORG4       71
     15      22                             ,99     6,8   4/ 5/65  6/17/65              10.6   27,7  ORG1       71
     18      22                             ,99     6,1   2/ 3/65  6/17/65              19.2   19,1  ORGJ       72
     20      25                             ,80    14,5   6/16/65  6/29/65               1.7   38,2  ORG1       11
     24      25                             ,99     5,6   5/ 5/65  6/29/65               7.7   32,2  ORG5       42
     23      26                             ,99     4,1   5/27/65  7/15/65               7.0   35,4  ORG2       22
     25      27                             ,78     5,5   7/26/65  8/ 5/65               1.7   43,7  ORG1       11
     26      27                             ,99     3,0   6/17/65  8/ 5/65               7.0   38,4  ORG5       23
     20      27                             ,99    13,6   6/10/65  8/ 5/65               8.1   37,3  ORG4       21
     23      27                             ,99     5,1   6/ 3/65  8/ 5/65               9.0   36,4  ORG1       21
     25      29                             ,99     2.0   6/30/65  8/19/65               7.1   40,2  ORG4       12
     27      28                             ,78     2,1   8/ 9/65  8/20/65               1.7   45.8  ORG2       23
     26      28                             ,99     3,1   6/18/65  8/20/65               9.0   38.5  ORG1       22
     28      30                             ,77     4,8   9/13/65  9/24/65               1.7   50,6  ORG4       23

                                         CLASSIFICATION
```

```
                                                           CLASSIFICATION
                                                                                                                PAGE   2
                                                          PERT/TIME
                                                       ACTIVITY REPORT
                                          REPORTING ORGN.            CONTRACT NO.
 PERT TIME TEST (PROBABILITY)N                                                0333                  TERM-
1ST SORT KEY    LATEST ALLOWED DATE                                                                 REPORT DATE-  9/17/64
2ND SORT KEY    LEAST SLACK                                                                         RELEASE DATE- 9/17/64
3RD SORT KEY    ORGANIZATION CODE
4TH SORT KEY    ACCOUNT NO.
     EVENT                                             ACTIV.       DATE            DATE           REMAINING
   PRED.   SUCC.     ACTIVITY DESCRIPTION        PROB.  TIME  EXPECTED  ALLOWED  COMP/SCHED SLACK  TIME   ORG.   ACCOUNT NO.
     29       30                                   .99   5.0   8/ 4/65  9/24/65              7.1   45.2 ORG1       12
     20       30                                   .99   8.6   5/ 6/65  9/24/65             20.0   32.3 ORG2       91
     30       31                                   .77   3.1  10/ 5/65 10/15/65              1.7   53.7 ORG6       91
```

```
                                                             CLASSIFICATION
                                                                                                                          PAGE   1
                                                              PERT/TIME
                                                          MILESTONE REPORT
                                              REPORTING ORGN.                CONTRACT NO.
 PERT TIME TEST (PROBABILITY)N                                                        333      TERM
                                                                                                   REPORT DATE-  9/17/64
LEVEL/SUMMARY ITEM       9/  0                                                                     RELEASE DATE, 9/17/64
                                                                              LATEST
                                                       MILESTONE  EXPECTED   ALLOWABLE  SCHEDULED   ACTUAL      SLACK
EVENT NO.            EVENT DESCRIPTION                    CODE      DATE       DATE       DATE       DATE
      01        EVENT  1                                    1                                       9/17/64      1,7
      02        EVENT  2                                    2     11/10/64   11/23/64   11/13/64                 1.7
      03        EVENT  3                                    3     10/13/64   11/24/64   10/16/64                 5,9
      04        EVENT  4                                    4     10/26/64   11/19/64   10/26/64                 3,7
      05        EVENT  5                                    5     12/11/64   12/23/64   12/14/64                 1,7
      06        EVENT  6                                    6     11/11/64   12/30/64   11/16/64                 6,6
      07        EVENT  7                                    7     11/10/64   12/ 8/64   11/16/64                 3,7
      08        EVENT  8                                    8     11/20/64    1/18/65   11/23/64                 7,7
      10        EVENT 10                                   10      1/12/65    1/22/65    1/ 8/65                 1,7
      11        EVENT 11                                   11     12/23/64    1/21/65   12/24/64                 3,7
      12        EVENT 12                                   12     12/22/64    2/17/65   12/24/64                 7,7
      13        EVENT 13                                   13     12/11/64    4/28/65   12/24/64                19,2
      14        EVENT 14                                   14      1/15/65    3/10/65    1/18/65                 7,7
      15        EVENT 15                                   15      2/16/65    2/26/65    2/12/65                 1,7
      16        EVENT 16                                   16      1/20/65    2/15/65    1/22/65                 3,7
      18        EVENT 18                                   18     12/18/64    5/ 5/65    1/11/65                19,2
      19        EVENT 19                                   19      3/ 8/65    3/18/65    3/ 5/65                 1,7
      20        EVENT 20                                   20      3/ 8/65    3/18/65    3/ 5/65                 1,7
      21        EVENT 21                                   21      3/ 5/65    4/29/65    3/15/65                 7,7
      22        EVENT 22                                   22      4/29/65    6/17/65    5/ 3/65                 7,0
      23        EVENT 23                                   23      4/29/65    6/17/65    5/ 3/65                 7,0
      24        EVENT 24                                   24      3/26/65    5/20/65    4/ 6/65                 7,7
      25        EVENT 25                                   25      6/16/65    6/29/65    6/18/65                 1,7
      26        EVENT 26                                   26      5/27/65    7/15/65    6/ 4/65                 7,0
      27        EVENT 27                                   27      7/26/65    8/ 5/65    8/ 2/65                 1.7
      28        EVENT 28                                   28      8/ 9/65    8/20/65    8/18/65                 1,7
      29        EVENT 29                                   29      6/30/65    8/19/65    7/12/65                 7,1
      30        EVENT 30                                   30      9/13/65    9/24/65    9/22/65                 1.7
      31        EVENT 31                                   31     10/ 5/65   10/15/65   10/15/65                 1,7



                                                            CLASSIFICATION
```

KNOWN DEFICIENCIES AND RESTRICTIONS

1. An extraneous line of characters follows the update history of type A reports.

2. Specifying the costlink option of the Y card (6000 PERT COST will not be available) causes hang-up in Seg. 14, requiring operator intervention to terminate the job. Also garbage is printed following the reports, so that the operator must terminate the output. Avoid using the costline option.

3. If the update codes for the two cards of a type B activity entry are in conflict, tape rocking may occur.

4. The names of the eighteen PERT TIME segments, PERT 66, Seg. 2,...Seg. 14, Reptgen, may not be used by any jobs running concurrently with PERT TIME.

5. When the network designation differs from that on the old master tape, the incorrect diagnostic "Update code other than A,R,D" is given.

6000 →

DOCUMENT CHANGES

6400/6600 FORTRAN
VERSION 1.1

CHANGES TO THE 64/6600 FORTRAN REFERENCE MANUAL, PUB. NO. 60132700

PAGE	SECTION	CHANGES
1-1	1.1	Remove "." from line describing "Comments". Add at the end of section: "Page Eject 1 . 2-80 contents are not listed"
1-2	1.1.5	Remove ", ." from list of comment designators.
2-2	2.2.1	Remove "Name list names" from list of alphanumeric identifiers.
2-2	2.3.1	Add at end of descriptive paragraph: "The result of integer addition or subtraction must not exceed $2^{59} - 1$. Subscripts, and DO-indexes are limited to $2^{18} - 1$"
2-3	2.3.3	In first paragraph, first line, remove "up to 15". Under forms, add "nE±s". The second paragraph, starting with the second line should read: "if s is positive. The range of a non-zero constant is, approximately, from 10^{-294} to 10^{+322}. If the range is exceeded, a compiler diagnostic is provided."
2-4	2.3.4	In first paragraph, first line, remove "up to 18". In second line, remove "the second is always zero". Under forms, add "nD±s. The second paragraph should read: "The D must always appear. The plus sign may be omitted for positive s. The range of a non-zero constant is, approximately, from 10^{-294} to 10^{+322}. If the range is exceeded, a compiler diagnostic is provided."
2-4	2.3.6	Replace the second sentence of the first paragraph with: "The maximum number of characters allowed in a Hollerith constant of H form depends on use--n is limited to 10 when used in an expression but is limited only by the number of characters that can be contained in up to 19 continuation lines when used in a DATA statement."
2-5	2.3.6	After the first complete sentence at the top of the page, add "The maximum number of characters allowed in expressions for these forms is 10. If more than 10 characters are given in a DATA statement for such a constant, only the last word will have the appropriate zero fill."

PAGE	SECTION	CHANGES
2-6	2.4.1	After first sentence of first paragraph, add the following: "The maximum allowable value of an integer variable depends on usage. The result of conversion from integer to real, of integer multiplication, or of integer division is limited to $2^{48}-1$; the result of integer addition or subtraction can be as great as $2^{59}-1$. Subscripts and DO indexes are limited to $2^{18}-1$".
2-6	2.4.2	The first sentence of the first paragraph should read: "A real variable can be given type explicitly or implicitly; the value must lie in the range $10^{-294} < \| r \| < 10^{+322}$ with approximately 15 significant digits." The possible values for r should read: "$-10^{+322} < r < - 10^{-294}$, r=0, $10^{-294} < r < 10^{+322}$"
2-7	2.4.3	The third line should read: "values in the range $10^{-294} < \| d \| < 10^{+322}$ with approximately 29 significant digits."
5-6	5.4	The third sentence of the first paragraph should read: "A multiply subscripted variable can also be represented using a single subscript." In the fifth sentence, "is" should be replaced by "can be".
5-9	5.5	After the semicolon in the fourth sentence of the second paragraph, the sentence should read "the constant 4. is discarded and B is unmodified."
6-3	6.2.2	The second sentence of the first paragraph should read: "If L is TRUE (not plus zero), the statement is executed".
6-3	6.2.3	The first sentence of the second paragraph should read: "The evaluated expression is tested for TRUE (not plus zero) or FALSE (plus zero) condition."
6-6	6.3.3	The left figure at the bottom of the page should be labeled "NOT ALLOWED", the right figure "ALLOWED".
6-6	6.3.3	The first explanation should have the second line changed to "contains it, but should not be made from the . . ."
6-7	6.3.3	Delete first two paragraphs and diagram from top of page.
9-5	9.3.1	In the first form given in the first paragraph, replace "308" by "512".
9-6	9.3.2	The second sentence of the third paragraph, "A field. . .digits" should be removed. The number "308" at the end of the sixth paragraph should be changed to "323".
9-10	9.3.5	In the first form given, replace "308" by "512".
9-10	9.3.6	The first sentence should be replaced by: "D conversion corresponds to E conversion except that the list variables must be double precision names."

PAGE	SECTION	CHANGES
10-1	10	The second paragraph under "i" should read: "integer variable name of no more than 6 characters whose value must be between 1 and 99". Delete "name name of NAMELIST record"
10-1	10.1	Under the heading "Action", the first, second, and fourth statements should have the word "after" replaced by "before".
D-1		Change first sentence of last paragraph to read: "The statement turns on sense light i; i must be an integer constant in the range 1 to 6."
D-2		The last sentence of the first paragraph should read: "The n_i are statement labels; i must be an integer constant in the range 1 to 6".
D-2		In the description of "IF DIVIDE CHECK" remove the phrase "the indicator is turned off".
D-5		Delete entire page.
E-1		Under structure of least significant half of double precision quantity, remove "(PRESENTLY ZERO)".
F-1		Replace "RUN (cm,fl,d,bl,if,of,fb)" by: "RUN (cm,fl,d,bl,if,of,fb, ,mp)". Under description of "fl" the wording should be ". . .; if omitted, it is set equal to the field length from the JOB card at compile time." Remove the following parenthetical expression. After the description of "fb" add: "mp if cm is L or S and if mp is not blank or zero, the cross reference maps will be printed." The eighth parameter is currently meaningless but is reserved for further use.
F-2		The first paragraph should begin: "Compiler output, except in the G or C modes, includes. . ."
F-10		From the illustration of the deck structure at the bottom of the page, remove the first card with a "-" in column 1.
G		Replace Appendix G by the following:

PAGE	SECTION	CHANGES
10-1	10.1	The third statement should have the semicolon and all that follows deleted.
10-2	10.1	In the first sentence of the fourth paragraph, the number "120" should be replaced by "136". The parenthetical expression in the last sentence of the paragraph should be removed.
10-6	10.4	The last line of the page should read: "0 for even parity (coded characters); non-zero for odd parity".
10-7	10.4	Replace last sentence of page by: "The action is described in detail in the documentation of BUFFEI."
10-8	10.4	Replace last sentence of first paragraph by: "The action is described in detail in the documentation of BUFFEO."
10-8	10.5	Replace the first line of the second paragraph by: "The first record within an area into which encoding or from which decoding will be done starts with the leftmost." The second sentence should read: "For ENCODE, if c is not a multiple of 10, the record ends in the middle of a word and the remainder of the word is blank-filled." In the last sentence of the third paragraph "(decoded)" should be deleted and the number "12" replaced by "150".
10-9	10.5	In example 2), record b should extend only to the end of "Z12345"; in example 3, to the end of "HEADbb02". Change "B6 (2)=HEAD02" to "B6(2)=HEADbb02".
C-1		The first section of this Appendix should be headed "IN-LINE FUNCTIONS". Change "AIMT" to AINT" and "COMPLX" to "CMPLX". After "AINT" add: "AND (X1, . . .XN) Boolean AND of $X_1 . . .X_n$ - Logical" After "CMPLX" add: "COMPL (X) Complement of X - Logical"
C-2		After "MOD" add: "OR (X_1, . . .X_n) Boolean OR of X_1 . . .X_n - Logical"
C-2		Remove "AND" and "COMPL" from Library Functions
C-3		Remove "OR" from list of Library Functions
C-3		Replace "SINGL" by: "SNGL(D) Double to real (unrounded) Double Real"
D-1		Remove "66", from title

FORTRAN ERROR PRINTOUTS G

VERSION 1.1

During a FORTRAN compilation, error printouts follow statements which are incorrect; other such printouts may follow END statements, indicating other types of errors in the program. This is a list of the error diagnostics along with a brief explanation of the type of error involved:

*****AC** ARGUMENT COUNT ERROR

Indicates that the number of arguments in this reference to a subroutine differs from the number which occurred in a prior reference.

*****AL** FORMAT ERROR IN ARGUMENT LIST

Indicates a format error in a list of arguments.

*****AS** FORMAT ERROR IN ASSIGN STATEMENT

Indicates a format error in an ASSIGN statement.

*****BC** FORMAT ERROR IN BOOLEAN CONSTANT

Indicates a format error in the designation of a FORTRAN boolean constant in a B-type expression.

*****BI** BINARY SUBROUTINE HAS INCORRECT HEADER CARD

Indicates that a subroutine of binary cards following the FORTRAN program and its FORTRAN-coded subroutines has an incorrect header card.

*****BO** COMMON BLOCK OVERFLOW

Indicates that the current requirements for a labeled block of COMMON storage exceed the length of the block as established in a preceding COMMON statement.

*****BX** FORMAT ERROR IN BOOLEAN STATEMENT

Indicates a format error in a B-type boolean statement.

*****CD** VARIABLE DUPLICATED IN A COMMON REGION

Indicates that a variable currently being assigned to the COMMON region has been previously assigned to this region.

*****CE** VARIABLES ASSIGNED TO COMMON ARE IMPROPERLY EQUIVALENCED

Indicates that two variables assigned to COMMON are improperly equivalenced.

*****CL** FORMAT ERROR IN CALL STATEMENT

Indicates a format error in a call statement.

*****CM** FORMAT ERROR IN A COMMON STATEMENT

Indicates a format error in a COMMON statement.

*****CN** TOO MANY CONTINUATION CARDS

Indicates that more than 19 continuation cards appear in succession or that one such card appears in an illogical sequence.

*****CO** COMMON STORAGE EXCEEDED

Indicates that the amount of COMMON storage required by the main program or specified to the compiler is less than required by the current program or subroutine.

*****CT** CONTINUE STATEMENT IS MISSING A STATEMENT NUMBER

Indicates a continue statement with no statement number.

*****DA** DUPLICATE ARGUMENTS IN A FUNCTION DEFINITION STATEMENT

*****DC** FORMAT ERROR IN A DECIMAL CONSTANT

Indicates a format error in the expression of a FORTRAN decimal constant.

*****DD** VARIABLE BEING DIMENSIONED HAS BEEN PREVIOUSLY DIMENSIONED

Indicates a variable has appeared in more than one DIMENSION statement.

*****DF** DOUBLY DEFINED FUNCTION NAME

Indicates that the function name in the current function-definition statement has occurred as the name of a previously defined function.

*****DM** FORMAT ERROR IN A DIMENSION STATEMENT

Indicates an error in the format of a DIMENSION statement.

*****DN** ILLEGAL DO TERMINATOR

Indicates the attempt to use a FORMAT, GO TO, arithmetic IF, or another DO statement as the termination statement of a DO.

*****DO** FORMAT ERROR IN A DO STATEMENT

Indicates an error in the format of a DO statement.

*****DP** MULTIPLY DEFINED STATEMENT NUMBER

Indicates the current statement has previously appeared in the statement number field.

*****DR** DATA RANGE ERROR

The range of the implied DO exceeds the named array.

*****DR** DATA RANGE ERROR-ATTEMPT TO STORE IN BLANK OR NUMBERED COMMON

A DATA statement has attempted to preset variables in blank or numbered COMMON.

*****DS** UNDEFINED STATEMENT NUMBER IN DO

Indicates that references have been made in DO statements to statement numbers which did not appear somewhere in the statement number field.

*****DT** FORMAT ERROR IN DATA STATEMENT

Indicates an error in the format of a DATA statement.

*****EC** CONTRADICTION IN EQUIVALENCE STATEMENT

Indicates that a variable currently appearing in an EQUIVALENCE statement cannot be equivalenced because of an inherent contradiction in the statement.

*****EF** END OF FILE CARD BEFORE END CARD

Indicates that an end of file card is detected before the last END card is encountered.

*****EM** ERROR IN FORMAT OF AN INDICATED EXPONENTIATION

Indicates the mode of the base or the exponent of an indicated exponentiation process is improper.

*****EQ** FORMAT ERROR IN EQUIVALENCE STATEMENT

Indicates an error in the format of an EQUIVALENCE statement.

*****EX** FORMAT ERROR IN EXPONENT

Indicates an error in the exponent portion of an indicated exponentiation process.

*****FL** FORMAT ERROR IN EXTERNAL OR F-TYPE STATEMENT

Indicates an error in an EXTERNAL statement or F-TYPE statement.

*****FM** UNRECOGNIZABLE STATEMENT

Indicates a statement whose type cannot be determined.

*****FN** NO STATEMENT NUMBER ON A FORMAT STATEMENT

Indicates that a FORMAT statement is missing a statement number.

*****FS** ERROR IN SPECIFICATION PORTION OF A FORMAT STATEMENT

Indicates a format error in the specification portion of a FORMAT statement.

*****FT** FORMAT ERROR IN FUNCTION TYPE STATEMENT

Indicates an error in a FUNCTION TYPE statement.

*****GO** FORMAT ERROR IN A GO TO STATEMENT

Indicates an error in the format of a GO TO statement.

*****ID** IMPROPERLY NESTED DO LOOPS

Indicates DO loops which are illegally nested.

*****IF** FORMAT ERROR IN AN IF STATEMENT

Indicates an error in the format of an IF statement.

*****IL** FORMAT ERROR IN AN INDEXED LIST OF I/O STATEMENT

Indicates a format error in an indexed list of the current input/output statement.

*****IN** ILLEGAL FUNCTION OR SUBROUTINE NAME

Indicates an apparent attempt to reference a subprogram whose name does not start with a letter.

*****IT** ILLEGAL TRANSFER TO DO TERMINATOR

Indicates an illegal jump to a statement number which terminates a DO loop.

*****LN** TOO MANY ARGUMENTS FOR LIBRARY ROUTINE

Indicates that a reference has been made to a standard library routine with more arguments appearing in the reference than are provided by the subroutine.

*****LS** FORMAT ERROR IN INPUT/OUTPUT LIST

Indicates an error in the format of an input/output list.

*****MA** ARGUMENT OF SUBROUTINE MISUSED IN EQUIVALENCE STATEMENT

Indicates that an argument of the subroutine or function being compiled has been misused in an EQUIVALENCE statement.

*****MO** MEMORY OVERFLOW, FIELD LENGTH TOO SHORT

Indicates that the compiler field length, as specified on the JOB card, is too short.

*****MR** MISSING SUBROUTINE

Indicates that a subroutine not in the standard library or not loaded with the program has been referenced by the FORTRAN coded routines.

*****MS** UNDEFINED STATEMENT NUMBER

Indicates that references have been made to statement labels which did not appear somewhere in the statement-label field of a line.

*****NC** NAME OF SUBROUTINE HAS BEEN PREVIOUSLY DEFINED

Indicates that the name of a subroutine or function conflicts with prior use of the name.

*****NM** IMPROPER HEADER CARD

Indicates an error in the formatting of the name (header) card.

*****OP** REFERENCE TO AN ARRAY BEFORE IT IS DEFINED IN A DIMENSION STATEMENT

Indicates that a reference to an array was made prior to the appearance of the array in a DIMENSION statement.

*****PN** UNBALANCED PARENTHESIS

Indicates an unequal number of open and closed parentheses in a statement.

*****RN** FORMAT ERROR IN A RETURN STATEMENT

Indicates an error in the format of a RETURN statement.

*****SB** ERROR IN SUBSCRIPT OF AN ARRAY

Indicates a format error in a subscript of an array reference currently being processed.

*****SE** FORMAT ERROR IN SENSE STATEMENT

Indicates an error in the format of a sense statement.

*****SF** FIELD LENGTH OF ROUTINE BEING COMPILED EXCEEDS THE SPECIFIED FIELD LENGTH

Indicates the required field length of the program exceeds the specified one.

*****SL** COMPILER FIELD LENGTH IS EXCEEDED AT THE TIME STANDARD LIBRARY ROUTINES ARE ASSEMBLED

Indicates the field length specified on the JOB card is exceeded by the library routines.

*****SM** FORMAT ERROR IN STATEMENT NUMBER

Indicates an error in the format of the statement label field.

*****SN** ILLEGAL CHARACTER IN STATEMENT NUMBER USAGE

Indicates an error in the format of the position where the statement label should appear.

*****SY** SYSTEM ERROR IN FORTRAN COMPILER

*****TM** SUBROUTINE HAS MORE THAN 60 ARGUMENTS

Indicates that a subroutine reference has more than 60 arguments or that the routine being compiled has more than 60 parameters.

*****TY** FORMAT ERROR IN TYPE STATEMENT

Indicates an error in the format of a TYPE statement.

*****UA** REFERENCE MADE TO AN ARRAY WHICH HAS NOT PREVIOUSLY BEEN DIMENSIONED

Indicates reference was made to an array which has not previously appeared in a DIMENSION statement.

*****UE** REFERENCE MADE TO AN I/O FILE WHICH DID NOT APPEAR ON HEADER CARD OF MAIN PROGRAM

Indicates a reference to an input/output file which was not listed in the header card of the main program.

*****US** A BINARY PROGRAM INCLUDED WAS NOT REFERENCED

Indicates that a subroutine on binary cards following the FORTRAN program and its FORTRAN subroutines has not been referenced.

*****VC** VARIABLE NAME CONFLICTS WITH A PRIOR USAGE

Indicates that a variable name appears which conflicts with some prior use.

*****VD** ARRAY WHOSE DIMENSIONS ARE ARGUMENTS TO THE SUBROUTINE OR FUNCTION HAS BEEN MISUSED

Indicates improper use of an array with variable dimensions.

*****XF** FORMAT ERROR IN THE EXPRESSIONS BEING PROCESSED

Indicates an error in the format of the expression currently being processed.

SUPPLEMENT TO FORTRAN REFERENCE MANUAL

The following FORTRAN object routine descriptions supplement the FORTRAN Reference Manual.

ALOG, ALOG10

These routines are complete revisions of their predecessors. The functions performed by the new versions, however, remain the same. Improvements have been made in speed, accuracy and storage requirements.

Method:

$$\text{ALOG10(X)} = \log_{10}(e) * \text{ALOG(X)}$$

$$\text{Let } X = 2^{K}*W, \quad .5^{1/2} \leq W \leq 2^{1/2}$$

$$\log_e(X) = K*\log_e(2.0) + \log_e(W)$$

$$\log_e(W) = \log_e\left(\frac{1+t}{1-t}\right) \cong 2t - Q*t$$

$$Q = \frac{n_1 Z + n_2 Z^2 + n_3 Z^3}{d_0 + d_1 Z + d_2 Z^2 + d_3 Z^3}, \quad Z = t^2$$

Other Information:

A. Time: 28 μs for ALOG on 6600
32 μs for ALOG10 on 6600

B. Storage: 33 words for ALOG
35 words for ALOG10

C. Accuracy: ALOG: In the range $.5 \leq X \leq 2$, the maximum observed absolute error was 3.2E-15; for values of X outside this range the maximum observed relative error was 5.8E-15

D. Restrictions: If $X \leq 0$, the result is set to infinity and the normal exit is taken

ERROR RETURNS:

For arguments listed in the left hand column in the table below, the result is set to the value given in the right hand column, and the normal exit is taken:

Argument	Result
$-\infty$	+ Indefinite
Negative	+ Indefinite
± 0	$-\infty$
$+\infty$	$+\infty$
± Indefinite	+ Indefinite

ATAN, ATAN2

These routines are complete revisions of their predecessors. The functions performed by the new versions, however, remain the same. Improvements have been made in speed, accuracy, and storage requirements.

Method: Let $A = \text{ATAN}(X)$, then $- PI/2 < A < + PI/2$.

Let $B = \text{ATAN2}(Y,X)$, then B is the argument of the complex number $X + iY$ and $- PI \leq B \leq + PI$.

$$B = \begin{cases} \text{Sign}(Y) * PI/2 & , X = 0 \\ \text{ATAN}(Y/X) & , X > 0 \\ \text{ATAN}(Y/X) + \text{sign}(Y) * PI & , X < 0 \end{cases}$$

Let $P = \tan(PI/16)$, $T = \tan(3PI/16)$

$\text{ATAN}(W) = \text{sign}(W)*\text{ATAN}(V)$, $V = \text{ABS}(W)$

$\text{ATAN}(V) = \text{ATAN}(R) + C$, R and C defined below

$0 \leq V < P$	$R = V$	$C = 0.0$
$P \leq V < 2^{1/2} - 1$	$R = (V-P)/(1 + V*P)$,	$C = PI/16$
$2^{1/2} - 1 \leq V < 1$	$R = (V-T)/(1 + V*T)$,	$C = 3PI/16$
$1 \leq V < 2^{1/2} + 1$,	$R = (1-V*T)/(V + T)$,	$C = 5PI/16$
$2^{1/2} + 1 \leq V$,	$R = (1-V*P)/(V + P)$,	$C = 7PI/16$

$\text{ATAN}(R) = R - R*Q$, $Z = R^2$,

$$Q = \frac{n_0 + n_1 Z + n_2 Z^2 + n_3 Z^3}{d_0 + d_1 Z + d_2 Z^2 + d_3 Z^3}$$

Other Information:

A. Time: 290 μs for ATAN on 6600

B. Storage: 52 words for ATAN

C. Accuracy: In ATAN whan X = .196 the maximum observed absolute error was 4.6E-16, and for all other values of X the maximum observed relative error was 1.0E-15.

In ATAN2 when $|Y/X| = .196$ the maximum observed absolute error was 1.6E-15, and for all other values of Y and X the maximum observed relative error was 3.7E-14.

D. Restrictions: In ATAN2 if X = Y = 0 the result is set to indefinite and the normal return taken.

BUFFEI BUFFER IN ROUTINE

This is a complete re-write of the BUFFER IN routine supplied with the COS system. The routine operates under the following revised specifications.

1. One and only one logical record is read each time BUFFEI is called. If the block length specified by the call is longer than the logical record, the excess block locations will not be changed by the read. If the logical record is longer than the block, the excess words in the logical record are passed over. They will be counted but not transmitted to the program area. The number of CM words in the logical record may be obtained by referencing LENGTH.

2. If the end of file flag has been set (and not cleared) by another routine, BUFFEI will abort the program with the diagnostic.

   *BUF IN**ENDFILE Filename

3. The FORTRAN statement IF (UNIT, i) must be used after each call to BUFFEI to insure that the read operation is complete and to check for a file mark. An IF(EOF),i will not detect a file mark read by BUFFEI.

4. If the last operation on the file was a write operation, no data is available to be read. BUFFEI will abort the program when this situation occurs with the Dayfile diagnostic:

   *BUF IN**LAST OP WRITE
   *CALL FROM ADDRESS XXXX

5. If the starting address for the block is greater than the terminal address, BUFFEI will abort the program with the Dayfile diagnostic:

   *BUF IN***FWA.GT.LWA
   *CALL FROM ADDRESS XXXX

6. The UNASSIGNED MEDIUM diagnostic has been extended to include a DMP of the program's file name table (Loc 0 thru RA+N+3). Prior to the DMP the name of the unassigned file is stored at RA+N+3. The diagnostic is

   *BUF IN**UNASSIGNED MEDIUM

BUFFEO BUFFER OUT ROUTINE

This is a complete re-write of the BUFFER OUT routine supplied with the COS system. The routine operates under the following revised specifications.

1. One logical record is written each time the routine is called. The length of the record will be LWA-FWA+1. The record will consist of standard 512 word physical records and a short record or just a short record if the block length is less than 512 words. In cases where the block length is a multiple of 512, the logical record will end with a zero length record (end of record mark).

   For BCD output on one-half inch tape the record consists of 136 character physical records. If the block length is less than 136 characters the physical record will be blank filled to 136 characters. If the block consists of several physical records the last record will be blank filled to 136 characters if necessary.

2. If the LWA is less than the FWA the program will be aborted and the following diagnostic will appear in the Dayfile:

   ```
   *BUF OUT**FWA.GT.LWA
   *CALL FROM ADDRESS XXXX
   ```

3. The FORTRAN statement IF (UNIT,i) must be used between successive calls to BUFFEO involving the same file to insure that the write operation has completed. Since BUFFEO changes the buffer arguments for the file to point to the CM block specified in the call, calls to other READ/WRITE routines involving the same file may not follow any buffer operation until the pointers have been restored by the IF (UNIT, i) check.

4. The UNASSIGNED MEDIUM diagnostic has been similarly extended.

COS, SIN

These routines are complete revisions of their predecessors. The functions performed by the new versions, however, remain the same. Improvements have been made in speed, accuracy and storage requirements.

Method:

Let $N = [X/(PI/2) + .5]$, and

$R = X - N*PI/2$, then $|R| < PI/4$.

Let $K = N\bmod 4$, $K = 0, 1, 2, 3$, then

$\sin(X) = \sin(R + N*PI/2) = \sin(R + K*PI/2)$

$= \sin(R)*\cos(K*PI/2) + \cos(R)*\sin(K*PI/2)$

A similar formula is used for COS (X).

Depending upon K, either the sine or cosine of R is evaluated and complemented if necessary.

The sine and cosine of R are evaluated by polynomials of degree 11 and 12, respectively. These polynomials were telescoped from a truncated Taylor-Maclaurin power series of degree 15 and 14.

Other Information:

A. Time: 243 μs on 6600

B. Storage: 46 words for both SIN and COS

C. Accuracy: The maximum observed absolute error for 10,000 values of X uniformly distributed in the basic range $|X| \leq PI/4$ was 5.3 E - 15 for both the sine and cosine. The maximum observed absolute error for 20,000 values of X uniformly distributed in the range $|X| \leq 31.4$ was 6.1E - 15 for both the sine and cosine.

D. Restrictions: If $|X| \geq 1.1$ E14 the result is set to indefinite and the normal return taken.

EXP

This routine is a complete revision of its predecessor. The function performed by the new version, however, remains the same. Improvements have been made in speed, accuracy and storage requirements.

Method:

Let $N = [X/\log_e (2.0) + .5]$, and

$R = X - N*\log_e (2.0)$, then $|R| = \log_e (2.0)/2$ and $e^X = 2^N * e^R$.

$e^R = 1 + R + Q$,

$Q = R*(R*B - Z*T)/(2*B + Z*T - R*B)$

$Z = R*R$, $T = 28*Z + 2520$,

$B = Z*Z + 420*Z + 15120$.

Other Information:

A. Time: 26μs on 6600.

B. Storage: 34 words.

C. Accuracy: The maximum observed relative error for 70,000 values of X uniformly distributed in the range $|X| < .347$ was 3.8E-15. The maximum observed relative error for 435,000 values of X uniformly distributed over the range $-675.0 \leq X \leq 741.0$ was 5.0E-15.

D. Restrictions: The result is set to zero if $X \leq -675.82$, it is set to +infinity if $X \geq 741.67$.

IOCHEK IF UNIT ROUTINE

This is a new version of the IF UNIT routine. The routine operates under the following specifications.

1. A Recall has been added prior to taking the operation-not-complete (X6=1) exit.

2. IOCHEK restores the original buffer pointers in the RA block (IN=OUT=FIRST), insures that an EOR status has been returned by CIO on read operations, and sets the length of the logical record read in RA+5 prior to taking the operation complete (X6=0) exit.

3. No data should have been transmitted if the end of file (X6=1) exit is taken. The exception to this rule occurs if the external file medium is a 1/2" BCD tape. Since there are no logical record separators defined, a Buffer In will cause the reading of one file and a Buffer Out may cause the writing of several physical records.

INPUTB BINARY INPUT ROUTINE

This is a modification of the binary input routine supplied with the COS system.

The modifications are:

1. One and only one logical record is read each time INPUTB is called. If the list is longer than the logical record the excess words in the list are ignored by the routine (i.e. no values are stored). Hence, these values will not be changed by the read statement.

2. An attempt to read past an uncleared end of file will cause a program abort. The end of file condition can be cleared by testing IF(EOF,i) for each file after the file mark has been read and prior to another attempt to read. The abort diagnostic sent to the Dayfile is

```
BIN    INPUT**ENDFILE    filename
```

3. The UNASSIGNED MEDIUM diagnostic has been extended as previously described.

INPUTC CODED INPUT ROUTINE

This is a modification of the coded input routine supplied with the COS system.

1. An attempt to read past an uncleared end of file will cause a program abort. The end of file condition can be cleared by testing IF(EOF, i) for end file after the file mark has been read and prior to another attemp to read. The end file condition for the file INPUT is set by either an end of file mark or a short record (end of logical record). It cannot be cleared within a FORTRAN program. The abort diagnostic sent to the Dayfile is

   *BCD INPUT**ENDFILE filename

2. The UNASSIGNED MEDIUM diagnostic has been extended as previously described.

3. Conversion follows the specifications of Section 9 of the COS FORTRAN Manual ammended to include an RW specification (Alphanumeric, right justified with zero fill if necessary).

4. A FORMAT error produces the following standard diagnostic in the Dayfile.

   *ERROR DATA INPUT*FORMAT NO. XX
   *(diagnostic line)
   *CALL FROM ADDRESS XXXX

   The diagnostic will be one of the following:

   *ILLEGAL FUNCTIONAL LETTER
   *PAREN GROUP NOT CLOSED
   *FIELD WIDTH ZERO
   *EXCEEDED RECORD SIZE
   *HOLLERITH FORMAT WITH LIST

5. A DATA error produces the following standard diagnostic in the Dayfile.

   *LINE NO. XXX
   *ERROR DATA INPUT*FORMAT NO. XXX
   *(diagnostic line)
   *CALL FROM ADDRESS XXXX

   The diagnostic will be one of the following:

   *ILLEGAL DATA IN FIELD
   *DATA OVERFLOW

INPUTS — DECODE ROUTINE

This is a complete re-write of the decode routine supplied with the COS system. The routine operates under the following revised specifications:

1. The parameter (c) specifying the record length may be an arbitrary number of BCD characters .LE.150. The record starts with the leftmost character of the location specified by V and continues 10 BCD characters per CM word for c BCD characters or until a zero (no bits) character is encountered. If the record ends in the middle of a CM word, the remaining characters are ignored. Each record begins with a new CM word. The number of records processed by each call to INPUTS depends on the format and the length of the list. If c is greater than 150 the routine aborts the program and enters the following diagnostic in the Dayfile:

   ```
   DECODE*CHAR/REC    )150.     CALL FROM XXXX
   ```

2. Conversion follows the specifications of Section 9 of the COS FORTRAN Reference Manual amended to include an RW specification (Alphanumeric, right justified with zero fill if necessary).

3. A Format error produces the following standard diagnostic in the Dayfile.

   ```
   *ERROR DATA INPUT*FORMAT NO. XX
   *(diagnostic line)
   *CALL FROM ADDRESS XXXX
   ```

   The diagnostic will be one of the following:

   ```
   *ILLEGAL FUNCTION LETTER
   *PAREN GROUP NOT CLOSED
   *FIELD WIDTH ZERO
   *EXCEEDED RECORD SIZE
   *HOLLERITH FORMAT WITH LIST
   ```

4. A DATA error produces the following standard diagnostic in the Dayfile:

   ```
   *LINE NO. XXX
   *ERROR DATA INPUT*FORMAT NO. XX
   *(diagnostic line)
   *CALL FROM ADDRESS XXXX
   ```

   The diagnostic will be one of the following:

   ```
   *ILLEGAL DATA IN FIELD
   *DATA OVERFLOW
   ```

LENGTH LENGTH OF RECORD FUNCTION

1. LENGTH returns the length in CM words of the last logical record transmitted from a file by a BUFFER IN statement.

2. LENGTH will not return a correct value unless it is preceded by an IF (UNIT,i) statement that indicates operation complete.

OUTPTB BINARY OUTPUT ROUTINE

This is a modification of the binary output routine supplied with the COS system.

The modifications are:

1. One logical record is written each time OUTPTB is called.
2. The UNASSIGNED MEDIUM diagnostic has been extended as previously described.

OUTPTC CODED OUTPUT ROUTINE

This is a modification of the coded output routine supplied with the COS system.

The modifications are:

1. For file OUTPUT only a line count is kept. If this count exceeds the count specified for the program it causes a program abort. The abort diagnostic entered in the Dayfile is

   **OUTPUT FILE*LINE LIMIT EXCEEDED*

2. The UNASSIGNED MEDIUM diagnostic has been extended as previously described.

3. Conversion follows the specifications of Section 9 of the COS FORTRAN Reference Manual amended to include an Rw specification (Alphanumeric, right justified in field if necessary), and the note, for Iw conversion, if the integer is greater than 2**48-1 a double R is placed right justified in the field.

4. A FORMAT error produces the following standard diagnostic in the Dayfile:

   *ERROR DATA OUTPUT*FORMAT NO. XX
   *(diagnostic line)
   *CALL FROM ADDRESS XXXX

   The diagnostic will be one of the following:

   *ILLEGAL FUNCTIONAL LETTER
   *PAREN GROUP NOT CLOSED
   *EXCEEDED RECORD SIZE
   *FIELD WIDTH ZERO
   *FIELD WIDTH .LE. DECIMAL WIDTH
   *HOLLERITH FORMAT WITH LIST

5. Data errors are flagged with an * on the listing.

OUTPTS — ENCODE ROUTINE

This is a complete re-write of the encode routine supplied with the COS system. The routine operates under the following revised specifications:

1. The parameter (c) specifying the record length may be an arbitrary number of BCD characters .LE.150. The record starts with the leftmost character of the locations specified by V and continues 10 BCD characters per CM word for c BCD characters or until a zero (no bits) character is encountered. If the record ends in the middle of a CM word (or is terminated by a zero character) the remaining characters are filled with blanks. Each record begins with a new CM word. The number of records processed by each call to OUTPTS depends on the Format and the length of the list. If c is greater than 150 the routine aborts the program and enters the following diagnostic in the Dayfile.

   ```
   ENCODE**CHAR/REC  )150.   CALL FROM XXXX
   ```

2. Conversion follows the specifications of Section 9 of the COS FORTRAN Manual amended to include an Rw specification (Alphanumeric, right justified in field if necessary), and the note, for Iw conversion, if the integer is greater than 2**48-1 a double R is placed right justified in the field.

3. A FORMAT error produces the following standard diagnostic in the Dayfile.

   ```
   *ERROR DATA OUTPUT*FORMAT NO.XX
   *(diagnostic line)
   *CALL FROM ADDRESS XXXX
   ```

   The diagnostic will be one of the following:

   ```
   *ILLEGAL FUNCTIONAL LETTER
   *PAREN GROUP NOT CLOSED
   *EXCEEDED RECORD SIZE
   *FIELD WIDTH ZERO
   *FIELD WIDTH .LE. DECIMAL WIDTH
   *HOLLERITH FORMAT WITH LIST
   ```

4. Data errors are flagged with an * on the listing.

RANF

This routine is a complete revision of its predecessor. The multiplicative congruential method for generating pseudo-random numbers is retained with the exception that a different multiplier is used. This multiplier eliminates a third order serial correlation in the unit cube which was present in the previous generator. Unlike the previous routine the argument of the function determines the action to be taken by RANF. If the argument is zero, the routine returns a new random number for the value of the function. If the argument is negative, the last random number is returned for the value of the function. Since any particular random number is determined by its immediate predecessor, the last random number produced can be useful for restart purposes. If the argument is positive, it is used as the beginning of a new sequence of random numbers. The value of the function under these conditions is meaningless.

SQRT

This routine is a complete revision of its predecessor. The function performed by the new version, however, remains the same. Improvements have been made in speed, accuracy and storage requirements.

Method: To calculate SQRT (X)

Let $X = 2^N * W$, $.5 \leq W < 1$, N is an integer.

$N = 2K + r$, $r * N \geq 0$,
$r = -1, 0, + 1$, then $X^{\frac{1}{2}} = 2^K * 2^{r/2} * W^{\frac{1}{2}}$

Let $B = .585786W + .420495$ be the
initial approximation to $W^{\frac{1}{2}}$,

Let C = result of two Newton's iterations using B as the
initial guess to $W^{\frac{1}{2}}$, then

$$4C = \frac{(B^2 + W)^2 + 4B^2W}{B(B^2 + W)}$$

$$D = 2X^{\frac{1}{2}} = 2^{K-1}2^{r/2}(4C)$$

$$X^{\frac{1}{2}} = .25*D + X/D$$

Other Information

A. Time: 17.5 usec on 6600

B. Storage: 26 words

C. Accuracy: The maximum observed relative error for 200,000 arguments uniformly distributed between .25 and 4.0 was 3.5E-15

In addition 199,999 of the 200,000 results gave exact agreement with the true value when it was rounded to single precision.

D. Restrictions: If $X < 0$, the result is set to indefinate and a normal return is taken.