VLBA Correlator Memo No. 101

Correlator Software Status, January 1991

Don Wells (for the correlator software team)
National Radio Astronomy Observatory, Charlottesville, Virginia

January 31, 1991

Abstract

A status report for the real-time software project for the VLBA correlator for the period mid-November 1990 to mid-January 1991 is given. During this period the team produced the first formal release of the integrated task structure. All of the software components in use are now under the NSE code management system, including code imported from the geodetic community and from the M&C [Monitor and Control] group. The "screens" facility is beginning to be used for the operator interface. An improvement in the delay model generator architecture has been decided, and is discussed here. The PBDs are now using version 5 of the RECON firmware from Haystack.

Contents

1 Tasks 2

2 First Release 2
- 2.1 Test Procedures and Results 4
- 2.2 Known Deficiencies and Not-Yet-Dones 5

3 Component Structure 6
- 3.1 Automatic procedures to fetch M&C Code 6
- 3.2 HCB Component 8

4 Operator Interface 8

5 Model-Crossbar-LTA Changes 9

6 PBD Firmware Upgrade Installed 9

1 Tasks

During a team meeting held 23-March-90 we decided to concentrate on developing an integrated task structure which will run jobs, ready to command the hardware as it becomes available, but also able to run independent of the hardware for software debugging purposes. During the current two-month period the team substantially achieved this goal of more than 9 months. This effort involved all of the tasks in Figure 1 below except for the archive and clock tasks on the right-hand side.[1] The present version of the figure differs slightly from the Figure 1 shown in the September 1990 status memo;[2] in particular, a number of the tasks which were then marked as "In progress" are now marked as "Done". "Done" does not mean that these tasks will not be changed in the future; it means that they are substantially complete in an initial operational form.

2 First Release

The team decided to produce a "release" of the software with the nominal date 15-January-91. The intent was

- to put a consistent set of all software components under NSE
- to load and operate these components together under vxWorks
- to practice team procedures for software release
- to provide a coherent basis for further development
- to produce a precise reference for a bug and deficiency list

During the weeks before the target date the team decided which work-in-progress items would be in the release, merged them together, and devised loading procedures to install them. The `new` environment was locked and several days of testing by team members followed (see below), and various corrections and minor enhancements were applied after testing. On 31-January-91 the code was "reconciled" (NSE technical term) to our "`release`" environment, and this memo documents the release.

[1] Figure 1 is an updated version—an "as built" schematic—of Fig. 4 on p. 36 in VLBA Correlator Memo No. 95 (Sept. 29, 1989).

[2] VLBA Correlator Memo No. 99

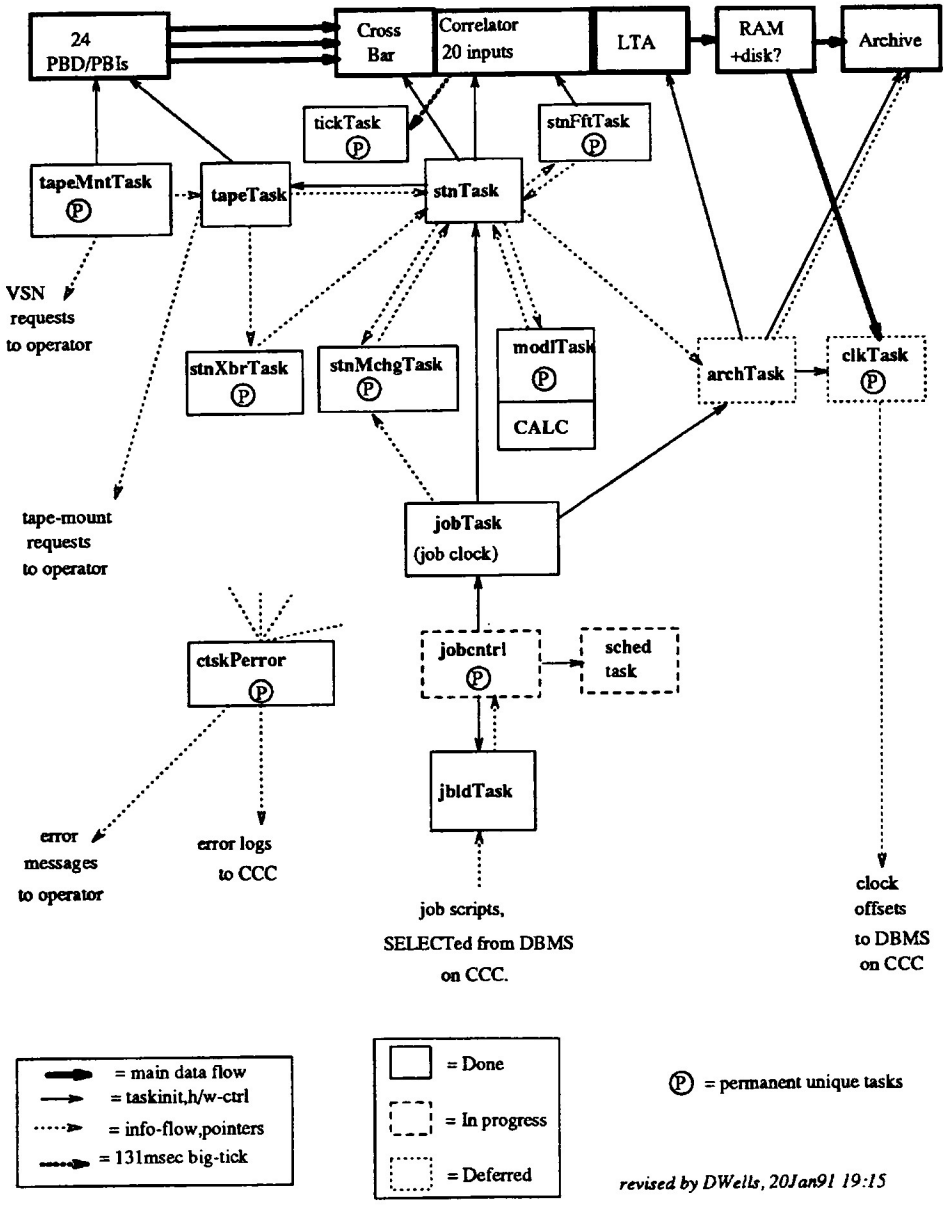

Figure 1: Real-Time Tasks (see Sect. 5 regarding `stnMchgTask`)

2.1 Test Procedures and Results

The test suite which was applied to this release was the set of component tests which were developed separately by the team members in their private NSE environments before the integration for the release:

tapeMntTask, tapeTask: The correlator software ran with all tasks active, including screens package and other M&C code necessary for the project. Two playback drives [PBDs] were available, although one had only the limited playback signal paths provided in acquisition recorders. The test script file contains information such as Job#, VSN#, Station#, head stack positions, etc. Two test sequences were performed:

Two PBDs with "mini-decoder": All tasks performed as expected, both PBDs followed the tapeTask commands to the letter, and the headers which were read from the tapes with the mini-decoders were found to contain the expected information.

One PBD with 18-track correlator playback interface: The current implementation supports Mark III mode. It worked properly, except that the time was not correct, but the AUX field appeared to be correct. The time was expected to be wrong.

stnXbrTask, stnMchgTask, jobTask: Two mock correlating jobs were executed, one with 7 stations and one with 2. The second job was started some time after the first job had been initiated. The state changes of the two jobTasks were monitored, and special attention was paid to the cycle of state changes in the crossbar and model change tasks to assure that the logic coordinated the 9 stnTasks while accounting for the differences between the two independent jobs. Crossbar operations were observed to take about 16 msec/station.

modlTask, stnTask: The released version was tested in the real-time system by running two mock correlating jobs simultaneously. The jobs ran successfully without interference. Delay models were calculated and downloaded into buffers for the HCB software at the proper times. Observing source changes and station electronics changes were tracked successfully in both test jobs.

Software interrupt frequency tests: The correlator hardware will produce a hardware interrupt at 7.629 Hz;[3] this will be the basic "ticking"

[3] this is $10^6/2^{17}$ Hz, 131.072 msec/tick.

frequency of the correlator system. When running with a software oscillator we are able to operate at other frequecies, and even to single-step the interrupts. The earliest tests of our "ticking" software 9-10 months ago were done at very low frequencies, e.g. 0.07 Hz, 100 times slower than normal. Gradually frequencies were increased; many tests late in 1990 were run at 2.0 Hz (they crashed or hung at higher frequencies, for various reasons). Several recent bug fixes have finally allowed this release to run reliably at full speed, 7.629 Hz. In fact, it also runs at higher frequencies! For example, the `modlTask` and `stnTask` tests qualified at 15 Hz. In another run which was not checked as thoroughly the same code was reported to run at 30 Hz, but began getting timeout alarms at 60 Hz (there are plausible arm-waving arguments that would explain this observation). Other tests of `tapeTask` showed that it would not operate at 30 Hz. We will investigate these matters further in the future, for two reasons: (1) high frequency operation stresses the logic and fixing the bugs disclosed in the tests tends to make the code more robust for lower frequency operation, and (2) we expect that it will be convenient to be able to run several types of software tests much faster than real time.

Memory utilization: The allocated memory before initialization was performed or permanent tasks were spawned was 910 Kbytes. After initialization and spawning the allocation was 1210 Kbytes. With two job scripts loaded and twenty more tasks operating (2 `jobTasks`, 9 `stnTasks` and 9 `tapeTasks`) the allocation reached 1560 Kbytes, and with some delay model structures attached to the jobs it reached 1590 Kbytes, at which point there were still 6480 Kbytes available on the free list.

2.2 Known Deficiencies and Not-Yet-Dones

The components of the system which have been developed for this release have the following deficiencies:

- the schedule and job-control tasks are still "In progress".
- the `stnTask` has not yet talked to any correlator hardware via the HCB (but note that the engineering test code has utilized the HCB device driver for almost-daily board testing for many months).

- the `tickTask` has not yet operated with the hardware interrupt source; all testing has been done with a software interrupt source.
- the transition of `jobTask` from its `job_STOPPED` state to its `job_RUNNING` state is not yet properly interlocked with the actions of the `tapeTask`.
- changes to FFT engine configurations by `stnTasks` do not yet cause the `stnXBrTask` to go through its crossbar-change (system-reset) cycle (but tape assignments do initiate crossbar cycles).
- job-termination logic is not yet fully implemented.
- only a few screens are implemented for operator functions.
- an integrated test suite has not yet been defined.
- embedded documentation is incomplete

3 Component Structure

Table 1 shows the NSE component structure as it stood in the "`new`" environment on 15-January-91; this table was adapted from output produced by "`nsecomp list -r :`". Note that Table 1 is not a complete listing of the actual component structure; redundant and minor components have been omitted for clarity.

3.1 Automatic procedures to fetch M&C Code

During the period a major effort was made to devise semi-automatic procedures for acquiring copies of the latest versions of M&C code from `vlbacc` and maintaining them on ccc under NSE. A new toplevel component `mc` was created, with subordinate components `New` and `Release`, each of which in turn has subcomponents. In particular, `New` contains components `code147New`, `codeSunNew`, `hNew` and `mv147`, which receive the various categories of code from Socorro, corresponding to the naming conventions used in the M&C code management system.

During the period Makefiles were devised which can compile shared code modules from the M&C code components and deliver them to be loaded under vxWorks with the correlator software. Makefiles for selected correlator components were augmented to reference the `include` files from the M&C code. The effect of this implementation is that a controlled subset

Component	*Nature*	*Status*
`:vxWorks:archive`		Deferred
`:vxWorks:calc`	GSFC library	Imported
`:vxWorks:clock`		Deferred
`:vxWorks:hcbus:hcbnew`	Device driver	Done
`:vxWorks:hcbus:hcbtestfix`	Test-fixture-code	Done
`:vxWorks:boot`	`bootCorrelator`	Done
`:vxWorks:jobcontrol`		In progress
`:vxWorks:jobloader`	`jbldTask`	Prototype
`:vxWorks:job`	`jobTask`, etc.	Done
`:vxWorks:model`	`modlTask`	Done
`:vxWorks:sched`		In progress
`:vxWorks:station`	`stnTask`, etc.	Done
`:vxWorks:tape`	`tapeTask`, etc.	Done
`:vxWorks:tick`	`tickTask`	Done
`:vxWorks:opint:menus`	Menu-tree	Prototype
`:vxWorks:opint:tape_scr`	PBD screens	Prototype
`:vxWorks:opint:job_scr`	Job screens	In progress
`:vxWorks:util:ctsk`	Tasking library	Done
`:vxWorks:util:efc`	Event flag library	Done
`:vxWorks:util:service`	Misc. library	Done
`:vxWorks:util:tables`	Table library	Done
`:mc:New:code147New`	MCB library, etc.	Imported
`:mc:New:codeSunNew`	`rscreen`, etc.	Imported
`:mc:New:hNew`	M&C includes	Imported
`:sun:dbms`	Schema, batch jobs	Suspended
`:sun:hcb`	Compute tables	Done
`:sun:util:scripts`	M&C update scripts, etc.	Done
`:init`	Initialize Delivery	Done
`:doc:memo95`	History	Done
`:doc:misc_memos`	This memo	In progress

Table 1: Principal NSE Components (as of 15-Jan-91)

of M&C software is now compiled with correlator procedures and options and is loaded with the correlator software, and that subset has known—and controlled—interactions with other components of the correlator software.

The procedures for fetching changes from the M&C code management areas are capable of running on a regular scheduled basis, and can also be executed manually as needed in order to pick up fixes and enhancements that are needed sooner than the scheduled update. Both the "New" and "Release" versions of M&C software are being fetched, but the current policy of the correlator project is to use the New version (the most recent code).

3.2 HCB Component

During this period the code in the HCB [Hardware Control Bus] component was made to be the same as the engineering test versions of the code. Furthermore, by the end of the period engineering tests were being performed with the executable versions of the driver code delivered by the Makefiles under NSE.

4 Operator Interface

From the inception of the correlator software project it has been our policy that we would use a version of the "screens" package from the M&C project in order to assure that VLBA operators would experience the same look and feel for the correlator as for the rest of the VLBA. This policy has been implemented in several ways.

Early in the project one of our team (JEH) spent many months porting the screens infrastructure (the "s-routines") to vxWorks and making other improvements; later the M&C group took over the maintenance of his code. Now, with the activation of automatic fetching of M&C code (Sect. 3.1), including the screens code, the correlator project is making use of that early development effort.

More recently another one of our team (RDG) began using the screens code to construct RT displays for his experiments with the tape subsystem. For our 15-Jan-91 release this display code has been brought under the NSE code management as the "`opint`" (operator interface) component, which currently contains sub-components `tape_scr`, `job_scr`, and `misc_scr`. The tape portions of these menus and displays are already useful, and we anticipate that other menus and displays will be operational for our next release.

A third member of our team (DCW) made some experimental enhancements to the `rscreen` display utility for Suns during recent months. These enhancements have recently been incorporated into the M&C version of `rscreen`, and we now use that version of the program, in the form that is delivered by the automatic-fetch process described in Sect. 3.1.

5 Model-Crossbar-LTA Changes

The correlator hardware and software teams held a special meeting on 10 January to review issues surrounding time, model generation, and crossbar operations. Ray Escoffier proposed several changes to the control firmware, and after discussion these changes have been accepted. The main effect of the changes will be that the active delay model can be changed for any station on any 131 millisecond tick. The master time variable will be reset to zero only on a crossbar operation (called "system-reset" by the hardware team), and new models being loaded between crossbar changes must have their polynomials shifted relative to the time of the last crossbar change. This is a *major* change of architecture, with far-reaching implications—mostly good ones for the software. In particular, model changing will now be done on a per-station and per-job basis, asynchronously, as needed. Numerous redundant model changing operations will be eliminated. Compute loads and dead times will be reduced. An unanticipated bonus is that source-switching will now be possible for up to four different positions, not just two.

An important effect of this change is that the `stnMchgTask` will no longer be needed; we will soon eliminate it.

In the course of this discussion the hardware and software teams exchanged considerable information about their respective implementations, and several misunderstandings were exposed, especially a serious one about the operation of the Long Term Accumulator during crossbar changes. The team discussed a possible minor change to facilitate reading out incomplete LTA integrations at a crossbar change.

6 PBD Firmware Upgrade Installed

At the end of the period the team received Version 5 of `RECON`, the recorder controller program which is the resident firmware of the PBD, from Roger

Cappallo of Haystack Observatory.[4] The README for version 5 lists 9 bug fixes and 14 new features, and many of these items are important to the correlator. We consider new feature 9 ("Separate functionality of digital write and monitor boards, so that either may be missing without affecting the other") to be the most important because PBDs are to be delivered without the write board.

[4] A-Technology-Sign-of-the-Times: Roger said he would send a floppy disk of the code with a paper copy of the documentation to us by mail, but then we rememembered that Haystack now has Internet service. After a phone call to find out the guest account, directory and filenames, we used FTP to fetch the S-record file for the firmware and the TEX files for the documentation. Within two hours we had the typeset documentation in hand and new PROMs were installed in our PBDs and were being tested. It is indeed a brave new world...