

Send all comments, criticisms and contributions to the editor: T.W. Lanzatella
University Computer Center, 2520 Broadway Drive, Lauderdale, MN 55113

TABLE OF CONTENTS

NOTICE OF CHANGES TO THE SYSTEM	1
PROPOSED CHANGES TO THE SYSTEM.	2
X.PROPOSAL(MIL) - W.T. Sackett.	2
A CATALOG MOD - B. Hanson.	2
ASCII PRINT FILES - T. Salo.	3
TRACKS COMMAND - K.C. Matthews and L.A. Liddiard	4
SYSTEM MAINTENANCE.	5
LAST WEEK'S SYSTEMS GROUP MEETING - T.W. Lanzatella.	5
CALLPRG AND LIBRARY TAPE NEWS - M. Riviere	6
FIELD LENGTH CHANGES - K.C. Matthews	6
CYBER 74 DEADSTART DUMP ANALYSIS - K.C. Matthews	8

NOTICE OF CHANGES TO THE SYSTEM

Marisa Riviere installed a change to WRITEUP processing in CALLPRG which remedies a long-standing and confusing situation. Until now, a WRITEUP command terminated prior to normal completion left the writeup file being printed, the writeup index and any additional writeup files which had not already been printed as local files to the user. A subsequent attempt by the user to execute a command with the same name as a previously specified writeup resulted in UNIDENTIFIED LOADER INPUT. CALLPRG will now return all files except a local copy of a user's own writeup index.

Brian Hanson uncovered a bug in his recent change to PACK, SORT and RESEQ which caused RESEQ to fail if the user's file was above a certain buffer size (see DSN 3, 14 p. 1). Brian also altered NODROP processing so that NODROPPed files are no longer LOCKed. Additionally, Brian installed a MFL= entry point in LINK. This change serves to ensure that local file loads of ABS type programs will always work.

Jeff Drummond installed the following changes.

1. DSDI was corrected to print on consecutive pages rather than to skip every other page. A long-standing bug.
2. Jeff installed a new common deck, COMSEFR, which defines the bits in the ECS flag register. This change is part of a series of changes to CPUMTR which adds a new ECSSM subfunction to set an ECS file busy (similar to SFBM). A portion of the new subfunction was also added to CPUMTR.

3. The long and cumbersome micros used to define origin types and file types have been replaced with a tidy little macro called TYPE which allows both easy declaration of types and easy concatenation into a micro string. The macro is PURGMACed in PPCOM so as not to clash with user defined TYPE macros.

Kevin Matthews installed his proposed ECS access changes (see DSN 3, 8 p. 4). The changes allow each separate path to ECS to be individually enabled or disabled. Five new DSD ENABLE/DISABLE functions were added:

ECS/STORAGE MOVE
ECS/CPU LOADS
ECS/PPU LOAD
ECS/DDP
ECS/ALL

A special document will be prepared for operations detailing the use of these new commands.

Kevin also installed some temporary changes to his performance measurement code designed to measure DDP usage. Additionally, Kevin repaired a bug in SET which caused central memory to be wrecked if illegal STK, RTK or TTK commands are entered in the CMRDECK.

Bill Sackett installed a Direct Access File size check in CIO (see DSN 2, 20 p. 3). The check has been used on MERITSS for several weeks without difficulty. The change will not affect MIRJE users since everyone has infinite limits. Bill also changed the explanation for the FC parameter on the LIMITS report.

Don Mears altered LTD so that the PDP-11 will not be switched off in the EST if the PDP-11 is not up at LTD initialization time. Additionally, Don repaired a problem in TELEX where an INTERRUPTED message could be generated when no program was running. This also prevents a possible TELEX A-status hang.

Tim Salo contributed a new PP program, LSB, which issues account file messages for SUPPIO. This provides an interlock on the dayfile buffer in SUPPIO's field length and reduces the amount of disk I/O performed by LSU. Additionally, SUPPIO now uses ECSW in the control point area to hold ECS RA and ECS FL and several routines in SUPPIO which allocate the CPU, perform housekeeping and resource allocation and search for disk I/O requests which were rewritten. This should reduce SUPPIO's CPU utilization.

PROPOSED CHANGES TO THE SYSTEM

X.PROPOSAL(MIL) - by W.T. Sackett

In order to remain compatible with MECC (if MECC decides not to use CALLPRG), MERITSS would like to make X. a control card in the system. While the choice of the name is debatably unfortunate, there is documentation throughout the country which would have to be changed if another name were used. X is a short program that cracks arguments after the . and runs the appropriate MIL program.

//////////

A CATALOG Mod - by B. Hanson

At the end of the second summer session, Link and Cyber loader will be changed to set the running FL to the value provided in the HHA field in Cyber loader produced

5400 tables and the Ends field in link produced 5300 tables. This value can be overridden by either a RFL= or a MFL= entry point in the absolute binary. However, the utility CATALOG, while it lists all the entry points, gives no mention of these important values HHA or Ends. I propose to change CATALOG to list the Ends and HHA values as psuedo entry points enclosed by parentheses.

//////////

ASCII Print Files - by T. Salo

The recent purchase of a 512 printer gives the Cyber the capability of printing ASCII files. These proposals will make this feature available to the users. The proposed external characteristics of ASCII print files are:

1. ASCII type print files will print only on the 512 printer.
2. Ordinary print files may print on the 512 if no ASCII print files are present in the system.
3. ASCII print files will be routed to the ASCII queue via the DISPOSE control card.
4. There will be a number of subtypes of ASCII files.
5. LCD will drive the 512 printer in addition to other unit record equipment.
6. Implementation of ASCII print files should not prevent the easy implementation of families or the ownership by account numbers of output queue files.
7. It will probably be possible to do a deadstart dump to the 512 printer.

Proposal 1 - The DISPOSE control card will be enhanced to allow a control card call of the form:

DISPOSE(lfn=Ax/...)

where: A indicates that lfn is to be disposed to the ASCII queue.
x specifies the ASCII file subtype.

Possible values for x might be:

- A - display code
- B - 64-character ASCII subset (TERM,TTY)
- C - 64-character ASCII (TERM,TTY/ASCII)
- D - 61-character (TERM,TTYD)
- E - 61-character ASCII (TERM,TTY/ASCII)
- F - 12-bit loose pack ASCII (transparent input)
- G - 7 * 8 bit tight pack ASCII
- H - 7 1/2 * 8 bit full pack ASCII

Proposal 2 - In order that IIO does not need to perform an inordinate amount of mass storage I/O, that output queue files may be owned by user numbers, and that families might be implemented with a minimum amount of work, the following information should be present in the output queue file's FNT/FST entry:

1. ASCII files must have some indication in the FNT/FST entry that they are ASCII files.
2. 18 bits is currently reserved by CDC in the FST. This space should be reserved for the user index owning the output queue file.

3. The EST ordinal of the family under which the file was created should remain in its current location.

Therefore a new ASCII queue file type, ASFT, should be added to the system.

This proposal may require the re-assembling of some CALLPRG or user binaries, but is less work and more consistent with the design of the system than alternate methods.

Alternate methods of implementing ASCII files include:

1. Use the 6 bits now holding the family EST ordinal to indicate that a print file is an ASCII print file. This would require that families have unique user indices if user numbers are to own output queue files.
2. The ID field could be used to indicate ASCII files. This is inconsistent with our current system making errors in implementation probable. This also restricts the sites which may receive ASCII files to Lauderdale.
3. The job name could be the user index hash, freeing 18 bits in the FST. This has the advantage of being consistent with NOS, but would require the re-education of operations and users (no small task).

//////////

Proposal of TRACKS Command for SET. - by K.C. Matthews and L.A. Liddiard

On the Cyber 74, the 808 disk is used for rollout files only. Although the 808 channel itself is kept busy, very few tracks are used up. In fact, out of 2048 tracks (4000B), we can never remember a case where more than 1/4 were ever used during normal production. This is probably good, since it hopefully minimizes head movement for the rollout files on the 808. (Ask Don Mears about 808 timings.) But Larry Liddiard noticed that this means that we are really wasting 3/4 (600B words) out of the 1000B words in the TRT for the 808. And since central memory is becoming more in demand both by the users and the system, this is a significant amount.

We therefore propose to cut down the size of the TRT by reducing the number of tracks we will use on the 808. In the spirit of generality we propose adding a new TRACKS command to the CMRDECK processing. The format of the TRACKS command is:

TRACKS,NNNN,E1,E2,...,EN.

This states that NNNN tracks are to be used on equipments E1, E2,...,EN. The corresponding TRT size will be, of course, (NNNN+3)/4. We make the proposal this way so that the 808 can be declared to have as many tracks as are needed. (If rollout tries to use more, any temporary device is used.) We also allow the number of tracks to be set more or less than the normal number for any equipment. This may be helpful someday for some use on systems time. The only real use we expect to make in the Cyber 74 CMRDECK is for the 808 with the following entry:

TRACKS,1000,4.

This selects 1000B tracks (a 200B TRT) for the 808. The 6400 may also elect to set the number of tracks to a lower figure when this command is installed.

This command requires about 40 lines of code in mod SETU2 to SET. SETU2 is the UCC mod which added commands like ON, OFF, and UNIT to the CMRDECK entries. I

expected that we would also have to change some other routines which have to know the TRT size - at least I expected to have to change something in the initialization routines IMS and MSI. But all the routines seem to accept the TRT length as left in memory by SET. If the TRT length in memory differs from that being required at deadstart time, deadstart complains. So the 808 will have to be initialized whenever the TRT size changes. This is no problem. Apparently no other system changes are needed.

SYSTEM MAINTENANCE: People and Procedures

Last Week's Systems Group Meeting - by T.W. Lanzatella

1. K.C. Matthews spoke briefly about some recent temporary system changes designed to help increase throughput.

a. IRO was altered so that rollout files are written to three different devices rather than two. If N is the rollout file size in thousands of words, here is the distribution used:

N<7K	ECS
7K≤N<20K	844
20K≤N	808

b. The AM (available memory) parameter for TXOT jobs was increased from 74K to 110K in hope of decreasing the amount of swapping.

Not enough statistics had been gathered at the time of this talk to determine the effect of these changes.

2. Kevin Fjelsted's proposal to install SSORT as a new CALLPRG type sort utility was accepted in principle, but nobody likes the control card format. Everyone felt that equivalenced parameters ought to be used and that a random sort didn't make very much sense. Kevin was not present at the meeting.

(Kevin has since withdrawn the proposal and is mediating with the program author, Mike Huck, over the specified changes. - ed.)

3. T.W. Lanzatella informed the systems staff that the number of high speed MIRJE ports for staff usage was being reduced to five from nine on the A-rotary (5730) because the number of C-rotary ports had been increased to four. Staff members should begin selecting the C-rotary (5740) before the A-rotary.

4. UCC recently acquired a 512 printer. We may or may not keep the printer depending on its condition. Little else was known at this time.

5. Persons of primary responsibility during systems time were identified.

Tues., Thurs. (4:00 a.m. - 7:45 a.m.)	DWM/TWL
Sat. (4:00 p.m. - 2:00 a.m.)	EJM
Sun. (6:00 a.m. - 12:00 noon)	Open
Sun. (12:00 noon - 4:00 p.m.)	Open

Unless a staff member signs up for the open slots, the building will be secured. The building locks will soon be changed so that when the building is secured, the building doors cannot be opened by systems staff.

//////////

Callprg and Library Tape News - by M. Riviere

On August 16, the following changes will take place among Callprg and Library Tape products:

Howard Kurs will be including a new Callprg index entry in the Cyber section of the index to be used as part of the FUTURE, MNF installation.

This entry will retrieve the current version of SYSIO renamed as MSUIO upon the use of the control statement FETCH,CDCIO. (WOW - ed.)

FUTURE,MNF uses as default library for input/output operations MSUIO (Michigan State University Input/Output). This library, although quite efficient for all the most commonly used tasks, does not process the handling of as many file and record types as SYSIO (CDC input/output library).

Providing a version of SYSIO renamed as MSUIO will direct the loader to load the input/output routines from the renamed CDC library.

Tim Salo will be removing the future entry for REBLOCK and replacing PACKMS with a version with several errors corrected.

The next Library Tape and Callprg modifications will take place on August 29. The deadline for modifications to be requested is August 18, by noon.

//////////

Field Length Changes - by K.C. Matthews

On Sunday, 21 August, the loaders at UCC will be changed so that absolute binaries will no longer load (by default) at the user selected RFL field length. You may recall that this change was proposed after a series of confusing and time-consuming systems meetings this spring. The August 21 date was selected because it was the first weekend after the end of the second summer session. It allows us the most amount of time before fall quarter to straighten out any problems that arise. The 21 August date was announced in the June UCC Newsletter.

There is currently a very short (less than 1 page) writeup called NEWFL which warns users again about the change. It refers them to a longer writeup called MEMORY which attempts to explain how field length is actually assigned. MEMORY explains how RFL= and MFL= entry points may be added to routines to select the proper FL for the routine. We would like to encourage here that persons who maintain things on CALLPRG test their packages to make sure they work correctly. If the program should not run at the ENDS/HHA declared FL in the binary, a correct entry point should be created as shown in the MEMORY writeup.

To put up a program under CALLPRG that can be most easily called by users requires a little care. Remember that the first time a program is called by CALLPRG for execution, the FL used will be the RFL of the job unless there is an FL or an MF parameter in the CALLPRG directive. (The FL parameter specifies a Field Length required by the job, just like an RFL= entry point. The MF parameter specifies the minimum FL needed by the program, just like an MFL=

entry point.) For the first program call, the program is loaded by a CALLPRG call to LDR, so the FL or MF parameter should be used to set up the correct FL before LDR is called. On subsequent calls to the same program, there will be a file around with the program call name. This is also true when past and future versions of programs are used. When the program is executed directly from a file, one of the loaders is called to examine the binary. In these cases, the ENDS/HHA fields or an RFL= entry point specified the required field length for the program. If the program has a minimum FL, the MFL= entry point will be interpreted by the loaders. In either case, the correct field length will be set by the loader for program execution.

Thus you must first determine whether your program always needs a certain fixed amount of FL to run (the Required FL), or whether it needs a certain minimum amount with larger amounts of memory needed for larger tasks. Most programs fall into the first category. If they are written in a higher level language, an absolute version will be created with one of the loaders. This binary will contain an ENDS or an HHA field, and the binary will not have to be changed by you. If you require a different FL than that declared by the loader, or if you require a minimum FL, you should create a special RFL= or MFL= entry point. If the program is written in COMPASS, you should make sure to declare the correct RFL= or MFL= entry point. Finally, each program called directly via CALLPRG (but not FUTURE, PAST or FETCH versions) should have an FL or MF entry in the CALLPRG directive.

Here again are the steps needed for an effective absolute binary program under CALLPRG.

1. Determine if the program needs a fixed amount of FL to run (usually the FL required to run in the loader map). If the program uses a variable amount of memory depending on the task, go to step 6.
2. If the absolute binary was created by either loader it will have an ENDS or an HHA field which specifies the required field length for execution. If the program works at this field length, go to step 4.
3. If the absolute binary was not created by a loader, or if you want the program to run a different FL than this, insert an RFL= entry point. If your program is not written in COMPASS, using program REENTRY is the easiest way to insert the entry point.
4. If your program is not PAST, FUTURE, or FETCH, use the FL parameter on the CALLPRG directive. Make FL = to the required FL you have determined.
5. STOP. Do not use an MFL= entry point or an MF parameter if the above steps apply to your program.
6. (From step 1 above.) Create an MFL= entry point in the binary. This is the minimum FL which the loaders will assign for the program. If the job RFL is greater, that FL will be used. If your program is not written in COMPASS, using program REENTRY is the easiest way to add an entry point.
7. If your program is not PAST, FUTURE, or FETCH, add an MF parameter to the CALLPRG directive. You should have MF = the minimum FL you have determined above.
8. STOP.

//////////

Cyber 74 Deadstart Dump Analysis - by K.C. Matthews

Saturday, 23 July - Monday, 8 August

Monday, 25 July

10:10 (DD-10)
Scopes went blank. The engineers examined the machine. It looks like another case where one bank of PPU's was unable to access central memory. This seems to happen every month or two.

Wednesday, 27 July

23:18 (DD-11)
Monitor busy. The operators could get no commands to be accepted. The dump revealed that CPUMTR was hung in a loop at address 77. It should only go there when CEJ/MEJ is disabled. The FL in the CPUMTR exchange package was also too low - 100K instead of 400K. No other exchange packages look bad.

23:48 (DD-12)
Errors on disk 13 showed up under BATCHIO. The errors were address errors, indicating that BATCHIO was trying to read an illegal unit. This error is usually caused by a bad linkage byte in a disk file, linking the current sector to an illegal sector. There was probably such an illegal byte on a Plot file.

00:08 (DD-13)
Same problem. Operators suspected a problem with one plot file and purged it. The error went away.

Wednesday, 3 August

12:29 The system hung up because some PPU program was not pausing for storage move. No dump was taken.

00:04 (DD-13)
Disk drive for device DN20 began giving solid errors. The disk would not work on other driver. The CDC engineer was called, and could find no problem with the drive. Device DN20 was loaded back to a scratch pack.

Sunday, 7 August

21:09 System hung with 1SJ trying to get field length for a new job. No dump was taken.

Monday, 8 August

14:56 (DD-13)
An experimental deadstart tape of KCM's was in the system. Both halves of the 808 were attached to the Cyber. A pointer (word 16B) was damaged in central memory, which tried to force rollout files to equipment 10 (the display console). KCM moved pointer word to a less volatile location.