SG 9700226-5
original copy

Repeatable Runout Free Servo Architecture in Direct Access Storage Device

Field of the Invention

This invention relates to direct access storage devices. More particularly this invention relates to a servo system architecture and a method of operating the servo system in devices such as a disk drive.

Background Art

High precision and low power electromechanical system design is required for future direct access storage devices (DASDs), particularly targeted for mobile computer systems. Without limiting operating vibration performance, servo architecture of a DASD should be capable of providing low positioning error without exceeding a competitive power budget.

Conventional methods attempt to achieve minimum positioning error by reducing the repeatable runout (RRO) component, that is any deviation from an ideal circular track, due to for example mechanical tolerances or servo writing, at the manufacturing level by imposing very severe manufacturing tolerance requirements, or by implementing gain enhancing algorithms, such as feedforward or narrow band filters, at RRO spectral frequencies. These approaches either require high cost manufacturing

This Sheet Has Been Replaced

methods or increased track following voice coil motor (VCM) power. In track following mode a disk actuator servo is designed to minimize head positioning error in the presence of repeatable and non-repeatable runout components (NRRO) that are present in the position error signal (PES). Traditional methods attempt to minimize either the RRO component at source by refining the manufacturing process or suppress RRO by implementing high gain servo schemes. Minimizing RRO at the source has a manufacturing cost penalty while a servo solution that effectively generates frequency specific current to the voice coil motor (VCM) has a power penalty.

A typical DASD servo as shown in Fig. 1 has a PES generating block 10, servo computation block 12, and a digital-to-analog converter (DAC) block 14. Figure 1 also shows a feedforward scheme including a misposition correction signal generator 16 augmenting the conventional servo to improve RRO track following capability. The output of generator 16 is added to the output of block 12 at a summing node 18. The added output of node 18 is then provided to DAC block 14. The analog output of DAC block 14 is provided to a current driver 20 which provides current to the VCM (not shown in Figure 1) of the actuator block 22, which represents the mechanical components that control the position of a transducer (not shown) which interacts with a data storage medium (also not shown in Fig. 1).

In the conventional servo architecture the control signal is first computed from PES signal derived from the data surface, and then the control signal generated by the servo computation block 12 is modified by the misposition correction signal (also called a feedforward signal) from block 16. The PES generating block derived from the data

This Sheet Has Been Replaced



surface invariably contains both RRO and NRRO components, represented by block 24. It will be understood that the signal sent to block 10 is the relative difference between a head position signal received from block 22 and the runout components of a track as represented by block 24. While a subtraction node 25 is shown, it will be understood that this is a theoretical node because it is only the difference, which is generated by block 10 that is available when the recording head signal is processed for use by the servo loop.

SUMMARY OF THE INVENTION

The presence of RRO not only impedes precision of track-to-track spacing but also increases idle mode power. A "No-RRO Servo Architecture" is defined where the servo algorithm avoids tracking the RRO component rather than suppressing the RRO component. The RRO component of the PES is removed by subtracting the "locked arm" RRO prior to generating the servo controller output. The No-RRO Servo Architecture produces high precision track spacing while maintaining conventional crossover frequency. The locked-arm RRO matrix for each sector (referred to as RRO*) can be either measured-and-stored at the manufacturing facility soon after the servo-sectors are written or it can be post computed from product PES and stored using an algorithm.

Basic to the invention is the ability to estimate the unknown RRO* components in the presence of the servo-loop using an inverse filtering concept. The invention simultaneously minimizes VCM power and PES magnitude, thus providing a new path for high track density DASDs at low cost and power. Depending on the

This Sheet Has Been Replaced



implementation resources available on board such as RAM space or computational power, the architecture can be invoked to either save only idle power, or improve both precision and power. The invention may be implemented, for example, in a 3.5" DASD to enhance precision in track following mode. The RRO* is measured and stored in RAM in this configuration. Approximately a 35% reduction in position error signal RMS value is achieved without sacrificing vibration rejection performance.

When RAM space is not available, the RRO* is estimated in real-time in order to implement the No-RRO Servo Architecture. This configuration requires extra spindle revolutions, and is effective only to save idle power and not for high precision tracking. The energy normally spent in tracking RRO components is conserved by this method in which approximately 45% of the VCM RMS power saving is achieved.

Brief Description of the Drawings

Fig. 1 is a block diagram of a prior art servo loop.

Fig. 2 is a conceptual block diagram of a servo loop in accordance with the invention.

Fig. 3 is a block diagram of a first embodiment of a servo loop in accordance with the invention.

Fig. 4A is a PES spectrum of a servo loop operating in accordance with the invention.

Fig. 4B is a PES spectrum of a servo loop operating in accordance with the prior art.

Fig. 5 is a representation of a second embodiment of the invention.

Fig. 5A illustrates a variation of the embodiment of Fig. 5.

Fig. 6 is a general flow diagram of a method which may be used to implement the invention.

This Sheet Has Been Replaced



Fig. 7A is a VCM current spectrum with and without no-RRO architecture.

Fig. 8A is a typical open loop transfer function used to estimate RRO* using an inverse filter.

Fig. 8B illustrates mismatch between the original RRO* and estimated RRO*.

Fig. 9A is a modified open loop transfer function used to improve the estimate of RRO* using an inverse filter,

Fig. 9B illustrates the original RRO* and Estimated RRO* being in good agreement.

Detailed Description of the Preferred Embodiments

Since the RRO component is known to be stable over the life of a DASD and has a well defined amplitude and phase characteristics, the new servo architecture in accordance with the invention exploits this fact by actively avoiding the RRO signal entering the servo computation block of the servo system. In this invention a conventional VCM servo is modified to achieve low power idle and high precision track-to-track spacing by eliminating the RRO component in the PES signal while preserving the desirable vibration rejection characteristics. Alternatively, the original track layout is electronically modified and mapped into "virtual tracks" that are free of RRO components.

The 'No-RRO Servo Architecture" is based on the fact that the RRO that is inevitably included in the PES stream need not be track-followed. This is the key to this

This Sheet Has Been Replaced



innovation and the concept is shown in Fig. 2. The only PES components that are random are the NRRO and vibration induced PES.

In Fig. 2, wherein like numerals are used to describe like blocks found in Fig. 1, it can be seen that the PES signal from the data surface is corrected by removing the RRO* component, represented by block 26, by being subtracted from the output of block 10 at node 28, by way of line 30, before being presented to the servo computation block 12. The output of servo computation block 12 is then presented to the DAC block 14 with no further correction such as feedforward compensation. The invention can be implemented in various forms where the method of generating and storing RRO* may differ from each other.

In order to achieve the No-RRO Servo Architecture the RRO* corresponding to each sector on every track in a DASD under servoless condition must be known in advance. For example, if a DASD has 5000 tracks with 50 sectors per track, then the RRO* will be a matrix of 5000 x 50 elements. The RRO* can be either "measured" and stored at the manufacturing site or else "estimated" and stored using product servo and microcode.

The RRO* can be simply measured while the arm is locked to an external reference during manufacturing, and then recorded in the same DASD for future use. This is called measured RRO*.

Figure 3 shows a RAM-based implementation of measured RRO* in which the RRO* (from a DASD 32, having a disc 34, which interacts with a head 36 at the end of an actuator 38, when the disk is rotated by a spindle motor 40 in a manner well known) is stored in a reserved data zone of the disk during manufacturing, and read into a RAM

This Sheet Has Been Replaced



42 during power up. The RAM-based RRO* is then subtracted from the PES in the track-following mode.

Fig. 4A and Fig. 4B show experimentally obtained PES spectra with and without the No-RRO Servo Architecture in which the RRO* for each sector is stored in a RAM and subtracted from the PES stream in track-following mode. A reduction in total RMS error of approximately 30% is demonstrated in this case. Almost all RRO spectral peaks have been effectively removed.

The "Locked-Arm RRO" can also be embedded in the servo pattern as shown in Fig. 5. In this case no RAM space is required, but more servo write time is needed.

A track center line 50, represented as a straight line in Fig. 5, is divided into a plurality of sectors 52. In the prior art, each sector 52 includes stored data for cylinder identification (ID) 54, servo signal data 56 and finally user data 58. In the practice of the invention, the RRO* data is stored in a block 57 between cylinder ID data 54 and user data 56. In practice, a decode electronics block 60 decodes the data from the disk 34 and provides it node 28 in a manner analogous to block 26 of Fig. 2 or RAM 42 of Fig. 3.

The digitally encoded locked-arm RRO data is decoded and subtracted from the PES stream by the microprocessor which controls the servo loop. The realization of No-RRO servo architecture according to Fig. 5 is identical to the performance of the RAM-based method except that the cost of RAM is exchanged for the complexity of servo writing the measured RRO* adjacent to the conventional servo signal band.

As illustrated in Fig. 5A, a compromise between excessive RAM space and excessive servo-write time can be made by estimating the RRO* using the inverse filter during the typical 24 hour test process that each DASD undergoes and then by recording the estimated RRO* in the user data zone as a reserved byte. For example if the standard

This Sheet Has Been Replaced



data block size is 512 bytes, then the 513th byte of the previous data sector will contain the estimated RRO*. This is illustrated in Fig. 5A by data block 62. For the nth sector PES, the previous data zone (n-1) will provide the necessary byte corresponding to RRO*. During the read mode of a file this is a continuous process. The hard disk controller must identify the reserved byte from the user bytes and present it to the servo controller operation. This may be accomplished in a hard disc controller chip 64. During the write process care should be exercised to allow for a quick read of the reserved byte. With the invention of magnetoresistive heads, the read and write operations are independent and the task of reading RRO* should not be of any concern.

A second approach is to use the product DASD servo system to position the head and then estimate the RRO* matrix. In this case an external reference is not required, but the DASD servo system gain must be accounted for in order to obtain RRO*. The equivalent of a "locked-arm RRO" can be estimated and stored using an "inverse filtering algorithm", which is an alternative means to implement the no-RRO servo architecture. The inverse filtering method is applied to generate the RRO* matrix and stored on the disk in a manner similar to the case shown in Fig.3. Both methods require RAM space for precision tracking. On the other hand RRO* can be estimated in real time on a track to track basis to conserve idle-mode power without requiring a large RAM space.

Fig. 6 shows an algorithm for extracting the RRO* in three stages. In the first stage the PES signal corresponding to a given track is averaged on a sector-to-sector basis to estimate the RRO with the servo loop active. A microprocessor (MPU) 70 performs not only this data handling, but also performs the functions of servo computation

This Sheet Has Been Replaced



block 12 (Fig. 2) to control the servo loop. This operation will produce m number of data points corresponding to an m sector DASD where m is typically about 50. This operation can employ N number of revolutions where N is about 20 revolutions. The RRO is the effect of yet unknown locked-arm runout components (earlier defined as RRO*) acting through the (1/(1+G)) rejection transfer function of the servo/mechanical system where G is the total open loop transfer function of the system. At the end of this first stage, the averaged RRO data for the m points is available in block 74 for use in the second stage.

In stage two the estimated m-point RRO (with the servoloop active) is passed through a digital filter 76 with the transfer function (1+G). The (1+G) can be viewed as a digital filter operating on the m-point RRO input to recover the unknown RRO* through the inverse operation, and hence the term inverse filter. The input RRO is concatenated for several revolutions as represented by successive data sets 74A, 74B, etc. (one for each revolution) so that any inverse-filter transients are allowed to decay prior to storing of the output. The transient filter output is represented by block 78, an intermediate output is represented by block 80, and the steady state output is represented by block 82.

The block 82 output of the filter corresponding to m-point RRO input is stored in a memory as m-point RRO* (i.e., locked arm RRO) for use in the third stage. While the third output set of filter 76 is used in Fig. 6, it will be understood by those skilled in the art that it may be desirable to use a later output set, depending on how quickly the output of filter 76 reaches a steady state.

This Sheet Has Been Replaced



In stage three, the stored values of block 82 are used as the estimated RRO* for operation of the servo loop. The estimated RRO* is subtracted from the PES signal in real time at node 28, so that the control algorithm operates on the difference between the PES and the estimated RRO*. For each track/sector this operation need to be repeated as warranted if RRO* is not generated and stored all at once following the manufacture of the DASD. When RRO* is computed in real time, a large memory space is not required, but the no-RRO servo architecture mode can not be invoked instantly.

The concept is implemented and evaluated on a 6000 TPI 3.5" DASD simulation model. The inverse filter is implemented with a proportional and derivative servo controller only (with no integral term) so that low frequency estimation errors are not adversely enhanced by the numerical operations required to produce RRO*. Fig.7A and Fig.7B show the conventional track following mode and RRO-free idle mode spectrum of actuator current. The nominal 6.6 mA RMS is reduced to 3.6 mA by this operation (45% reduction). Frequency spectrum shows reduction in current amplitude at frequencies around 500 Hz. The amount of power saving that is made possible by this scheme depends on the magnitude of RRO present. Larger RRO components will result in greater power savings.

To identify limitations of the inverse filter-based RRO* estimation process two numerical experiments are conducted with a comprehensive dynamic simulation model. Fig. 8A and Fig. 8B show a conventional track-following transfer function used to estimate the RRO* from the originally servo-written in RRO*. As can be seen in Fig. 8B, the match is not exact, and it demonstrates the non-obvious nature of the inverse filter concept. The random processes associated with a disk file causes the servo-on

This Sheet Has Been Replaced



PES to drift due to hysteresis and electronic noise in the feedback system. The conventional controller (Fig. 8A) exaggerates the drift. As can be seen in Fig. 8B, the match between estimated RRO* and actual RRO* is not good, particularly at low frequencies. This is attributed to a complex interaction of actuator hysteresis and controller transfer function characteristics. However, when the open loop transfer function is modified to have crossover and roll-off characteristics as shown in Fig. 9A, the reconstruction of RRO* was accurate. The controller of Fig. 9A still has sufficient vibration rejection characteristics, but due to its transfer function characteristics, advantageously does not amplifying the higher harmonics. For example, the gain at the fundamental frequency of 70 Hz. (the frequency of rotation of the disk) is 0 db in the controller of Fig. 9A, which is considerably lower than the 20 db or so of the controller of Fig. 8A.

The modified versions of RRO minimization concept can be achieved in many other ways. For instance, if a single frequency RRO is found to be dominating the PES then a band pass filter is used to compute the RRO and then the corresponding RRO* is generated knowing the gain of (1 + G) at that frequency. Therefore, variations of this concept of removing RRO either totally or partially can be traded-off with simplicity of computational algorithm.

When a major disk shift is encountered after the manufacture of a DASD, it is critical to eliminate the PES error due to disk-shift while in high precision mode by means of a gain enhancing filter. On the other hand, in the idle mode the VCM power is reduced by turning off the high gain filter, or further reduced by notching the fundamental harmonic component of the PES in the PES generation block.

This Sheet Has Been Replaced



* C L M S *

CLAIMS

What is claimed is:

1. In a direct access storage device including a rotating disk, a head positioned for interaction with said disk, an actuator for carrying said head and a servo system for operating said actuator so as to follow a substantially circular track on said disk by responding to a position error signal, a method for reducing response of the servo system to radial runout of the circular tracks, comprising the steps of:

a. obtaining radial runout data;

b. filtering the data;

c. storing the filtered data;

d. subtracting the filtered data from the position error signal to obtain a corrected position error signal; and

e. using the corrected position error signal to control the servo system.

2. The method of claim 1 wherein several repetitions of the radial runout data is stored and a steady state version of the radial runout data is subtracted from the position error signal.

This Sheet Has Been Replaced

* C L M S *

3. The method of claim 1 wherein the radial runout data is stored in RAM.

4. The method of claim 1 wherein the radial runout data is stored on the disk.

5. The method of claim 4 wherein the radial runout data is stored on a user data portion of said disk.

6. The method of claim 1 wherein said step of filtering includes:

obtaining runout data for a plurality of rotations of the disk;

passing the data through an inverse filter; and

selecting a steady state output of the inverse filter.

7. In a direct access storage device including a rotating disk, a head positioned for interaction with said disk, an actuator for carrying said head and a servo system for operating said actuator so as to follow a substantially circular track on said disk by responding to a position error signal, apparatus for reducing response of the servo system to radial runout of the circular tracks, comprising:

a. means for obtaining radial runout data;

This Sheet Has Been Replaced

* C L M S *

b. means for filtering the data;

c. means for storing the filtered data;

d. means for subtracting the filtered data from the position error signal to obtain a corrected position error signal; and

e. means for controlling the servo system using the corrected position error signal.

8. The direct access storage device of claim 1 wherein said means for obtaining data obtains several repetitions of the radial runout data and said means for storing the data stores a steady state version of the radial runout data.

9. The device of claim 1 further comprising a RAM for storing data wherein the radial runout data is stored in said RAM.

10. The device of claim 1 wherein the disk is configured with portions for storing said radial runout data and said data is stored on said portions.

11. The method of claim 10 wherein the said portions are user data portions of said disk.

12. The method of claim 7 wherein said filter comprises:

means for obtaining runout data for a plurality of rotations of the disk;

This Sheet Has Been Replaced

* C L M S *

an inverse filter through which the data is passed; and

means for selecting a steady state output of the inverse filter.

This Sheet Has Been Replaced