

## ELSoP: 리눅스 기반 임베디드 시스템용 전력소모분석 소프트웨어 프로버 설계\*

신동석<sup>1</sup>, 남영진<sup>2</sup>, 서대화<sup>1</sup>, 이종열<sup>3</sup>, 김성률<sup>3</sup>

<sup>1</sup>경북대학교 전자전기컴퓨터학부

{zzang9, dwseo}@ee.knu.ac.kr

<sup>2</sup>대구대학교 컴퓨터 · IT공학부

yjinam@daegu.ac.kr

<sup>3</sup>임베디드소프트웨어 연구센터

{theoslee, srkim}@cest.re.kr

## ELSoP: Power Consumption Analysis Software-Prober Design for Embedded Linux System

Dong Seok Shin<sup>1</sup>, Young Jin Nam<sup>2</sup>, Dae-Wha Seo<sup>1</sup>, Jong-Yeol Lee<sup>3</sup>, Sung-Ryul Kim<sup>3</sup>

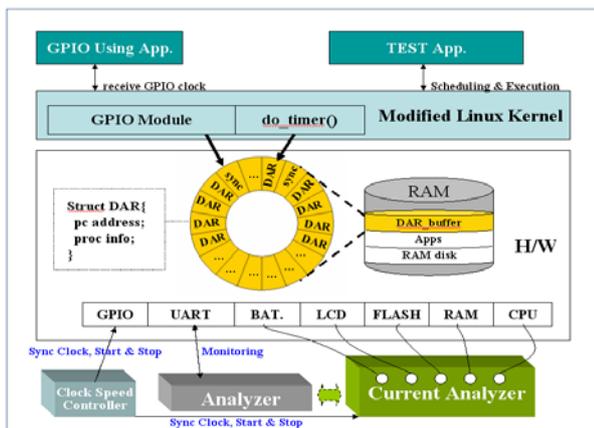
<sup>1</sup>School of Electrical Engineering & Computer Science Kyungpook National University

<sup>2</sup>School of Computer & IT Daegu University

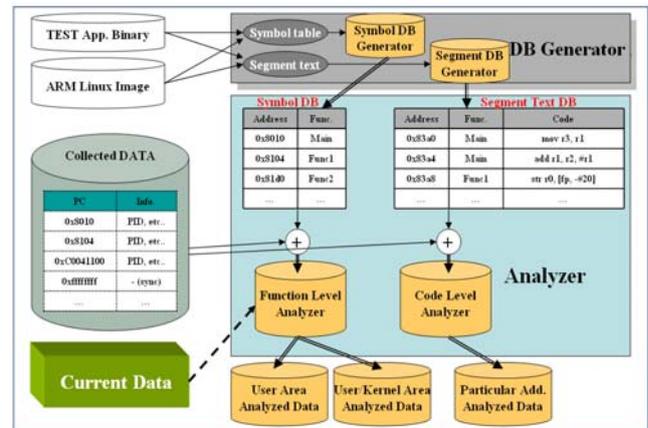
<sup>3</sup>Center for Embedded Software Technology

임베디드 시스템은 수행하고자 하는 소프트웨어의 특성에 맞는 하드웨어와 운영체제를 필요로 한다. 임베디드 시스템은 PC 환경과 달리 특정한 기능을 수행하고 설계 시 전력 소비, 메모리 량, CPU 성능, 실시간성과 같은 제약 사항들이 있다. 특히, 배터리 기반으로 동작하는 대부분의 임베디드 시스템에서 전력 소비는 중요한 이슈이다. 리눅스는 우수한 안정성과 폭넓은 개발 용이성을 바탕으로 산업용, 사무용, 가정용 등의 임베디드 시스템 분야에서 널리 사용되고 있으며, 오픈소스 기반이므로 최적화에 대한 접근성이 다른 운영체제에 비해 높다. 저 전력이 요구되는 임베디드 시스템에 리눅스를 적용하기 위해서는 커널과 커널에서 수행되는 소프트웨어의 저 전력 설계가 요구된다. 본 논문에서는 임베디드 리눅스에서 수행되는 소프트웨어를 함수 및 세그먼트 단위로 분석하는 도구인 ELSoP(Embedded Linux Software Prober)를 설계하고, 실제 타겟시스템의 소비 전력을 측정하는 멀티 레벨 하드웨어 전력측정 시스템과 통합하는 기법을 소개한다.

ELSoP의 전체적인 구조는 <그림 1>과 같다. ELSoP는 타겟시스템의 데이터 수집도구, 클럭 속도 컨트롤러, 전력분석 시스템으로 구성된다.



<그림 1> ELSoP의 구조



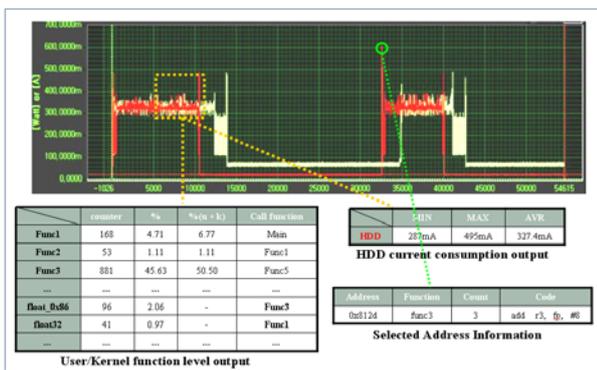
<그림 2> 분석 시스템 구조

\* 본 연구는 정보통신부 및 정보통신연구진흥원의 대학IT연구센터 지원사업의 연구결과로 수행되었음 (IITA-2006-C1090-0603-0045)

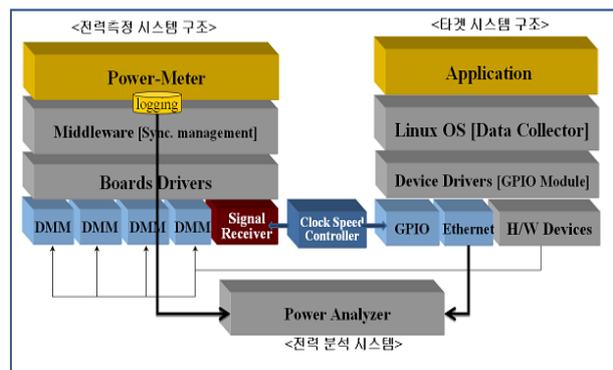
ELSoP의 실행 과정은, 우선 시리얼 통신을 이용하여 타겟시스템과 전력 분석 시스템의 네트워크 파일 시스템 설정, 프로그램 실행, 모듈 적재 등의 타겟시스템을 컨트롤 한다. 클럭 속도 컨트롤러의 첫 번째 신호가 GPIO 디바이스를 통해 들어오면, 타겟 시스템의 데이터 수집도구와 전력 측정 시스템이 구동된다. 수집도구는 클럭 속도 컨트롤러로부터 주기적인 신호를 받아 동기화 데이터를 입력한다. 전력 측정 시스템은 지속적으로 하드웨어의 전류를 샘플링하고, 동기화 신호를 저장한다. 타겟시스템의 수집 도구는 동기화 신호와는 상관없이 주기적으로 수행중인 프로그램의 PC값과 정보를 버퍼(DAR: Data Acquisition Record)에 저장한다. 프로그램의 실행이 종료되면 클럭 속도 컨트롤러는 클럭 입력을 통해 데이터 수집 도구와 전력 측정 시스템을 종료 시킨다. 종료 후, 데이터 수집 도구는 버퍼에 있는 수집된 자료를 파일로 저장하고 네트워크 파일 시스템을 통해 전력분석 시스템으로 전송한다. 전력분석 시스템은 수집도구와 전류 측정 시스템으로부터 받은 자료를 통합, 분석하여 이를 출력한다.

ELSoP의 데이터 수집도구는 리눅스 커널내의 PC값 저장 루틴, 데이터 버퍼 및 GPIO 모듈로 구성된다. 데이터 수집 처리 루틴은 PC값과 프로세스를 식별 할 수 있는 기타정보를 버퍼의 구조체 배열에 저장한다. 버퍼는 메모리 사용량을 최소화 하기위해 원형큐 구조로 구현하였다. 데이터 버퍼는 커널 심볼 테이블을 이용하여 GPIO 모듈에서도 동기화 데이터를 빠르게 저장 할 수 있으며, 커널 패치와 GPIO 모듈 적재를 통해 모든 리눅스 시스템에서 데이터 수집이 가능하기 때문에 이식성이 높다. GPIO Using App.는 GPIO 디바이스를 통해 지속적으로 동기화 신호를 받을 수 있다.

ELSoP의 분석 시스템은 크게 데이터베이스 생성도구와 분석도구로 나뉜다. 또 데이터베이스 생성도구는 심볼테이블 데이터베이스 생성기, 코드 세그먼트 데이터베이스 생성기로 구분되며, 분석도구는 요구에 따라 유저 어플리케이션 분석, 유저/커널 통합 분석 및 특정 주소 정보 분석 3가지로 나뉜다. 심볼테이블과 코드 세그먼트 데이터베이스는 어플리케이션 실행파일과 리눅스 커널 이미지 파일을 이용하여 생성한다. 심볼테이블 데이터베이스는 함수 주소정보를 포함하므로, 유저/커널 함수 수준 분석에 사용된다. 코드 세그먼트 데이터베이스는 코드 세그먼트의 모든 코드 라인에 대한 주소정보와 어셈블리 코드를 포함하며, 특정 주소 정보 분석에 사용된다. 유저 어플리케이션 분석도구는 각 유저영역 주소 값을 DB에서 찾는다. 결과 파일의 각 함수에 대한 가중치를 적용하고, 최종적으로 실행비율을 출력한다. 분석 결과를 이용해 유저 어플리케이션의 함수 실행 비율을 알 수 있으며, 오버헤드나 지연이 발생하는 함수를 확인 할 수 있다. 유저/커널 통합 분석은 수집된 모든 주소값들을 대상으로 분석한다. 분석 결과를 이용해 시스템 콜과 같은 커널 함수를 호출하는 함수에 대한 분석이 가능하다. 또한, 어플리케이션이 운영체제의 어떤 기능을 사용하는가와 같이, 유저 어플리케이션과 커널을 연계하여 분석 할 수 있다. 주소 정보 분석기는 수집된 PC주소값 중 사용자가 선택한 특정 주소에 대한 정보를 세그먼트 데이터베이스를 이용해 열람하고자 하는 주소값을 검색하고, 코드가 위치하는 함수, 해당 어셈블리 코드 및 추출 빈도를 출력한다.



<그림 3> 분석 결과 예



<그림 4> ELSoP & H/W 전력 측정 시스템의 통합구조

<그림 3>의 그래프는 전력 측정 시스템을 이용하여 실제 PMP의 동영상 실행 중 HDD의 전류 샘플링 결과이다. 그 외 각 테이블은 유저/커널 통합 분석, HDD 전력 측정 분석 및 특정 주소 정보 분석의 결과이다. 그래프 상의 점선으로 표시된 부분과 결과 테이블의 연계 분석은 함수 연구 결과에 대한 예시이다.

측정 시스템을 제외한 분석 도구 실험에서 실제 코드라인과 수행횟수에 대한 예상 비율과 분석 비율의 오차는 1% 미만으로 나타났다. 향후 연구에서는 멀티 레벨 하드웨어 전력 측정 시스템과의 연계를 통해 ELSoP의 좀 더 정확한 검증을 수행 할 것이다. 통합된 시스템은 프로그램이 사용하는 하드웨어 모듈에 대한 소모 전력과 사용 패턴과 같은 철저한 분석을 통해 저 전력 소프트웨어 설계에 대한 구체적인 방법을 제안할 수 있을 것으로 예상된다. <그림 4>는 ELSoP과 H/W 전력측정 시스템의 통합구조를 보여준다.