This document describes the features and facilities available under the proposed DISPDOS dispersed DOS system. The basic intent of DISPDOS is to provide a comprehensive package of system software to permit a group of 5500s to access disk files physically located on one adjacent 5500 as though those files were directly attached. The goal is to provide multiple processing resources on a single coordinated data base, allowing SORTs, Datashare, RPG, COBOL and DATABUS programs read-only access to a set of shared files while still allowing for read/write/update access to a set of privately held files.

Communication Technique

All communication between 5500s within the DISPDOS system is implemented over a Fast Resource Intercommunication Link (hereafter referred to as FRIL) operating at a nominal speed of 2.5 megabaud over coaxial cable. The FRIL combox has two 512-byte buffers; buffer transit time through the FRIL network is on the order of two milliseconds. Because the FRIL net is inherently multidropped, only one FRIL combox per processor is required. It is believed that the use of the FRIL net for interprocessor communication should result in relatively slight degradation of system performance over the use of strictly local disk attachment, for most operations.

Programming Compatibility

One extremely important aspect of DISPDOS is the speed with which it can be implemented and made a part of the product line. Existing software and hardware must be used as much as possible. Minimization of special software will also greatly reduce the support effort required for the maintenance of DISPDOS. It is believed that most existing software can be integrated into the DISPDOS scheme without modification.

Program Access to Files

Under DISPDOS, it shall be possible to run, for example, two SORTs and two sixteen-tube DATASHAREs off of one Datacentral machine in a five-processor network. Datacentral may use either a twenty-megabyte WANGCO or a 9370-series MEMOREX/TELEX disk system. Anywhere from two to eight physical drives are supported for Datacentral. Dataslave machines need not be equipped with a disk controller at all; however, it will be possible to attach a

controller and one or more disk drives
(WANGCO/TELEX/MEMOREX) to each Dataslave. Programs running
within the Dataslave machine can access both files contained
on the disks maintained by Datacentral and at their own
local disks simultaneously.

In addition to the remote/local file access (where
local disks might be used for, for example, SORT scratch
files or the like), several classes of files exist within
the Datacentral machine. The first class of files consists
of those files which are accessible to every Dataslave.
Normally, programming convention will relegate these files
to read-only status, which can be enforced for specific
files, if desired, through the use of the DOS write/delete
protect mechanism. By not using this convention,
programmers will be free to provide shared updating of files
in this class using any enqueue/update protocol they deem
appropriate. The remaining classes of files at Datacentral
are normally private to a single Dataslave. Programs may
create temporary scratch files, and the like, without fear
of conflicting with files in use by another Dataslave.
(Such files will be automatically access-reserved to the
creating Dataslave processor.) However, for special
purposes it will be possible to establish files which are
update-available to only a selected subset of Dataslaves.
This will permit, for example, multiple Datashare Dataslaves
to update a shared database at Datacentral without loss of
integrity. The number of such selected Dataslave subnets
available is limited by the number of disks present at
Datacentral.

Datacentral files which are access-reserved to a
Dataslave can be transferred either to another Dataslave's
access-reserved status or can be made available to all
Dataslaves via a command from the console of the owning
Dataslave processor. No special file naming conventions are
required to guarantee access-reserved status of a file
created by a Dataslave within Datacentral's disk system.
Files being created by a Dataslave may be created either
within that Dataslave's local disk system (if present) or at
Datacentral as determined either by drive number or by
symbolic volume reference under the DOS system commands
which support the symbolic volume naming facility.

Bringup/Takedown of DISPDOS

DISPDOS is brought up at Datacentral by booting DOS.D
in the normal way and issuing a console command to invoke
the Datacentral system monitor. Dataslaves are brought up
in one of two different ways. If a local disk system
exists, DOS.D is booted up in the normal way and the
Dataslave system monitor invoked via a console command. If
no local disk system exists, the Dataslave is established by

placing a tape in the rear deck and depressing the
RUN/RESTART keys. DISPDOS operation at a Dataslave is
terminated via a console command at the Dataslave;
Datacentral operation is likewise terminated via a console
command issuable only at Datacentral. Datacentral operation
is expected to entirely consume the processing and memory
resources of the Datacentral processor; therefore, no "DOS
partition" or problem programs can be run at Datacentral
while DISPDOS operation is in progress. However, when
DISPDOS is not running, the Datacentral machine can be
booted and DOS.D run in the normal fashion without
impairment. When Datacentral is not running, the
Datacentral processor can not only access (and BACKUP) all
files regardless of the owning Dataslave, but can identify
the owning Dataslave or Dataslave subnet associated with
each of the files in its disk system.

It is also possible at Datacentral to additionally
restrict the disks to which each Dataslave has access; if,
for example, one subset of the Dataslave processor net is
not desired access to files on disk "A" and another subnet
of Dataslave processors is desired to only be able to access
files on disk "B", this is easily established by console
command issued at Datacentral.

Memory Availability

In Dataslave processors, 48K of user memory space will
be available for program use on those systems with the
optional local disk system. On processors without the
optional local disk system, 44K of user memory space will be
available. (However, SNAP, LINK, and most other programs
which test for the end of memory should work within the
reduced space without difficulty). Since system memory is
used for the Datacentral/Dataslave system monitors, DISPDOS
and PS cannot be used at the same time. On the other hand,
the DISPDOS facility largely satisfies the multiple
partition requirement and so this restriction is not
considered significant.

DISPDOS Network Size

Up to six Dataslaves are supported when a 5500
processor is used as Datacentral. Use of a Tower processor
should allow supporting at least eight Dataslaves. It is
possible that fifteen Dataslaves could be supported under
Tower. At this early stage of system development, it would
be premature to state firmly the maximum size of the network
obtainable. It is anticipated, however, that the amount of
disk processing activity at Datacentral will tend to be the
limiting factor due to performance reasons; the use of
local disk storage at Dataslaves is expected to relieve this
congestion to some degree.

Another design goal for DISPDOS regards system performance under varying load. It is considered desirable to achieve a "graceful degradation" as system utilization increases, rather than a catastrophic thrashing characteristic of some other vendors' systems.

Anticipated Development Time Required

It is believed that the system outlined above can be developed, coded, debugged, documented and ready for release within two months from project initiation, assuming reasonable availability of hardware. This figure is believed to be realistic and allowing sufficient leeway for timely completion of the project.