

RTOS 기반의 소프트웨어 2D BitBLT 엔진의 설계

김 봉 주*, 홍 지 만*

A design of Software 2D BitBLT Engine based on RTOS

Bong-Joo Kim*, Jiman Hong*

요 약

본 논문은 소프트웨어 기반의 2D BitBLT 엔진을 pSOS 운영체제에서 구현하는 것을 제안하였으며, 제안된 BitBLT엔진을 환자감시장치에서 동작을 검증하였다. 본 논문에서 제안한 방법의 검증을 위해, 환자감시장치 보드를 기반으로 하는 별도의 프로토타입 PCB 보드를 제작하고, 동작을 검증하였다. 메인 보드는 ARM9 기반의 CPU로 설계하였으며, 하드웨어 기반의 BitBLT 모듈을 소프트웨어 기반의 모듈로 동작하면서 가중되는 CPU의 부하문제의 해결을 위해 200Mhz 프로세서 대신 400Mhz 프로세서로 변경하였다. 본 논문에서는 환자감시장치에서 GUI를 구현하는데 있어 그래픽 컨트롤러의 핵심요소 중의 하나인 2D BitBLT 모듈을 커널의 디바이스 드라이버로 구현하였다.

▶ Keywords : BitBLT, LCD 디바이스 드라이버, 래스터 연산, 생체 신호 감시 장치

Abstract

In this paper, we proposed the implementation of software-based 2D BitBLT engine on the pSOS operating system and the operation of the BitBLT engine on patient monitoring device was verified. To verify the proposed method on the patient monitoring device, we designed prototype PCB board, and verified the operation. We designed the motherboard by using ARM9-based CPU. Because hardware-based BitBLT module was replaced with software-based one, CPU load problem was weighted. To solve this problem, we changed 400Mhz processor instead of 200Mhz processor. We implemented 2D BitBLT kernel module as a device driver which is one of the key elements of a graphics controller GUI in patient monitoring device.

▶ Keywords : BitBLT, LCD device driver, Raster operation (ROP), biosignal monitoring device

•제1저자 : 김봉주 •교신저자 : 홍지만

•투고일 : 2014. 2. 10, 심사일 : 2014. 2. 20, 게재확정일 : 2014. 3. 20.

* 숭실대학교 컴퓨터학과(Department of Computer Science and Engineering, Soongsil University)

I. 서론

환자감시장치는 대표적인 생체 신호 감시 장치이며, 그림 1과 같이 환자의 ECG, SpO₂, NIBP 그리고 체온등과 같은 생체 신호를 측정하고 LCD또는 VGA 장치에 표시하거나 외부 시스템과 통신하는 전자 의료 기기이다[1]. 이러한 장치들은 이동성 보장을 위해 포터블 디바이스로 개발과 함께 10인치 이상의 LCD 화면을 적용하려는 연구들이 이루어졌다. 따라서 고성능 임베디드 프로세서와 그래픽 컨트롤러를 이용한 환자감시장치의 개발이 요구된다.

지금까지의 연구는 임베디드 시스템의 GUI구성을 위해 LCD 컨트롤러와 같은 장치를 CPU와 함께 사용하여 빠르게 그래픽 데이터를 처리 하였다[2, 3, 4]. 최근 소프트웨어를 이용한 자원 관리 기술의 발달로 하드웨어 주변 장치를 사용하지 않게 할 수 있는 방법이 주목 받고 있다[5]. 하지만, 전자 의료 기기에서 주로 사용되는 2D BitBLT의 소프트웨어 기반 엔진 기술에 대한 연구는 미비한 실정이다. 대부분의 연구가 하드웨어의 성능향상을 위한 방향으로 진행되고 있으며, 이러한 연구들과 본 연구의 차이점은 하드웨어의 성능향상에 따른 소프트웨어 자원의 효율적 활용을 위해 소프트웨어 기반의 그래픽 엔진 설계를 시도했다는 점이다.

기존 시스템의 경우, S1D13806F와 같은 하드웨어 기반의 전용 프로세서와 128KB 용량의 SRAM[15]을 비디오 메모리로 사용하여 구현되어 있어, 시대적인 변화에 따라 시스템을 사용하는 환경이 변화하더라도 알고리즘의 변경이 불가능하다는 단점이 있으며, 환자감시장치의 시장에서 원가가 상당히 민감한 부분을 차지하고 있는 것을 감안하면, 두 개의 소자의 사용은 상당한 약점으로 볼 수 있다. 반면 본 논문의 경우, 알고리즘을 소프트웨어로 구현하여, 필요시 알고리즘을 상황에 맞게 수정할 수 있으며, 하드웨어로 구현된 두 소자를 사용하지 않게 됨에 따라 원가도 절감되는 장점이 있다.

본 논문에서는 환자감시장치와 같은 생체 신호 감시 장치에서 주로 사용되는 LCD 제어기의 특화된 기능 중의 하나인 2D BitBLT 엔진을 소프트웨어 기반으로 설계하고 구현하였다. 또한, 제안하는 소프트웨어 기반 2D BitBLT 엔진의 동작 확인을 위해 프로토타입 환자 감시 장치에 pSOS 운영체제를 탑재하여 정상 동작함을 검증하였다.

논문의 구성은 다음과 같다. 2장은 환자감시장치와 관련된 연구에 대해 기술하고 3장에서는 하드웨어 기반의 2D BitBLT 엔진의 기능과 소프트웨어 기반의 2D 그래픽 엔진의 설계에 대해서 설명한다. 4장에서 제안하는 소프트웨어 기

반의 2D BitBLT 프로토타입의 구현 및 결과를 보여준다. 마지막으로 5장에서 논문의 결론을 내린다.

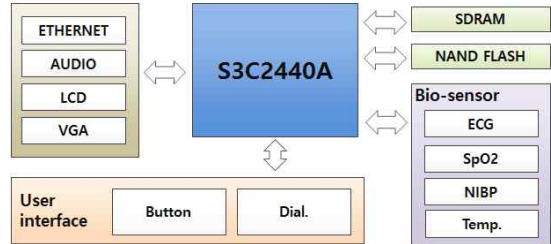


그림 1. 환자 모니터 아키텍처 블록 다이어그램
Fig. 1. Patient monitor architecture block diagram

II. 관련 연구

환자감시기의 구현을 위한 연구로는 스마트 폰과 같은 모바일 장치를 사용한 환경에서 TinyOS 기반의 의료용 메시지 구조를 설계하고, NesC를 사용하여 HBE-Ubi-ZigbeX에 포팅하였으며, 백프로퍼게이션 신경망에 기반한 상황인식을 하여 모바일 환경에서 심장병 환자를 위한 편재형 헬스 케어 시스템을 구현하였다[18, 19]. 대규모 데이터 병렬성을 활용할 수 있는 GPU를 이용하여, 많은 고성능 컴퓨팅 응용 분야에서 사용되고 있으며, 행렬의 고유벡터를 구하는 power method를 GPU에서 구현하여 커널 호출 회수를 줄이고, 스레드 배치를 최적화함과 동시에 수렴 여부 판단을 위한 연산을 최적화하였으며, 영상 생성의 고속화를 위해 GPU를 이용한 Gaussian Hole-Filling Algorithm의 병렬처리 구조를 제안하고 이를 이용한 DIBR Algorithm 생성과정을 제시하였다[20, 21].

그래픽 프로세서는 최근 고성능 프로세서를 이용하여 환자감시장치의 구현과 고해상도의 그래픽 인터페이스를 지원하는 장치의 개발에 관한 연구가 활발하다. 이들 연구의 대부분이 임베디드 시스템의 GUI구성을 위해 LCD 컨트롤러와 같은 장치를 CPU와 함께 사용하여 빠르게 그래픽 데이터를 처리하는데 중점을 두고 있다[2, 3, 4]. ECG 관리를 위한 표준화된 해결방안에 대한 연구[8]가 있으며, 고성능 프로세서를 이용하여 환자감시장치 구현에 관한 연구는 TFT-LCD를 갖춘 이동형 환자 감시장치에 대한 연구[6]와 고속 신호 처리 및 LCD 인터페이스를 위해 DSP와 ARM9코어가 내장된 듀얼 코어 CPU를 이용한 연구[7]가 있다.

이와 같이 최근에 고가의 고성능 프로세서들이 다양하게 출시되고 있으며, 고가의 FPGA를 이용한 IP 제품들이 설계

되어 출시되고 다양한 제품에 응용되고 있다. 하지만, 제품의 원가에 민감한 의료용 환자감시기 시장의 특수성을 감안해 볼 때, 고가의 고성능 프로세서를 이용하지 못하는 것이 현실이다. 본 연구에서 제안한 방법은 가격이 저렴한 ARM9 코어만을 이용하여 시스템을 구성하여 [6, 7]과 차별화하였으며, 하드웨어 기반의 2D 그래픽 엔진에 관한 연구가 주로 이루어진 시점에서 소프트웨어 기반의 연구를 진행한 것이[2, 3, 4, 20, 21]와의 차별성을 가지고 연구라고 할 수 있다.

III. 2D 그래픽 엔진을 위한 소프트웨어 설계

1. 2D 그래픽 엔진의 BitBLT 기능

BitBLT는 Bit BLock Transfer를 의미하여 컴퓨터 그래픽스 기술로 메인 메모리에서 디스플레이 메모리에 비트 블록을 이동하는 기능이다. 이것은 화면에서 움직이는 객체(애니메이션, 스크롤)를 고속으로 디스플레이 할 수 있게 해준다. 소프트웨어 BitBLTs는 non-graphics 시스템에서 사용되기도 하지만 하드웨어 BitBLT가 가장 빠른 속도를 제공하여 사용에 제약이 있었다.

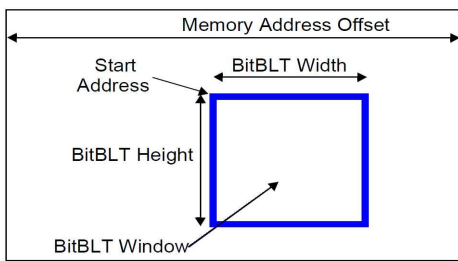


그림 2. BitBLT 관련 용어들
Fig. 2. BitBLT Terms

그림 2는 BitBLT에서 함수에서 사용하는 용어에 대해 정의하고 있으며, 세부 내용은 다음과 같다. Memory Address Offset은 화면의 폭을 의미하며, 단위는 16-bit word이다. Start Address는 BitBLT 윈도우의 좌측 상단을 의미하며, 단위는 Byte이다. BitBLT Width는 BitBLT의 폭을 의미하며, BitBLT Height는 BitBLT의 높이를 의미하며, 이들의 단위는 Byte이다. BitBLT Window는 BitBLT 연산이 수행되는 비디오 메모리 영역을 의미한다[17].

하드웨어 기반의 그래픽 제어기에서 제공하는 2D

BitBLT 엔진은 아래의 BitBLT 함수들을 제공한다[9].

- o Write BitBLT with ROP
- o Color Expansion
- o Read BitBLT
- o Move BitBLT in positive direction with ROP
- o Move BitBLT in negative direction with ROP
- o Transparent Write BitBLT
- o Transparent Move BitBLT in positive direction
- o Pattern Fill with ROP
- o Pattern Fill with Transparency
- o Color Expansion with Transparency
- o Move BitBLT with Color Expansion
- o Move BitBLT with Color Expansion and Transparency
- o Solid Fill

각각의 제공된 BitBLT 함수들은 다음의 기본 BitBLT 함수와 데이터 함수의 조합으로 구성되었으며, 기본 BitBLT 함수의 세부 기능은 다음과 같다. Read BitBLT 함수는 비디오 버퍼에서 CPU로 픽셀데이터를 이동한다. 즉, Read BitBLT 함수를 이용하여 호스트 CPU는 비디오 메모리에서 데이터를 읽어낼 수 있다. Move BitBLT 함수는 비디오 버퍼의 한 곳에서 다른 곳으로 픽셀데이터를 이동한다. Pattern Fill BitBLT 함수는 비디오 메모리의 원하는 영역에 일정한 패턴으로 채우는 기능이다. Solid Fill BitBLT 함수는 메모리의 특정 공간에 한 가지 색으로 채우는 기능이다.

데이터 함수는 두 가지가 있으며, 그 기능은 다음과 같다. ROP 함수는 픽셀 데이터와의 부울 연산을 수행한다. 제공하는 부울 연산은 AND, OR, XOR, NAND, NOR 등이 있다. 그림3은 ROP 함수에서 XOR 연산의 수행을 도식화 한 것이다. Transparency 함수는 투명한 색으로 지정한 색상과 다른 픽셀 데이터를 기록한다.

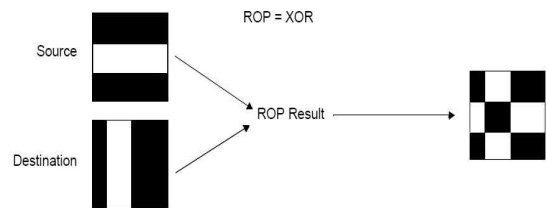


그림 3. ROP 예제(XOR)
Fig. 3. ROP Example(XOR)

2. 소프트웨어 기반 2D 그래픽 엔진 설계

그림 4는 GUI를 위한 그래픽 제어 함수와 명령 세트에 대한 계층 구조와 새롭게 정의한 소프트웨어 2D Graphic 엔진의 BitBLT 함수에 대한 설명이다. 이것은 디바이스 독립적인 어플리케이션 작성을 위한 기능을 제공하는 HAL (Hardware Abstraction Layer)(10)와 유사한 기능을 한다. 소프트웨어 기반 2D 엔진 계층에서는 데이터 연산에 대한 정의를 하였으며, 이것은 상위 계층인 high level device driver로 통합하였다. High level device driver의 기능에 따라서 API function group으로 통합되고, 최종적으로 어플리케이션 인터페이스를 위한 API library로 통합하였다.

소프트웨어 기반 2D 엔진 계층에서 2D 그래픽 데이터 업데이트를 위한 연산은 두 이미지 source(S)와 destination(D)을 가지고 처리한다. 그래픽 데이터를 직접 업데이트 하는 기능(Direct)과 래스터 연산 [ROP]에 따른 비트 연산이 필요한 명령에 대해서는 분리하여 계층이 재정의 되어있다. ROP는 기본적으로 부울 공식이며 소스와 대상을 덮어 쓰거나 AND, OR, XOR, 그리고 NOT 연산을 비트 단위로 수행한다(11, 12). 이것은 그림 4의 소프트웨어 기반 2D 엔진 계층 중 하위 단계로 표시하였으며 Boolean operation으로 표시하였다.

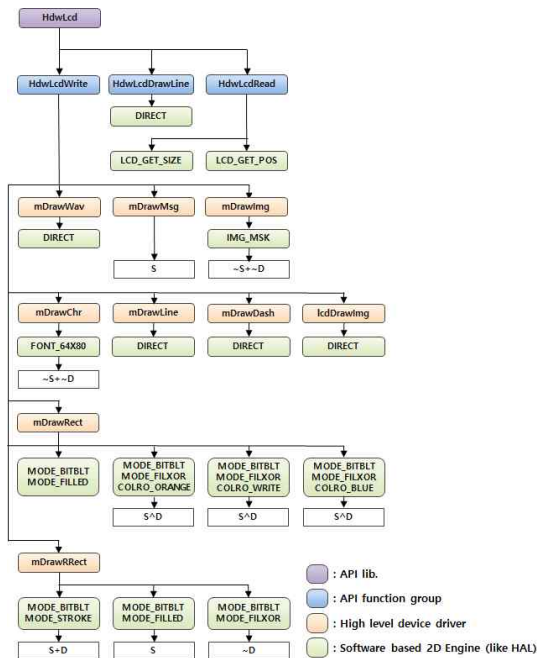


그림 4. 소프트웨어 2D 그래픽 엔진의 BitBLT 함수 트리
Fig. 4. Function Tree for Software 2D BitBLT Module

IV. 구현 및 결과

제안된 소프트웨어 기반의 2D BitBLT 모듈의 기능 및 성능을 검증하기 위해서 프로토타입 환자감시장치의 하드웨어에 탑재하였다. 프로토타입 환자감시장치의 하드웨어는 최대 400MHz로 동작하는 ARM9T 코어 기반의 S3C2440A를 이용하여 설계하였다. 제안된 시스템의 참조 보드는 표 1에서와 같은 하드웨어 기반의 2D BitBLT 엔진이 내장된 S1D13806F를 탑재한 ARM9T 코어 기반의 S3C2410으로 구성되어 있다. 2D BitBLT 엔진을 소프트웨어로 구현하면, 시스템 소프트웨어의 확장성을 고려하여 2MB NOR FLASH 메모리를 가격 대비 용량이 큰 128MB NAND FLASH 메모리로 변경하였다. 이에 따른 부트로드의 설계 또한 변경된 하드웨어에 맞추어 다시 설계하였다. SDRAM은 구현된 소프트웨어 기반의 2D BitBLT 모듈의 실행 코드 및 메모리 버퍼 데이터를 저장하는 기능을 추가하여 용량이 16MByte에서 64MByte로 변경하고, 하드웨어 기반의 BitBLT 제어기가 그래픽 데이터를 저장하는 데 사용하는 SRAM은 제거하였다.

제작된 메인 보드의 성능을 검증하기 위해서 그래픽 엔진이 적용된 하드웨어를 제작하고, 입출력 인터페이스 (IO) 및 LCD 패널과 연결하고 환자감시장치의 기본 기능인 그래프 출력 각종 폰트 출력과 GUI 출력을 확인하였다. 환자감시기의 기능별 동작 확인을 위해서 기존 시스템과 제안된 시스템을 1:1로 동작을 하여 기능별로 이상 유무를 확인하였다.

표 1. 시스템 환경
Table 1. System Environment

ITEMS	Ref.	Proposed
CPU	ARM9(S3C2410)	ARM9(S3C2440A)
FLASH MEMORY	2MB (NOR FLASH)	128MB (NAND FLASH)
SDRAM	16MB	64MB
SRAM	128KB	-
LCD CONTROLLER	SED13806F (hardware based 2D BitBLT engine)	Embedded LCD controller (without 2D BitBLT engine)
OS	pSOS	pSOS

1. ARM920T 기반 32-bit RISC 마이크로프로세서 S3C2440 32 비트 RISC 마이크로 프로세서는 ARM920T 코어를 사용하고 마이크로 컨트롤러 버스 아키텍처 (AMBA)[14]로 알려진 새로운 버스 아키텍처를 채택하였다. LCD-전용 DMA와 LCD 컨트롤러 (최대 4K 컬러 STN 및 256K 컬러 TFT)를 내장한 것이 특징이다. LCD 컨트롤러는 외부 LCD 드라이버에 시스템 메모리에 있는 비디오 메모리에서 LCD 이미지 데이터를 전송하는 회로로 구성되어 있다. 이들은 수평 및 수직 픽셀, 데이터 인터페이스를 위한 데이터 라인 폭, 인터페이스 타이밍과, 업데이트 속도의 수에 관한 화면에 서로 다른 요구 사항을 지원하기 위해 프로그래밍이 가능하도록 되어 있다.[13] 본 논문에서는 2D 그래픽 엔진의 구현을 위해서 직접 프로그래밍을 진행하였다.

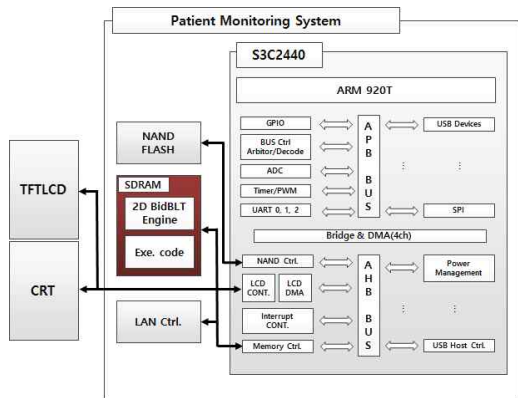


그림 5. 소프트웨어 기반 2D BitBLT 엔진을 이용하여 제안한 환자감시장치의 하드웨어 블록 다이어그램
Fig. 5. Hardware block diagram of proposed patient monitoring system using software based 2D BitBLT Engine

2. 환자감시장치의 기능적 요소

본 논문에서 구현된 환자감시장치는 입력(ECG, SPO2, NIBP, 체온 신호, 버튼, 다이얼)과 출력(모니터, 스피커) 그리고 통신(이더넷) 기능을 갖는다. 10.4 인치 컬러 고해상도 (800x600 이상) TFT 스크린 [16]과 VGA 외부 모니터는 모든 환자 모니터링 상태를 GUI로 표시한다.

3. 소프트웨어 기반의 그래픽 엔진의 구현

소프트웨어 기반의 그래픽 엔진은 운영체제의 디바이스 드라이버 형태로 구현 하고 프레임의 동기 및 타이밍은 내장된 하드웨어 장치의 기능을 사용하였다. 구체적으로는 그림 6과

같이 S3C2440에 내장된 LCD 컨트롤러 블록의 VIDEO MUX에서 발생하는 디스플레이 장치의 프레임 동기 신호를 사용하고, 프레임 버퍼 업데이트 속도를 보장하기 위해 최적화한 2D BitBLT 엔진을 구현하고 SDRAM 내부에 실행 코드 영역과 비디오 디스플레이 버퍼 영역을 분리하였다.

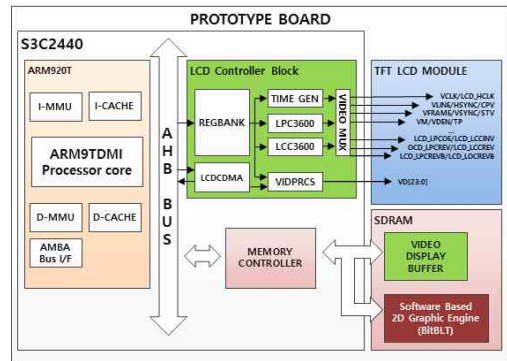


그림 6. 2D 그래픽 엔진, 비디오 디스플레이 버퍼, TFT-LCD간의 그래픽 데이터 흐름
Fig. 6. Graphic data flow among 2D Graphic engine, video display buffer and TFT-LCD

4. 구현 결과

제작된 메인 보드의 성능을 검증하기 위해서 NOR 플래시 메모리와 그래픽 엔진이 적용된 하드웨어를 제작하고, 그림 7과 같이 입출력 인터페이스 (IO) 및 LCD 패널과 연결하고 환자감시장치의 기본 기능인 그래프 출력 각종 폰트 출력과 GUI 출력을 확인하였다. 환자감시기의 기능별 동작 확인을 위해서 기존 시스템과 제안된 시스템을 1:1로 동작을 하여 기능별로 이상 유무를 확인하였다.

표 2에서는 기존 시스템과의 성능 비교를 위한 수치를 나타내고 있으며, 제안된 시스템도 기존 시스템과 같은 ARM920T Core를 사용하고 있다. 기존 시스템에서 하드웨어 기반의 2D BitBLT 엔진을 사용하면서 AHB 버스의 클럭이 100Mhz를 사용하였으나, 2D BitBLT 엔진을 소프트웨어로 구현하게 됨에 따라 약 30% 정도의 성능 향상이 필요함을 다양한 실험을 통해 확인 할 수 있었다. 특히, 30%보다 낮은 성능에서 구현한 경우, Write BitBLT with ROP를 사용하는 전체 화면을 지우는 기능에서 제한된 시간에 화면을 모두 갱신하지 못하는 등의 부작용이 있어 안정적인 구현을 위해 S3C2440의 Core 클럭을 399.65Mhz로 설정하였다.

표 2. 기존 시스템과의 성능 비교
Table 2 System Performance

ITEMS	Ref.	Proposed
Core	ARM920T	ARM920T
Core Clock	200Mhz	399.65hz
AHB Clock	100Mhz	133.33Mhz
LCD 제어기	SED13806F (하드웨어2D BitBLT)	내장 LCD 제어기 (소프트웨어2D BitBLT)
Video 버퍼	128KB SRAM	Code Memory와 공유

참조 보드에서 하드웨어 기반의 2D 그래픽 엔진을 사용하여 구현된 기존의 고해상도 LCD 제품의 기능과 비교하여 모든 기능이 구동함을 확인하였다. 기능을 테스트 하는 과정에서 몇 가지 예상치 못한 오류들이 있었다. 대표적인 예로 일정 시간동안 화면이 갱신되지 못하는 등의 오류들이며, 이러한 오류의 주요한 원인은 다양한 분석결과 타이밍과 버스의 성능으로 판명되었다. 이들의 해결을 위해 AHB 버스의 클럭 향상 및 Core 클럭의 향상으로 해결하였다.



그림 7. LCD GUI가 출력된 전면 패널
Fig. 7. Front panel with LCD GUI

V. 결론

환자감시장치와 같이 가격에 민감한 제품의 제작과 업그레이드에 있어서 코드의 재사용과 하드웨어로 구현된 특수 기능을 소프트웨어로 구현하여 가격 경쟁력이 높은 제품의 개발 방법이 요구되었다.

이에 따라서 본 논문에서는 2D 그래픽 엔진의 특수 기능인 BitBLT 기능을 소프트웨어로 구현하고 LCD controller가 내장된 범용 프로세서인 ARM920T 코어 기반의 32-bit

RISC microprocessor를 이용하여 고가의 하드웨어 기반의 2D 그래픽 엔진을 구현 하는 방법을 제안하였다. 제안된 방법에서는 기존에 사용된 어플리케이션의 호환성을 보장하기 위해 소프트웨어 그래픽 엔진 인터페이스와 코어를 설계하고 라이브러리로 제작하였다. 하드웨어 호환성 검증을 위해 메인 보드를 제작하였고 800x600 해상도를 갖는 TFT-LCD 모니터와 연결하여 동작을 확인하였다.

제안된 2D 그래픽 엔진 라이브러리와 환자감시장치의 성능을 TFT-LCD에 표시된 잔상, 색 번짐 현상이 없음을 확인하여 제안된 방법의 효율성을 검증하였다. 제안된 방법은 향후 그래픽 해상도나 디스플레이 컨트롤러 장치의 수급 환경 변화에 따른 제품 개발에 소프트웨어 그래픽 엔진을 사용하여 어플리케이션 코드 재활용이 가능하고 하드웨어 검증 시간을 단축하여 제품 개발 기간 단축에 따른 비용 절감 효과가 기대된다.

향후 고해상도 디스플레이 장치에 적용 가능한 시스템 개발 방법을 연구하고 다양한 개방형 운영체제(예: linux) 환경에 적용이 가능하도록 하여 코드 검증에 관한 연구가 추가로 필요하다고 판단된다.

참고문헌

- [1] Weifeng Liu, Zhiling Dai, Xinfen Hu, Xiaoying Tang, "Design of User Interface in Portable Health Monitoring System," ICBBE 2009. 3rd International Conference, Bioinformatics and Biomedical Engineering, pp.01-03, 2009.
- [2] Chein-Liang Chen, Chia-Hsing Lin, Hwan-Rei Lee, Chein-Wei Jen, "A Raster Engine for Computer Graphics and Image Composition," Circuits and Systems, IEEE Asia-Pacific Conference, APCCAS, pp.103-108. 1994.
- [3] Maoteng Xing, Zhanhe Wang, "The Low Power Design of 2D Graphic Engine Based on the AMBA Bus," 2010 10th IEEE International Conference, Solid-State and Integrated Circuit Technology (ICSICT), pp.647-649, 2010.
- [4] Pinto, A., Harish, Y.S., "Maximizing Efficiency in Reference Model Based Verification of 2D Graphics Engine," Emerging Trends in Engineering and Technology (ICETET), 2011 4th International Conference, pp.290-295, 2011.

- [5] Min Jin, Xiang Zhou, Peng Duan, Zefeng Tian, Jihui Zhou, "The Design and Implementation of Embedded Configuration Software Based on Embedded-Linux," Computer Science and Software Engineering, 2008 International Conference , pp.98-101, 2008.
- [6] Xuwen Li, Qiang Wu, Shuicai Wu, "Hardware Design of ADSP-BF548 Based Portable Patient Monitor," Electrical and Control Engineering (ICECESystem Based on OMAP," CSE '09. International Conference, pp.1002-1006, 2009.
- [7] Jing Liang, Yinqin Wu, "Wireless ECG Monitoring System Based on OMAP," CSE '09. International Conference, pp.1002-1006, 2009.
- [8] Bitblt, <http://encyclopedia2.thefreedictionary.com/bitblt>
- [9] S1D13513 Display Controller, <http://vdc.epson.com/>
- [10] Hardware abstraction, http://en.wikipedia.org/wiki/Hardware_abstraction
- [11] Sanchez, Julio, Maria P. Canton., "Software solutions for engineers and scientists", CRC Press, pp.690, 2007.
- [12] Bitblt, <http://en.wikipedia.org/wiki/Bitblt>
- [13] S3C2440A 32-BIT CMOS MICROCONTROLLER USER'S MANUAL, <http://www.samsung.com/global/business/semiconductor/>
- [14] ARM920T Product Overview, <http://www.arm.com/>
- [15] 128Kx8 bit Super Low Power and Low Voltage CMOS Static RAM, <http://www.samsung.com/global/business/semiconductor/>
- [16] OMAP5912 Applications Processor, SPRS231E, <http://www.ti.com>, 2011 International Conference, pp.5948-5951, 2011.
- [17] S1D13719 Display Controller, <http://vdc.epson.com/>
- [18] Kyoung-Mok Kim, "Implementation of Patient Monitoring System based on Mobile Healthcare", Journal of The Korea Society of Computer and Information Vol. 17 No. 12, 2012.12
- [19] Jeong-Won Kim, "Implementation of a pervasive health care system for cardiac patient on mobile environment", Journal of The Korea Society of Computer and Information Vol. 13 No. 5, 2008.09
- [20] Junghwan Kim, Jinsoo Kim, "Implementation of Efficient Power Method on CUDA GPU", Journal of The Korea Society of Computer and Information Vol. 16 No. 2, 2011.02
- [21] Jun-ho Park, Tack-Don Han, "Accelerating Gaussian Hole-Filling Algorithm using GPU", Journal of The Korea Society of Computer and Information Vol. 20 No. 2, 2012.07

저 자 소 개



김 봉 주
 1992: 동의대학교 화학과 이학사
 1995: 경성대학교
 응용통계학과 이학석사
 2009: 숭실대학교
 컴퓨터학 박사과정 수료
 현 재: 숭실대학교
 컴퓨터학과 박사과정
 관심분야: 시스템소프트웨어,
 임베디드시스템
 Email : ghostpl@paran.com



홍 지 만
 1994: 고려대학교
 컴퓨터공학과 공학사
 1997: 서울대학교
 컴퓨터공학과 공학석사
 2003: 서울대학교
 컴퓨터공학과 공학박사
 현 재: 숭실대학교 컴퓨터학과 교수
 관심분야: 시스템소프트웨어,
 임베디드시스템
 Email : jiman@ssu.ac.kr