MARK IV MEMO #256

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
HAYSTACK OBSERVATORY

WESTFORD, MASSACHUSETTS 01886

27 March, 1998

Telephone: 508-692-4764
Fax: 617-981-0590

To: Mark4 Development Group

From: Joel I. Goodman

Re: Initial results decoding data from tape recorded at 84Kfci

Related Documents

[1] ‘A DSP read channel for increased linear density in VLBI tape recording,’ Mark4 Memo #247
[2] ‘Timing recovery and gain control for proposed DSP read channel,’ Mark4 Memo #259

Introduction

Pseudo-random data recorded on a Mark3 data-acquisition system at 84 K flux changes per inch (84 Kfci = 1.5 x 56Kfci) was played back through standard equalizers and digitally sampled at high speed. Digital data corresponding to approximately 45,000 recorded samples were captured and processed through advanced software decoding algorithms with no detected errors. Extensions of this technique are applicable to densities as high as 112 Kfci.

Discussion

In July 1997 Hans Hinteregger and I recorded two Sony D1K tapes at 84Kfci at 80ips using a standard Mark3 data-acquisition system with standard headstacks. One tape was recorded with the standard pseudo-random test pattern available from the Mark3 formatter and the other with an ‘all-zeros’ data pattern replacing the pseudo-random test pattern. The tapes were subsequently played back on the same transport at 80 ips through an analog Mark3 equalizer, whose analog output was digitally sampled at 25Msamples/sec (sample = 8 bits) using a Lecroy 9304 digital oscilloscope. Approximately 0.25 MB of contiguous data, spanning ~ 45,000 recorded data bits, was captured into scope memory from each tape and written to floppy disk for analysis on a standard PC computer.

Starting in August, I began to develop a software model to decode the data captured on disk. The software components included a maximum likelihood sequence detector (MLSD) [1], adaptive equalizer [1], timing recovery [2], gain control [2] and baseline recovery [2]. I also developed software that removed the effect of the analog equalizer (deconvolution) and resampled the data (multi-rate filter). Data resampling was necessary given the fact that the scope's internal sampling rate used to capture data was an asynchronous non-integer multiple of the recorder clock.

Using visualization software I developed in Microsoft C++, a partial response was chosen that did not require shaping the raw response. The MLSD, timing recovery (also Maximum Likelihood) and gain control were refined to operate on the 'isolated transition' all-zeros data (odd parity provides this isolated transition). Adaptive equalization was also trained on all-zeros data and was very effective in truncating the tails of the response not included in the MLSD model. In the all-zeros case two sync blocks were successfully decoded without error (only two were available). Closer analysis of the sync block, which contains back to back transitions, showed that linear superposition does not hold perfectly, and that the equalizer (linear) would need to be trained on pseudo random data to obtain a 'compromise filter' that truncates the tails of the response. All in all, the sync blocks and roughly 45,000 bits were decoded without error.

Using only the MLSD, timing recovery and gain control software models, I processed pseudo-random data generated by the Mark3 formatter. Alan (Whitney) gave me the Mark3 formatter template pattern used to write pseudo random data on tape that I read back and decoded in software. With help from Hans, we were able to find the proper match in the data between the decoded software output and Alan's template. Over 2500 bits (of the 45,000) were checked by hand (before we stopped), and these matched up without error. Future work will automate this checking process. Figure 1 gives a block diagram representation of the processing involved to decode data at 84Kfci.

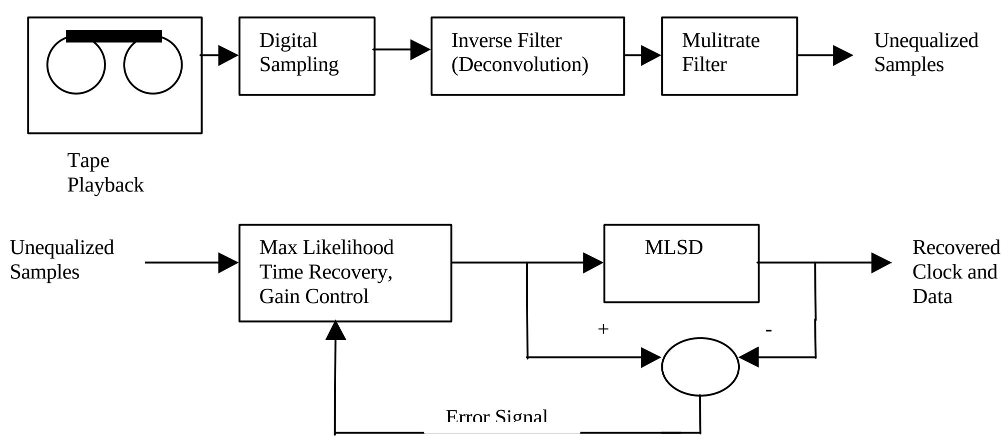

Figure 1.

Projection to Even Higher Density

Though the described technique appears to work well at 84Kfci, non-linear superposition is likely to be a stumbling block to decode data at linear densities greater than 112Kfci. However there are simple techniques that can be applied to account for this phenomenon, including increasing the number of states and paths in the MLSD to encompass data dependent noise and non-linear superposition.

Further refinement of the model includes incorporating adaptive equalization and processing data with a high degree of non-gaussian data dependent noise. Including adaptive equalization simply involves the process of training the existing software model on a known pseudo random pattern. Handling non-linear superposition requires more research into characterizing the nature of interference at high densities, and then augmenting the MLSD to account for this.

Summary

The file of 45,000 all-zeros data with associated sync blocks recorded at 84Kfci was successfully decoded without error by the software model I developed. Over 2500 bits (of the 45,000) of pseudo random decoded by software matched perfectly with the template used in their generation. Because this was checked by hand (initially by two people) and is labor intensive, I did not check all 45,000 samples. Future work will include automatic procedures to measure error rate. Initial indications are very positive with no errors detected.

The model used to decode the data is very simple, a 4 state MLSD, digital timing recovery with 4 multipliers and adders, and gain control with 2 multipliers and adders. Equalization (adaptive or fixed) will only enhance performance, and in all likelihood is not necessary at this density. Implementing a single channel in hardware in a standard FPGA should pose no technical difficulty or risk.