

NAND 플래시 메모리용 파일 시스템 계층에서
프로그램의 페이지 참조 패턴을 고려한 캐싱 및 선반입 정책

Caching and Prefetching Policies Using Program Page Reference Patterns on a File
System Layer for NAND Flash Memory

김경산, 김성조

kskimc@nex1.co.kr, sjkim@cau.ac.kr

Abstract

In this thesis, we design and implement a Flash Cache Core Module (FCCM) which operates on the YAFFS NAND flash memory. The FCCM applies memory replacement policy and prefetching policy based on the page reference pattern of applications. Also, implement the Clean-First memory replacement technique considering the characteristics of flash memory. In this method the decision is made according to page hit to apply prefetched waiting area. The FCCM decrease I/O hit frequency up to 37%. Compared with the linux cache and prefetching policy. Also, it operated using less memory for prefetching(maximum 24% and average 16%) compared with the linux kernel.

1. 서론

임베디드 시스템의 활용과 역할에 대한 수요가 증가하면서 저장장치로서 NAND 플래시 메모리가 널리 사용되고 있으며, NAND 플래시 메모리의 특성을 반영한 플래시 메모리용 전용 파일 시스템(YAFFS, JFFS2 등)의 개발도 활발히 진행되고 있다. 본 논문에서는 프로그램의 페이지 참조 패턴을 고려하지 않은 리눅스 커널의 캐싱 및 선반입 정책을 개선하고자[1] NAND 플래시 메모리용 파일 시스템 계층에서 동작하는 Flash Cache Core Module(이후 FCCM 이라 칭함)을 YAFFS[2] 상에서 설계 및 구현하였다. 커널의 안정성과 호환성을 최대한 확보하고 새로운

커널로의 이식에 소모되는 오버헤드와 커널과의 의존성을 최소화하기 위해 파일 시스템 계층상의 독립적인 모듈로서 설계되었다. FCCM의 성능을 측정된 결과를 제시한 후 결론과 향후 연구과제에 대해 논의한다.

2. 본론

FCCM은 메인 메모리와 플래시 메모리간의 I/O 횟수를 줄이고 보다 지능적으로 캐시 및 선반입된 메모리 공간을 관리하기 위해 플래시 메모리용 파일 시스템 계층에서 제안한 프로그램의 네 가지 페이지 참조 패턴(순차 참조, 반복 참조, 시간지역성 참조, 확률 참조[3])을 이용하는 새로운 캐시 모듈 관리 기법이다. FCCM은 프로그램의 이러한 페이지 참조 패턴을 동적으로 발견하고, 이를 플래시 메모리의 특성이 고려된 캐시 메모리 관리 정책과 선반입 정책에 반영한다. FCCM은 또한 캐시 페이지의 히트 여부를 고려한 차등적인 캐시 메모리 관리 기법을 도입하였다. FCC는 <표 1>과 같이 기능상 크게 세 개의 모듈로 구성된다.

<표 1> FCCM 모듈 구성

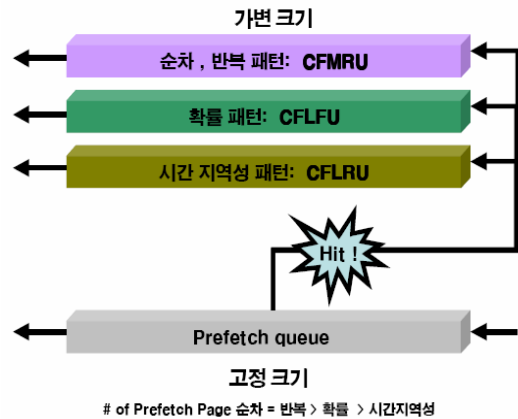
모듈	기능
패턴 탐색 모듈 (Pattern Detection)	일정 주기 동안 참조된 페이지들의 속성정보와 미래 참조 주기를 이용하여 프로그램의

Module)	페이지 참조 패턴 탐색
캐시 핵심 모듈(cache core Module)	캐시기능 수행 모듈 -페이지 캐시 기수 트리 -메모리 회수 리스트 -선 반입 대기 리스트
연결 모듈	-메모리 관리 모듈

패턴 탐색(Pattern Detection) 모듈은 프로그램의 실행 중에 페이지 접근 정보를 분석하여 프로그램의 페이지 참조 패턴을 찾아낸다. 캐시 코어(Cache Core) 모듈은 FCCM 이 파일 시스템 계층에서 캐시 기능을 수행할 수 있도록 지원한다. 이 모듈에는 페이지 캐시 기수 트리(page cache radix tree), 메모리 회수 리스트(CFLRU, CFMRU, CFLFU)[4], 선반입 대기 리스트(Prefetch Waiting List) 등이 포함된다. 마지막으로 리눅스와 연동하여 FCCM 이 동작하기 위한 연결 모듈이다. 이 모듈은 주로 리눅스 커널의 메모리 할당/해제/관리를 담당한다. (그림 1)은 FCCM 모듈의 전체 구성을 나타낸다.

3. 실험 결과

성능 테스트는 리눅스 커널 2.6.10 에서 S3C2440 MCU 기반의 OMAP5912OSK 보드에서 오리지널 리눅스 캐시 모듈과 FCCM 을 대상으로 수행되었으며, YAFFS 파일 시스템의 정규파일의 I/O 작업에 대해 횡수의 증감비율과 선반입되는 메모리양을 측정하였다. 리눅스 캐시 및 선반입 정책과 비교해 본 결과 논문에서 구현한 FCCM 모듈은 최대 37%, 평균 13%가량 I/O 비율을 줄일 수 있었다. 또한 선 반입에 소모되는 메모리양은 최대 24%, 평균 16% 줄었다. 이는 캐시의 히트 율이 증가했음을 의미하며 캐시 메모리 자원이 보다 효율적으로 관리됨을 알 수 있다.



(그림 1) FCCM 모듈

참고문헌

- [1] Robert Love, *임베디드 개발자를 위한 리눅스 커널 심층 분석*, Developer`s Library, 2004;
- [2] Aleph One Company, "Yet Another Flash File System," <http://www.aleph1.co.uk/yaffs/>.
- [3] J. Choi, San. H, "Design, Implementation, and Performance Evaluation of a Detection-Based Adaptive Block Replacement Scheme", IEEE Transactions on Computers, Vol. 51, No. 7, pp. 793-800, 2000.
- [4] E.G Coffman and P.J. Denning, *Operating System Theory*, Prentice-Hall, Englewood Cliffs, New Jersey, 1973.