
임베디드 리눅스 시스템에서의 GUI 구현을 위한 그래픽 플랫폼 성능 평가

김영주*

*신라대학교

Performance Evaluation of Graphic Platforms for GUI Implementation in Embedded Linux System

Young-ju Kim*

*Silla University

E-mail : yjkim@silla.ac.kr

요 약

임베디드 리눅스 시스템은 하드웨어 성능 측면에서 기존 리눅스에서 지원하는 X-window 기반의 GUI 플랫폼을 그대로 사용할 수 없다. 따라서 GUI를 지원하기 위해서 적은 자원을 사용하면서 단순화된 동작 방식을 지원하는 그래픽 라이브러리가 요구되며, 이에 오픈소스 그래픽 라이브러리들의 성능을 평가하여 멀티미디어 임베디드 시스템을 위한 효과적인 그래픽 플랫폼을 선정하고 개선점을 제안한다.

ABSTRACT

Embedded linux system can't take up the X-window-based GUI platform commonly used in PC platforms because of low performance in hardware. For the GUI implementation in embedded linux system, the lightweight graphic library is needed, which supports simplified functions and the usage of little resource. In this paper, several of popular open-source graphic libraries are selected and evaluated for the adoption of an efficient library in the GUI implementation for multimedia embedded system requiring the simple and intuitional user interface.

키워드

임베디드 GUI, 오픈소스 그래픽 라이브러리, 경량화, 멀티미디어 임베디드 시스템

1. 서 론

Post-PC 시대가 오면서 임베디드 시스템에 대한 관심이 고조되고 있다. 임베디드 시스템은 하드웨어 구성 및 사용 환경이 매우 다양하여 임베디드 GUI(Graphic User Interface) 방식도 이에 따라 매우 다양하게 전개되고 있다[1,2]. 예를 들면, 커피자판기처럼 간단한 LCD 패널과 몇 개의 LED를 이용한 문자 기반의 사용자 인터페이스에서부터 PC 데스크탑 GUI와 같이 복잡하고 다양한 기능의 사용자 인터페이스 등이 있다.

본 논문은 DVR(Digital Video Recorder), 셋탑 박스 그리고 디지털 TV 등과 같이 주로 임베디드 리눅스 운영체제를 채택하는 멀티미디어 임베디드 시스템을 위한 GUI 플랫폼 구현에 관심을 가진다. 멀티미디어 임베디드 시스템에서의 GUI는 640x480 이상의 높은 해상도 및 전체화면(Full-screen) 기반의 사용자 인터페이스를 지원하여야 하며, PC 윈도우 시스템 수준의 복잡하고 다양한 기능보다는 비교적 단순하면서 직관적인 사용자 인터페이스가 요구된다. 또한, 하드웨어 구성에서 멀티미디어 데이터 처리 부분은 높은 성능이 지

표 1. 오픈소스 그래픽 라이브러리 종류 및 주요 특징

구분	툴 명	설 명	라이선스	지원 현황	비 고
저수준 GUI 라이브러리	DirectFB	<ul style="list-style-type: none"> - 저수준의 경량 그래픽 라이브러리 - 하드웨어 그래픽 가속기능, 입력장치 이벤트 처리, 통합 윈도잉 시스템 지원 - 응용간의 프레임버퍼 공유 및 사운 출력 등을 지원하는 패키지 제공 - 고수준 그래픽 라이브러리의 저수준 계층으로 활용 - 멀티미디어 응용 및 단순 그래픽 응용 등에 적용 	<p>GPL 2.1</p>	<p>활발한 업데이트</p>	<ul style="list-style-type: none"> - 적용 사례가 증가 추세 - www.directfb.org
	SDL	<ul style="list-style-type: none"> - 원래 게임 개발용으로 설계, - 그래픽 디스플레이, 입력장치 이벤트 처리, 사운드, CD-ROM 오디오, 타이머 등을 지원 - X11, DirectFB, 프레임버퍼 위에서 동작 가능 - 일반적으로 작은 데스크탑 게임 등에 활용 	<p>GPL</p>	<p>활발한 유지 관리</p>	<ul style="list-style-type: none"> - 다양한 게임의 저수준 그래픽 라이브러리로 적용 - www.libsdl.org
	KDrive	<ul style="list-style-type: none"> - 임베디드 시스템용으로 개발된 X-server의 단순화 버전(기존의 Tiny-X로서 X.org server에 통합) - 충분한 X-server 기능 제공 -> X11 프로토콜을 완벽하게 지원함으로써 기존의 X11 응용 및 라이브러리 사용이 가능 - 고수준의 툴킷(Gtk+, Qt, 등)과 같이 사용 	<p>X11 License</p>	<p>활발한 개발 및 유지 관리</p>	<ul style="list-style-type: none"> - DirectFB가 경량 이면서 같은 수준의 API를 지원함으로써 DirectFB로 대체되고 있음. - www.x.org
	Nano-X	<ul style="list-style-type: none"> - 소형장치를 위한 경량 그래픽 윈도우시스템을 제공 - 2가지 형태의 API 제공하나(Win32 및 Xlib), 호환성은 부족 - client/server 모델로 동작, 윈도우 관리자를 지원하나 widget 라이브러리를 지원하지 않음 - Nano-X 기반의 몇몇 툴킷 라이브러리가 지원되고 있음(e.g tnw2000 etc.) 	<p>Mozilla Public License</p>	<p>2005년 최종 릴리즈</p>	<ul style="list-style-type: none"> - 현재 사용자 기반이 축소되고 있고, 추후 지속적인 유지 관리를 보장하지 못함 - www.microwindows.org
고수준 GUI 라이브러리	Gtk+	<ul style="list-style-type: none"> - 데스크탑 환경에서 가장 널리 사용되고 있는, 위젯 기반의 고수준 API를 지원하는 툴킷 - C API 지원하면서 C++를 포함한 다양한 언어에 대한 바인딩을 지원 - Xorg server 및 DirectFB 기반 위에서 동작 - 윈도잉 시스템을 지원하지 않음 -> 다중 윈도우를 지원하기 위해서는 경량의 윈도우 관리자 필요 	<p>GPL</p>	<p>활발한 개발 및 유지 관리</p>	<ul style="list-style-type: none"> - 다양한 위젯 기능 지원 등으로 실행 시에 많은 자원들을 요구 - GNOME - www.gtk.org
	QT	<ul style="list-style-type: none"> - 데스크탑 및 임베디드 환경에서 널리 사용되고 있는, 위젯 기반의 고수준 API를 지원하는 툴킷 - 윈도우 관리자 기능 포함 - Qt for Embedded Linux/ Qt Entended - C++ API 지원 - 프레임버퍼, X11, DirectFB 상에서 동작 가능 - 그래픽 툴킷 및 완전한 개발 프레임워크를 제공 	<p>GPL, Commercial License</p>	<p>활발한 개발 및 유지 관리, 문서 지원</p>	<ul style="list-style-type: none"> - 모바일폰 플랫폼을 제공, 많은 기능 제공으로 실행 시에 많은 자원 요구 - www.trolltech.com - doc.trolltech.com
	FLTK	<ul style="list-style-type: none"> - 경량화를 목표로 설계된 위젯-수준의 그래픽 툴킷 - 정적 링크 시에 적은 메모리 사용하도록 설계 - Xlib 위에서 동작하나 다른 저수준 그래픽 라이브러리에 포팅되어 지원 - C++ 기반의 표준 API를 지원 	<p>GPL</p>	<p>활발한 개발 및 유지 관리</p>	<ul style="list-style-type: none"> - 다양한 저수준 그래픽 라이브러리에 포팅이 진행 - www.fltk.org
	WxEmbedded	<ul style="list-style-type: none"> - 그래픽 호환성을 가진 응용 개발 위해 설계된 기존 툴킷에 대한 추상화 라이브러리 - 다양한 저수준 그래픽 라이브러리에서 동작 가능 - C++ API 지원 - 실행 시에 공유 라이브러리 및 정적 라이브러리에 대한 비교적 큰 메모리 용량을 요구 	<p>GPL</p>	<p>활발한 개발 및 유지 관리</p>	<ul style="list-style-type: none"> - WxEmbedded의 경우 기본적인 위젯만 지원 - www.wxwidgets.org
	LiTE	<ul style="list-style-type: none"> - DirectFB를 위해 설계된 툴박스 엔진 - 두 계층으로 구성: Lite-위젯 구현을 지원하는 저수준 계층, Leck-위젯 구현 계층 - 현재 단순한 위젯만 사용 가능 - 실행 시 메모리 요구량이 매우 적음 - 단순 GUI 구현에 적합 	<p>LPGL 2.1</p>	<p>개발 진행중 (0.8.1)</p>	<ul style="list-style-type: none"> - 제한된 사용자 계층 - www.directfb.org
	MiniGUI	<ul style="list-style-type: none"> - 실시간 임베디드 시스템을 위한 경량 GUI 시스템으로 설계 - 프레임버퍼를 포함한 다양한 저수준 라이브러리 위에서 동작 가능하며 Win32 호환 API 지원 - 다중 윈도우 및 메시징 메커니즘 지원 	<p>GPL, Commercial License</p>	<p>활발한 개발 및 유지 관리</p>	<ul style="list-style-type: none"> - GPL 버전은 기능이 제한적 - www.minigui.org

원되나 사용자 인터페이스 처리 부분의 성능은 상대적으로 낮아 빠른 GUI 처리를 위하여 경량성(light-weight)과 안정성을 가진 GUI 플랫폼이 요구된다[1, 2, 3, 4].

이에 본 논문은 멀티미디어 임베디드 시스템의 GUI 플랫폼 구현에 적합한 그래픽 라이브러리를 선정하기 위해 최근 임베디드 시스템 개발에 많이 채택되거나 넓은 사용자 기반을 가진 몇몇 오픈소스 그래픽 라이브러리들을 선택하여 성능을 평가하고 비교한다. 또한 선정된 라이브러리에 대한 개선점을 제안한다.

II. 오픈소스 그래픽 라이브러리

임베디드 리눅스 시스템에서 GUI 플랫폼 구현을 위해 사용되는 그래픽 라이브러리는 라이선스 형태에 따라 상용 그래픽 라이브러리와 오픈소스 그래픽 라이브러리로 나눌 수 있다[5, 6]. 상용 그래픽 라이브러리는 성능과 안정성이 어느 정도 검증이 되었을 뿐만 아니라 개발에 따른 기술 지원을 받을 수 있는 반면에 라이선스에 따른 비용이 발생한다. 오픈소스 라이브러리는 상용 라이브러리와 비슷한 수준의 성능을 지원하나 안정성을 보장하지 못하는 문제점이 있는 반면에 라이선스 비용이 요구되지 않으며 용량과 크기 면에서 다양한 종류가 있어 최적의 그래픽 라이브러리 선택이 가능한 장점이 있다.

본 논문은 멀티미디어 임베디드 시스템의 GUI 플랫폼 구현을 위해 오픈소스 그래픽 라이브러리만을 고려하며, 최근 임베디드 시스템 개발에 많이 적용되고 넓은 사용자 기반을 가진, 특히 활발한 업데이트 및 유지 관리가 지원되는 오픈소스 그래픽 라이브러리를 선정하여 각각의 주요 특징을 표 1에 기술하였다. 표 1에서는 오픈소스 라이브러리를 지원되는 API 특성에 따라 저수준과 고수준의 라이브러리로 분류하였으며, 저수준 라이브러리는 임베디드 시스템의 그래픽 입출력 장치에 대한 추상화 계층을 제공하면서 기본적인 그래픽 출력 기능과 입력 장치 이벤트 처리 등을 지원한다. 고수준 라이브러리는 저수준 라이브러리 기능을 포함하거나 이에 기반하여 응용 프로그램 구현에 용이한 추상화된 그래픽 출력 기능 및 위젯 등을 지원하는 툴킷 라이브러리를 의미한다.

III. 오픈소스 그래픽 라이브러리의 성능 평가

본 논문은 표 1에서 제시된 오픈소스 라이브러리 중에서 멀티미디어 임베디드 시스템의 GUI 플랫폼에 대한 요구사항, 라이브러리 유지 관리 지원 그리고 GUI 구현을 위한 자료 접근성 등을

고려하여 다음과 같은 4 가지의 그래픽 라이브러리 패키지를 선정하여 성능 평가를 진행하였다.

- (1) DirectFB + GTK+
- (2) DirectFB + LiTE
- (3) nano-X + TNW2000
- (4) Qt/Embedded-linux(Qtopia-Core)

그래픽 라이브러리의 성능 평가를 위해 GUI 구현에서 기본적으로 사용되면서 빈도가 높은 위젯 출력 기능과 이미지 출력 기능을 테스트하는 2개의 테스트 프로그램을 작성하여 성능을 측정하였다. 첫 번째 프로그램은 그림 1의 (a)과 같이 기본 위젯인 레이블, 푸시버튼, 체크버튼 그리고 입력박스 등을 출력하는 프로그램이며, 두 번째 프로그램은 그림 1의 (b)와 같이 JPEG 이미지를 단순히 디코딩하여 출력하는 프로그램으로 이미지 출력 성능을 평가한다. 그리고 테스트 프로그램을 표 2와 같은 실제 임베디드 하드웨어 플랫폼에서 실행하고 메모리 요구량과 실행 시간을 측정하여 라이브러리간의 성능을 비교하였다.

표 2. 성능 평가 실험 환경

H/W	CPU	Intel Xscale PXA255 400MHz
	Memory	128 MB
	Flash Memory	32 MB Intel Strata NOR Flash Memory
S/W	O.S	Embedded Linux Kernel 2.4.20
	Compiling Tool	Cross-Toolchain(gcc 3.2.1/glibc 2.2.5) - supports software emulator for floating point operation
	JPEG Codec S/W	IJG JPEG Library version 6b[15] + arm patch

그림 1은 각각의 그래픽 라이브러리에 대해 2개의 테스트 프로그램을 실행한 후에 나온 출력 결과 중에 일부만 보여준다. 위젯 테스트 프로그램 출력 결과를 통해 그래픽 라이브러리가 제공하는 위젯의 그래픽 특성을 파악할 수 있었으며, 이미지 출력 프로그램 결과는 거의 모두 유사한 결과를 보여주었다.

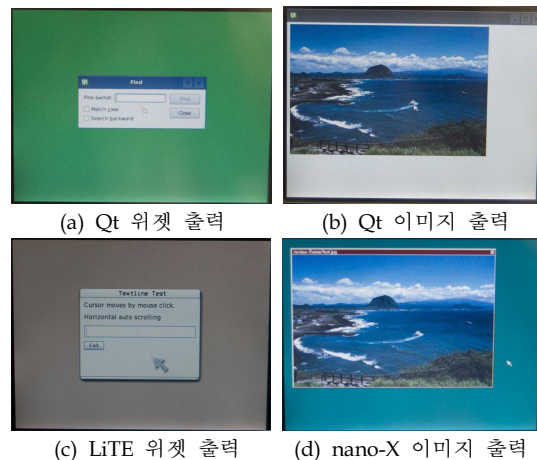


그림 1. 그래픽 라이브러리별 출력 결과

그림 2는 그래픽 라이브러리 별로 테스트 프로그램의 실행 시간을 비교한 것으로 GTK+와 Qt가 상대적으로 처리 오버헤드가 커서 실행이 지연됨을 알 수 있으며, LiTE와 nano-X의 실행 시간이 상당히 빠름을 알 수 있다.

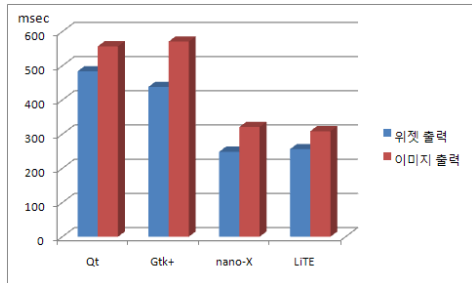


그림 2. 실행 시간 성능 비교

그림 3은 그래픽 라이브러리 별로 테스트 프로그램 실행에 요구되는 메모리 크기를 측정하여 비교한 것으로 LiTE가 상대적으로 매우 적은 메모리를 요구함을 알 수 있다.

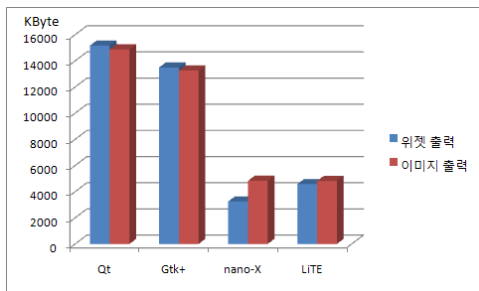


그림 3. 메모리 요구량 성능 비교

성능 평가에서 알 수 있듯이 DirectFB 기반의 Gtk+와 Qt/Embedded-linux 라이브러리는 다양한 위젯과 고수준의 그래픽 출력 기능을 지원하여 깔끔하고 매우 익숙한 사용자 인터페이스 구현이 가능하나 경량성과 빠른 처리에 대한 요구 조건을 만족하기에는 오버헤드가 훨씬 크다.

이에 반해 nano-X 기반의 tnw2000 툴킷 라이브러리는 적은 메모리 요구량과 빠른 처리로 경량성에 대한 요구를 만족시키고 익숙한 사용자 인터페이스 구현이 가능하나 업데이트가 진행되지 않아 사용자 기반이 적을 뿐만 아니라 접근 가능한 구현 자료가 부족한 문제점을 내포하고 있으며, OSD(Over Screen Display) 그래픽 출력에는 부적합하다.

DirectFB 기반 LiTE 라이브러리는 nano-X 라이브러리와 유사한 장점을 가지고 있으면서 DirectFB 사용자를 중심으로 사용자 기반이 확대되고 있고, 높은 경량성은 단순하면서 직관적인 사용자 인터페이스를 요구하는 멀티미디어 임베디드 시스템의 GUI 구현에 적합하다. 그러나 부족한 위젯 기능과 검증되지 않은 성능 등은 개선되어야 할 점이다.

V. 결 론

본 논문은 멀티미디어 임베디드 시스템을 위한 경량 GUI 플랫폼을 구현하기 위하여 오픈소스 그래픽 라이브러리 중에 사용자 기반이 넓고 성능이 검증된 라이브러리들을 선정하여 성능을 평가하였다. 실제 임베디드 하드웨어 플랫폼에서 위젯 출력 및 이미지 출력 기능에 대해 성능을 평가하여 비교한 결과, DirectFB 기반 LiTE 라이브러리가 다른 라이브러리와 비교하여 적은 메모리 사용과 빠른 실행 시간의 경량성을 지원하여 단순하면서 직관적인 사용자 인터페이스를 요구하는 멀티미디어 임베디드 시스템의 GUI 구현에 적합한 것으로 평가되었다. 그러나 부족한 위젯 기능과 검증되지 않은 성능 등의 문제점을 포함하고 있다.

향후에는 DirectFB 기반의 LiTE 라이브러리를 참조 모델로 하여 멀티미디어 임베디드 시스템의 GUI 플랫폼 구현에 적합한 경량의 그래픽 라이브러리를 구현할 예정이다.

참고문헌

- [1] 김지용, 광지영, 설동명, 안성호, 이은령, 이경희, 김두현, "임베디드 그래픽 사용자 인터페이스 기술 동향," <http://www.itfind.or.kr/WZIN/jugidong/1107/110701.htm>.
- [2] 설동명, 안성호, 광지영, 김지용, 이경희, "임베디드 리눅스 시스템용 경량 그래픽 라이브러리 생성 시스템," 한국정보처리학회 논문지, 제11권, 제 2호, 2004.
- [3] 윤기현, 김용희, 박희상, 이철훈, "실시간 운영체제를 위한 Graphic User Interface의 설계," 한국정보과학회 2002년도 가을학술발표논문집, 제29권, 제2호(I), 2002.
- [4] 정진경, "Embedded 시스템에서 GUI 구현을 위한 그래픽 성능 연구," 영남대학교 대학원 석사학위논문, 2005.
- [5] The GUI Toolkit, Framework Page, <http://www.atai.org/guitool>, 2007.
- [6] Embedded Linux Graphics Quick Reference Guide, <http://www.linuxfordevices.com/c/a/inux-For-Devices-Articles/Embedded-Linux-Graphics-Quick-Reference-Guide>, 2002.