July 10, 2003

A New Method for No-Clock-Head (NCH) Servo Track Writing

1 Background

A fundamental recording problem which distinguishes servo track writing from data storage and retrieval is the requirement for track to track coherence. Prior art depends on the use of a reference clock head which drives the recording process for all servo tracks. Timing signals derived from the reference clock establish an accurate angular positioning reference so that adjacent tracks can be written synchronously.

In the following paragraphs, a new method for writing coherent tracks is described. This method does not require a separate clock head, and is therefore one of the family of NCH technologies. Such techniques can be used to completely eliminate the need for external servo writer hardware, as in self-servo writing, if sufficient DSP capability is built into the drive.

2 Overview

The steps required to implement the method using one of the drive data heads as a servo writer are:

- Write a closed reference track at the OD crash stop.
- Initialize angular partition counters to section the track into alternating 'A' and 'B' blocks.
- Step 1/2 track away from crash stop.
- Rewrite the 'B' blocks with progressively increasing delays.

- Step back to the crash stop.
- Scan the 'B' blocks to determine the one with maximum amplitude.
- Save delay associated with above 'B' block.
- Rewrite the 'B' blocks with optimal delay.

At the conclusion of the above sequence, the correct delay required to write and read at the same physical location has been determined. This process and the supporting hardware are described in detail in the following paragraphs.

3 Track Closure

Track closure at the crash stop is achieved by repeatedly invoking a *write-test-modify* sequence until satisfactory results are obtained. This is similar to the method used in a conventional servo writer and no additional constraints are imposed by the NCH method.

4 Phase Detector

The phase detector[1] required is similar to the detector used in existing hardware. A suitable design is suggested in the following figure.

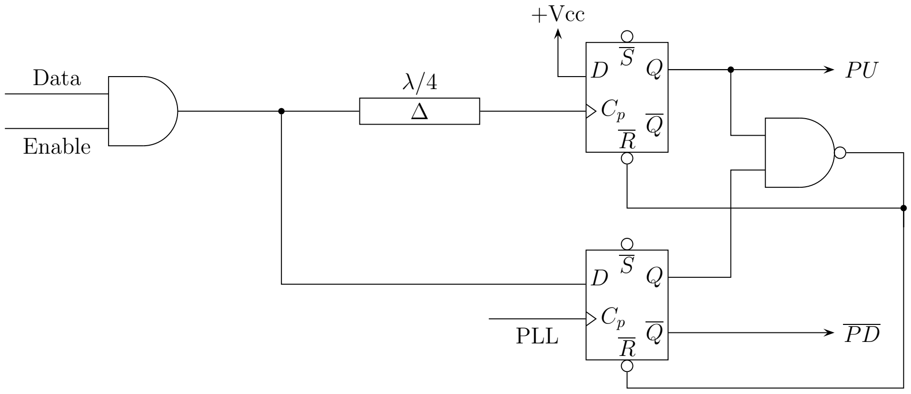

Figure 1. Gated phase/frequency detector (PFD).

[1] Actually, phase/frequency detector.

Note that this detector is capable of skipping over missing pulses and gaps without error, as long as the reappearance of pulses is synchronous with the previous pulses.

During servo track writing, the PFD will be periodically disabled by gating functions generated in the timing logic, causing the PLL to *float* or *coast* for short periods.

5 Clock Counters/Comparators

Timing signals are generated with counters and gates driven from a master clock counter. Since the recorded data will contain gaps, the master clock is driven by the VCO, which runs continuously. Counters, comparators and latches are used to signal the following events:

- Start of track (index).
- Location of 'A' blocks.
- Location of 'B' blocks.
- Phase detector enable/disable gates.
- Programmable delay change.

6 Track Layout

A simplified track layout which will be used to explain the method is shown in Figure 2.

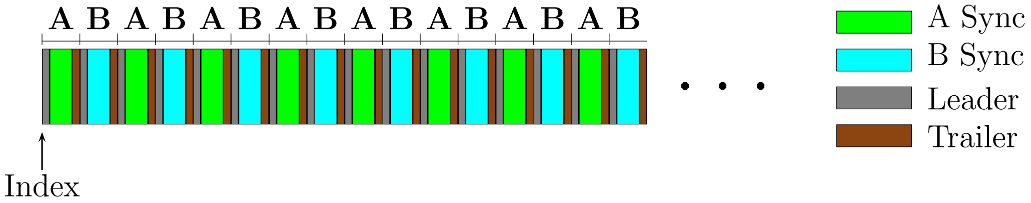

Figure 2. Sample Track Layout

As will be shown later, the 'A' and 'B' sync regions are for PLL locking. The leader and trailer regions provide guard bands around the sync regions to guarantee distortion free boundaries for them. There are also some erase bands at the start of leaders and end of trailers. These are shown in the expanded view of Figure 3.

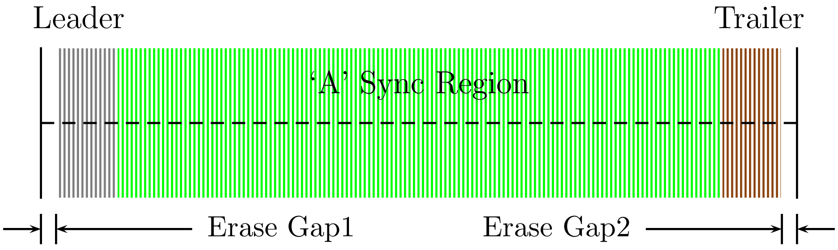

Figure 3. Detail of 'A' Block.

The algorithm used to establish track coherence involves rewriting the 'B' blocks with progressive delays. An accurate timing reference for writing is possible by floating the PLL during write operations and resyncing the PLL in the 'A' blocks, as shown in Figure 4.

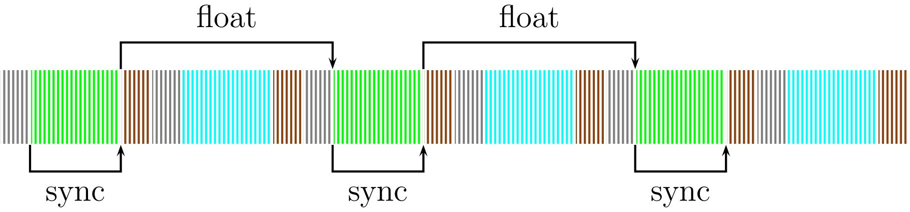

Figure 4. PLL Control Sequence

7 Signal Modeling

In order to quantify the signal properties it is useful to devise an analytic model of the coherence problem. We begin with a modified expression for an infinite train of Lorentzian pulses,

$$V(x,s,\theta) = k\frac{\sinh(k)\cos(\theta+\pi x/s)}{\cosh^2(k)-\cos^2(\theta+\pi x/s)}$$

where s is the spacing between pulses normalized to PW_{50} and θ is the angular displacement of the sequence. Using this equation, we can generate the signals corresponding to any combination of track offsets and adjacent track incoherencies, such as the cases $s = 6$ in Figure 5.

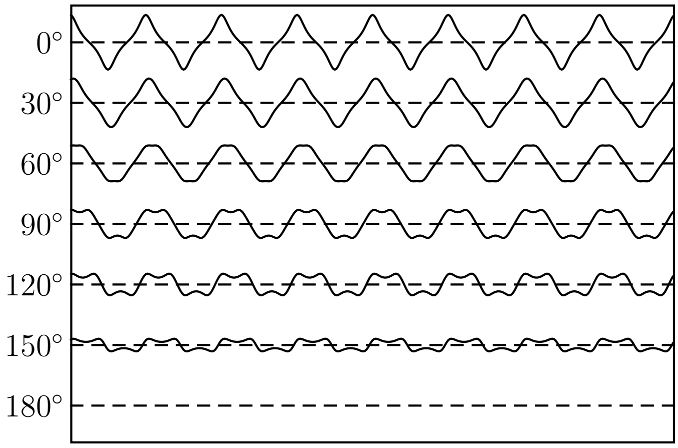

Figure 5. Summation of Signals with Variable Lag

8 Coherence Function

Another view of the signal space is shown in Figure 6, where a family of curves for differing pulse spacings is plotted. The plots show the variation of maximum amplitude of the signal vs. angular lag.

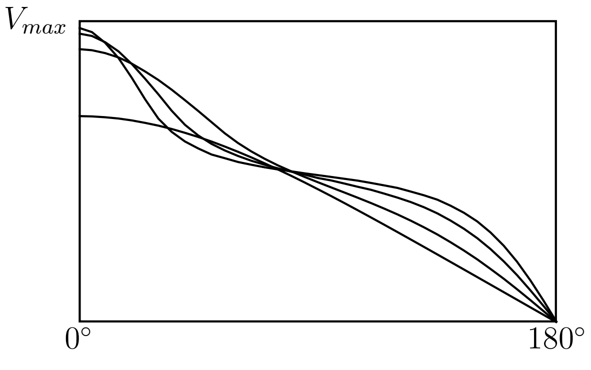

Figure 6. Coherence Function for $2 \leq s \leq 8$

9 Determining Write/Read Delay for Maximum Coherence

Using the previous model, we can determine the signal properties for a test case. The following parameters are chosen for this example:

- RPM: ≈ 6000.
- Clock cycles per revolution: 100,000
- Clock pulses per revolution: 200,000
- Clock frequency: 10MHz
- Pulse spacing: 50ns
- PW_{50}: 12.5ns
- Number of 'A' blocks: 200
- Size of 'A' block 1000 pulses

Given these parameters, a suitable programmable delay line is the Data Delay Devices part no. PDU18F. This device is available with 255 delay values in 0.5nsec steps. Hence a range of delayed signal bursts can be written for the 200 'A' blocks covering delays from 0 to 100ns.

In operation, after achieving tracks closure and determining the 'A' block counts, the head is moved 1/2 track away from the OD crash stop and the 'B' blocks are rewritten with progressively increasing delays. The head is then returned to the crash stop. After moving back to the crash stop, the observed signal will resemble that in the track fragment shown in Figure 7.[2]

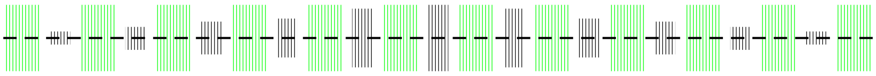

Figure 7. Signal Variation with Programmed Delay

[2] This fragment is compressed to present delay steps of about 30°. For the example, steps would be about 2°.

It is clear that a gated peak detector which is activated in the 'B' blocks can find that block which has maximum amplitude. This block is the one with best coherence. Since the relation between delay and block ID is fixed, the correct delay for track coherency can now be determined.

Depending on the thickness of the head pole which separates the read and write transducers, the spacing involved in this delay measurement may exceed one or two pulses. The above algorithm solves for the fractional delay, but it still remains to find the number of additional whole pulses (if any) that are involved.

If the rewritten 'B' blocks are preceded by an erase gap, it will be possible to enable a latch which catches the clock count associated with the first written pulse in the coherent block. This suffices to construct the exact delay required.

10 Checkerboard Patterns

The delay required for track coherence is only valid over a small region of the disk. The measurement must be corrected for track skew, or retaken periodically as servo writing progresses.

The incremental method which places servo tracks on the media begins with the process just described. Once the reference track has been used to find the correct delay, the track can be rewritten with appropriate servo patterns which must include PLL sync regions.[3] Then the head can be moved 1/2 track and, periodically relocking the PLL in the sync regions, alternating blocks of new patterns can be written using the correct delay.

[3] These can be in data blocks.

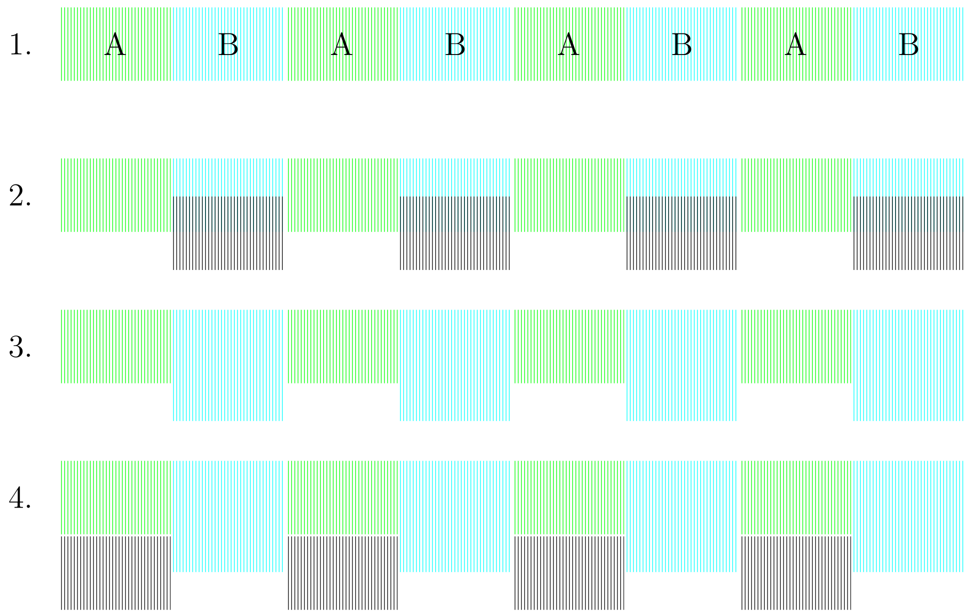

Figure 8. Progressive Servo Writing Steps

Summarizing the steps:

1. Re-record reference track after determining coherence delay.
2. Step 1/2 track and record synchronous ‘B’ blocks.
3. (Illustrates synchronous ‘B’ blocks.)
4. Step full track from OD crash stop, sync on ‘B’ blocks and write synchronous ‘A’ blocks.

After this, the servo writing process can proceed by moving in full track steps alternately synching in ’A’ regions and writing ‘B’ blocks or synching in ‘B’ blocks and writing ‘A’ blocks.

This process should continue until a recalibration point is reached or the drive is done.

11 Conclusion

A simple strategy for solving the problems associated with NCH servo writing has been described. Although some details have been omitted in the interest of brevity, sufficient information has been disclosed to implement the method.