

체외형 자동 제세동기(AED)와 심전도계(ECG)를 위한 통신 프로토콜의 설계 및 사용자 인터페이스 구현

Communication Protocol Design and U/I Implementation for Automated External Defibrillator and ECG System

김 남 훈* 권 희 중* 권 호 열** 나 학 록***
Kim, Nam-Hun Kwon, Hee-Joong Kwon, Ho-Yeol Na, Hak-Lok

Abstract

In this paper, we studied Automated External Defibrillator and Electro-Cardiogram System. Proposed System, CU-EX1, is a PC-based AED/ECG system that manages the data acquired by Paramedic CU-ER1. The system controls the transfer of data from the Paramedic CU-ER1 and the storage of the data in a local host computer. Patient data could be entered as soon as the rescue data is stored in a local computer. The system also displays the data on a local computer and facilitates review of rescue events.

키워드 : 체외형 자동 제세동기, 프로토콜 설계, ECG 디스플레이 소프트웨어
Keywords : AED, Protocol Design, ECG Display Software

1. 서론

오늘날 의료기술의 발달로 인한 사망률 감소하고 있지만 관상동맥 질환은 여전히 사망의 주요 원인으로 남아 있을 뿐만 아니라 식생활의 변화와 환경여건의 악화로 인해 심장에 관련된 질환으로 사망하는 인구는 오히려 증가하고 있는 추세이다. 국내에서는 응급의료에 관한 법률이 1994년에 제정되었으며, 심장박동이 정지되었을 때 이를 정상으로 회복시키기 위한 제세동기와 같은 응급의료 시스템을 갖춘 응급의료 전문팀을 가동하는 노력이 이루어지고 있다[1].

제세동기(Defibrillator)란 심실 세동과 맥박이 없는 심실 빈맥 현상을 제거하기 위해 흉곽을 개방하여 직접 심근에 전류를 인가하거나 비간접적으로 흉곽외부에 전류를 인가하여 심근에 전기적

충격을 가하는 의료장비이다. 1899년 Prevost 와 Batelli 에 의해 처음으로 전기충격을 이용한 제세동 개념이 소개된 이래 현재 여러 형태의 제세동기가 개발되었다 특히 진단장비의 발전과 함께 1979년에 소개된 자동 제세동기는 심박리듬을 인식하는 데 개인별 훈련이 필요없는 제세동 기술이 가능하며 환자의 소생 확률을 개선할 수 있음을 보여주었다. 최근에는 제세동기에 필요한 ECG 감시시스템을 고속 디지털 신호처리 프로세서를 이용하여 구현한 연구[2] 등으로 제세동 기능의 고도화와 동작의 안정도 및 신뢰도의 개선이 이루어지고 있다.

심전도는 개개의 환자에 따라 또는 측정부위 및 측정 시간에 따라 다르게 나타난다. 심근경색(myocardial infarction)의 경우 실제 심근경색(acute myocardial infarction) 상황에서 첫 심전도에서 전형적인 소견이 나오는 경우는 50% 정도이며 정상 심전도 소견도 10% 정도 된다. 그러나 계속해서 검사한 심전도의 진단정확도는 95%에 육박한다. 그러므로 심전도를 이용하여 심근 경색(myocardial infarction) 진단을 위해서 가장 중요

* 강원대학교 전기전자정보통신공학부 학사

** 강원대학교 전기전자정보통신공학부 교수, 공학박사

*** (주)씨유메디칼시스템

한 것은 시간에 따른 심전도의 변화를 보는 것이다[3].

본 연구에서는 체외형 자동 제세동기와 심전도계를 위한 통신 프로토콜을 설계하고 사용자 인터페이스를 구현하였다. 제안된 시스템, CU-EX1,은 개인용 컴퓨터 기반의 소프트웨어 시스템으로서 외부 제세동기 CU-ER1 으로부터 획득한 환자의 병리자료에 대한 전송과 저장을 통제하고 이들 자료의 검토를 위하여 그래픽 디스플레이 기능을 제공한다.

본 논문의 구성은 제 1 장의 서론에 이어, 2 장의 본론에서는 제안된 시스템의 기본구조, 통신 프로토콜 및 사용자 인터페이스의 설계, 제안된 시스템의 구현된 결과를 보였으며, 끝으로 3 장의 결론에서는 연구 결과의 요약 및 차후 연구과제에 대하여 제시하였다.

2. 본론

본 연구의 대상은 체외형 자동 제세동기 (AED: Automated External Defibrillator)의 통신 및 그래픽 디스플레이 모듈로서, 사용자들의 요구 사항일 뿐만 아니라, AED 를 개발하는 회사라면 없어서는 안 될 PC쪽의 필수 서비스 사항이며, CU-EX1 이라는 프로젝트 이름으로 개발되었다.

CU-EX1 의 주된 목적은 소프트웨어의 다양한 기능 보다는 사용자(의사, 소방대원 등) 입장에서 생각하여, 심전도 (ECG: Electrocardiogram) 를 PC상에서 보다 쉽고 편리하게 관리 및 분석 할 수 있도록 하는 것이다. 아무리 편리하고 다양한 기능이 제공된다고 해도 그 정보 표현에 사용자들의 눈에 맞지 않는다면 쓸 모가 없기 때문이다. 현재 CU-EX1 에는 심전도 디스플레이, CU-ER1 과의 통신, 이벤트 목록 디스플레이, 환자 정보 관리 및 프린트 기능 등이 구현되어 있다.

이 연구 개발의 최종 단계는 심전도 데이터베이스 전산화이다. 추후에 개발될 CU-ER2, CU-PH1 도 CU-ER1과 CU-EX1 사이에서 개발된 통신 프로토콜을 이용하여 심전도 데이터를 다운로드할 수가 있다. 이러한 점들을 고려했을 때, 밀지 않아 개발될 CU-EX2 에서 이 모든 심전도 데이터를 웹상에서 관리하여 심전도 데이터베이스 전산화의 기초를 이룰 수가 있는 것이다. 이때 사용자들은 웹상에서 디스플레이 되는 심전도를 볼 수가 있으며, 자신의 심전도 데이터에 관한 많은 정보를 쉽게 얻을 수 있다.

2.1 제안된 시스템의 기본 구조

본 연구에서 제안된 시스템은 그림 1 과 같은 기본 구조를 갖고 있다. 제안된 시스템 설계는 크

게 통신 모듈과 사용자 인터페이스모듈로 나누어진다. 통신 모듈은 휴대용 제세동기인 CU-ER1 등에서 ECG 관련 데이터를 다운로드하여 미리 정해진 양식의 파일로 생성한다. 이렇게 생성된 파일을 사용자 인터페이스부에서 열어 여러 가지 기능을

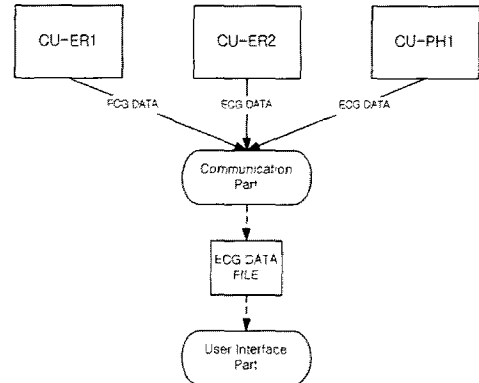


그림 1 ECG 데이터의 흐름

제공하여 사용자로 하여금 편리하게 원하는 정보에 접근할 수 있도록 한다.

2.2 통신 프로토콜 설계

2.2.1 직렬 통신

최대 115 kbps 를 지원하는 통신 모듈을 위한 직렬 인터페이스의 사양은 다음과 같다.

- 보오(Baud)율 : 9600, 19200, 38400, 57600, 115200 [bps]
- 데이터비트 : 8 비트
- 정지 비트 : 1 비트
- 패리티비트 : 없음
- 핸드셰이크 : Xon, Xoff (소프트웨어제어 방식)

컴퓨터와 휴대용 제세동기 사이의 통신 설정을 위한 제어신호 교환 프로토콜은 그림 2 와 같다.

PC와 장비는 리셋(Reset) 되기 전까지는 9600 bps 의 기본 보오(Baud)율을 사용하여 장비 정보와 통신설정을 수행하게 된다. 장비정보가 확인되지 않을 경우 통신은 중단되지만, 장비정보가 확인되면 보오율 설정정보의 교환이 이루어진다. 리셋이 이루어지면 보오율 설정 상태로 다시 들어간다. 리셋 후 데이터 전송이 이루어지는데 본격적인 데이터 전송이 이루어지기 전에 장비의 일련번호 확인 절차를 한번 더 거치게 된다. 일련번호가 확인되지 않으면 통신포트가 폐쇄된다.

체외형 자동 계세동기(AED)와 심전도계(ECG)를 위한 통신 프로토콜의 설계 및 사용자 인터페이스 구현

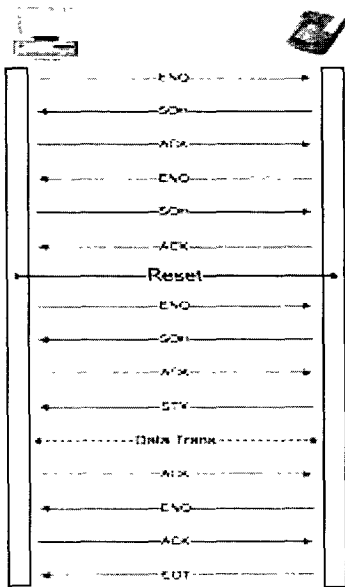


그림 2 통신 프로토콜의 흐름도

확인 된 일련번호는 CRC 방식으로 암호화하여 ECG 데이터와 함께 파일에 기록된다 암호화 된 일련번호는 ECG 데이터를 디스플레이 할 때 사용되는 키 값이 된다.

데이터 블록 및 메시지에 대한 장비와 PC간의 전송은 3초 간격으로 10번 반복적으로 이루어진다. 10번 전송 후에도 응답이 없으면 EOT 신호를 보내고 통신포트를 폐쇄한다.

2.1.2 데이터 블록 포맷

통신선로를 통하여 전송되는 데이터 블록의 크기는 모두 261 바이트이며, 데이터블록의 포맷은 표 1 과 같다.

표 1 데이터 블록 포맷 (단위: 바이트)

Block Command	Block Number	Check Sum	Node Data	CRC HI	CRC LOW
:	1	1	256	1	1

Block Command는 제어신호 바이트로 사용되고 Block Number는 이름이 의미하는 대로 Block Number로 사용된다. Block Number는 1 바이트 형식을 사용하므로 255 까지 증가한 후에는 256으로 넘어가지 않고 0번으로 된다 Node Data는 ECG 데이터를 전송하는 바이트로 사용된다.

CRC는 헤더를 포함한 모든 데이터를 CRC 비

트를 생성하여 데이터의 손상 여부를 확인하는데 사용된다.

2.1.3 제어 코드

각 제어 코드는 1 바이트를 사용하며, ACK, ENQ, NAK, EOT 는 DC2 와 조합하여 2 바이트로 전송되며 SOH, STX, ETX 는 261바이트 형태로 전송된다, SOH는 보오(Baud)율[bps] 설정, 장비정보, PC정보 전송 시에 사용되고 STX, ETX는 ECG Data 전송 시에 사용된다.

표 2 제어 코드

문자	10진값	16진값	설명
ENQ	5	0x05	Enquiry
SOH	1	0x01	Start of Header
ACK	6	0x06	Acknowledgment
NAK	21	0x15	Negative ACK
DC2	18	0x12	Device Control 2
STX	2	0x02	Start of Text
ETX	3	0x03	End of Text
EOT	4	0x04	End of Transmission

2.1.4 메시지 포맷

메시지 포맷은 제어신호 SOH가 보내질 때 블록 데이터의 Node Data에 적재되어 전송된다 메시지의 포맷은 다음과 같다.

제조회사의 식별을 위한 Company ID 는 CU 로 하고 메시지 블록 크기는 8 바이트로 하였다 여기서 Device ID는 색인값(Index)으로 사용된다.

표 3 메시지 포맷 (단위: 바이트)

Flag Index	Company ID	Device ID / Baud Rate Index	System Version
1	2	1	4

PC와 장비가 정보를 교환하기 위한 제어 신호 및 플래그는 표 4 와 같다.

표 4 제어 플래그

색인값	설명
0	ID
1	보오율
2	이벤트 데이터
3	ECG 데이터
4	음성 데이터

색인값 0은 PC와 장비가 서로 정보(장비정보, EX1정보) 교환을 할 때 사용되며, 색인값 1은 보오울 설정 시 사용하는 제어 신호이며, 색인값 2, 3, 4는 이벤트 데이터, ECG 데이터, 음성 데이터를 의미 한다. 즉 Node Data에 각 색인값에 대한 정보가 들어 있다는 제어코드로 사용된다.

장비 분류를 식별하는 색인값은 표 5 와 같다.

표 5 장치 ID

장치 ID	장치명
0	ER1
1	ER2
2	PH1
4	EX1

보오울 설정을 위해 사용되는 색인값은 표 6 과 같다.

표 6 보오울 [bps]

색인값	설명
0	9,600
1	19,200
2	38,400
3	57,600
4	115,200

표 7 ENQ, ACK, NAK, EOT 포맷 (단위: 바이트)

Block Command.	Device Control 2
1	1

2.3 사용자 인터페이스 구현

2.3.1 스크롤바 구현 방식

스크롤바의 구현에 난해한 문제점은 심전도 디스플레이에 접근하는 방법이 여러 가지가 있지만, 스크롤바는 그러한 접근을 모두 고려하여 동기화가 이루어져야 한다는 것이다. 스크롤바에 접근 방식은 크게 3가지로 나눌 수가 있다.

첫 번째로 각각 버튼에 의한 접근, 이 접근 방식은 기존의 파일 포인터의 수치를 일정량 변경하여 접근하는 방식이고, 두 번째로 이벤트 트리에 의한 접근 방식으로 이 방식은 파일 포인터를 직접 모두 전달하여 해당 위치의 심전도를 디스플레이하는 방식이다.

마지막으로 스크롤바 방식은 위에 말한 두 방식을 모두 합한 방식으로 생각할 수가 있지만, 기본 동작 원리는 첫 번째 방식에서 그 파일 포인터의 변화량이 빠르게 처리된다고 볼 수 있다. 그래서

이러한 모든 방식에 만족하는 알고리즘으로 CU-EX1에서는 공유 버퍼링 방식을 사용하였다. 이 공유 버퍼링 방식은 하나의 공유 버퍼를 세 가지 접근 방식에서 공유한다.

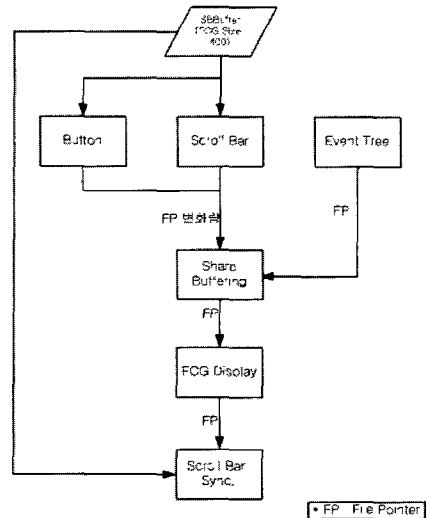


그림 3 스크롤바 동기 알고리즘

공유 버퍼링 방식은 그림 3 에서 보는 것과 같이 ECG 길이를 스크린 크기(= 스크롤 바의 최대 크기)로 나눈 몫으로 기준값을 설정한다. 이렇게 설정된 기준값을 위에서 언급한 첫 번째, 세 번째 방식으로 접근하여 오버헤드가 생기게 되면 스크롤 바 위치가 카운트되며, 또한 공유 버퍼에 있는 값과 파일 포인터의 변화량으로 파일 포인터의 값을 계산하게 된다. 하지만, 범위를 오버하지 않으면 스크롤 바 위치는 카운트 되지 않으며, 공유 버퍼에 값이 더해져 저장된다. 단, 두 번째 이벤트 트리에서는 파일 포인터가 바로 주어졌기 때문에 파일 포인터를 바로 공유 버퍼에 넣어 쓰게 된다.

파일 포인터의 변화량은 공유 버퍼에 쌓이며, 이렇게 쌓인 버퍼의 값을 기준값으로 나누어 스크롤바의 위치를 정하게 된다. 위의 그림 3 과 같이 각각의 접근 방식은 다르지만 공유 버퍼와 기준값을 공유하며 동기화는 이루어지게 된다. 즉, 어떠한 방식으로 접근해도 해당 위치에서의 공유 버퍼와 기준값으로 계산되는 값은 같게 된다는 것이다.

2.3.2 그리드와 ECG 데이터의 정렬

심전도는 QRS (심실중격, 좌·우심실이 차례로 탈분극 할 때 그려지는 곡선) 크기가 보통 1 mV 정도이다. 임상적으로 환자의 QRS 크기를 정확히 측정하는 것이 중요한 요소이기 때문에 감시 장치의 화면상에 기준 전압으로 1mV를 표시한다[4].

그러므로 화면의 격자를 표시하는 그리드는 기본 사이즈를 사용자들의 눈에 익숙한 1mV당 1cm로 가로·세로 0.5 cm의 정사각형으로 표현한다. 또한 전체 화면 크기는 CU-ER1 에서 한번에 표현할 수가 있는 최대 크기, 즉 4 sec 로 표현한다. ECG의 표현은 전체 화면인 경우, 400개의 픽셀을 표현하기 위한 [400 바이트 * 2] (ECG 픽셀 하나 당 2 바이트로 표현)의 버퍼를 가지고 있다.

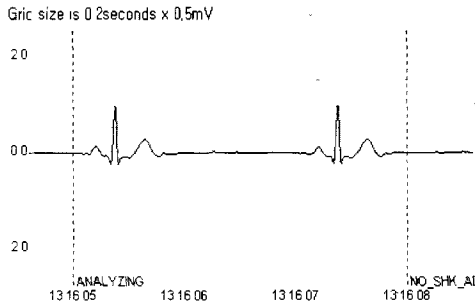


그림 4 그리드와 ECG 표시의 정렬

CU-EX1에서는 이러한 그리드와 ECG를 정렬시키기 위하여 400 바이트의 플래그 버퍼를 하나 더 두었다 현재 경우는 3가지 경우로 압축되며, 서로 선을 그릴 때, 이 플래그가 '0'일 경우 아무것도 그리지 않으며, '1'일 경우에는 1초마다 라인을 그리며, '2'일 경우에는 0.2 초마다 라인을 긋는다 이러한 플래그는 변화된 파일포인터의 위치를 감지하여 세팅되며, ECG 또한 이 파일 포인터로 접근하기 때문에 자동으로 정렬되어 표현된다

2.3.3 윈도우 크기의 최적화

CU-EX1은 사용자가 국내에서만 국한되는 것이 아니라 해외시장도 고려를 해야 한다 이것은 현재 사양으로 정해진 운영체제 (Windows 98/2000/xp)를 한글 버전 이외에도 영문, 중국어, 독일어 버전 등도 생각해야 한다는 뜻이다. 이때 중요한 것은 언어가 다른 운영체제마다 기본적으로 인식하는 사이즈가 달라 심전도 디스플레이 시에 그 사이즈가 다르게 나온다는 것이다. 본 연구에서는 이를 위하여 CU-EX1에 사용되는 모든 윈도우의 크기를 재조정하였다.

2.3.4 환자 데이터파일 관리

파일의 이름을 고치기 위해 현재 READ / WRITE Mode이기 때문에 파일을 닫는다 Last or First Name을 획득하여 새로운 파일이름을 우선 생성한다 혹시 Name이 모두 공백 일지도 모르기

때문에 Name의 길이를 체크하여 '0'이면 기본 파일명을 가진다. '0'이 아니면 현재 디렉토리에 변경될 같은 이름을 가진 파일이 없는지 체크한다. 만약 그런 파일이 없다면 파일명을 변경하고, 있다면 파일명을 카운트한다 카운트는 1부터 99999 까지 순차적으로 검색하며, 만약 검색 중에 같은 파일이 없다면 파일명 변경을 시도한다 단, 99999 까지 카운트를 해도 같은 이름을 가진 파일이 있다면 파일변경이 불가능하다.

이 알고리즘의 핵심은 복잡한 과정 없이 고쳐질 파일명에 순차적으로 번호를 붙여 같은 디렉토리 내에 지금 수정할 파일명이 있는지 비교하는 것이다.

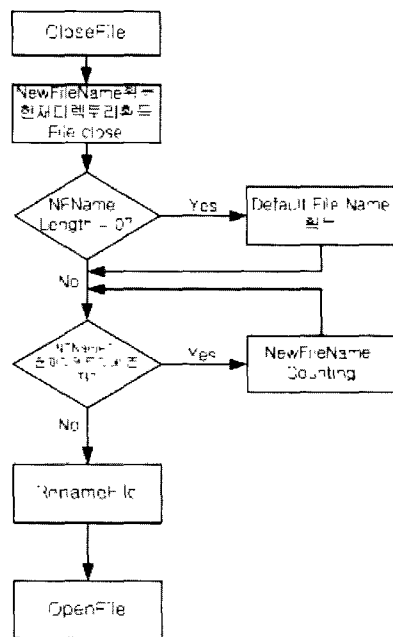


그림 5 Rename File 플로우차트

2.4 구현 결과

그림 6 은 좌측의 CU-ER2 과 우측 노트북의 CU-EX1 이 연결되어 IrDA 통신 방식을 사용하여 동작하는 모습을 보여준다. 그림 6에서 보는 것과 같이 CU-ER2 에서는 IrDA 통신 포트를 내장하고 있어 직접 통신 연결이 가능하다 이러한 통신은 22 절에서 구현한 통신 프로토콜로 가능하며, 구급차와 같이 충격이 심한 곳에서도 안전하게 통신을 할 수가 있는 것이다. 물론 CU-ER1 이나 CU-ER2 등에서도 같은 프로토콜로 동기화를 이루고 있다.

그림 7 과 그림 8 은 CU-ER2 에서 받은 심전도 파일을 해당 형식에 맞추어서 디스플레이하는

모습이다.



그림 6 CU-ER2와의 통신

그림 7에서 좌측에 보이는 트리는 장비를 사용하였을 경우 발생한 해당 이벤트이며, 우측에 보이는 화면은 환자 심전도에 이상이 생겨 전기 충격을 가했을 때의 심전도 모습이다.

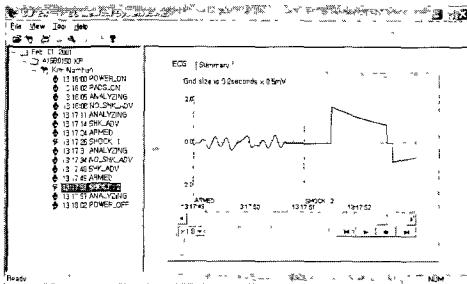


그림 7 CU-EX1의 심전도 표시화면

그림 8에서 우측에 보이는 화면이 환자 정보와 장비 정보를 보여준다.

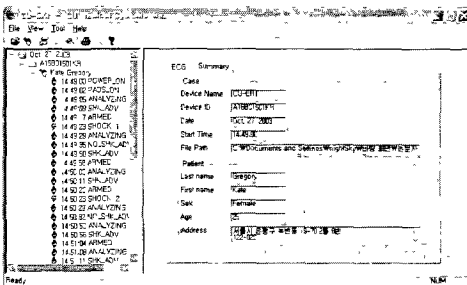


그림 8 CU-EX1의 환자정보 관리

본문에서 열거한 내용은 CU-EX1을 개발하면서 예상하지 못했거나, 다른 프로그램에서는 사용되지 않았던 방식 등을 나타낸 것이다. CU-EX1은 미리 정의한 포맷 형태로 심전도 데이터 및 이벤트 데이터를 CU-EX1으로부터 다운로드하여 심전도를 PC

상에서 디스플레이해주는 소프트웨어이다. 이 CU-EX1이 타사 제품과 비교하였을 때, 스크롤바에서 더 정교한 동작을 보인다. 타사 제품은 스크롤 박스가 심전도 디스플레이 후 계산되어 뒤늦게 움직이지만, CU-EX1은 위에서 언급한 공유 버퍼링 방식으로 심전도 재생 시에도 완벽한 동기화로 움직인다. 또한, 적은 메모리의 사용으로 현재 보이고 있는 화면에서 스크롤 하였을 때, 심전도를 해당 스크롤 박스의 위치 정보를 읽어 바로 디스플레이한다. 이것은 사용자로 하여금 보다 간편하고 보기 쉬운 인터페이스를 제공하는 것이다. 이밖에도 현재 영문 윈도우 OS 뿐만 아니라, 한글 윈도우 OS에서도 똑같은 사이즈로 디스플레이하여 앞으로 해외 시장에 초점을 맞추었을 때에도 문제가 없다.

3. 결론

본 연구에서는 체외형 자동 제세동기와 심전도계를 위한 통신 프로토콜을 설계하고 사용자 인터페이스를 구현하였다. 개발된 시스템, CU-EX1 ver. 0.91, 은 번들 버전으로써 원주 기독교 병원과 호주에서 사용 중이며, 이것은 베타 테스트의 역할도 포함하고 있다.

본 연구와 관련하여 차후 연구해야 할 과제는 통신 모듈 및 사용자 인터페이스 모듈을 표준화와 프린터 연결 기능을 포함한 사용성의 개선이다. 현재 ver. 0.91에서 미흡했던 여러 가지 문제점과 MEDICA 2003에서 요구되었던 사항을 개선한 ver. 1.00 을 2004년 3월초에 출시할 예정이며 2005년 3월경에는 더욱 기능이 보강된 CU-EX2 ver.2.00을 출시할 예정이다.

참 고 문 헌

- [1] 윤해연, 고훈우, "제세동기 최적 파라미터 설계", *센서학회지*, 제6권 제3호, pp.74-80, 1997.
- [2] 나학록, 최성환, 권호열, "DSP를 이용한 심장 세동기의 ECG 모니터링 시스템", *CMC2001 학술 논문집*, pp. 225-229, 2001. 6.
- [3] Jin-Ho Choi 's *Beginner's Guide to ECG*, http://heartkorea.com/ecg_guide/ecg_main.htm
- [4] 박상희, *생체신호처리 및 응용*, 에드텍, 1999.
- [5] 이상엽, *Visual C++ Programming Bible Ver. 6.X*. 1판, (주)영진출판사, 2003.
- [6] C. Petzold, *Programming Windows 95*, 8판, (주)교학사, 1999.
- [7] B.A. Forouzan, *Data Communications and Networking*. 2nd Ed., McGraw-Hill, 2001.
- [8] H. Schildt, *알기쉽게 해설한 C++*, 이한출판사, 1996.
- [9] <http://www.devpia.com>