

실시간 임베디드 리눅스 기반의 영상진단 솔루션 설계 및

구현에 관한 연구

이상훈^o 이기화 문승진
수원대학교 정보공학대학 컴퓨터학과
{vkfka1^o, noline38, sjmoon}@mail.suwon.ac.kr

A Study on Design and Implementation of The Image Diagnostic Solution based on Real Time Embedded Linux

Sang-Hoon Lee^o Ki-Hwa Lee, Seung-Jin Moon
Dept. of Computer Science, The University of Suwon

요 약

임베디드 시스템의 발전으로 인하여 고성능화, 소형화, 편의성 등이 강조되어 지는 시점에서 의료장비 또한 임베디드 시스템 기술 도입이 빠르게 진행되어 지고 있다. 본 논문에서는 초음파 영상진단 단말기 (Magic-Probe)를 구현하기 위해 SA-1100 마이크로 프로세스 레퍼런스 보드에 리눅스 커널을 포팅하고 포팅된 플랫폼을 기반으로 초음파 영상진단을 처리하기 위한 응용프로그램의 설계 및 구현에 관하여 논하도록 하겠다.

1. 서 론

초기 임베디드 시스템은 단순한 순차적인 작업을 수행하는 프로그램에서 점차 임베디드 시스템에 맞는 운영체제가 도입 되었고 ARM, StrongARM 프로세서들과 같은 저전력으로 높은 클럭을 보이는 프로세서의 출현과 이를 사용한 레퍼런스 보드의 출시로 임베디드 디바이스의 저변이 확대되면서 고성능화, 소형화, 편의성 등의 강조되었고 이러한 흐름은 의료장비와 상통하여 의료장비에 임베디드 시스템의 기술을 도입하여 저렴한 가격으로 정확하고 손쉽게 진단과 치료를 할 수 있도록 노력하고 있다.

본 논문에서는 StrongARM SA-1110을 사용하여 임베디드 리눅스 기반의 초음파 영상 처리 단말기 Magic-Probe를 구현 하였고 현재 임베디드 시스템의 소형화, 고성능화 등에 대한 요구에 따라 적합하한 응용프로그램을 구축 하고자 한다.

논문의 구성으로는 2절에서는 관련연구로 임베디드 리눅스와 파일시스템에 대해 설명하고, 3절에서는 초음파 영상처리 단말기 응용 소프트웨어 설계 및 구현에 대해 논하며, 마지막으로 4장에서는 결론 및 향후 연구과제를 기술하겠다.

2. 관련 연구

2.1 임베디드 리눅스

현재 개발된 임베디드 운영체제로는 Window CE,

pSoS, VxWorks, uC/OS, QNX 등이 있지만 이러한 상용 운영체제들은 사이즈가 크고 커널이 재구성 되기가 어려우며 로열티가 비싼 편이다. 이에 임베디드 리눅스의 경우 커널을 개발 목적에 맞게 최적화 과정을 거쳐 크기를 줄일수 있으며, 공개된 운영체제로서 비용을 절감할 수 있는 장점이 있다.

이에 본 논문에서는 Embedded Linux를 본 연구의 기반 운영체제로 선택하였다.

2.2 임베디드 리눅스 파일 시스템

리눅스는 기본적으로 디스크와 파일에 기반을 하는 운영체제이다. 시스템의 모든 디바이스들은 /dev 디렉토리의 파일들을 이용해 접근이 가능하고, 심지어 CPU내부의 레지스터들의 상태 값들조차 /proc 디렉토리의 파일들을 통해 액세스 할 수 있다.[1] 따라서 임베디드 디바이스 역시 내부적으로 디스크에 기반을 하는 파일시스템이 존재해야 했고, 디스크가 없는 임베디드 시스템에선 이를 위해 사용가능한 메모리의 일부를 포맷하여 램 디스크로 사용해야만 한다는 제약조건이 있을 수밖에 없다.

표1 갱신빈도에 따른 파일시스템 할당

갱신빈도	용도	파일시스템	읽기/쓰기
극히 적음	root파일시스템, 각종 Library	CramFs	읽기전용
적음	시스템 S/W, 환경설정자료	JFFS	읽기/쓰기
아주 잦음	화상데이터, 브로드캐스팅된 정보	Ext2 (Ramdisk)	읽기/쓰기

이로 인해, 현재는 커널과 램 디스크 압축 이미지 모두 물리적으로 Flash ROM에 위치해 있다가, 부팅 시 램 디스크 압축 이미지를 RAM에 해체 및 적재해서 사용하는 방식이 임베디드 리눅스 디바이스의 일반적인 패러다임으로 형성되었고, 최근에는 여기에서 더 발전을 이뤄, 클라이언트 디바이스에 내장된 S/W 및 데이터들이 얼마나 빈번히 갱신 되는가에 따라 표1 과 같이 구분을 하고, 특성에 따라 각기 다른 파일 시스템을 적용시켜 최적화를 하는 추세이다.

3. Maigic-Probe 응용 소프트웨어 설계 및 구현

3.1 개발 환경

3.1.1 개발 환경 및 도구

Magic-Probe 는 팜팜테크(주) [3] 의 TynuxBox-II를 기반으로 다음과 같은 환경과 도구들을 사용하였다.

- RedHat Linux version 9.0
- RedBoot Boot Loader
- GNU tools version 2.96 (cross compiler)
- Qt/Embedded version 2.3.0
- JPEG library version 6b
- W3C Libwww version 5.4.0 library
- MP3 Decoder

3.1.2 개발환경 구성도

Magic-Probe의 개발 환경으로 타겟보드 보드와 호스트간의 Jtag 케이블, 직렬 케이블, 네트워크로, 시리얼 케이블로 연결되어지며 타겟보드의 자원 한계를 극복하기위해서 Cross Development Environment 환경과 NFS 환경을 구축한다. 이는 호스트의 자원을 활용하여 최적화 시킨후 부트 로더를 이용하여 Flash 메모리로 기록, 삭제, 수정 등의 기능을 TFTP와 BOOTP를 연계하여 처리 하게 된다.

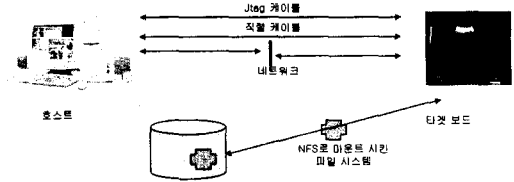


그림 1 개발환경 구성도

3.2 Magic-Probe

Magic-Probe는 임베디드 리눅스 기반의 초음파를 이용한 영상 진단 장비이다. 리눅스 기반의 QT Library를 기본으로 프로그램을 설계하였으며, 단말기 메인 화면에 여러 가지 응용 프로그램을 추가하여 성능을 극대화 할 수 있고, 특히 영상을 출력함에 있어 Probe를 부착함으로써 기존의 영상 진단 장비 기능을 갖춘 시스템을 구현 할 수 있다.

3.2.1 Magic-Probe 설계

리눅스 환경에서 램 디스크 이미지를 마운트 한 후 Magic-Probe 메인 화면 환경 프로그램을 설계하고 Calibration을 설정한 후 단말기로 포팅 한다. Magic-Probe 메인 화면은 480X320, 640X480의 해상도를 나타내기 위해서 설계 하였고, Probe가 연결이 되었을 경우 grayscale 비율을 테스트하기 위해서 임의의 픽셀을 출력 하도록 설계 하였다.

Calibration의 경우 단말기를 부팅 할 때 보정 값을 정해주기 위해서, 처음 좌표를 설정해 줄 경우 그 값이 /etc/calibration 파일에 저장이 되는데 그 좌표 넘버가 있는 Calibration 파일을 저장 하였다.

3.2.2 Magic-Probe 모드 구현

Magic-Probe 시스템을 부팅하면 단말기 영상 환경에 맞는 640X480 크기의 Magic-Probe 메인 화면이 출력 되고, 그림 5에서 하단에 위치해 있는 여러 버튼 중 Probe Select 버튼을 누르면 램 디스크에 있는 여러 Probe Image들 중 하나를 선택하는 창이 뜨면서 Scan 버튼을 누르면 그림 6처럼 화면에 480X320 크기의 화면을 출력할 수 있게 된다.

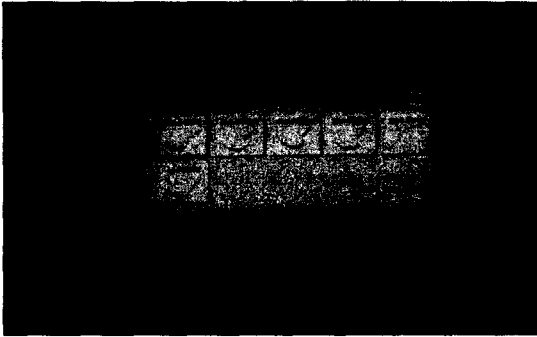


그림 2 Probe Selection 화면

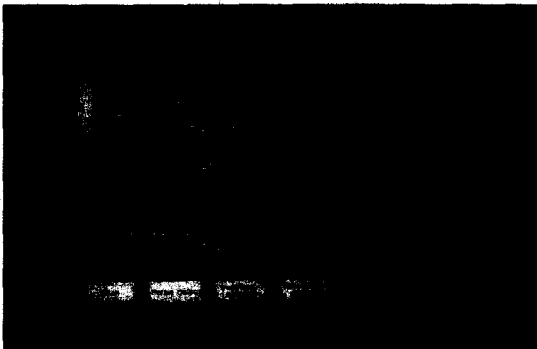


그림 3 480X320 출력 화면

출력 방법은 직접 Probe가 연결되지 않은 상태로 Test하여 그림 7에서 480X320 화면 크기의 경우 x와 y의 픽셀을 정하고, makeGrayPixel() 함수를 불러들여 gray_level을 한 단계씩 늘여 화면에 픽셀을 출력 할 수 있는 gray_level의 범위를 구현 하도록 하였다. 또한 640X480크기의 화면도 출력할 수 있도록 설정하여 전체 화면에서 볼 수 있도록 하였다.

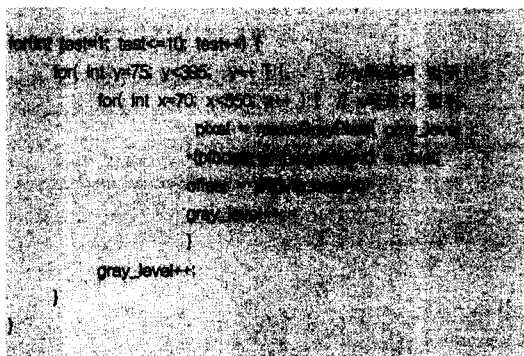


그림 4 480X320 화면 소스코드

4. 결론 및 향후 연구과제

본 시스템은 현재 타겟 보드에 리눅스가 포함되어 실시간 이미지 영상 구현을 위한 설정 기능을 가지고 있고, 사용자 인터페이스를 위한 기능을 추가하여 Probe를 직접 연결해서 화면에 출력하는 기능을 개발 중이다.

이 모듈은 휴대용 초음파 영상진단 장비로의 확대 발전으로 의료용으로의 영상진단 장비로서 그 가치가 더욱 높아질 것으로 기대된다.

참고문헌

- [1] KELP, <http://www.kelp.or.kr>
- [2] 김정기외, "임베디드 플래시 파일 시스템", 정보처리 학회지, 제9권 제1호, 2002년. 1월
- [3] 팜팜테크 (주) <http://www.palmpalm.co.kr>
- [4] 이연조, "임베디드 리눅스 프로그래밍", PC'BOOK, 서울, 2002
- [5] 팜팜테크(주), TynuxBox-II User Guide
- [6] KESL, <http://www.kesl.or.kr>
- [7] Inte Corporations, "Intel StrongARM SA-1110 Microprocessor Advanced Developer's Manual", 278240-003, June, 2000
- [8] ARM, "ARM Architecture Referece Manual", ARM DDI 0100B
- [9] Kurt Wall, "Linux Programming by Example", que
- [10] <http://linuxfromscratch.org> Linux From Scratch Project
- [11] Jeremy Bentahm, "TCP/IP LEAN : Web Servers for Embedded Systems", CMP Books, 2002. 9
- [12] 박재호, "IT Expert 임베디드 리눅스", 한빛미디어
- [13] <http://arm.linux.org.uk>
- [14] <http://www.kldp.org>
- [15] Trolltech Corporation, "Qt On-Line Reference Documentation"
- [16] <http://www.qtcenter.co.kr>
- [17] 이재일, "QT Programming Guide", <http://sonegyv.home.uos.ac.kr/programming/qt2.x>