저전력 모바일 드라이브에서의 멀티미디어 데이터 재생

최정완[0] 원유집
한양대학교 전자통신전파공학부
(chrys[0], yjwon)@ece.hanyang.ac.kr

ARM : Multimedia data retrieval in low power mobile disk drive

Jung-Wan Choi[0] You-Jip Won
Division of Electrical and Computer Engineering Hanyang University

Abstract

In this work, we present the novel scheduling algorithm of the multimedia data retrieval for the mobile disk drive. Our algorithm is focused on minimizing the power consumption involved in data retrieval from the local disk drive. The prime commodity in mobile devices is the electricity. Strict restriction on power consumption requirement of the mobile device put unique demand in designing of its hardware and software components. State of the art disk based storage subsystem becomes small enough to be embedded in handheld devices. It delivers abundant storage capacity and portability. However, it is never be trivial to integrate small hard disk or optical disk drive in handheld devices due to its excessive power consumption. Our algorithm ARM in this article generates the optimal schedule of retrieving data blocks from the mobile disk drive while guaranteeing continuous playback of multimedia data.

1. Introduction

1.1 Motivation

Electricity is the prime commodity in mobile device, e.g. smart phone, PDA, MP3 player and etc. This strict restriction on power consumption requirement of the mobile device puts unique demand in designing hardware and software components of the device. While disk based storage devices, e.g. hard disk and optical disk, provide superb storage capacity and/or portability, they have rarely been considered as the choice of storage in mobile devices. Primary reason is the power consumption behavior. Because it has mechanically moving component, e.g. disk arm, servo, etc., it consumes far more power than any other semiconductor based memory device.

Supporting the real-time playback of multimedia data in the local storage has been subjects of intense research efforts from the academia as well as the industry during the past decade. The major issue is how to guarantee the continuous flow of data. The related works address the issues in file system technology[8, 10], disk scheduling strategy[11], and task scheduling and resource allocation in operating system[3, 12]. Most of these works assumes that the storage subsystem is always in the steady state and focuses on scheduling of the data retrieval for the continuous flow of data. However, this does not hold in modern disk based storage system for the mobile device. Modern disk based storage system is not always in the steady state and in fact, it goes into idle state if it is not in use. This storage system is to minimize the power consumption of the device. While the feature of this storage system makes significant contribution in extending the battery life of the respective device, it adds another dimension of complexity in scheduling of the data retrieval.

In this work, we present the novel algorithm called Adaptive Round Merge(ARM) scheduling algorithm, which guarantees a certain disk bandwidth for the multimedia playback while minimizing the power consumption of the storage device.

1.2 Related Works

Power management on mobile devices attracts significant interests in the academia as well as in the industry. Recently, numerous works related to the hard disk power management have been done in order to minimize the power consumption[1,7]. These papers propose techniques to determine the length of active states of the disk. Simunic[9] developed a Semi-Markov model for minimization of system level power consumption. Kravets[5] developed transport level protocol to achieves power savings in wireless environment. Furthermore a software centric approach has been proposed[6]. In their works, the operating system analyzes the behavior of the tasks and shutdowns the hardware.

2. Operational Characteristics of Mobile Disk Drive

Fig. 1 illustrates the power consumption profile of the mobile disk drive. The disk drive designed for the mobile device stops rotating the platter when there are no I/O requests. This is to minimize the power consumption.

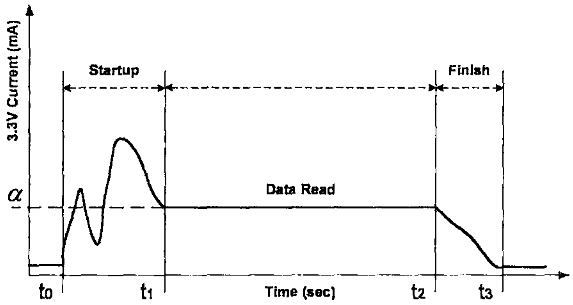


Fig. 1 Schematic View of Power Consumption Profile of Data Read Operation

The entire period of reading data blocks from the disk drive consists of three phases: Startup, Read and Finish phases. The Finish phase is followed by an idle period. Startup phase includes the operation of spin up, focus and tracking. In read phase, the disk drive transfers the data. The Finish phase includes the spin down operation to gradually stop the platter and park the head. The small size storage can be categorized into two disk drive types: non-removable media and removable media. In the non-removable medium disk, the storage media is permanently housed with the mechanical components, e.g. microdrive[2]. The storage device based on the non-removable media delivers the higher data transfer rate and superior storage capacity. In the removable medium disk drive, the storage media is detachable from the device itself. The legacy CD-ROM and the DVD-ROM belong to this category. The removable media based storage device has better $/byte ratio. However whether the storage media is detachable or not, commodity small size storage devices retrieve the data based on startup, read and finish phases. In the present paper, the Adaptive Round Merge(ARM) scheduling algorithm can be used in both environments.

3. Multimedia Data Retrieval

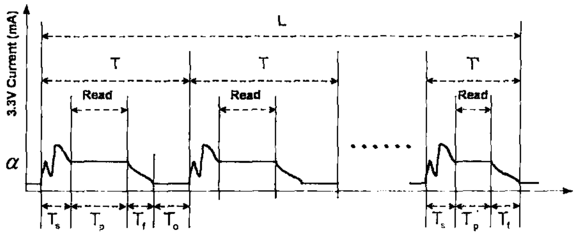


Fig. 2 Retrieval of Multimedia Data

Fig. 2 illustrates the power consumption profile when the disk retrieves the data block in the mobile environment. Let L be the playback length of the file. The entire playback period is made up of sequence in active period where the disk reads the data block at the maximum rate. Each active period is made up of startup, read and finish phases. The length and the power consumption at startup and finish phases are determined by the physical characteristics of the disk drive. Our objective is to determine the length of the read phase and the amount of data blocks during each read phase.

4. Adaptive Round Merge(ARM) Scheduling

4.1 Problem Formulation

L, r and R, respectively, denotes the playback length, multimedia playback rate, and the maximum data rate of the disk. P_s and P_f denotes the power consumption for startup and finish phases, respectively. T_s, T_p, T_f and T_o denotes the length of startup, read, finish and idle phases, respectively. B^* is a buffer size. α denotes the power consumption rate in read phase. Round is a single period, which is made up of startup, read, finish and idle phases. We first introduce the concept of Full Buffering. In Full Buffering, the disk drive retrieves the data block at the maximum rate until the available buffer space is full and then goes into finish phase. Thus, T_p corresponds to $\frac{B^*}{R-r}$ in Full Buffering. Video playback consists of sequence of rounds which have the same length except the last one. Let N be the number of rounds of the same length. Then, the entire playback consists of $N+1$ rounds. We can compute the total power consumption in Full buffering as in Eq (1).

$$P = N(P_s + P_f + \alpha T_p) + (P_s + P_f + \alpha T_p')I \qquad (1)$$

$N=\left\lfloor \frac{L}{T} \right\rfloor$ where $T=T_s+T_f+T_p+T_o$. I is an index function, which is 0 if $L \bmod T = 0$ and 1, otherwise. If playback length, L, is integer multiples of T, i.e. $I=0$, full buffering yields minimum power consumption[3].

4.2 Adaptive Round Merge

Once the buffer is full, the application only can retrieve the data blocks at playback rate. Otherwise, buffer overflow will occur. Thus, the efficiency of the power consumption is significantly degraded when the disk reads the data block continuously after the buffer is full. Due to this reason, full Buffering stops reading the data block when the buffer is full and goes into idle phase. However, if the amount of data read in the last round is very small, it may be more efficient to read this data in its immediately preceding round avoiding one startup and finish phase.

We assume that the power consumption during idle phase is negligible. B, the amount of data blocks read during T_p corresponds to $\frac{B^*}{R-r}R$ if $T_p \le \frac{B^*}{R-r}$ and $\frac{B^*}{R-r}R+(T_p-\frac{B^*}{R-r})r$, otherwise. Subsequently, the length of idle period, T_o, can be computed as $\frac{B}{r}-(T_s+T_f+T_p)$. In Full Buffering, the amount of data blocks read in the last round, B_l, corresponds to $L \cdot r - NB$ and it consumes $P_s+P_f+\alpha\frac{B_l}{R}$ of power. However, it is also possible that last B_l data is read in the preceding round, i.e. N^{th} round. We call this a round merge. If we merge the last round with its immediately preceding round, we can save the power consumption involved in startup and finish phases of the last round. However, additional power consumption to read B_l data in preceding round entails and whose amount corresponds to $\alpha\frac{B_l}{r}$. We define P^* as function to determine whether to merge the last round to its immediately preceding round as in Eq (2).

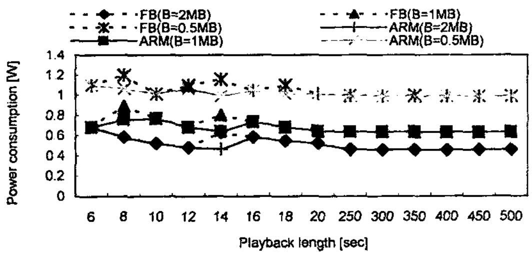
$$P^* = P_s + P_f + \alpha B_l(\frac{1}{R} - \frac{1}{r}) \qquad (2)$$

If $P^* > 0$, the last round is merged with the preceding one. Otherwise, last B_l data is retrieved in the separate round. This adaptive round merge algorithm achieves the minimum power consumption in retrieving given multimedia data[4].

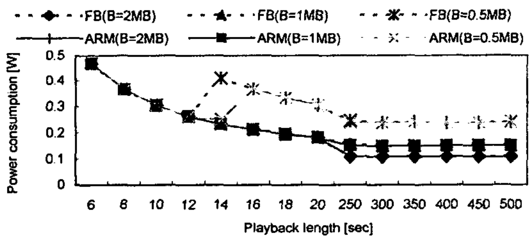
5. Experiment

The disk in this experiment is modeled after micro-optical disk DPMO-501B from DataPlay. Simulation parameters are as follows: P_s=1.42(*Watt*), P_f=1.04(*Watt*), T_s=1(sec), T_f=1(sec) and α=1.58(*Watt*/sec). We compare the power consumption rate in Full Buffering strategy and ARM.

We consider two playback rates, 1.4Mbits/sec and 300Kbits/sec and three different buffer sizes, 0.5 MB, 1 MB and 2M. In Fig.3 (a) and (b), the playback rate is 1.4Mbits/sec and 300Kbits/sec, respectively. We plot the power consumption rate of the drive based on different length video clips. We find that larger size buffer can reduce the power consumption significantly. Increasing the buffer size from 0.5 MB to 2.0 MB, we can save up to 40% of the power consumption. Our ARM algorithm can save up to 23% of the power consumption compared with full buffering. Particularly, ARM algorithm manifests itself with relatively short video clips, typically less than 20 sec.



(a) Playback Rate = 1.4Mbits/sec



(b) Playback Rate = 300Kbits/sec

Fig. 3 Power Consumption vs. Playback Length (Max Transfer Rate = 8Mbits/sec)

6. Conclusion

In this work, the ARM(Adaptive Round Merge) scheduling algorithm is presented to minimize the power consumption for the multimedia data playback. The electricity is the prime commodity in mobile devices, e.g. PDA, smart phone, MP3 player, and etc. While the disk based storage devices, e.g. hard disk and optical disk can be comfortably used in mobile devices due to their compact size, the practical usage of which leaves much to be desired due to the stringent power consumption restriction of the mobile device. To reduce the power consumption in the mobile disk based storage device, it goes into the idle mode when there is no more request to be processed. The playback of multimedia data requires that data blocks are delivered to the destination in periodic fashion. Introduction of the idle mode adds another dimension of complexity in scheduling multimedia data retrieval.

In mobile disk drive, reading the data block is preceded by startup phase and is followed by finish phase. When the buffer is full and there remains more blocks to read, we have to decide whether to continue reading or to read the data blocks in the next round. In this work, we elaborately model the power consumption of both of these situations and develop an ARM scheduling algorithm which adaptively merges the round to minimize the total power consumption.

7. Reference

[1] Adaptive power management for mobile hard drives. Technical report, IBM Co. Storage Systems Division, 1999.
[2] IBM family of microdrives. Technical report, IBM Co. Storage Technology Division, 2000.
[3] Jiandong Huang and Ding-Zhu Du. Resource Management for Continuous Multimedia Database Applications. In Proceedings of the 15th IEEE Real-Time Systems Symposium, Puerto Rico, December 1994.
[4] J.Choi and Y.Won. Low power disk scheduling for continuous media retrieval. Technical report, Division of Electrical and Ccmputing Engineering, Hanyang University, 2001.
[5] P.Kravets, Robin; Krishnan. Application-driven power management for mobile communication. Wireless Networks, 6:263-277, Sep 2000.
[6] Yung-Hsiang Lu, Luca Benini and Giovanni De Micheli. Operating system directed power reduction. In Proceeding of the International Symposium on Low Power Design, 2000. ACM, New York, NY, USA, 2000
[7] Yung-Hsiang Lu and Giovanni De Micheli. Adaptive hard disk power management on personal computers. In Proceeding of the IEEE Great Lakes Symposium on VLSI, 1999. IEEE, Los Alamitos, CA, USA, 1999.
[8] Prashand Shenoy, Pawan Goyal and Harrick Vin. Architectural considerations for next generation file systems. In Proceeding of ACM Multimedia Conference, pages pp.457-467, Orlando, FL, USA, Nov 1999.
[9] Tajana Simunic. Dynamic management of power consumption. Technical report, HP labs, 2001
[10] C.Wang, V. Goeble and T.Plagemann. Techniques to increase disk access locality in the Minorca multimedia file system. In Peoceedings of the 7th ACM Multimedia, 1999.
[11] Youjip Won and Jaideep Srivastava. "SMDP: Minimizing buffer requirements for continuous media servers". ACM/Spinger Multimedia Systems Journal, 8(2):pp. 105-117, 2000.
[12] W.Zhao and S.K.Tripathi. A resource reservation scheme for synchronized distributed multimedia sessions. International Journal of Multimedia Tools and Application, 7(1/2):133-146.