

임베디드 단말기를 위한 범용 리눅스 배포판의 적용 방법 분석

남현우[○], 장병근^{*}, 박영욱^{*}

^{○*}한국스마트카드 T-money 기술연구소

e-mail: hw.nam@koreasmartcard.com[○] {bgjang, yw.park}@koreasmartcard.com^{*}

An Analysis of General Linux Distribution Install Method for Embedded Terminal

Hyunwoo Nam[○], Byunggeun Jang^{*}, Youngwook Park^{*}

^{○*}T-money Tech. Unit, Korea Smart Card

● 요약 ●

최근 모바일 임베디드 단말기에서는 iOS나 안드로이드와 같이 다양한 S/W 플랫폼들이 경쟁을 하고 있다. 이에 반해 전통적인 리눅스 기반 임베디드 단말기의 경우 busybox에서 제공하는 기본 명령어 외에는 필요한 Application 및 라이브러리를 모두 빌드하여 사용하였다. 하지만 PC에서는 범용적인 리눅스 배포판을 사용하여 필요한 Application이나 라이브러리를 패키지 매니저를 통해 쉽게 설치가 가능하다. 이에 따라 임베디드 단말기에서도 범용 리눅스 배포판을 설치하여 PC 에서와 같은 장점을 취하려는 방법들이 개발되는 추세이다. 본 논문에서는 현재 개발된 방법들을 비교 분석 해보고, 앞으로 개선해야 할 사항들을 도출하고자 한다.

키워드: 임베디드(Embedded), 리눅스(Linux), 배포판(Distribution)

I. 서론

현재 많은 임베디드 단말기에는 다양한 OS 및 S/W 플랫폼이 탑재되고 있다. 또한 전통적인 리눅스 기반 임베디드 단말기들도 안드로이드와 같은 플랫폼들을 탑재하고 있는 추세이다. 이는 대형 벤더들에 의해 개발되어 제공되는 개발 리소스들을 그대로 활용할 수 있고 관련 개발 인력들을 쉽게 구할 수 있기 때문이다. 특히 앱(App) 마켓과 같이 수많은 Application들을 활용할 수 있다는 장점이 부각되고 있다.

하지만 이는 개발도구나 언어, 그리고 디바이스 활용과 관련하여 플랫폼에서 제공되는 범위 안에서만 구현을 해야 한다는 제약 사항이 있다. 따라서 특정 플랫폼에 종속되기 보다는 순수 리눅스 임베디드 단말기가 개발의 높은 자유도를 보장해줄 수 있는 방안이 될 수 있다. 하지만 기존 임베디드 리눅스의 경우 busybox에서 제공하는 기본 명령어 외에는 필요한 Application 및 라이브러리 등을 모두 빌드하여 사용해야만 하는 단점이 있었다.

이와 달리 PC에서 리눅스를 사용하는 경우 페도라, 우분투와 같은 범용 리눅스 배포판을 사용하여 MS 윈도우즈 정도의 편리한 설치가 가능해졌으며, Application이나 라이브러리를 RPM, DEB 와 같은 패키지 매니저를 이용하여 쉽게 설치 및 사용이 가능해졌다. 또한 개발 도구나 하드웨어 접근과 관련하여 제약사항이 없으므로 자유로운 개발이 가능하다.

이에 따라 임베디드 단말기에서도 PC와 같은 범용 리눅스 배포판을 설치하여 기존 문제점들을 해결하고자 하는 방법들이 해킹이

나 베타 서비스 정도의 수준에서 이뤄지고 있는 추세이다. 이와 같은 노력으로 상용 모바일 단말기에 테비안이나 우분투와 같은 리눅스 배포판을 설치가 가능해졌으며, 수많은 기존 리눅스의 Application 및 라이브러리 등을 임베디드 단말기에서도 활용할 수 있게 되었다.

하지만 임베디드 단말기에 범용 리눅스 배포판의 적용 방법이 공식적으로 제공되기 보다는 해킹수준에서 이뤄져 일반화된 방법이 존재하지 않고 있다. 따라서 현재까지 시도된 다양한 방법들을 일반화 할 필요가 있으며, 각 방법들의 비교를 통해 효율적인 방안을 모색할 필요가 있다.

본 논문에서는 현재까지 시도된 다양한 방법들을 비교 분석하고자 하며, 이를 통해 임베디드 단말기에 범용 리눅스 배포판의 적용 타당성을 검증하고자 한다. 그리고 추가적인 활용 방안에 대해 모색하고, 향후 개선해야 할 항목들을 도출하고자 한다.

II. 관련 연구

1. 스마트폰 S/W 플랫폼

최근 스마트폰의 보급이 증대되면서 이에 사용되는 S/W 플랫폼의 중요성도 커지고 있는 추세이다. 세계적으로 스마트폰 플랫폼은 애플사의 iOS 기반의 iPhone과 안드로이드 기반의 스마트폰이 현재 시장에서 대부분의 점유율을 차지하고 있다.

이에 대항하여 PC 운영체제에서 강자인 MS사에서 Windows Phone을 내세우고 있지만 시장 점유율 면에서 고전을 면치 못하고 있다. 이 외에도 단말기 제조업체에서 개발하는 S/W 플랫폼도 출시되었지만 크게 성공하지 못하였다. 예를 들어보면 삼성전자의 바다 플랫폼과 노키아 사의 Maemo와 같은 플랫폼이 있다. 최근 모바일 단말기 환경에서는 S/W 플랫폼에 의해 시장이 좌우되어 그 중요성이 커지고 있다.

2. 리눅스 배포판

리눅스는 오픈소스로 개발되어 일반 데스크탑 PC뿐만 아니라 서버와 같은 엔터프라이즈 시장에서 주요 운영체제로 사용되고 있다. 리눅스는 커널이라고 하는 운영체제의 핵심 코드를 말하며 실제 운영체제로서 사용자가 사용하기 위해 필요한 라이브러리, Application, UI등의 모든 요소들을 사용자가 쉽게 설치 및 사용할 수 있도록 패키징 된 것이 리눅스 배포판이다.

주로 사용되는 배포판의 종류로는 우분투(Ubuntu), 민트(Mint), 페도라(Fedora), 오픈수세(openSUSE), 데비안(Debian) 등이 있다. 이외에도 다양한 특징들을 가진 수 많은 리눅스 배포판들이 존재한다.

대부분의 리눅스 배포판들은 x86을 타겟으로 패키징 되어 배포가 되는데, 최근 우분투의 경우 ARM 계열의 스마트폰과 같이 임베디드 단말기를 위한 배포판을 제공하기도 한다. 이는 임베디드 단말기에서도 PC에서와 같이 배포판에서 제공하는 수많은 패키지들을 사용할 수 있다는 장점이 있다.

III. 임베디드 단말기에서의 범용 리눅스 배포판 설치 방법

하이퍼바이저나 에뮬레이션과 같은 가상화 기술을 사용한 방법을 제외한 Native로 형태로 수행되는 임베디드 단말기를 위한 리눅스 배포판 설치 방법에 대해 정리하였다.

방법으로는 크게 기존 OS를 교체하여 새로운 배포판으로 부팅하여 사용하는 방법과 기존 OS 기반에서 배포판 루트 파일 시스템 이미지를 로드하여 병렬적으로 사용하는 방법이 있다.

1. 리눅스 배포판으로 직접 부팅하는 방법

이 방법은 안드로이드 기반의 도시바 AC100 넷북(ARM CPU 적용)에서 수행한 방법이다. AC100에서 우분투 리눅스 배포판을 사용하기 위해서는 SD카드를 이용한 멀티부팅 방법을 이용하는 데, 이를 위해 부트로더를 업그레이드 해야만 한다.

① 부트로더 업데이트

```
$ sudo LD_LIBRARY_PATH=. ./nvflash --bl
./prebuilt/fastboot.stock.bin --download 5
tegra-2.6.29-arm2-ac100-boot.img
```

② 미리 생성된 우분투 이미지를 sd 메모리 카드에 저장한다. 이 이미지는 우분투 공식 사이트에서 제공되는 우분투 이미지이다.

```
sudo tar xzvf tegra-rootfs.tgz --numeric-owner -C /mnt/sd
```

③ 부팅하기

sd 메모리 카드를 삽입한 후 "HOME" 버튼을 누른 상태에서 전원을 켜다. 그럼 메뉴가 뜨고, 이때 "1"번 버튼을 눌러 sd 메모리 카드로 부팅을 시작한다. 아래 그림은 부팅 과정을 보여주고 있다.

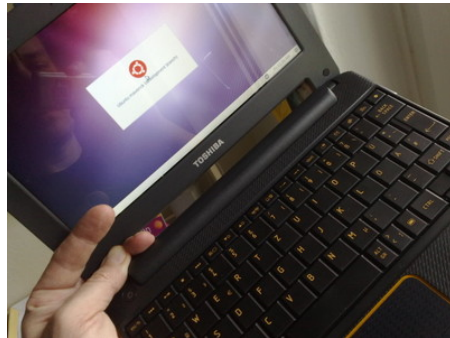


그림1. AC100의 우분투 부팅 과정

2. 기존 OS위에서 리눅스 배포판을 운영하는 방법

이 방법은 안드로이드 기반 삼성 GalaxyS와 Maemo 기반 노키아 N900 스마트폰에서 적용할 수 있는 방법이다. 기존 운영되고 있는 안드로이드나 Maemo 운영체제 위에서 우분투 배포판을 병렬적으로 운영할 수 있는 방법이다.

① root 권한 사용을 위한 루팅 작업을 수행

갤럭시S에서는 ordin이란 프로그램을 이용하여 루팅된 펌웨어로 교체해야 한다. 루팅이란 관리자인 root 권한을 사용할 수 있도록 공식 펌웨어를 해킹한 버전을 적용하는 과정을 말한다. 루팅이 완료되면 root 권한을 사용할 수 있으며, 이를 통해 우분투 배포판 설치시 권한이 필요한 작업들을 수행할 수 있다.

② 우분투 이미지 생성

이미지 파일은 아래의 명령어처럼 rootstock 라는 유틸리티를 이용하여 생성할 수 있다.

```
$ sudo rootstock -f localhost --login ubuntu --password
ubuntu --imagesize 1G --notarball
```

③ 생성된 우분투 이미지를 마운트

```
$ sudo mount -o loop IMG-FILE_NAME /mnt
sudo mount -t proc proc /mnt/proc
```

④ 우분투 이미지를 루트 파일시스템으로 변경

마운트 된 우분투 이미지를 chroot 명령어를 사용하여 루트 디렉토리로 변경해준다. 변경이 완료되면 해당 셸에선 우분투의 루트 파일 시스템의 이미지로 운영이 되며, ssh 서버 또는 vnc 서버 등을 사용하여 사용자가 접속을 할 수 있도록 기능을 제공할 수 있다.

chroot 명령어는 새로운 가상의 루트 디렉토리를 생성해주는 명령어로서 일반적으로 보안 등을 위해 원격 서비스로부터 접속할 경우 이 디렉토리의 상위로 이동을 막고자 할 때 사용되는 명령어

이다.

```
$ chroot /mnt /bin/bash
```

⑤ 원격 접속

ssh 서버를 가동시켜 터미널을 통해 우분투를 접근할 수도 있으며 vnc viewer를 설치하여 데스크탑 UI로 접근 방법도 있다.

아래는 기존 OS인 안드로이드에서 vnc 클라이언트를 이용하여 설치된 우분투 이미지로 접속한 화면을 보여준다.

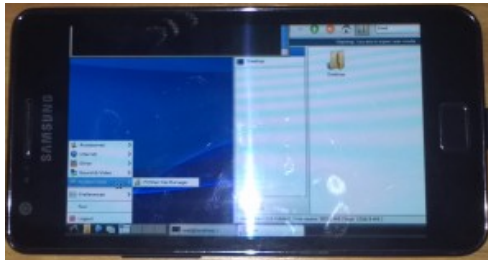


그림 2. 안드로이드 폰에서 접속

⑥ 우분투 종료

아래와 같이 우분투 이미지의 마운트를 해제하면 종료가 완료 된다.

```
$ sudo umount /proc
$ sudo umount /mnt
```

3. Rootstock

Rootstock는 리눅스 배포판인 우분투의 루트 파일시스템의 핵심 이미지를 생성해주는 도구이다. 이 도구를 사용하여 호스트 PC에서 타겟에 해당하는 우분투 이미지를 생성할 수 있다.

실제 생성시 옵션을 통해 로그인 ID나 패스워드를 설정할 수도 있으며, 이미지에 포함할 패키지의 선택 및 전체 이미지의 크기, 이미지 파일의 형태 및 lxde, kde, gnome과 같은 UI 매니저 등을 선택할 수 있다.

최종 우분투 생성된 이미지는 마운트 하여 루트 파일시스템의 내용들을 탐색 및 수정이 가능해지며, 필요한 추가 설정 작업들을 수행할 수 있다.

표 2. 배포판 설치 방법에 따른 비교 분석

구분	직접 배포판으로 부팅하는 방법	기존 OS 기반에서 배포판을 사용하는 방법
장점	<ul style="list-style-type: none"> - 하나의 OS만 로드되는 심플한 구조, - 기존 OS 기반에서 배포판을 운영하는 경우보다 필요한 저장 공간이 적음. 	<ul style="list-style-type: none"> - 안정적인 공식 지원 OS 기반에 배포판에서 제공하는 패키징 기능만을 부가적으로 사용 가능. - 기존 리눅스 계열의 OS라면 제공된 디바이스 드라이버를 그대로 활용 가능.
단점	<ul style="list-style-type: none"> - 배포판을 외부 저장 장치(SD카드)에 설치할 경우 멀티 부팅을 위해 	<ul style="list-style-type: none"> - 기존 OS에 추가적인 배포판 이미지가 모두 설치되어야 함으로 저장 공간이

<ul style="list-style-type: none"> - 부트로더를 수정하거나 기존 플래시 메모리 공간에 설치할 경우 플래시 쓰기 기능이 활성화 되어야 함. - 공식 OS에서 제공하는 디바이스 드라이버를 사용하기가 어려움. 	<ul style="list-style-type: none"> - 추가적으로 필요함. - 기존 OS가 부팅 완료 된 후 배포판을 초기화 해주는 과정이 필요함.
---	--

IV. 배포판 설치 방법에 따른 평가

범용 리눅스 배포판을 임베디드 단말기에 적용함으로써 상용 배포판에서 제공하는 수많은 Application과 라이브러리 등을 패키지 매니저를 통해 쉽게 설치를 할 수 있었으며, 또한 배포판에서 제공하는 보안 패치 등을 임베디드 단말기에도 그대로 적용이 가능해졌다. 이를 통해 ARM CPU 기반 넷북 및 패드와 같은 임베디드 단말기에서 리눅스에서 제공하는 오피스 제품군, 웹브라우저와 게임등의 Application등을 그대로 활용할 수 있게 되어 제품의 경쟁력을 높일 수 있을 것으로 예상된다.

추가 장점으로 기존 임베디드 단말기의 Application의 빌드를 PC에서 크로스 컴파일러 환경을 구축하여 수행하였는데, 범용 리눅스 배포판을 사용할 경우 임베디드 단말기 자체에서 GCC와 같은 개발 도구의 설치가 쉽게 가능해져 직접 Application의 빌드를 수행할 수 있다. 이를 통해 크로스컴파일 환경이 없어도 빌드가 가능해지며, 실 동작 환경에서 빌드가 이뤄져 빌드 후 Application을 타겟 단말기에 옮겨주는 수고를 덜어줄 수 있다. 또한 현재와 같이 듀얼코어 이상의 CPU가 사용되는 임베디드 환경에서 빌드 속도도 만족할 만한 수준으로 가능할 것이다.

하지만 단점으로 PC와 같이 범용 CPU를 환경을 고려하다 보니 패키지의 빌드시 컴파일러 최적화 옵션이 대부분 비활성화 되어 실행파일이 최적화 되지 못한 상태로 패키징 된다는 것이다. 따라서 최신 CPU를 사용 하더라도 제공되는 최신 명령어들을 활용할 수 없게 되어 CPU를 100% 활용하지 못한다. 따라서 이를 해결하기 위해선 타겟 단말기의 CPU가 정해진 후 가능한 모든 최적화 옵션을 활성화 하여 모든 패키지들을 다시 빌드한 후 패키징 하는 것이다. 이를 통해 생성된 패키지들은 개인 패키지 저장소를 사용하여 운영할 수 있을 것이다.

또한 배포판을 설치하기 위해서는 루트 권한이 필요하거나 플래시 쓰기를 위해 상용 시스템의 경우 해킹을 통해 루팅 등의 작업이 필요하다는 단점이 있다.

V. 결론

순수 임베디드 리눅스를 사용하면서 발생했던 단점들을 범용 리눅스 배포판을 적용하면서 해결할 수 방안들을 도출할 수 있었다.

하지만 대부분의 임베디드 모바일 단말기 제조사에서는 공식적인 지원을 하지 않고 있다는 단점이 있다. 하지만 이러한 단점에도 불구하고 범용 리눅스 배포판을 사용함으로써 얻을 수 있는 이점들이 많아 모바일 단말기 외에도 교통 단말기나 기타 상용 임베디

드 제품에 적용하는데 가치가 있다고 결론 낼 수 있다. 또한 ARM CPU를 사용하는 넷북이나 패드 형태의 단말기에서는 범용 리눅스 배포판을 적용하는 것이 다양한 Application들을 수용할 수 있는 좋은 방법이라고 예상된다.

향후 기존과 달리 표준화된 임베디드 리눅스 설치 방안 마련이 필요 하다고 볼 수 있다.

참고문헌

- [1] CW. Baek, "A compulsory utility of Unix & Linux", Hanbit Media, Nov, 2004.
- [2] <http://www.ubuntu.com>
- [3] <http://fedoraproject.org>
- [4] <https://wiki.ubuntu.com/ARM/RootStock/>

감사의 글

이 논문은 2010년도 지식경제부의 지원을 받아 수행된 연구 사업임(WBS AFC 표준 SW 솔루션 개발, 과제번호 10038474)