

# "PIPELINE" SYSTEM REVIEW

Version 7 - May 15, 1984

## 1. INTRODUCTION

The purposes of this document are:

- 1- To describe the Pipeline system as it is currently planned and, by so doing, to elicit suggestions and criticisms.
- 2- To describe the current status of the system, the nature of work in progress and the "game plan" for completion.
- 3- To introduce potential users to system components that are currently usable.
- 4- To list known problem areas and to report solutions when they have been achieved.
- 5- To draw attention to design or implementation issues that have not yet been resolved.

To meet these objectives this review has been divided into the following parts:

- 1- Introduction.
- 2- The Pipeline System Plan
- 3- Planned Use of the Pipeline: Step-by-step
- 4- Current Status of the System.
- 5- A User's Guide through the Current System
- 6- Plans for the DISPLAY Subsystem
- 7- Miscellaneous System Implementation Details
- 8- A List of Unresolved Issues
- 9- Appendix: Changes from the Previous Version

## 2. THE PIPELINE SYSTEM PLAN

### 2.1. Purpose of The Pipeline

The main objective of the VLA Pipeline System is to provide the mapping throughput necessary to operate the VLA at full efficiency, including the full spectral line capacity designed into the hardware.

The most important factor governing the Pipeline throughput is obviously the computer capacity as measured in I/O bandwidth and in compute cycles. The major concern in the design of the Pipeline has been to provide this capacity. However, this alone is not enough to make the Pipeline successful. If the Pipeline is to have its full impact on the amount and the quality of science done with the VLA, it also has to be one of the astronomer's tools. It will have to be able to cope with less than perfect data. It must have the flexibility to perform the kind of mapping functions desired by the astronomer based on his assessment of the data and his objectives. Finally, it must enable him to select a subset of data to take away for further analysis. This subset should contain all the relevant information but be sufficiently reduced in volume to take away on a reasonable number of magnetic tapes. For example, a spectral line observer should be able to see the full field observed in all of the channels observed with all significant instrumental errors removed before he has to make the (unfortunately) necessary decision to reduce the database until it only contains the object of interest.

### 2.2. Quantitative Objectives

Please note that these are OBJECTIVES, not currently achieved performance figures. For a description of what is available at this time turn to section 4 of this report.

The quantitative goal that has been set for the Pipeline is expressed in terms of the number of input data values that should be accepted by the system in a continuous stream while it is concurrently producing, at a specified rate, maps of a given size.

At this time the goal is to be able to accept 6000 complex numbers per second while producing one 2048 by 2048 map per minute.

By way of comparison, 256-channel data from all 351 baselines, averaged over 10 seconds, corresponds to a data rate of 8985.6 complex values per second.

Note that the stated objective does not provide for any other simultaneous activity such as the transfer of data or maps between tape and disc or between the Pipeline and any other machine.

The Pipeline system will be capable of producing continuum maps that are up to 4096 x 4096. The user will be able to copy maps of this size or put them onto FITS tapes but most other functions in the system will be limited to processing a 1024x1024 subsection.

Work on 4096x4096 maps can be supported only if it is done rarely. The typical map size for which the system is designed is 1024x1024. The system will be capable of producing rectangular maps.

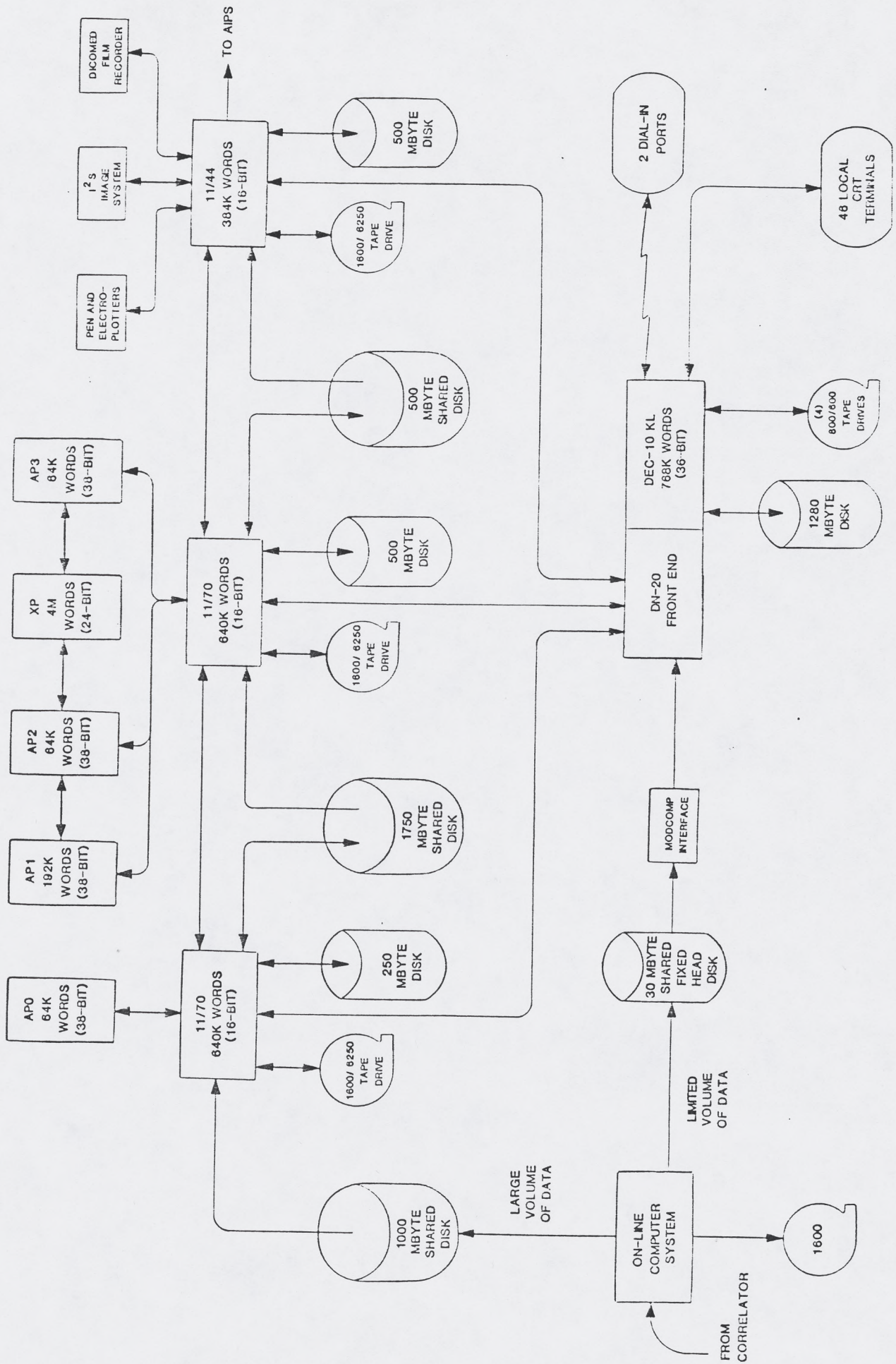


FIGURE 1

### 2.3. Hardware Configuration

The "Pipeline" is a system rather than a piece of hardware. Components of the Pipeline software reside on 6 different computers. Figure 1 is a diagram of the Pipeline hardware configuration. The computers involved in the Pipeline system are:

1- A ModComp which serves merely as an interface to the on-line (data-acquisition) system. (The on-line ModComps also do some preliminary sorting of spectral line data for the Pipeline.)

2- A DEC-10/KL1091 used for calibration and editing. Control of Pipeline operations (scheduling) is exercised by the user at terminals connected to this machine.

3- A PDP-11/70 (SORTER) used for sorting the visibility data. This machine is also used for miscellaneous "housekeeping" functions.

4- A PDP-11/70 (GRIDDER) which acts as a host machine for array processors and transpose memory which do the actual gridding and mapping.

5- A PDP-11/44 (DISPLAY) which supports various display devices. Links from this machine to the DEC-10 let the user participate in an iterative process wherein data is edited, mapped, evaluated, then re-edited for another round of processing.

6- A PDP-11/44 (WORKER; not shown in Figure 1) which is intended for program development and as a DECnet gateway to the outside world.

The name "Pipeline" is sometimes used (misleadingly) to refer to just the first three PDP-11's. That practice will NOT be followed in this document. In particular, since the DEC-10 is necessary for data calibration and user communication, it is an integral part of the Pipeline. Each of the four PDP-11's is identified by a name that reflects the specialized task it performs within the system. Those names, SORTER, GRIDDER, DISPLAY and WORKER will be used in this document.

The amount of memory on each machine is as follows:

SORTER	PDP-11/70	640 K 16-bit words	(1.28 Mbytes)
	AP-0	64 K 38-bit words	
GRIDDER	PDP-11/70	640 K 16-bit words	(1.28 Mbytes)
	AP-1	192 K 38-bit words	
	AP-2	64 K 38-bit words	
	XPOSE MEM	4,000 K 24-bit words	
	AP-3	K 38-bit words	
DISPLAY	PDP-11/44	384 K 16-bit words	(0.76 Mbytes)
WORKER	PDP-11/44	704 K 16-bit words	(1.41 Mbytes)
DEC-10		768 K 36-bit words	

The control information that is required to make these individual machines act as one logical unit flows through the system via a set of DECnet links. (See Figure 2.) The DEC-10 is linked through its DN20 "front end" with each of the PDP11's. There are also 3 direct DECnet links between GRIDDER and SORTER, DISPLAY and WORKER. Links to the outside world go through WORKER. All of these links operate at 56 kilobaud; it is planned to replace the existing communications hardware between the PDP-11's, the DEC-10 and the VLA VAX computers with Ethernet hardware which will significantly increase the communication speed. (NOTE: It might appear that there are redundant links to and through the DEC-10 but the TOPS10 operating system supports only an outdated version of DECnet that does not provide any "routing" facility.)

The connection between the on-line ModComp system and SORTER consists of four shared-access disks. One ModComp processor is dedicated to communication with a DN20 which is the DEC-10 "front end" machine. That ModComp is buffered from the on-line ModComp machines by a shared-access fixed-head disk. The buffer capacity is several hours worth of data if (as planned) only calibration data is transferred through that route to the DEC-10. The bulk of the data moves from the on-line ModComp system to the SORTER machine via 4 250-megabyte shared-access disks.

Data flow from SORTER to GRIDDER and from GRIDDER to DISPLAY also uses shared-access disks.

The Pipeline system contains a total of 17 disk drives exclusive of the fixed-head disk in the ModComp system and exclusive of the disk space used on the DEC-10. All of the drives are essentially identical. Each has a capacity of approximately 250 megabytes. Four of the disk drives are used for the "mail box" link between the on-line ModComp system and SORTER. SORTER places the data on six drives which can be read (but not written) by GRIDDER. GRIDDER is equipped with three disk drives which can be read (but not written) by DISPLAY. The remaining drives are accessible by only one machine each. There is one such drive on SORTER, on GRIDDER, on DISPLAY and on WORKER.

Each of the four PDP-11's has its own 1600/6250 bpi tape drive. ModComp tapes will probably not handle archiving at the maximum designed rate of 6000 line-channels/second. No decision has been made on alternative equipment.

Four FPS-120B array processors are used in the Pipeline. One is attached to SORTER; the other three are available to GRIDDER. Each of the AP's on GRIDDER contain IOP devices whereby it communicates directly with the transpose memory and with the other APs without host intervention.

A detailed list of the graphics devices intended for the Pipeline will be presented later in a section devoted to the DISPLAY subsystem.

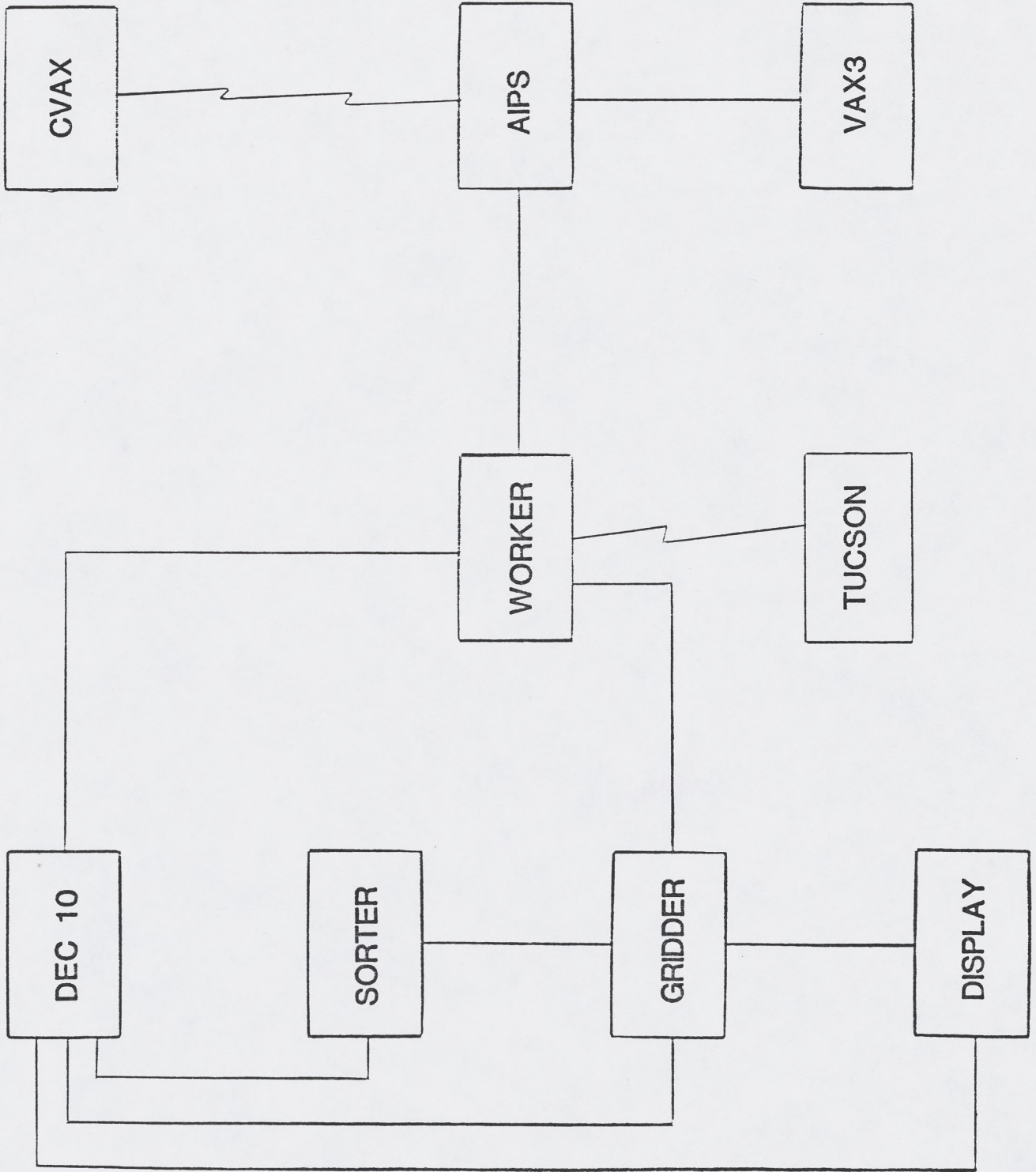


FIGURE 2

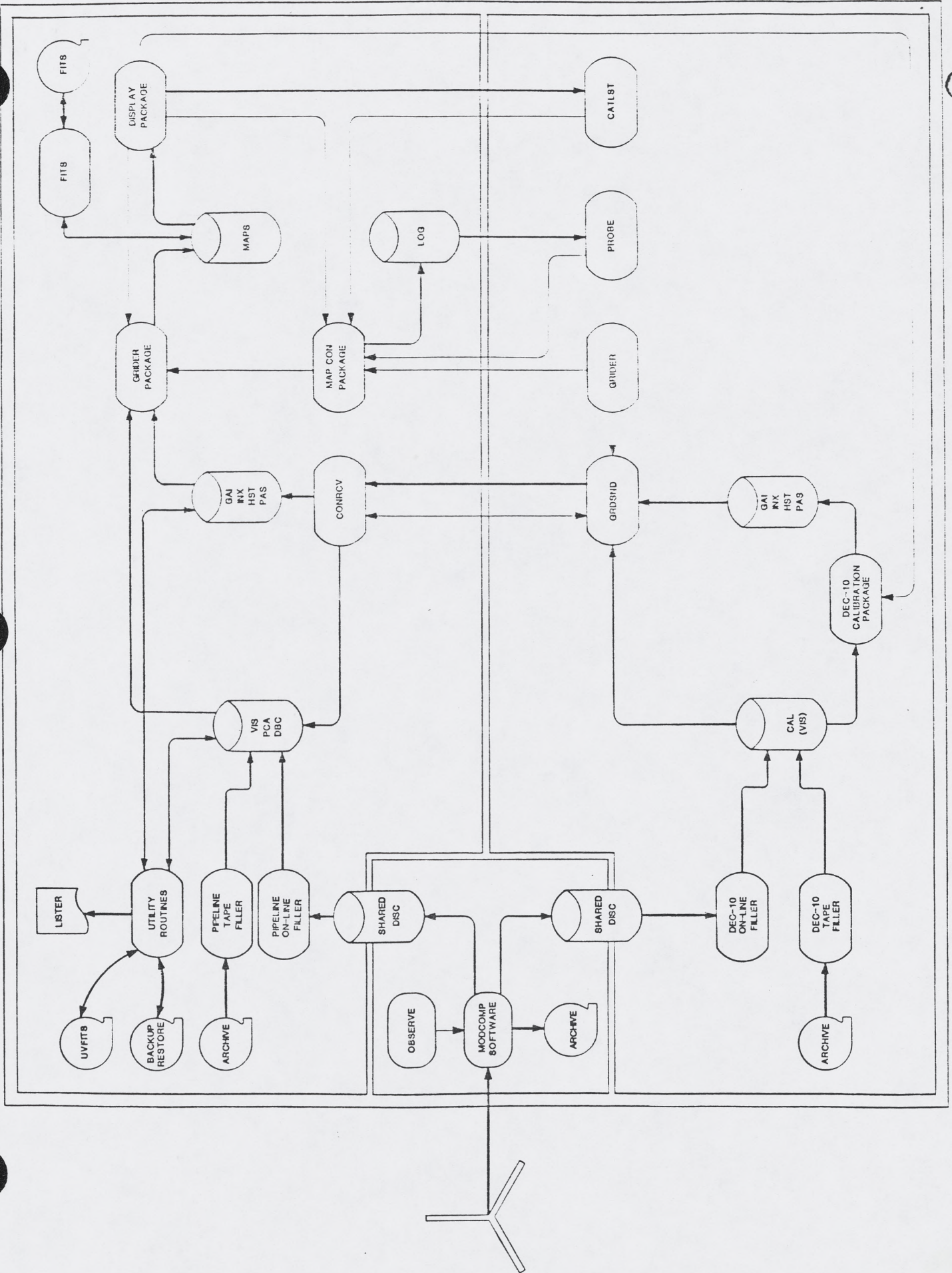


FIGURE 3



## 2.4. Overview of Data Flow Through the System

The following paragraphs may be more meaningful if the reader refers to Figure 3 which is a schematic representation of data flow through the Pipeline. The lighter lines in Figure 3 show the interaction (flow of control) between various Pipeline programs.

During observing, all visibility data will be transferred from the on-line ModComp system (where it will have been partially ordered) to the Pipeline's SORTER machine where further ordering will occur before the data is stored on disks. Visibility data for calibrator observations only will be sent to the DEC-10 (as well as to SORTER). All normal information, but no visibility values, on all the other sources that were observed will also be sent to the DEC-10.

At the same time that it is transmitting data directly to SORTER and to the DEC-10, the ModComp system will write a copy of all visibility data (for sources and calibrators) on an Archive tape. The tape can be read by programs on the DEC-10 and on the Pipeline machines as an alternate (asynchronous) data input source.

Calibration, correction and flagging will be done in the DEC-10 under the observer's direction and initiative (this is NOT an automatic process). The quality of the visibility data stored on SORTER's disks (the data for sources) can be ascertained by displaying various pseudo-maps (baseline vs time plots for example) that can be created in GRIDDER for presentation on DISPLAY. Flagging information can be sent from DISPLAY back to the DEC-10.

The results of calibration and editing are recorded in a set of gain, index and history files which will reside initially on the DEC-10 disks.

When a map is to be made, the observer must send a request to the Pipeline scheduler (MAPCON). A DEC-10 program (called GRIDER) is available for this purpose.

The Pipeline system will respond to a mapping request by copying the gain, index and history files from the DEC-10 to the GRIDDER disks. Actual mapping can begin as soon as those files have been copied but, if the Pipeline is busy answering earlier requests, it might not begin until much later. Some requests may tie up the mapping facility for hours on end.

The observer can monitor the progress of mapping requests via a DEC-10 program (called PROBE) that reads the Pipeline's log file.

When the Pipeline is ready to process a request, the visibility data for sources will be obtained from the disks shared by SORTER and GRIDDER. The appropriate selection of data, like almost all of the gridding and mapping computations, will take place in the array processors that are peripherals on GRIDDER. The maps (and beams) that are produced will be placed on GRIDDER's disk.

The DISPLAY machine has direct (shared) access to the disks upon which GRIDDER stores its output maps. The DISPLAY machine can also be used to submit a new mapping request to the Pipeline scheduler thereby closing a loop between data editing, mapping and displaying. Requests for new maps submitted from the DISPLAY machine will have priority over requests originating in the DEC-10.

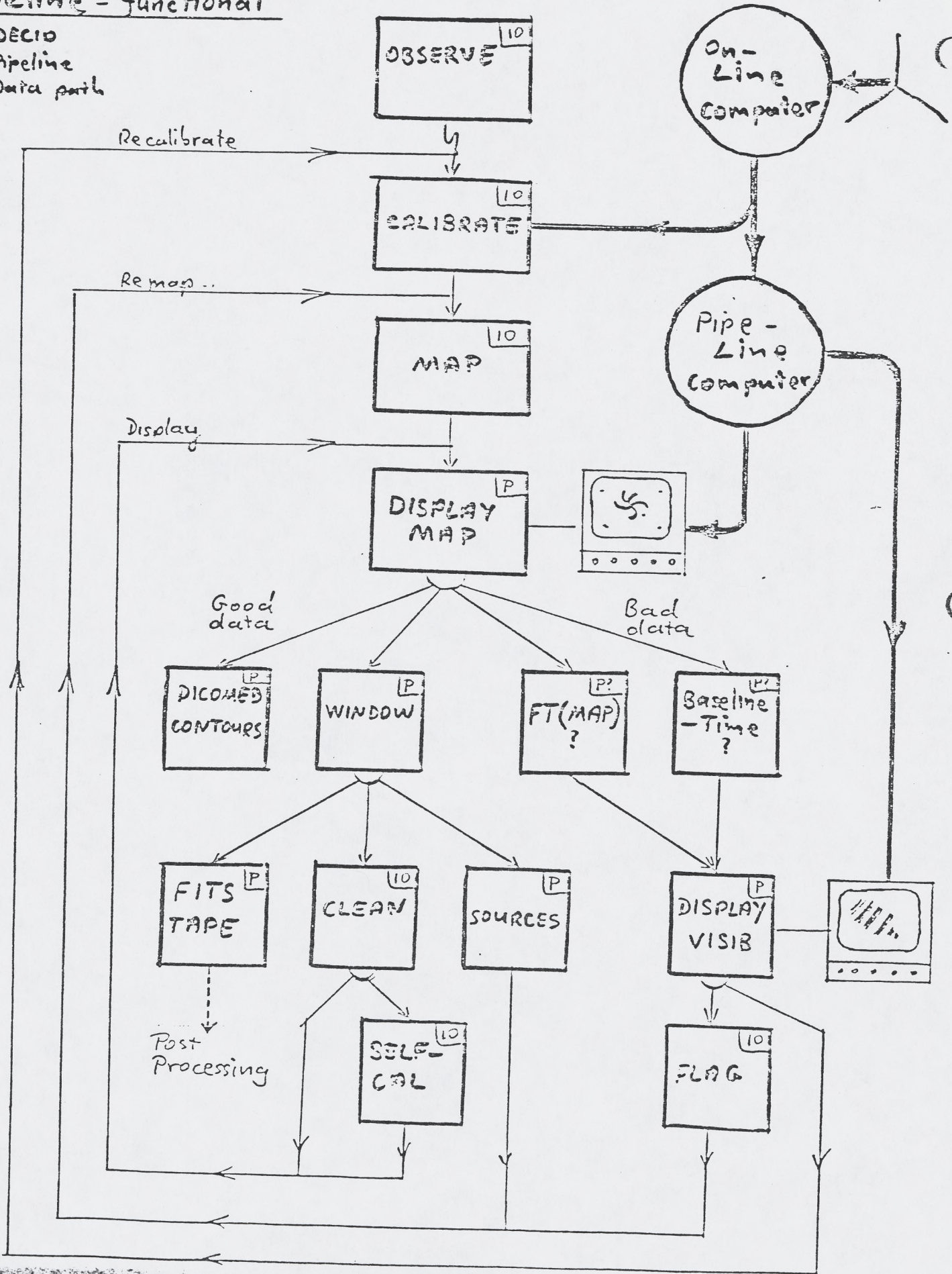
Finally, maps will be written to FITS tapes either on GRIDDER or DISPLAY. Calibrated visibility data can be written on UVFITS tapes by a program in the SORTER machine. (Raw visibilities can also be written from disk to tape by a utility routine on SORTER but this will not usually be done since re-filling onto the SORTER disks from the ModComp archive tape will be just as fast as restoring visibilities using the utility program).

There is no selfcalibration program planned for the Pipeline system. For spectral line data it might be worthwhile to combine the facilities of the AIPS system with those of the Pipeline. Visibility data and a dirty map for one channel could be copied from the Pipeline system to the AIPS system. There one could use the "selfcal" procedure to get a gain table which would be returned to the Pipeline in order to create maps for all the other channels.

Figure 4

Pipeline - functional

DECIO  
Pipeline  
Data path



### 3. PLANNED USE OF THE PIPELINE: STEP-BY-STEP

#### 3.1. Overview

Figure 4 is a functional overview of the Pipeline system from the user's viewpoint. In some ways the Pipeline will appear to be similar to the older MAPPER system. The user will run a program on the DEC-10 to request a map making operation. The resulting map will be placed on the disk connected to a PDP-11 (GRIDDER). In some ways displaying the maps that are produced by the Pipeline will be similar to the use of the former IMPS system. However, use of the Pipeline system will be different from use of MAPPER/IMPS in several ways. It will not be necessary to transfer maps from one PDP-11 to another via DECnet links or FITS tapes in order to display the maps since the DISPLAY system will have direct shared access to the GRIDDER disks which store the finished maps. Also, after looking at a map with the DISPLAY system, the user will be able to submit additional mapping requests directly from the DISPLAY system rather than being required to go back to the DEC-10. The following sections describe the appearance of the Pipeline system from the point of view of the user.

#### 3.2. Input to Programs (Standard Commands)

Pipeline system functions which are invoked by running programs on the DEC-10 will use the DEC-10 standard commands.

Some Pipeline system functions involve the initiation of programs that run on one of the PDP-11's. This will include such things as FITS tape reading/writing and loading visibility databases from tape. All of these programs will be run from a DEC-10 terminal using a program called PIPELN which will give the user simple and direct communication with the required PDP-11 programs. The Pipeline user will never need to use a PDP-11 terminal (except for the DISPLAY system, of course). The PDP11 programs will be written to use a command scanner which provides standard commands that work exactly like the DEC-10 standard commands.

The user interface for DISPLAY system functions will also use standard commands but it will also contain display-oriented and menu-selection input. This is described in another section.

#### 3.3. Preparation of Observing Specification

The user will prepare his observing specification by running the DEC-10 OBSERV program just as is currently done.

### 3.4. Getting Visibility Data into the Pipeline

#### 3.4.1. DEC-10 AND SORTER VISIBILITY DATABASES

The Pipeline system will make use of visibility databases on both the DEC-10 and the SORTER computers. These databases will differ in both content and organization.

The user's visibility database on the DEC-10 will be very similar to the current DEC-10 visibility database. It will consist of an index file, a gain file, a visibility file, an antennas file, a history file and (for spectral line observations), a passband calibration file. The index and gain information will be identical to the current format. Complete visibility data for calibrators will be stored just as in the current system. However, for non-calibrator sources, no visibility data will be stored. The data will be stored in baseline-time order as is currently done on the DEC-10.

Complete visibility data for both calibrators and non-calibrators will be stored in SORTER. Also, the data stored in SORTER will be partially sorted according to the absolute value of the "U" coordinate. The database for each user will consist of a visibility catalog file (which is different from a map catalog file) and a set of visibility data files. Each visibility catalog record will describe all the data for a set of scans which have unique values of source name, user ID and frequency. For spectral line data, each visibility catalog record will describe a group of eight frequency channels. Thus, a spectral line observation may require multiple entries in the visibility catalog. Note that this organization will make it possible to delete visibility data that pertains to just a single source.

#### 3.4.2. FILLING VISIBILITY DATABASE DURING OBSERVING

During normal operations, all of the observed visibility data will be automatically sent from the ModComp system to SORTER. Also, the visibility data for calibrator observations will be automatically sent from the ModComp system to the user's disk area on the DEC-10.

The full data rate produced by the ModComps will be about 6000 channel-baseline measurements per second, where each measurement is a four byte number. At this rate, SORTER's disks can be filled in about ten hours. It is expected that 20 second or 30 second averaging will often be done in the ModComps, which will reduce the data rate by a factor of two or three over the maximum rate.

### 3.4.3. FILLING VISIBILITY DATABASE FROM TAPE

If for some reason the on-line visibility database filling did not occur, then the database can be filled from tape. To fill the DEC-10 database, the user will run a modified version of the current DEC-10 tape FILLER program. To fill the SORTER database, the user will run a SORTER program called DBFILL. This program will not be able to do data averaging as it fills the database. In the full spectral line case with 10-second averaging, the data rate may be too high to be written out on ModComp archive tapes. In this case the data will have to be sent to SORTER in real time. In this situation, SORTER can be considered to be an essential part of the on-line system. Although the ModComps will be able to do data averaging if desired, they will not be able to send unaveraged data to SORTER at the same time averaged data is being written to tape, i.e., the same averaging interval must be used for data sent to SORTER and data sent to tape.

### 3.4.4. DELAYED DATA PROCESSING

For observations involving very large databases, processing will have to be done immediately after the observed data is transferred from the ModComps to SORTER. However, for small databases, it may be possible to dump the raw visibility data from SORTER to tape and then restore the data to SORTER's disks and process it at a later time.

## 3.5. Listing Visibility Data

The user will be capable of listing visibility data stored in the SORTER ("Pigeon Hole") database. Currently it is possible to list parts of the SORTER visibility database by running the SORTER program DBUTIL. The crude listing capability of this program is designed for program debugging and not for listing in a usable fashion for astronomers. It is currently possible to display baseline-time data using the DISPLAY system.

## 3.6. Calibration and Correction

### 3.6.1. APPLYING KNOWN CORRECTIONS - GTBCOR

Applying known corrections to the visibility data will be done by running the DEC-10 program GTBCOR just as is currently done. This will modify the gain file stored on the DEC-10 disk to correct for time errors, position errors, frequency errors, shadowing, etc.

### 3.6.2. DETERMINING ANTENNA GAINS - ANTSOL

Determining antenna gains will be done by running the DEC-10 program ANTSOL just as is currently done. This will examine the calibrator visibility data stored on the DEC-10 disk and produce a .CAL file on the DEC-10 disk.

### 3.6.3. APPLYING CALIBRATION - GTBCAL

The calibration information contained in the DEC-10 .CAL file will be applied to the DEC-10 gain file by running the DEC-10 program GTBCAL just as is currently done. The gain table will be shipped to SORTER at the time map requests are made. Application of the gains will occur on the fly during gridding in the APs.

### 3.6.4. PASSBAND CORRECTION

The passband corrections for spectral line data will be calculated by SPECTRE just as is currently done. The passband file will be shipped to SORTER at the time map requests are made. Application of the corrections will occur on the fly during gridding in the APs.

### 3.6.5. POLARIZATION CALIBRATION - POLCAL

The polarization corrections will be calculated from calibrators stored in the DEC-10 database. The corrections will be stored in the gain file in exactly the same manner as is currently done.

## 3.7. Pipeline Control from the DEC-10

### 3.7.1. MAPPING REQUESTS

The user will request that maps be made by the Pipeline system by running a program on the DEC-10. The mapping request will be entered into a queue of requests to be processed. After the request has been processed, the finished map will be stored on GRIDDER's disk.

In a single request, the system will be capable of producing just a map, just a beam (antenna pattern), or both a map and a beam. Also, a single request will be able to produce all polarization maps. For the spectral line case, a single request will make an entire spectral line map data cube.

The mapping requests shipped to GRIDDER will result in the transfer of the gain table (file) to GRIDDER for subsequent application to the data.

### 3.7.2. OTHER PROCESSING REQUESTS FROM THE DEC-10

The user will be able to make other requests for Pipeline processing by running programs on the DEC-10. This will include such things as cleaning, writing magnetic tapes and deleting maps and databases.

### 3.7.3. CHECKING PIPELINE SYSTEM STATUS - PROBE

The DEC-10 PROBE program will allow the user to check on the status of the Pipeline system (i.e., GRIDDER processing). Log files of the status of various jobs will be created on each system. This file will be accessed by PROBE.

## 3.8. Visibility Data Editing

Another function of the DISPLAY system will be to allow the user to display and interactively edit visibility data. To do this the user will first make a "mapping" request which specifies that a baseline-time array is to be produced rather than a map array. This will result in an array of data in which each row of the array will correspond to a different correlator. The pixels across a row will be the visibility values as a function of time for that correlator. This array can then be displayed on the IIS with intensity-hue encoding, i.e., phase will be encoded as the displayed hue and amplitude will be encoded as the displayed intensity. A bad correlator will then show up as a horizontal pattern on the display. Interference will show up as a vertical pattern on the display. The user will then be able to edit out the bad data by pointing to it with the IIS cursor. It will be possible to flag individual correlators on a scan basis, although it is not clear where these correlator flags will be stored. The flagging of individual visibility records may be supported in the future.

## 3.9. Removing Data and Maps from the System

### 3.9.1. REMOVING VISIBILITY DATA FROM THE SYSTEM

It will be possible to dump raw uncalibrated SORTER visibility data to tape by running the SORTER program DBUTIL. This program will also allow deleting data from SORTER's disk.

The UVFITS program will provide a standard format tape that can be used to transport calibrated visibility data to other machines at any site.

### 3.9.2. REMOVING MAPS FROM THE SYSTEM

Maps will be written to tape by running the FITS program on GRIDDER or DISPLAY.



### 3.10. Miscellaneous

#### 3.10.1. PDP-11 DISK USAGE

Within each of the PDP-11's (SORTER, GRIDDER, and DISPLAY), the data belonging to a given user will be kept separate from the data belonging to all other users. Although each computer will have more than one disk drive, the user will not be concerned with keeping track of which drive contains his data. From the user's point of view, the collection of disk drives on a single computer will appear to be a single large pool of disk space.

The user of the DISPLAY system will of course be aware of the fact that some maps are being stored on GRIDDER's disk pool and some maps are being stored on DISPLAY's disk pool.

#### 3.10.2. MAP NAMING CONVENTION

(See the May 19th, 1983 memo distributed by R. Ekers on this subject.)

#### 3.10.3. DISPLAY SYSTEM USE

(See the next Section)

#### 3.10.4. ADDITIONAL MAP PROCESSING

The Pipeline system is not expected to provide all of the processing and analysis that will be needed by the users. Thus, it is expected that users will write their maps (or appropriate map subsections) onto FITS tapes and then continue their processing at their home institution or on one of the AIPS systems at NRAO.

## 4. CURRENT STATUS OF THE SYSTEM

### 4.1. Hardware

The Pipeline hardware is now in its final configuration as shown by figure 1. Since the last issue of this document, the main computer room at the VLA site has been re-arranged to make the final hardware connections.

There is no DECnet link between DISPLAY and the DEC-10. (There is a DECnet link to the SORTER/GRIDDER combination).

The Telex tape drives do not work reliably at 6250bpi. They do work at this density most of the time and hardware improvements are underway but users are encouraged to use 1600bpi for the present.

The image storage system and the control panel for the DISPLAY machine are still under construction.

### 4.2. Software

This section lists the major software elements that await completion. It also describes temporary replacements that have been implemented. But this section is not meant to be an operations manual; a separate document will describe how the existing software can be used in its present state.

The "on-line" filler program for SORTER is not yet available. This means that data must be filled from tape using the DBFILL program.

The tape filler for SORTER, DBFILL, is working but at a disappointingly slow rate: 2000 visibility values per second for continuum data, and only half of that rate for spectral line data. (Recall that the goal for on-line filling is 6000 values per second.) The tape filler for the DEC-10 has been modified to accomodate the needs of the Pipeline. In particular, only calibrator visibilities are filled into the DEC-10 but proper gain tables and index files are created. All normal DEC-10 programs work properly.

There is no way to flag visibility records in the Pipeline data base on SORTER. There is not even a way to get a useful listing of source data on SORTER since a LISTER program has not been written for that machine. A utility program can be used to check the integrity of files but it was never intended as a lister program, it cannot do flagging and it cannot yet patch "broken" files. There exist programs on GRIDDER and DISPLAY which will make and display baseline-time plots of visibility data.

Limited display software is working on the DISPLAY system. As yet, there does not exist a way for getting automatic transfer of data flags from the DISPLAY system back into the DEC-10.

There is no way to flag visibility data in the Pipeline by creating "history" records. (The DISPLAY system doesn't write into the HISTORY file in any case.)

Only the MAPCON (scheduler) program creates files that can be accessed by PROBE. Other programs need to send messages to MAPCON and/or they need to write files that PROBE can read. There are no individual log files, merely a system file for programmer use (in debugging).

The DEC-10 database organization needs to be revised to accommodate spectral line data. This involves changes to a great deal of software; as many as 12 man-months may be required for this alone.

## 5. A USER'S GUIDE THROUGH THE CURRENT SYSTEM

This section has been completely re-written and issued as a separate document. It is called "Pipeline User's Manual" and is in the VLA Cookbook and the Pipeline Reference Manual (Observer's Reference Manual, Vol.2).

## 6. PLANS FOR THE DISPLAY SUBSYSTEM

### 6.1. The Hardware

#### 6.1.1. DEVICE LIST

The DISPLAY computer peripherals include the following graphics devices:

Image Display	IIS Model 70F. Two 512x512 image planes, 4 graphic overlays, cursor, look-up tables for color & gray scale, ALU and Videometer.
(Image Storage System)	NRAO. Storage for 600 512x512 images with film loop playback on IIS display. ** Not yet available **
Refresh Vector Display	DEC VT-11. Line drawing graphics.
Data Tablet	Summagraphics. IIS and VT-11 cursor positioning.
(Control Panel)	NRAO. Switches, knobs, etc. for controlling image storage system and IIS look-up tables etc. ** Not yet available **
Gray Scale Copier	Honeywell VGR-4000. Gray scale copy of TV image without use of software.
Film Recorder	Dicomed. 4Kx4K resolution color or gray scale images on 4x5 film. 1024x1536 resolution on 35mm film.
Dot Matrix Printer/ Plotter	Versatec D-1200A. 200 dots per inch. 10.5 inch wide plots.
Pen Plotter	Zeta Model 1453B. 12 inch wide drum plotter with four pens.

### 6.1.2. IMAGE STORAGE SYSTEM

The image storage system is being designed and built by Ray Escoffier in Charlottesville. The term "image" here means an IIS image, consisting of IIS 8-bit pixels, not disk maps. The images will be stored digitally on two small Winchester disks and there will be a programmable microcomputer (a Motorola 6803) to control the transfer of images between these disks and the PDP-11/44 host computer or the IIS. The image storage system will be connected to the IIS in such a way that images can be transferred in either direction between the PDP-11 and the image storage system, between the PDP-11 and the IIS, and between the image storage system and the IIS. The program in the PDP-11 will be able to send commands to the image storage system to do such things as load a given set of images into the IIS as a movie presentation with a given frame rate. The microcomputer will then control the loading without the need of constant program execution in the PDP-11. The system will be able to store 600 512x512 images and play them back at a frame rate of up to seven frames per second. Since the system will be controlled by the programmable microcomputer, we will be able to add new functions to the system as they are needed.

The image storage system will initially be used in one of the AIPS systems. At some later time, a copy of the hardware will be built for use with the Pipeline DISPLAY system.

### 6.1.3. CONTROL PANEL

The control panel is currently being designed by Don Wells in Charlottesville. It will be connected to the same microcomputer that is incorporated into the image storage system. This microcomputer will monitor the control panel and then send appropriate commands or data to the IIS when the user changes something on the panel. This organization will thus provide smooth and rapid response to the user without putting a burden on the host computer. By changing the microcomputer program we will be able to modify the functions of the devices on the control panel. Also, the control panel itself will consist of modules which can be replaced or modified. Thus, we will be able to modify the control panel as new ideas for its use are developed.

In addition to controlling the IIS display functions, the control panel will also control the functions of the image storage system.

The control panel will initially be used in one of the AIPS systems. At some later time, a copy of the hardware will be built for use with the Pipeline DISPLAY system.

## 6.2. Software

### 6.2.1. IIS SOFTWARE

We will make use of the IIS interface driver for RSX-11M which is supplied with the hardware. We will also make use of the set of basic Fortran callable interface routines which is supplied with the hardware. And, we will make use of the set of "Primitives" routines which was purchased as a separate option. These routines provide higher level applications oriented functions. Some of these routines will need to be modified since we will be using our data tablet rather than the IIS trackball.

### 6.2.2. VT-11 SOFTWARE

We will make use of DEC's subroutine package for handling the VT-11 display. Although this is not a very good package, there seems to be no practical alternative to using it. We have ordered the latest version of this package. Perhaps this will provide an improvement over the version that we currently have in terms of functionality or implementation efficiency.

### 6.2.3. VERSATEC SOFTWARE

We will make use of Versatec's subroutine package for handling the Versatec electrostatic printer/plotter. We have ordered the latest version of this package. It should be functionally identical to the version that we currently have, but it will hopefully include a fix for the bug that causes the output task to die occasionally.

### 6.2.4. DI3000 Graphics Package

The DI3000 graphics package has implemented the GKS graphics standards. This allows the possibility of writing device independent application software.

### 6.2.5. DATA TABLET SOFTWARE

We will make use of the data tablet driver that Al Braun implemented years ago but which was never incorporated into IMPS. This will provide improved response to the data tablet compared with that of IMPS.

### 6.3. User Interface to the DISPLAY System

#### 6.3.1 GENERAL

The user interface on the Pipeline DISPLAY system will be an improved version of the IMPS user interface. The system will display a tree-structured menu of available functions on the VT-11 display screen. The user will point to the desired function with a cursor that is controlled by the data tablet stylus. In addition to the tree-structured menu, there will be an "expert" mode of operation in which a single page (or a small number of pages) on the screen will briefly list all of the available functions. This will allow experienced users to by-pass the tree structure.

After the user has selected a function, a list of current parameters pertaining to that function will be displayed on the VT-11. This will be similar to the "inputs" listed by the DEC-10 standard commands. The current parameters will be the values that were used on the previous execution of the function by the current user. For parameters which can take on only a discrete set of values, all of the possible values will be listed, and the currently selected value will be highlighted. To select a different value for the parameter, all the user will have to do is point to the desired value with the cursor. If the user points to a parameter involving map coordinates, a cursor will appear on the IIS image display and the user will be able to interactively specify the image coordinates. This will be useful for entering such things as the map subsection to be contoured. To change the value of one of the remaining types of parameters, the user will point to the parameter and then type in the desired new value. It will also be possible to reset all parameters to their default values and to save/restore sets of parameter values in named files.

Another part of the user interface on the DISPLAY system will be the control panel which is being built in Charlottesville. This device will consist of a variety of switches, knobs and buttons for interactively controlling the IIS display functions such as look-up table modification for image enhancement, blinking, zoom, roam, etc. The control panel will also be used for controlling the image storage system which is being built in Charlottesville. This will include such things as speed and direction of movie loop play-back of stored images.

#### 6.3.2. INTERACTIVE IMAGE DISPLAY

The DISPLAY system user will of course be able to access maps that are stored on the disks associated with the DISPLAY PDP-11. However, the DISPLAY system user will also have read-only access to maps stored on GRIDDER's disks. This will give the user the ability to immediately see the maps produced by the Pipeline without transferring maps over DECnet connections or FITS tapes. The following is a summary of the display functions that will be available:

1. Load a map into an IIS image plane for display.



2. Load a reduced map (skipping an integral number of pixels) into an IIS image plane.
3. Generate axis labelling and annotated wedge.
4. Intensity-hue display of two images.
5. Compare two images using blink and/or split screen.
6. Display horizontal, vertical, or arbitrary orientation cross sections on the VT-11.
7. Display spectrum at selected sky coordinate on VT-11.

The image storage system will allow the user to see a sequence of stored images displayed on the IIS as a movie loop presentation. This will be useful for looking at a sequence of spectral line channel maps. The image storage system will also allow the user to rapidly save and restore selected IIS images.

#### 6.3.3. HARD COPY OUTPUT

The following is a summary of the types of hard copy output that the system will produce:

1. Versatec or pen plotter contour plot.
2. Honeywell gray-scale copy of IIS image.
3. Versatec copy of VT-11 image.
4. Dicomed gray-scale or color film output.
5. Versatec or line printer print-out of numerical values for a map subsection.

#### 6.3.4. IMAGE PROCESSING AND ANALYSIS

The following is a summary of the functions that will be available for extracting information from maps:

1. Interactively display coordinates and pixel value of the IIS cursor location.
2. Fit discrete source parameters (amplitude, position, size).
3. Calculate mean, rms, histogram for a map subsection.

The system will also allow the user to process maps to produce new maps. The following is a summary of the functions that will be available:

1. Produce map subsection.

2. Map arithmetic, e.g., +, -, \*, /, spectral index, etc.
3. Average or sum of a group of spectral line channels.

#### 6.3.5. MAPPING/CLEANING REQUESTS FROM THE DISPLAY SYSTEM

The various display and analysis functions listed above will be executed entirely within the DISPLAY PDP-11/44. In addition to these functions, the DISPLAY system user will be able to request the same Pipeline processing operations that the DEC-10 user can request, e.g., mapping, cleaning. These requests will be entered into the processing queue along with requests from DEC-10 users. However, the requests from the DISPLAY system user will be given a higher priority than the DEC-10 requests.

## 7. MISCELLANEOUS SYSTEM IMPLEMENTATION DETAILS

### 7.1. ModComp to DEC-10 Connection

Visibility data will be passed from the ModComp system to the DEC-10 through the fixed-head disk just as is currently done. The program on the ModComp side (which runs in BOSS) will be modified so that it does not send complete visibility data for non-calibrator sources to the DEC-10. The DEC-10 real-time FILLER will create the index, gain and visibility files on the DEC-10 disk just as is currently done.

The DEC-10 tape FILLER program will be modified so that it will not put complete visibility data for non-calibrators onto the DEC-10 disk when it reads a ModComp archive tape.

### 7.2. ModComp to SORTER Connection

The connection between the ModComp system and SORTER will consist of four shared access disk drives. On the ModComp side, two of these will be connected to CORA and two will be connected to CORBIN. On the SORTER side, all four will be connected to SORTER.

The files on the shared disks will not be in standard PDP-11 FILES-11 format. Instead, blocks of data will be written and read at pre-defined physical locations on the disks. The data items on the disks will also not be in PDP-11 format. The byte order will be the ModComp byte order and floating point numbers will be in ModComp format. The SORTER program will do the byte swap and conversion to PDP-11 floating point format.

The data transfer system will be initialized by running the PGI program in CORA and CORBIN. This will clear two special locations at the beginning of each disk which are used for communicating control information between CORA/CORBIN and SORTER.

A program called PIGEON will run in CORA and CORBIN to write the visibility data out to one of the shared disks. CORA will output one group of spectral line channels and CORBIN will output the others. Each time there is a source change PIGEON will check the special location on the other disk to see if SORTER is done reading the data from the disk. If so, it will set the special location on the first disk to indicate that it is done writing to that disk and then begin writing to the second disk. A program called DEC-PIGEON will run on SORTER to read the data from the shared disks. At startup, MC-PIGEON will check the special location on the first disk connected to CORA which indicates when it is done writing to the disk. When PIGEON sees that CORA is done writing, PIGEON will begin reading and processing from the first disk. When it finishes processing the data it will set the location on the disk which indicate that it is done reading. It will then process the data from CORBIN in a similar manner.

### 7.3. Visibility Data Sorting on SORTER

The PIGEON programs will write the visibility data onto the shared disks in a partially sorted order. There will be 64 "pigeonholes," where a pigeonhole consists of a range of values in the absolute value of the "U" coordinate. The data within a pigeonhole will not be in any particular order. The PIGEON program in SORTER will further sort the data so that there will be a total of 256 pigeonholes in the SORTER disk database and splits line databases into 8 channels.

### 7.4. Visibility Database Format on SORTER

For each user that has visibility data on SORTER's disks, there will be a single "Database Control" file. The name of this file will be the four digit user number (leading zeroes are required) and the extension will be .DBC. Each record in the DBC file will contain header information and a "pointer" to a separate file in which the actual visibility data is stored. Each visibility data file will be entirely contained on one disk, but a single user's catalog will be able to point to visibility files that are stored on any of SORTER's eight disks.

A detailed description of the SORTER visibility database format can be found in a document written by Bob Duquet. A discussion of the visibility data format can be found in Computer Memo Number 161, "Floating Point Format for Visibilities" by Barry Clark.

### 7.5. SORTER/GRIDDER Disk Packs

It has been recognized for some time that we needed some mechanism to cope with three problems related to disk usage by the Pipeline. The first problem is how to make use of more than one pack for a large multi-file database. The same problem crops up as the need for flexibility in utilizing available disk space wherever it might be. The second problem is how to minimize the impact that would result from the loss of one or more disk drives. The third is what to do about private packs.

The problem is solved by giving a pack name to each disk pack and having system utilities to maintain the pack name to physical disk drive translation tables. Then the Pipeline programs at the user level deal with physical pack names and inner level routines figure out what physical drive the pack is on.

### 7.6. Map I/O

There will be a set of standard subroutines for accessing a map and its associated header information. All programs in the system will use these subroutines for accessing maps.

## 7.7. MAPCON - Control of Operations in GRIDDER

MAPCON is the name given to a set of tasks that control map processing activity on SORTER. MAPCON was originally developed on MAPPER with the following objectives:

- 1) To accept map processing requests from other machines.
- 2) To invoke whatever tasks are needed to fulfill the request. For a given operation the user may specify what class of software is used: SYS, a mature set that has been tested for some time; NEW, software that has been recently changed to provide some new feature or correct some recently discovered bug; or EXP, those tasks that are radically different from SYS or NEW.
- 3) To schedule tasks to make best use of CPU and disk resources, with turnaround enhancement for users who have used the system least in the recent past. Users may specify a relative priority for their requests, in order to achieve a particular execution order, or to minimize their impact on others, if they do not need the results for a day or two. MAPCON provides a measure of disk space management. It always tries to clean the disk of unwanted or temporary files, and if one disk is full it will try and use another. Application tasks need not be concerned with disk space management.
- 4) To prevent different tasks from using the same file or some other non-sharable resource, such as an array processor at the same time. Different tasks communicate with NQTASK which arbitrates contentions as well as handling disk assignments.
- 5) To provide error logging and recovery for tasks that fail, while attempting to maintain useful throughput without human intervention. If a task fails for a reason that MAPCON determines is correctable, the task will be rescheduled with lower priority. If this is not possible the request is dispatched to a queue of bombed jobs. The status of bombed jobs may be examined in great detail, facilitating location of the problem that caused the failure.
- 6) To provide a statistics gathering facility. The execution times for each task invoked to service a request is maintained, as well as a log of significant events that occur while MAPCON is running. The log of each event does not contain all information in requests.
- 7) To provide a utility for other machines to interrogate the system and change its operation. This is provided by the tasks CATLST and PROBE. CATLST permits the transfer of maps to other machines and may be used to delete maps from SORTER. PROBE may be used to examine the queues of requests. It can provide detailed information on the parameters used to define a request and on the causes of an error. PROBE can also be used to delete an unexecuted request or to abort an executing one. It may also be used to gracefully terminate a clean operation. The user interface counterpart tasks currently exists only on the DEC-10 (user CATLST does exist on PDP11).

8) To provide management utilities for manipulating priorities, debugging, and keeping the disks clear of unwanted files. Requests that have failed can often be manually restarted after the cause of the problem has been removed. The management task is called MAPCON.

MAPCON is a flexible framework and is subject to occasional changes as new needs become apparent.

#### 7.7.1. THE REQUEST QUEUE SYSTEM

MAPCON maintains a multiple queue system for scheduling requests. There are (at present) four distinct queues. The first queue is for requests that need, but have not completed database transfer from another machine. The second is for requests that are in the process of executing or are ready to execute. The third is for requests that have successfully been completed. Requests are automatically aged off this queue at a rate that depends upon the activity of the particular user. The last queue is for jobs that have failed for reasons that MAPCON is unable to rectify.

#### 7.7.2. THE MESSAGE AND LOG SYSTEM

Tasks within the MAPCON system can make use of its message and error logging system. When application tasks complete, either successfully, or after detecting an error in input data, or even catastrophically, such as when aborted by the console terminal, status is returned to MAPCON so that appropriate action may be taken. Tasks can also send a number of different types of messages, for example clean requests return data from time to time that may be interrogated by the DEC 10 via the probe utility. Other messages may be useful in debugging application tasks. By providing this service MAPCON removes the need for application tasks to perform FORTRAN formatted output, with a substantial saving in code size. Messages are accumulated by MAPCON with timestamps and are stored in a large circular file that may be interrogated by CONCON.

#### 7.7.3. THE MANAGEMENT SYSTEM

CONCON may be used to interrogate the MAPCON system at a very low level. It can be used to examine and modify both the queue system and the individual request files. In addition, CONCON can turn the individual request streams on and off and initiate shutdowns and startups of the MAPCON system. If the queue system, which is a multiply linked disk resident list structure, becomes tangled as the result of a hardware error, or a system boot at an inappropriate time, MAPCON will normally rectify the problem when it restarts. If this fails, CONCON can strip the queue system down to its individual elements and then relink them, although the original ordering within queues will be lost.

## 7.8. Map Making

The user will submit a mapping request from the DEC-10 using the GRIDDER program. The request will then be enqueued by MAPCON. GRIDDER will obtain the appropriate gain file from the DEC-10 (MAPCON initiates the transfer of the gain file prior to the time when the array processors become available to make the map).

After the gain file is transferred into GRIDDER's disk and the array processors are available, MAPCON will initiate the mapping program in GRIDDER. Data read from the disks shared by SORTER and GRIDDER will be sent to AP-1, which will perform data selection, gain corrections, gridding and convolution. When AP-1 finishes the gridding of a line in the UV plane it will send the line to AP-2, which will perform the FFT on the line and write the result out to the transpose memory. When enough data has been written out to the transpose memory, AP-3 will begin reading columns from the transpose memory and doing the column FFT's. As output map lines become available, they will be transferred to GRIDDER and written out to the disk.

The above description applies to the case where the entire gain table can be loaded into the memory of AP-1. If the gain table is too big, it will be necessary to transfer it from GRIDDER's disk to AP-1 multiple times. The above description also applies to the case where the entire map will fit into the transpose memory, i.e., the map is 2048x2048 or smaller. For larger maps, multiple passes through the system will be required. This will involve transferring the visibility data from SORTER to GRIDDER to AP-1 multiple times.

Additional details on the mapping system can be found in the following Computer Memos by Wim Brouw:

VLA Computer Memo Number 157      February 1981.  
Remarks on the GRIDDER Development.

VLA Computer Memo Number 158      September 15, 1981.  
GRIDDER System. Part 1 - Array Processor Programs.

VLA Computer Memo Number 159      September 15, 1981.  
GRIDDER System. Part 2 - PDP 11/44 Programs.

VLA Computer Memo Number 160      September 15, 1981.  
GRIDDER System. Part 3 - Changes made to FPS Software.

### 7.9. Making Baseline-Time Arrays

Producing baseline-time arrays of visibility data will be similar to producing maps. The data will be gridded onto a baseline-time plane instead of a UV plane. This gridded array will then be output without the need for any FFT's. The B-T files can currently be displayed on the IMPS system (and will be displayed on DISPLAY). The program BASPLT on the DEC-10 is used to initiate the generation of baseline-time files.

### 7.10. Map Cleaning

The B. Clark CLEAN algorithm is implemented and will be the standard CLEAN available on GRIDDER. The current version does not use the transpose memory.

### 7.11. Self-Calibration

Self-calibration will not be implemented on the Pipeline. It will, however, be possible to ship gain tables from AIPS to the DEC-10, probably via tape, and have the gain tables applied during map making on GRIDDER.

### 7.12. Visibility Data Editing

Visibility data will be flagged in two ways. The FLAGER program on the DEC-10 can be used to flag antennas by scan. This flagging information is stored in the index record which is transferred to GRIDDER as part of the map requests made from the DEC-10. Secondly, it will be possible to flag data by correlator from the DISPLAY system. These flags will be shipped to GRIDDER as part of the mapping requests.



## 8. UNRESOLVED ISSUES

### 8.1. Mapping During Observing

How will the user be able to make maps while his observation is still in progress?

### 8.2. Storage of Antennas Files

Will antennas files be stored in the DEC-10 as is currently done? Will they be stored in SORTER? In GRIDDER? In DISPLAY?

### 8.3. Generalized Lists of Coordinates

Should the database design include provision for using a generalized list of coordinates for uniformly storing such things as clean components, fitted sources, star positions, etc?

### 8.4. Transposed Spectral Line Map Cubes

Should the DISPLAY system be able to handle transposed spectral line cubes that are read in from a FITS tape? Should the Pipeline be able to transpose a spectral line cube? Should the Pipeline be able to make a transposed spectral line cube?

### 8.5. Transfer of Saved Mapping Parameters

If the user makes a mapping request from the DEC-10, looks at the resulting map on the DISPLAY system, and then wants to submit another mapping request from the DISPLAY system, it would be desirable for him to be able to easily submit a request which has just a single parameter changed. How will the previous parameter values from the DEC-10 mapping request be communicated to the DISPLAY system? Should this be automatic, or should we require the user to enter a specific command to get parameters from the DEC-10? What about going back to DEC-10 and getting the last parameters that were used in a request from the DISPLAY system?

### 8.6. Mapping Priority for DISPLAY System User

Should the system be capable of suspending a mapping (or cleaning, etc.) operation that was requested from the DEC-10 in order to immediately do an operation that was requested from the DISPLAY system?

### 8.7. Visibility Data Editing

If the user does visibility data editing on the DISPLAY system, how will the flag information become associated the the visibility data? Will scan level flags be incorporated into the index records in the DEC-10? Will visibility record flagging be supported? Will visibility record flagging into the SORTER database, be possible? Or, will flags from DEC-10 index records, flags from the DISPLAY system, and flags from the SORTER database just be collected together and OR'd at map making time?

### 8.8. ANTSOL Solution Display

Will the DISPLAY system need to be able to display the ANTSOL solutions? If so, how will it get access to this data?

### 8.9. Monitor Data Display

Will the DISPLAY system need to be able to display monitor data? If so, how will it get access to this data?

### 8.10. Batch Mode DISPLAY System Operations

Should the DISPLAY system allow concurrent execution of a queue of batch mode operations while the user is doing interactive things? If so, how flexible should this capability be? Should it be possible for a user who is using the display devices to make entries in the queue at the same time that another user is making entries in the queue by running a program on DISPLAY which uses standard commands? Should a log of batch queue processing be produced?

### 8.11. Initiation of Batch DISPLAY Functions from the DEC-10

Should the user be able to initiate batch mode display functions such as making Versatec contour plots or making Dicomed images from the DEC-10? If so, how should this be done? Should the DEC-10 user also be able to examine and modify the DISPLAY system batch queue?

## APPENDIX

### CHANGES TO THE PIPELINE PLAN (as of Version 7)

- 1- Various minor typographical changes or corrections have been made.
- 2- Colloquial names for the computers have been added in section 2.3.
- 3- The document has been updated in a couple of places to reflect the fact that it has been decided that calibrator data will be sent to both the DEC-10 and SORTER.
- 4- The user guide (Section 5) to the present system has been moved to a separate document.
- 5- Section 4 (Hardware and Software status) has been updated.
- 6- Section 6.1.2 has been changed to more clearly imply that the image storage device will store and retrieve IIS images, not maps.
- 7- Self-calibration will not be implemented on the Pipeline, except as noted in the revised section 7.11.