222

VLB ARRAY MEMO No. 394

VLBA CORRELATOR: CONTROL COMPUTER AND OPERATING SYSTEM

T. J. Pearson and M. S. Ewing

October 3, 1984

SUMMARY

This memorandum presents the recommendations of the VLBA Correlator design team for the choice of the Control Computer, Operating System, and Programming Languages for the VLBA Correlator.

The preferred configuration consists of one or two computers of the VAX-11 series for control, data-formatting and program development, with an attached Array Processor and a number of special-purpose computers based on an Motorola 68020 32-bit (or similar) processor with fast floating-point hardware.

The VAX-11 computers will run the DEC VMS operating system. The special-purpose processors will use a commercial real-time operating system.

It is not possible to choose a single computer language to use throughout the Correlator system. Modules which require close integration with the operating system (*e.g.*, handling interrupts, memory allocation) will be written in either VAX-11 Pascal or VAX-11 C; the final choice between Pascal and C has not yet been made. Modules which perform intensive floating-point arithmetic will be written in standard Fortran-77. The VMS operating system allows mixing of subroutines written in different languages. The special-purpose processors will be programmed in either Pascal or C, depending on the availability of cross-compilers for VAX/VMS.

This memorandum does not address quantitative requirements: how many cpu cycles per second or what I/O bandwidth are required. Thus it cannot recommend a specific cpu model. The quantitative requirements will be determined during the design of the VLBA Correlator, currently scheduled to be completed by July 15th, 1985.

COMPUTER CONFIGURATION

The proposed hardware configuration for the VLBA Correlator is described in a forthcoming VLBA Memorandum. In addition to the hardware of the correlator proper, there are four computer systems:

1. Control Computer: the Control Computer controls and monitors the real-time operation of the entire correlator system. It provides the interface through which the operators control the correlation process, and it is responsible for obtaining experiment setup information from the Monitor and Control Database, for controlling and monitoring the Data Playback System (including making sure that the correct tapes are mounted at the right time and computing geometric delays for each station), for controlling and monitoring the Correlation Assembly (including computing model phases to control the lobe-rotators), and for controlling and monitoring the Fringe Processor and Archive Writer.
2. Calibration Processor: the Calibration Processor is required to carry out global fringe fitting of data from calibration sources in essentially real time, so that the calibration

parameters may be determined and applied to subsequent program observations in the Fringe Processor. This processor will use algorithms developed for the AIPS Post-Processing System, and will therefore require AIPS-compatible hardware.

3. Fringe Processor: the Fringe Processor performs routine calibration and filtering operations on the output from the Correlation Assembly before the data are recorded on the archive medium.
4. Archive Writer: the Archive Writer is responsible for collecting and formatting the data output of the Correlator, and for updating the experiment database with information pertaining to the correlation process. At present we assume that the primary archive medium will be magnetic tape, but by the time the VLBA is operational, optical disks may be more attractive. Depending on operational requirements, it will also be possible to make some of the data available for immediate access by the Post-Processing System, either by way of shared disks or through a special computer-interconnect bus.

In addition to the operational functions enumerated above, the Correlator computer system must support the program development required to build the control programs.

It is clear from this description that the Control Computer and Archive Writer must be general-purpose computers. Until the design of the Correlator is complete, it is not possible to determine whether the Control Computer and Archive Writer functions can coëxist in a single computer. The Calibration Processor function, which will use AIPS software, is best carried out in an Array Processor attached to one of the general-purpose computers. The Fringe Processor function is a cpu- and I/O-intensive task which is better suited to a special-purpose computer. It is possible that some other tasks, such as the geometric model computation, could also be assigned to special-purpose processors or an Array Processor.

REQUIREMENTS

We wish to minimize the proliferation of computer types, operating systems and languages in the VLBA project. In particular, we wish to observe the following constraints:

- The interface presented by the Correlator Control Computer to the Correlator operators, who may also be responsible for controlling the antennas, should be similar or identical in syntax to that presented by the Array Control Computer. This is most easily achieved if both computers run the same operating system.
- The Correlator Control Computer has to be able to read and update the Monitor and Control Database. This is most easily achieved with a disk shared by the Correlator Control Computer and the Array Control Computer, which may constrain both computers to use the same operating system.
- The Archive Writer has to write the data archive in a form acceptable to the Post-Processing (AIPS) system. If tape is used for the output, this is not difficult, but if a shared disk or computer-interconnect bus is used, this may constrain the choice of operating system for the Archive Writer.
- The Correlator Control Computer must be furnished with a real-time operating system, *i.e.*, one capable of servicing the attached hardware (responding to interrupts)

with a short and predictable delay. The manufacturer (or a third party) must provide hardware and software suitable for connecting the correlator hardware to the control computer.
- The Correlator Control Computer must provide excellent facilities for program development, including the debugging of real-time processes.
- So far as possible, familiar languages and operating systems should be used to facilitate program development and maintenance.

OPERATING SYSTEM AND CPU

The Correlator Control Computer and Archive Writer will both have interfaces with a variety of hardware, including disks, tape-drives and operators' terminals. This implies that they should be general-purpose computers in which such interfaces are supported by the manufacturer. The requirement that the operating system be familiar to those who will be working on the project and to other astronomers leads us to consider two operating systems: (a) UNIX, and (b) VMS. UNIX is available on many different computers in a variety of dialects, but we are not aware of any implementations which provide the support required for our real-time application. VMS is only available on DEC VAX computers, but our experience with VAX/VMS for both general-purpose computing and the real time control of the Block-II VLBI Correlator gives us confidence that VMS is capable of supporting the VLBA Correlator requirements. This choice of operating system thus constrains our choice of computer to members of the DEC VAX-11 series. Until we have completed the design of the VLBA Correlator, however, we cannot choose a particular member. Our experience with the Block-II Correlator suggests that the largest VAX computer, the VAX-11/785, should be adequate for the VLBA Correlator Control Computer, although a smaller model may suffice.

There are fewer constraints on the choice of micro-computer for use in the Fringe Processor. The Fringe Processor hardware will be designed and assembled as part of the VLBA project, and the design can be adapted to a wide variety of microprocessors. Our experience with the Block-II Correlator's Tensor processor and with the Caltech/JPL Concurrent Computations Project suggests that a "board-level" computer with the M68020 32-bit processor and fast floating point hardware is very attractive, and will be able to perform 32-bit floating point operations at a rate of more than 1 million per second. The number of boards will be chosen to accommodate the anticipated fringe-processing requirements. Commercial ROM-based operating systems will be available for such microprocessors and will allow programs to be developed in a high-level language on the host VAX and loaded into the Fringe Processor memory.

PROGRAMMING LANGUAGES AND ENVIRONMENT

For programming the Control Computer and Archive Writer we wish to use standard programming languages which are well integrated with the host operating system and supported by the manufacturer. For VAX/VMS, we may choose from APL, BASIC, BLISS, C, COBOL, CORAL, DIBOL, Fortran, LISP, Pascal, and PL/I. Some of these languages are clearly inappropriate for our application, and we narrow the choice to C, Fortran, Pascal and PL/I. We have also considered FORTH, and rejected it owing to its lack of standardization and lack of manufacturer support. The language ADA would merit consideration if a good VAX/VMS compiler were available. Our greatest experience is with Fortran, and we can say that it would be possible to program the entire control system in Fortran-77, with the addition of a few non-Fortran routines for accessing parts of the operating system (*e.g.*, for interrupt handling). Fortran is very poor for handling dynamic data structures which require manipulation of memory addresses and allocation of memory from a pool, and for such applications we should prefer to use one of the other languages. The choice between these is less clear. For no very good reason, we dismiss PL/I, and consider only Pascal and C. The VAX/VMS Pascal compiler is very good for building data structures and for accessing operating-system functions, but achieves this versatility by adding many "extensions" to the language defined by international standard. The VAX/VMS C compiler adheres very closely to the (informal) standard for C, but is less well integrated with VMS (after all, C is most at home in UNIX), and has the serious drawback that it performs all floating-point arithmetic in double precision. In summary, if we had to choose a single language for programming the Correlator system, and were not constrained by considerations of compatibility and portability, we should probably choose VAX-11 Pascal. However, to maintain as far as possible compatibility with the rest of the VLBA project, we prefer to develop a hybrid system in which modules that require close integration with the operating system will be written in either VAX-11 Pascal or VAX-11 C (the final choice between Pascal and C has not yet been made), and modules which perform intensive floating-point arithmetic will be written in standard Fortran-77. The VMS operating system allows one to link subroutines written in different languages.

The microprocessors which constitute the Fringe Processor will also be programmed in a high-level language. Cross-compilers are expected to be available for C and Pascal, and possibly for Fortran. We expect to choose either C or Pascal, depending on the choice of language for the Correlator Computer. At present, C implementations are more "standard" than Pascal implementations, so it appears likely that we shall choose C over Pascal for both the Fringe Processor and the Control Computer.