National Radio Astronomy Observatory
Charlottesville, Virginia

3 November, 1986

To: VLBA Correlator Memo Series

From: John Benson

Subject: Computer Simulations of the VLBA FX Architecture

I. Introduction

The FX simulator is a set of computer programs that simulate the complete VLBA FX correlator architecture and data path beginning with the data quantizers and following through to the cross-correlation accumulators. The simulator routines correlate model data from one I.F. channel in one interferometer baseline. Up to 2048 spectral line channels are availiable simultaneously across the upper and lower sidebands.

The FX simulator is being used to quantitatively test various correlator architectural configurations, and to search for unexpected problems that may have eluded us. Currently, the simulator is being used to help determine how many bits of precision are required within the correlator FFT, and to measure spurious correlations due to station-based digital lobe rotators and two bit sampled data. The tests are mostly centered around spectral dynamic range measurements in the presence of very strong, narrow spectral lines.

The simulator programs exist on the NRAO Convex C-1 computer in Charlottesville, and on a Cray X-MP/48. These programs correlate data that was generated in the computer. We do not correlate actual radio astronomy data. A typical run would correlate 16 Msamples during an overnight job on the Convex C-1. The Cray based programs generally provide a speed increase of about a factor of 8 to 10. The Cray simulator generates and correlates 16 Msamples in about one hour of cpu time. The Cray execution times are about 1.20 times the cpu times.

By means of menu driven inputs, each simulation run may be configured with a specific VLBA observing model : correlation coefficients in line and continuum sources, spectral line frequencies and line widths, and station-based fringe rates. Also, options may be set that separately control the operation of each stage in the correlator architecture.

II. Description

The Convex simulator software consists of two separate programs that perform three logically separate functions. The first routine (and function) is called the raw signal generator. It manufactures the noise-like signals that will be correlated. These signals are held in permanent disk files and are reuseable. The data are created with random number generators and digital filters. The data are single sideband noise with normal distributions, and may optionally include spectral lines. In the most recent versions of the software, we have eliminated the module that forces the data into a normal distribution. The additions in the digital filter itself cause the data distribution to approach a normal distribution (via Central Limit Theorem). A block diagram of the raw signal generator is shown in Figure 1.

In the Cray simulator software, the raw signal generator is included within the correlator simulator program itself. Each time the correlator simulator program is run, the raw data sets are regenerated and loaded into a very large 'solid state disk'. The solid state disk has extremely fast access times. The program speed-up that is due to reduced disk I/O compensates for having to rerun the raw signal generator.

Generally four data sets are used in a simulation run: one continuum sky signal, one spectral line sky signal, and two system noise signals. The continuum sky signal and the two system noise signals are statistically and spectrally identical. The spectral line data set(s), however, may have any line frequency and line width specied by the generator control parameters. Absorption as well as emission spectra may be generated.

One of the principle reasons for using the simulator software is to search for subtle errors in the FX correlator output. So, basically we wish to correlate enough samples to get down to noise levels of 1 percent or less of the model correlation coefficient. This requires correlating tens of millions of data samples. Raw data sets of this size would take many hours to create, and they would be costly and difficult to store permanently on disk. Therefore, we use a scheme of multiply rewinding and re-reading relatively short data files. Typically the raw data files are 528000 samples in length. After the correlator simulator has read through them the first time, the files are rewound and they are read through again. The system noise files (stations A and B) are read with 'data time' offsets from each other, and from the read pointers of the sky signal files. On each rewind pass, the offsets are increased by a multiple of n data samples, where n is the number of lags in the digital filter in the raw signal generator. In this way, uncorrelated and independent system noise is added to the sky signals. As long as the model cross-correlation coefficients are kept low, the system noise drives the quantizers and independent signals are generated. If the correlation coefficients are high, then there is less advantage in rewinding and correlating more samples. The samples on each rewind are less independent.

The workhorse part of the FX simulator system is the correlator simulator program. It may be most simply described in terms of two separate logical parts. The first section of the program reads the raw sky signal files and the system noise files, and manufactures two data streams according to an input model. This part of the program is called the model data generator (see Figure 2a). The user specifies the continuum correlation coefficient, the spectral line correlation coefficient, and the fringe rates at each station. The two model data streams are 32 bit floating point complex words.

The second part of the program simulates the FX correlator architecture. The block diagram is shown in Figure 2b. A brief description of each block follows.

Quantizer. The quantizer samples the 32 bit floating point data into either one bit, two bit or 16 bit fixed point words. The two bit thresholds are set for an equal probability of a sample falling in any of the four states (default threshold = 0.674 sigma). The noise level in the model data is measured when the program starts up. 10000 data samples are generated by the model data generator, and the signal variances are measured directly. There is a provision for allowing the quantizer thresholds to vary with time.

Digital Lobe Rotator. The digital lobe rotator may operated with 2 to 12 bits in the phase, and 2 to 16 bits in the output words. When the program starts up, a table of output words is calculated for all combinations of input data words and phase states. The calculations are simple multiplications with the sines/cosines of the quantizied phase, and integer rounding. Future versions of the simulator will be able to access externally calculated tables. We plan to experiment with optimized lobe rotator tables.

FFT. The FFT is a pipeline subroutine that is based on a single Radix-2 butterfly. The butterfly is shown in Figure 3. The input data to the first FFT stage is converted from fixed point complex words into floating point complex words which share a single exponent. The number of bits in the floating point mantissa and in the FFT coefficients may be specified in the program inputs. There is a provision for a window function on the FFT input and a scaling spectrum on the FFT output, although they have not yet been implimented. We may include a Radix-4 butterfly in the future.

Baseline Multiplier. The complex baseline multiplier will operate with any number of floating point mantissa bits specified, or in regular 32 bit floating point words.

Accumulator. The accumulator operates in 32 bit floating point only. The baseline spectra are normalized by the product of the rms's measured in each quantizer output. All of the cross-correlation spectra from one simulation run are accumulated in one output spectrum.

Upon completion, the simulator writes the accumulated cross-correlation spectrum into a text file, and updates an existing simulator log file.

III. Simulator Control Parameters.

The software switches that the user may specify when starting up the program are listed below.

› input data models

continuum correlation coefficient,
file name of spectral line data set,
spectral line data multiplier.

› output file name

› data quantization

n bit sampling, 1, 2, or 16 bits,
quant. threshold in sigmas.

› lobe rotators

n bits lobe rotator phase, 2 to 12 bits,
n bits in the output quantization, 2 to 16 bits,
n samples/turn lobe phase on #1, 8 to 64 M,
n samples/turn lobe phase on #2, 8 to 64 M.

› FFT size and total integration. The FFT may be either the FX-style FFT which duplicates that of the proposed VLBA correlator, or a substitute 32 bit floating point FFT written for the Convex C-1. The FX FFT uses fewer bits in the data words and in the sine and cosine multipliers. The FFT data words are complex and each complex pair shares a common exponent. The number of bits in the mantissas and sine/cosine twiddle factors may be specified.

n point FFT, powers of 2 through 2048,
n FFT iterations per rewind, use 250 for 2048 pt FFT,
n rewinds, use 62 to get 32 Msample run,
FFT type, Convex R*4 (QXFOUR) or FX model FFT (VectFFT),
n bits in sines, cosines for FX FFT,
n bits in floating point mantissas for FX FFT.

› baseline multipliers, accumulators

R*4 floating point or FX style floating point,
n bits in mantissa,
n bits in exponent.

IV. Execution timimg.

The FX model FFT dominates the run time of the simulator program on the Convex C-1. A 16 Msample run (2048 pt FFT) will run from 6 hours to 8 hours on the Convex (during the night). A run with identical configuration except using the Convex FFT (QXFOUR) will run to completion in about 1.5 hours. Overnight runs of 16 Msamples are pretty much the practical limit for the Convex based simulator.

The simulator on the Cray is organized differently. When the program begins, it creates the raw data signals and stores them in a large 'solid state disk'. Building these data files costs about 500 cpu seconds for four 528k word files. Beyond that, the simulator runs at about 5k samples per cpu second (2048 pt FFT). A 16 Msample run on the Cray requires about 3500 cpu seconds.

V. Algorithm Verification.

The simulator algorithms were checked by isolating key subroutines and running them with test signals whose results were to some degree predictable. The most difficult modules to write and debug were the FFT subroutine itself, and the fixed point to floating point conversion module. The operation of the FFT has been verified by comparing it's results with those of a real*4 FFT for various single frequency signals. The FX FFT produces spectra that become nearly identical to those of the real*4 FFT as the number of bits allowed in the FX FFT approach 16.

VI. Simulator Programs.

The simulator program is called FXSIMULATE, and resides in the Charlottesville Convex in the subdirectory :

/vlb/jbenson/vlba/modelcorr/public.

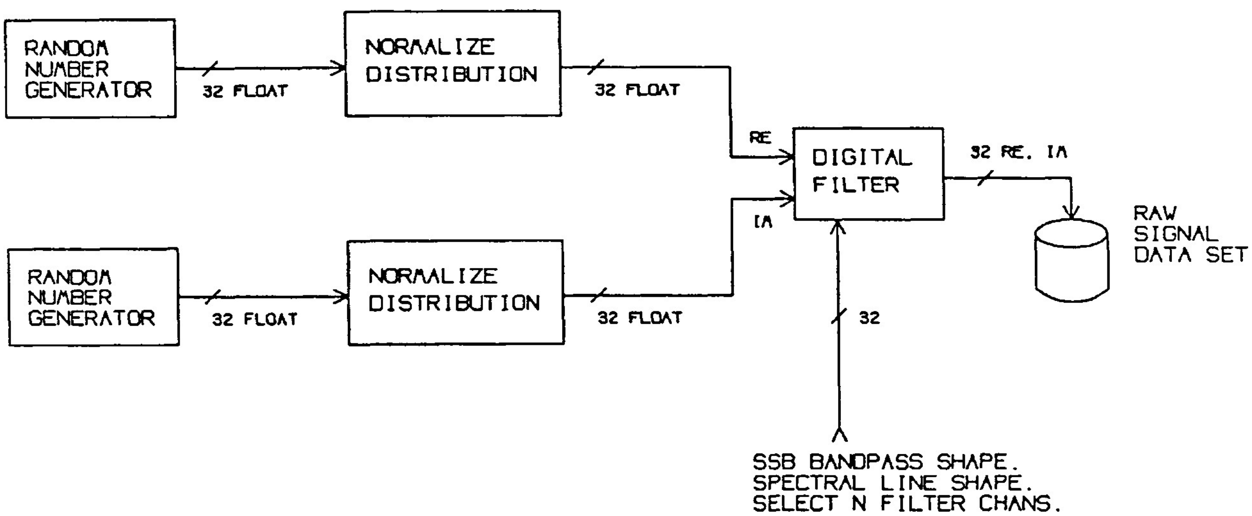
The FFT subroutine is called NEWVECTFFT, and the fixed-to-floating point convertor is called VFL554. All subroutines used by the simulator are in fastsubs.o (and .f) and fftlib.a. In addition, a program called FXLINE in /public tests the FFT only. It is useful for injecting simple test signals into NEWVECTFFT.

There are a few programs that perform rudimentary analysis on the simulator output cross-correlation spectra. They are also located in the /public subdirectory. They are :

ANALYZE : calculates the mean and sigma in amplitude and phase over a specified channel range.

DIFFSPEC : calculates the absolute and relative differences between two specified spectra.

VIEWNOIS : calculates the total power spectrum for a given model data set.



FX SIMULATOR
RAV SIGNAL GENERATOR
J. BENSON OCT 1986

Figure 1.

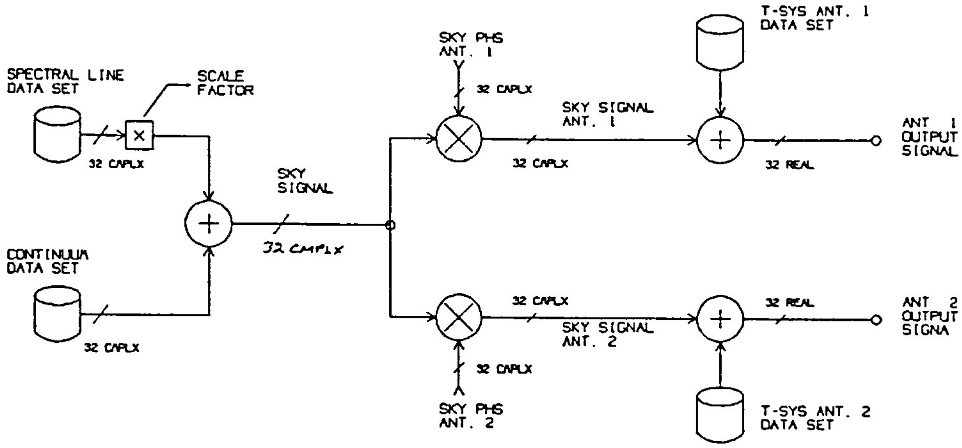


Figure 2a.

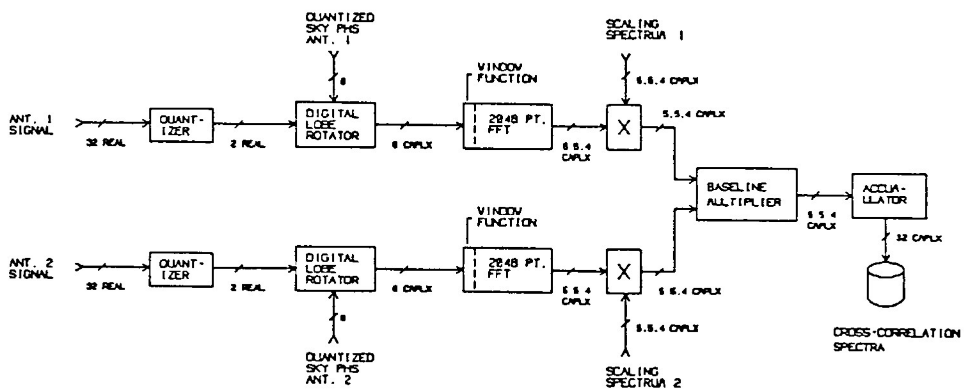


Figure 2b.

FX CORRELATOR - SIMULATOR
J BENSON OCT 1988

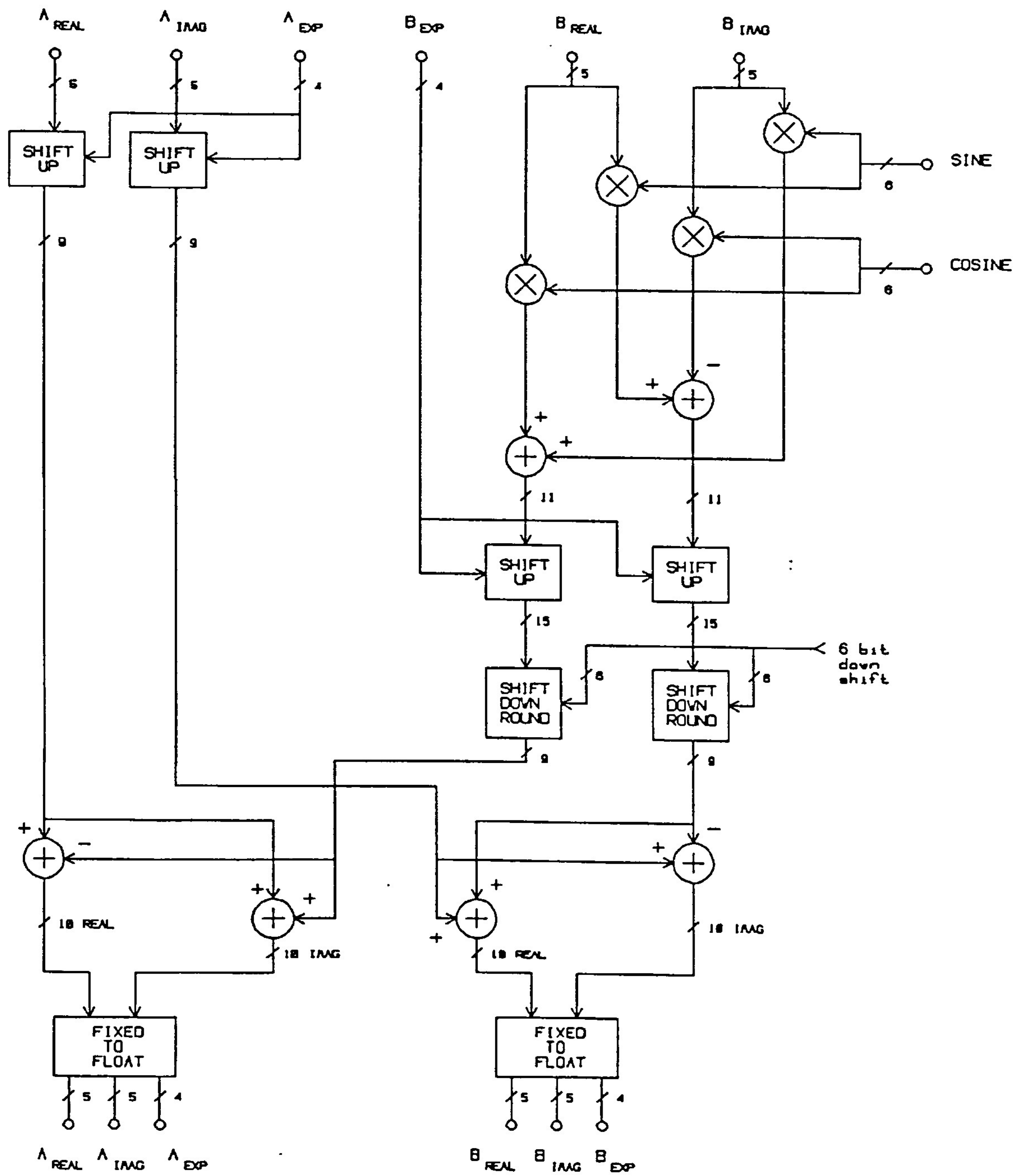


Figure 3.

FX SIMULATOR
5.5.4 BUTTERFLY
J. BENSON OCT 1986