CALIFORNIA INSTITUTE OF TECHNOLOGY
VLBA Correlator Memorandum

To: VLBA Correlator Group
From: Steve Kator, Tim Pearson, Martin Ewing Date: 31 July 1985
Subject: Computing Load for SMP

In our current design, there are four Station Model Processors (SMPs). Each SMP computes geometric phase and delay models for a 4-channel "quadrant" of the Station Electronics. (Refer to §4.2 and §4.5 in the Architectural Design Report, Correlator Memo VC041.) Input to the SMP from the control computer are the interferometer parameters station position, source position, time, etc., and output are delay, phase, and phase rate suitable for driving the hardware delay and phase generators. A secondary task is to control the phase calibration detector and to accept its output for transmission to the control computer.

These tasks may be subdivided as follow:

1. Communications with control computer
2. Control and diagnostics of SE
3. Model Calculation
4. Model Interpolation
5. Communications with hardware delay/phase generators
6. Phase calibration control and output

We anticipate that tasks 3-5 will account for the bulk of the work for the SMPs. This Memo provides estimates for these tasks.

Model Calculation

The model calculation in the SMPs must include at least the following terms:

(a) Change of the baseline orientation w.r.t. the source;
(b) Retarded baseline correction;
(c) A crude atmospheric model (e.g., slab).

For the moment we shall ignore (b) and (c) as they are slowly varying compared with (a). Other terms (e.g. earth tides and precession) are updated infrequently and should add a negligible load (except when model switching).

The calculation for (a) must be done every 10 s. We need to compute the phase at the center of the 10-s interval and its first four derivatives. The derivatives could be obtained by differencing successive computed phase values, but this doesn't work very well at scan boundaries where there is a discontinuity in phase. It is easy to compute the derivatives directly, so

we shall assume that we shall do so.

At each update, the phase and derivatives must be calculated for 384 models (24 stations x 16 channels). In the general case these are all independent, although in most cases there is scope for optimization as different channels will share the same geometry. A single phase calculation involves the following:

(a) Advance Hour Angle by 10 s (UT). This is a single floating-point addition, the required DELTA in radians having been precomputed:

```
TIHA = TIHA + DELTA
```

(b) Compute u, v, and w (relative to center of earth). U and V are not used but are required for output to the FITS archive. Here index IB is the station/channel number, and BXY*SINDEC, BXY*COSDEC, BZ, COSDEC and SINDEC are precomputed constants for the scan. This involves a computation of a SIN and a COS (of the same argument - faster than independent COS and SIN), 5 floating multiplies and 2 floating additions:

```
U = BXY(IB)*COS(TIHA)
V = BZ(IB) *COSDEC + (BXY(IB)*SINDEC)*SIN(TIHA)
W = BZ(IB) *SINDEC - (BXY(IB)*COSDEC)*SIN(TIHA)
```

W is the required phase, in some units (depending how the constants are defined). We only need the fraction of a cycle, however:

```
PHASE = MOD(W,1.0)
```

All the derivatives of phase are clearly multiples of SIN(TIHA) or COS(TIHA), which have already been computed. Thus each derivative is calculated with a single extra multiplication. To compute u, v, w, phase, and four derivatives (assuming all constants have been precomputed), requires a total of

4 floating additions
9 floating multiplies
1 floating divide (in the MOD if you do it that way)
1 SIN+COS calculation

We are not sure how many "floating-point operations" are required for the sin-cos calculation. Assuming it is 10, the total requirement is 24 operations per model, or 9216 for 384 models; a rate of ~ 1 kFLOPS.

All this should be done in double precision, so the FLOPS are double precision ones.

On the VAX-11/780, a test program to simulate this computation for 100 10-s intervals (38,400 sin-cos pairs) took 8.9 s of CPU time (optimized). Assuming that a 68020 with 68881 floating point coprocessor is comparable to a VAX-11/780, an SMP using these chips would also experience ~1% loading. Actually, we propose to employ 4 SMPs, each of which would experience only about 1/4% loading from model calculations.

[These benchmarks and the kFLOPS estimate above indicate that the VAX effectively runs at 110 kFLOPS in double precision. This seems low, and indicates the limited accuracy of our estimates. A benchmark on the intended SMP hardware should be undertaken as soon as possible.]

Interpolation

For the purposes of estimating the computational load due to the fringe phase and delay interplation in the Station Electronics, the following general algorithm is assumed.

The current phase value is generated by adding a slope to the previous value. The rate at which the phase is updated is determined by the peak fringe frequency and the phase error which is allowed. The slope value is in turn updated at some slower rate, also determined by the allowable error. Thus we have a piece-wise linear approximation to the ideal phase function, built up from p-w.l. approximations to the derivatives of the function. Higher-order derivatives are approximated at slower and slower update rates, and the highest-order derivative employed is updated by some more sophisticated algorithm which can accurately account for the subtleties of the model. Clearly, the number of derivatives that are approximated is arbitrary, and more derivatives corresponds to a slower update rate required for the highest derivative. Since the update rate for the highest derivative can be made arbitrarily slow, the algorithm that provides those updates can be arbitrarily complex.

By this method we are in effect approximating the function by a Taylor series with remainder:

$$\phi(t) = \phi(0) + (1/1!)\phi'(t)*t + (1/2!)\phi''(t)*t^2 + \ldots + R_n(t)$$

The worst case error due the approximation of each derivative can be found by assuming a peak fringe frequency and an update rate for each of the derivatives, then evaluating each term of the series.

In our analysis we assumed a peak fringe rate of 250 kHz, which corresponds to a 10000 km baseline at an observing frequency of 100 GHz. The first derivative is

$$\phi' = 250000 * \sin(2\pi t/86400) \text{ Hz}.$$

Taking the worst case (where the sine term equals one),

$$\emptyset' = 250000 \text{ Hz}.$$

Likewise, the worst case for $\emptyset''$ is

$$\emptyset'' = 250000 * 2\pi / 86400 = 18 \text{ Hz/s},$$

for $\emptyset'''$ is

$$\emptyset''' = 0.00132 \text{ Hz/s}^2,$$

and for $\emptyset''''$ is

$$\emptyset'''' = 9.6*10^{-8} \text{ Hz/s}^3.$$

The peak error associated with updating $\emptyset'$ at 500 ns intervals is 45^0. Assuming that the phase is expected to be quantized to eight levels, this value equals the precision and hence should be sufficiently good. (This should be looked at some more.) The peak error due to $\emptyset''$ being updated at 2 ms is

$$\Delta\emptyset = (1/2) * 18 * (0.002^2) = 0.013^0.$$

For $\emptyset'''$, if the values are updated every 0.125 s then the peak error is

$$\Delta\emptyset = (1/6) * 0.00132 * (0.125^3) = 0.00015^0,$$

and for $\emptyset''''$ with updates at 10 s, it is

$$\Delta\emptyset = (1/24) * 9.6*10^{-8} * 10^4 = .014^0.$$

These update rates were chosen arbitrarily to produce a total worst case peak error of less than 0.1^0. With these rates we can evaluate the computational load in calculating the updates. It is assumed that the 500 ns $\emptyset'$ update is done in hardware, and the 10 s $\emptyset''''$ updates are supplied by the CCC. Also we assumed that each update corresponds to one floating point operation.

For each independent model, we require 500 + 8 + 0.1 = 508.1 FLOPS. Thus if we generate independent models for every channel of every station, the total computational load for 20 stations is 508.1 * 16 * 20 = 162.6 kFLOPS.

If a similar algorithm is assumed for computing delay, the load might double, since in some cases (e.g., wide field of view) separate delay models could be required on each channel. Thus, a peak load of no more than about 320 kFLOPS is anticipated, or ~80 kFLOPS for each SMP.

Note that, although we assume floating point operations in the interpolation algorithm, we could use scaled integer calculations. Floating point is preferred, however, for ease of programming and maintenance. Of course, the model calculations could be done in integer arithmetic as well, but we are even less comfortable with that suggestion.

Hardware Communications

There will be 384 hardware delay/phase generators updated at a 500 Hz rate. We assume each update will require 6 16-bit word transfers (delay, fringe phase, and fringe rate, each of 32 bits). The total I/O transfer rate

is then about 1.2 Mword/s, divided among 4 SMPs: 300 kword/s each. Such a rate is probably feasible with the 68450 DMA controller and a 68000 (16-bit) processor. However, the 68020 32-bit processor provides the option to halve the word rate using 32-bit transfers. (If pin-out limitations constrain the delay/phase generators to 16-bit transfers, a simple hardware 32-to-16 bit multiplexor may be required.)

Conclusions

If we plan to do the programming the "easy" way, with maximum use of floating point, we expect a floating point load of somewhat less than 100 kFLOPS per SMP, dominated by the model interpolation function. Coupled with this is a rather heavy I/O burden in communicating results to the hardware. In addition, there will be the other tasks outlined at the beginning. The floating arithmetic rate demands a hardware floating point capability, which (in the 68000 family) is only available in the 68020. This 32-bit processor will also be much more comfortable with the I/O and other tasks than the alternative 16-bit processors. The cost of a system built on the 68020 should only be a few thousand dollars greater per SMP than a 68000 system. In summary, we feel that the 68020 with 68881 floating point chip is an appropriate selection for the SMP.