VLA COMPUTER MEMORANDUM (106)

ASSIGNMENT OF TASKS TO CPU'S

Preliminary Specification

B. G. Clark

The items listed below are tasks, which run asynchronously with minimal interaction. The descriptions given here are obviously inadequate for the more complex tasks, some of which will include thousands of lines of code in several overlays. Discriptions are given here at these extremely varied levels because at this level, the temporal and spacial relation between various peices of code are fundamentally undefined. At all lower levels, code can be described by flow charts or discussions in Backus-Naur format. At the task level, on the other hand, relationships become dependent on various external events, and should be handled by a few, very formal, procedures to avoid unforseen interrelationships.

The presumption is made that the correlator handling CPU's will be equipped only with our in-house operating system, which will only schedule the CPU and will be very primative indeed. Therefore, for this pair of CPU's I have listed in some detail the tasks which are part of the operating system. The first two listed tasks are the interupt handling program. They operate with interupts inhibited, and essentially are the operating system. The third task listed is essentially the task for which we coded programs during the computer evaluation.

We further presume that the Boss CPU is run under the control of a very full Max III operating system, with all options. Monitor CPU will also be under control of a fairly advanced operating system, and there will be good software tying the two together. I anticipate that only very minor changes would be needed to, for example, run a peripheral connected to the other CPU, rather than the one in which the task is running. Therefore, tasks can be readily moved back and forth between these two to level the load in time or core. This flexibility should be kept in mind, and the division of tasks between these two systems given below is far from fixed.

Since these CPU's will be operating with the full services of Max III, I make no attempt to list the specifically operating system tasks.

CPU's CORA and CORBIN

The correlator handling CPU's, hereafter called Cora and Corbin (or generically called Cora), will run only two time-consuming tasks, one which accepts data from the correlators and integrates, and one which performs the reductions and corrections and passes the data on. It is anticipated that, for efficiency, more than one version of these task programs will be written. One version will handle the array as a unit, all antennas pointing at the same source, all correlators in use. When the array is split into subarrays, there are at most 301 baselines (instead of 351), so that the time saved in not handling the extra baselines may now be spend in handling the bookkeeping involved with the subarrays.

1. CLOCKC. This task is given control by the clock interupt, calculates the true time of day, and sets bits in the various task control blocks (TCB's) to control the other time-synchronous tasks as needed. Program runs with interupts inhibited.

2. DISPCH is activated at the end of any task. It itself runs with interupts inhibited. It searches a compiled-in priority table of tasks to find the highest priority task which needs execution, and it then starts or restarts that task, making its last step be the turning on of the interupts.

3. INTAKE. This task receives data from the correlators, integrates it, and, for interference, etc., detection, accumulates an RMS.

4. BUFSWC. This task switches buffers, passing a fresh and empty accumulation buffer to INTAKE, and passing the full one to RED for reduction.

5. RED. This task takes the accumulated data from INTAKE, performs the correlator-by-correlator reduction, and formats the output record. The reduction consists of a) shifting to put the interesting part into a 16 bit word. b) Multiplying by the appropriate antenna voltage sensitivity. c) Applying (if necessary) a correction for a fault in sine/cosine component quadrature. d) Analyzing (in an as yet undetermined fashion) the sum of square of residuals.

6. PASS. This task passes the output data from RED to the Boss computer. It consists mainly of code to handle the DMP and Modcomp-Modcomp link.

7. TIMSET. This task maintains the time synchronization between the time-of-day clocks in Cora and in Boss.

8. INIT1. This task initializes all or a subset of the correlator buffers for start of an observation. It will halt operation of INTAKE and RED (at least for the subset of the buffers being initialized). It will clear a buffer, pass it to INTAKE, and wait for an even 10 seconds to activate INTAKE.

9. INIT2. This task completely initializes everything, including reading in new versions of the programs INTAKE, RED, and INIT1.

10. VOLTS accepts from Boss a new set of antenna voltage sensitivities.

11. DEBUGC is initiated by a request from Boss, and causes a given core word to be passed to Boss. Boss custodiet Cora, et nos custodiem Boss, atque quis nos ipsos custodiet?

MONITOR CPU

The monitor and communications handling CPU, hereafter called Monty, is presumed to have available also most of the resources of Boss. Specifically, we will imply disk files being used, ignoring the fact that the disk is physically attached to Boss, and the decision as to whether printer and CRT terminals are attached to Monty or to Boss is reserved for future resolution.

The tasks specific to Monty are listed below.

1. CLOCKM updates and maintains synchronism of the time-of-day clock in this CPU.

2. DMPX demultiplexes the data received from the serial system, and performs any indicated averaging. It would also pass to Boss various pieces of control information, such as the console switch registers.

3. LOG makes various log records and inserts them in the appropriate disk files.

4. DISPL takes selected monitor points and places them at selected points on a CRT screen as a decimal number.

5. RUN decodes a keyboard input and outputs it to a selected address on the DCS.

6. HOOKUP changes the various pointers in DISPL. It is an operator initiated task, and once the pointers are selected, it goes away again. It also inserts the pointer in RUN, being careful to disconnect the device from whatever software would otherwise be running it.

7. CHECK interprets the monitor definition dictionary, and, on finding errors, makes an internal memorandum describing them, and flags the data appropriately.

8. COMPIL compiles and enters new entries into the monitor dictionary. This is an operator initiated task, usually accepting input from a terminal keyboard.

9. TECHN takes the internal memoranda generated by CHECK and converts them into messages for CRT's and printer, using a file of old messages to avoid producing excessive amounts of output.

10. MORMON accepts additional monitor data from Boss (actually from Cora via Boss) and inserts it into the list of demultiplexed variables with the other monitor data.

11. CMD collates the DCS command information received from various sources, formats them appropriately, generates some simple command cycles itself, inserts any necessary filler or dummy commands, and outputs the works to the DCS.

THE BOSS CPU

The list of tasks provided here is by no means complete. It is fully expected that there will be other tasks, yet to be specified, which perform engineering test exercises, computer assisted diagnosis, or additional convenience features. The list given here is intended to describe the set of programs provided for initial array operation, and gives only those things fundamental to the control of the array.

There are, in this listing, four separate tasks associated with the array geometry, as discussed in VLA Computer Memorandum #103. These are associated with the four time scales needed.

1. CLOCKB. This is a general clock supervision task, entrusted with counting the 19.2 Hz interupts to make a time of day clock, comparing it with the hardware time-of-day clock, and maintaining the synchronization of the time-of-day clocks in the subordinate CPU's.

2. GEOM10 performs the every-ten-second calculations of Memorandum #103.

3. GEOMDL runs the delay lines, outputting new delay values every 50 ms.

4. GEOMA runs the antennas themselves, issuing a new azimuth and elevation for each antenna every 100 ms.

5. GEOMLR runs the lobe rotators, passing new values of rate and phase to the antennas every second or so.

6. DATA accepts correlator data from Cora and distributes it to the various tasks and files which require it. The correlator self monitor words will go to Monty, as will the sum-square information and, for calibrator observations, the amplitudes themselves. Various record header information will be associated with the data, and the results written on the fixed head disk, and simultaneously, on the magnetic tape. Information will also be directed to the system log.

7. REVIEW is an operator initiated task for making inquiries to the system log. It will initially have three modes: a) List mode, in which the named monitor points are displayed as numbers on the screen, b) Plot mode, in which the operator states the timescale of interest and a list of monitor points, which will then be plotted as a function of time in a printer plot on the CRT terminal, and c) Limit mode, which will search the system log backwards from the present, and tell the operator the last time at which the monitor point of interest exceeded the limit given.

8. INIT is an extremely complex task which is run at the start of each observation. It will, among other things: a) Fetch a new observation request card from the observing queue for the subarray of interest, b) Perform the ephermeris and baseline calculations for this observation, c) Update all of the system control blocks, d) Alter switch settings and parameters at the antennas (set front end selection

switches, rotate subreflector, tune upconverter, etc.) and e) Inhibit data recording until at least some of the antennas involved are pointed at the source.

9. IMP reads observation request cards, checks for errors, and enques them in the appropriate observation queue. The card images may come from the card reader, a CRT terminal, console digit switches, or a file on disk left by the Asynchronous computer. The same task would have programs for the display of the observing queues, and some facility for editing them. It is an operator initiated task.

10. CONNEC is an operator initiated task which is used to make major changes in the array connectivity, changing the affiliation of antennas among subarrays, or to inform the computer of antenna moves or address changes.

11. ALTER is run every ten seconds for those observation modes which require major adjustments during the course of the observation (for instance pointing or making room for eventual implementation of frequency switching--the equivalent of interferometer D mode observations).

12. TABLES is an operator initiated command which updates various disk tables, such as the nominal antenna locations, the limit checking source commands, the antenna pointing constants, etc.

13. COLD is the cold start routine - it restores everything from disk. (Part of this service comes free with MAX III.)