
임상검사장비를 위한 안정적 인터페이스의 구현

이종혁*

Implementation of Stable Interface for Clinic Laboratory Equipment

Jong-hyeok Lee*

이 논문은 2009학년도 경성대학교 학술연구비지원에 의하여 연구되었음

요 약

디지털 병원이 지향하고 있는 병원내의 정보화 시스템으로는 검사정보시스템을 비롯한 여러 시스템들이 있다. 인터페이스 프로그램은 검사장비로부터 검사결과를 취합하고 검사정보를 효율적으로 관리 운영하는 핵심적인 작업이며, 이를 구현하는 방법은 멀티스레드 방법과 멀티태스킹 방법이 있다.

본 연구에서는 위에서 제시한 두 방법을 시뮬레이션을 통하여 비교 분석한 후 안정적인 방법을 제안하였으며, 제안한 방법을 적용하여 사용자 요구사항을 만족할 수 있는 인터페이스 프로그램을 설계, 구현하였다. 구현한 검사장비 인터페이스 프로그램을 병원 현장에 적용한 결과 안정적으로 동작함을 확인할 수 있었다.

ABSTRACT

As for information system in hospital which digital hospital aims at, there are LIS(Laboratory Information System) and so on. Interface program gather the results from the clinic laboratory equipment and manage the results efficiently as well. To implement interface program, multi tasking and multi thread method is used.

In this paper, we proposed the stable method after comparing and analyzing these two methods we present above through the simulation. Also we designed and implemented Interface program which satisfy the users' requirements applying the proposed method. The result of using the built interface program in the hospital field confirms that the program operates safely.

키워드

임상검사정보 시스템, 인터페이스 프로그램, 멀티태스킹

Key word

Laboratory Information System, Interface program, Multi tasking, Multi thread

I. 서 론

발전된 정보통신 기술 및 각종 신기술을 결합하여 병원을 21세기 디지털 시대에 부응하는 수준으로 정보화한 형태를 디지털 병원이라고 할 수 있다.[1] 디지털 병원이 지향하고 있는 병원내의 정보화 시스템으로는 분석 장비를 이용하여 검체에 대한 검사를 실시한 후의 결과를 처리하는 임상 검사정보시스템(LIS: Laboratory Information System),[2] 의사가 환자 진료 후 환자에 대하여 내리는 적절한 치료 및 추가적인 검사의 처방전을 발행하는 과정을 정보화한 처방전달시스템(OCS : Order Communication System),[3] 병원내에서 발생하는 각종 의료영상을 저장하고 전송하는 영상저장 및 전송시스템(PACS : Picture Archiving and Communication System),[4] 검사실 결과 정보 교류를 위해 CDA(Clinical Document Architecture) 문서를 생성 처리하는 방법과[5] 임상 정보를 공유하고자 하는 외부기관과의 자료 교환 및 관리가 가능한 HL7(Health Level 7)-CDA 기반 전자의무기록 시스템[6] 등이 있으며, 이에 대한 연구들이 지속되어져 왔다. 한편 디지털 병원을 구축을 위해 사용된 주요 정보통신 기술로는 인터넷, 무선 컴퓨팅, 보안기술(security), 네트워크 컴퓨팅, 요소 및 객체 기술(components and object technology), 원격의료, 데이터웨어하우스(data warehouse), 전자적 컴퓨팅 등이 있다.[7]

진단검사의학과에서 시행하는 대부분의 검사가 임상 검사 장비를 이용한 검사이고 이렇게 검사된 검사결과를 일일이 수작업으로 OCS에 입력한다는 것은 시간이 많이 걸릴 뿐만 아니라 수작업에 의한 오류가 발생할 수 있다. 그러므로 많은 병원들이 임상검사장비와 LIS를 연동하여 사용하고 있다.[3]

임상검사장비의 인터페이스 프로그램은 검사장비로부터 검사결과를 취합하고 얻은 검사정보를 효율적으로 관리하고 운영하는 핵심적인 작업이며, 인터페이스 프로그램에서 임상 검사 장비를 지원하는 방법으로는 둘 이상의 스레드를 동시에 실행할 수 멀티스레드 방법과 같은 시간에 여러 개의 프로그램을 띄우는 멀티태스킹 방법이[8] 있다.

본 연구에서는 위에서 제시한 두 방법 중 어느 방법이 안정적인지 시뮬레이션을 통하여 알아보고 이를 적용하여 사용자 요구사항을 만족할 수 있는 임상검사장비

인터페이스 프로그램을 설계, 구현한 후, 이를 병원 임상 검사실에 적용하여 확인하고자 한다.

II. 설 계

LIS의 블록도는 그림 1과 같다. 각 임상검사 장비, 인터페이스 프로그램, Main DB로 구성되어 있으며, 인터페이스 프로그램이 실행되는 클라이언트 내에 MDB와 Log File를 둔다.

클라이언트 컴퓨터의 최소 요구사항은 메인클럭 2.8 GHz, 메인메모리 1GB, 운영체제 윈도우 XP에서 이더넷 통신은 10 내지 100 Mbps, 시리얼 통신은 9600 bps를 지원하는 것으로 한다.

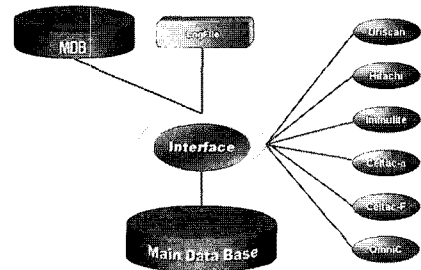


그림 1. LIS의 블록도
Fig. 1. Block diagram of LIS

인터페이스 프로그램은 각 장비에서 검사결과 값이 나왔을 때 결과 값을 파싱하여 화면에 보여주고 MDB파일과 Log File파일에 저장 한 후 Main DB에 최종 전송하도록 하며, MDB에서 각 장비별 결과를 보관하는 테이블에 기록시간 컬럼을 추가하여 Main DB에 삽입하는 일자와 시간을 넣도록 한다.

인터페이스 프로그램과 Main DB와의 통신은 TCP/IP 프로토콜을 이용하고, 임상검사장비와 인터페이스 프로그램간의 통신은 RS-232C 프로토콜을 이용하며, COM1 ~ COM9 까지 지정할 수 있도록 한다.

로그는 일별로 생성되는데, 단순로그는 일자, 시간, 채널No, 환자ID, 검체ID 및 송수신했던 내용을, 상세로그는 분할선, 단순로그내용, 파싱결과, 업데이트 SQL문을 각각 문단을 나누어 표시하도록 한다.

하나의 인터페이스 프로그램에 코딩을 통하여 여러

장비에 대한 환경설정을 할 경우, 다른 장비의 추가가 있을 시 프로그램상의 오류 또는 충돌 등의 문제가 발생할 수 있다. 이러한 문제를 해결하기 위해 초기 설정은 INI파일과 MDB 파일을 동시에 참조하여 환경설정을 하도록 한다. 환경설정에서는 Main DB접속 정보 저장, Log저장위치 설정 및 경로, MDB 저장 경로 지정 등을 할 수 있어야 하며, 환경설정 품은 modal로 실행되어야 한다. 그리고 메인화면에서 장비를 선택하면 환경설정(config.ini)에 지정된 포트번호는 자동으로 설정되도록 한다.

화면디자인은 가로, 세로 길이의 비율 4:3 또는 황금비율을 지향하며, 메인 폼은 리사이즈 되도록 하고, 리사이즈 시 내부 컨트롤(로그 데이터, 하단에 작업 모니터링 스프레드 쪽)도 같이 리사이즈 되도록 하며, 폼의 트립 단위를 가로 세로 모두 30으로 지정하고, 메인 폼에서 로그 데이터와 하단의 작업 모니터링 스프레드 사이에 구분자를 넣어 상하로 로그 데이터 높이와 작업 모니터링 높이가 조절이 가능하도록 한다.

III. 구현

구현을 위하여 컴퓨터는 CPU P-CEL 2.8GHZ, 메인 메모리 1GB, 윈도우XP Service Pack2을 사용하였다. 응용 프로그램 작성을 위하여 마이크로 소프트웨어 Visual Basic 6.0, 데이터 저장을 위하여 MDB는 마이크로소프트 오피스 Access 2003을, Main DB는 오라클 10g를 사용하였다. 임상검사장비는 뇨분석기 URISCAN Pro2(이하 URISCAN), 자동혈액분석기 CELLTAC-a, 자동혈구 계산기 CELLTAC-F, 혈액 개스 장비 OmniC, 생화학분석기 HITACHI 7180(이하 HITACHI), 면역학분석기 IMMULITE 2000를 대상으로 하였다. 먼저 단방향 통신을 지원하는 URISCAN과 양방향 통신을 지원하는 HITACHI에 대하여 시뮬레이션을 하고 그 결과를 이용하여 전체 프로그램을 구현하였다.

HITACHI의 프로토콜은 그림 2와 같으며, 시간당 720 검체를 처리할 수 있는 대형병원용 뇨 분석기인 URISCAN은 RS232c 통신 프로토콜을 이용하는 장비로서 단방향 통신을 지원하며, 실험에서는 통신속도 9600 bps, 패리티 비설정, 데이터 8비트, 중단 1비트로 설정하고 통신포트는 1로 하였다. 위와 같은 통신 프로토콜을

근거로 하여 HITACHI와 URISCAN 장비에 대한 에뮬레이터를 구현하였고 이를 인터페이스 프로그램의 시뮬레이션에 이용하였다.

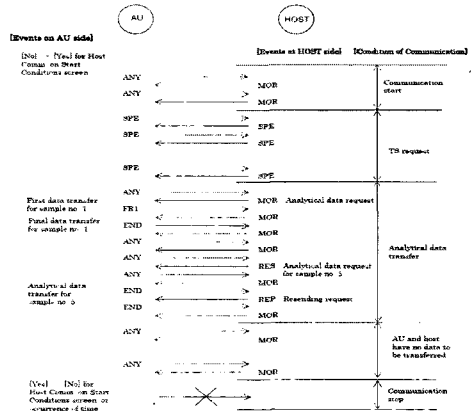


그림 2. HITACHI 통신 프로토콜
Fig. 2. Communication Protocol of HITACHI

3.1 멀티태스킹 방법

임상검사장비 인터페이스 프로그램 2개와 HITACHI, URISCAN 장비 에뮬레이터를 실행시킨 결과를 그림 3에 나타내었다. 스레드 증가와 프로그램 개수 4만큼 프로세스가 증가됨을 확인 할 수 있었다.

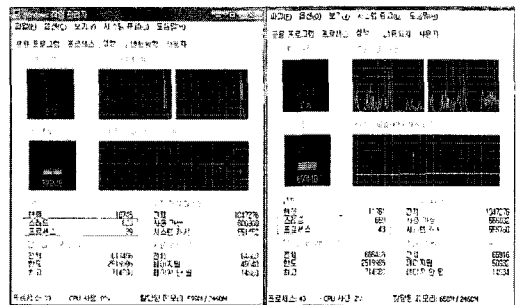


그림 3. 멀티태스킹 실행 결과
Fig. 3. Execution result of Multi tasking

먼저 URISCAN 에뮬레이터에서 전송을 시작한 뒤 조금 후 HITACHI 에뮬레이터에서 데이터를 전송 하였을 때의 결과를 그림 4에 나타내었다. 두 개의 프로그램이 독립적으로 실행되면서 각각의 검사 결과 값이 이상 없이 전송 완료하는 것을 확인할 수 있었다.

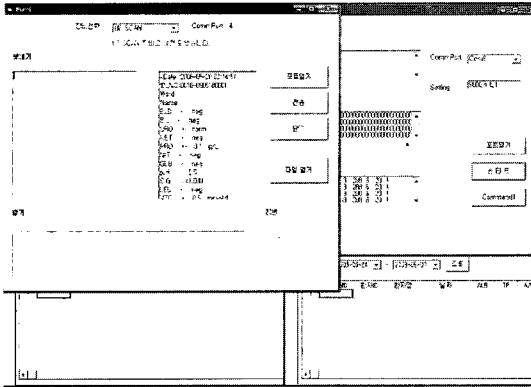


그림 4. 멀티태스킹에서 데이터 처리 결과
Fig. 4. Result of data processing at multi tasking

3.2 멀티스레드 방법

임상검사장비 인터페이스 프로그램 1개와 HITACHI, URISCAN 장비 에뮬레이터를 실행시킨 결과 스레드 증가와 프로그램 개수 3만큼 프로세스가 증가됨을 확인 할 수 있었다.

멀티태스킹 방법처럼 데이터 전송을 하였으며 이의 결과를 그림 5에 나타내었다.

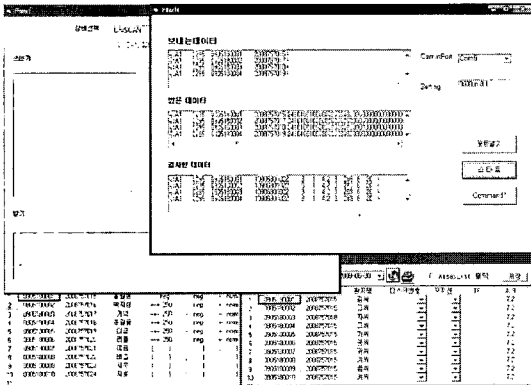


그림 5. 멀티스레드에서 데이터 처리 결과
Fig. 5. Result of data processing at multi thread

HITACHI 장비는 핸드 셰이킹 방식으로 데이터 전송을 하므로 클라이언트에서 데이터 처리가 늦어지더라도 정상적으로 DB에 저장이 되지만, URISCAN 장비 경우 단방향 통신으로 인하여 빠른 데이터 처리가 되지 않을 경우 데이터 손실을 가져올 수 있음을 확인 할 수 있었다.

표 1에 프로그래밍 방법에 따른 데이터 처리율을 나타내었다. 멀티태스킹 방법은 정상완료 했지만 멀티스레드 방법은 URISCAN에서 처리율이 86.7%이었다.

표 1. 프로그래밍 방법에 따른 처리율
Table 1. Processing rate at programing method

프로그래밍 방법	장비	송신	수신	처리율(%)
멀티태스킹	HITACHI	90	90	100
	URISCAN	300	300	100
멀티스레드	HITACHI	90	90	100
	URISCAN	300	260	86.7

여러 실험 결과를 분석한 결과 URISCAN 장비와 통신이 되는 도중 HITACHI 장비와 통신이 이루어지게 되면 URISCAN 장비와의 통신은 중단하고 HITACHI 장비와 통신이 진행됨을 알 수 있었다. 그러므로 URISCAN 장비에서 앞서 보낸 데이터를 클라이언트 컴퓨터에서 처리하지 못한 상태에서 다음 데이터를 버퍼에 보내게 되고 이를 인터페이스 프로그램에서 처리하게 되므로 늦게 들어온 데이터를 처리하지 못하게 되었다는 것을 알 수 있었다. 이와 같은 문제를 처리하기 위하여 소프트웨어적으로 버퍼에 들어온 데이터의 길이를 분석한 후 1번에 수신한 길이보다 클 경우 이에 대한 예외처리를 하여 에러를 감소시킬 수는 있을 것으로 생각된다. 그러나 본 연구에서는 예외처리를 하지 않고 동일한 프로그램 환경에서 실행하는 것을 전제로 한 것이므로 이를 에러로 간주하였다. 따라서 하나의 프로세스가 한 가지 일만 하는 멀티태스킹 방법 이 하나의 프로세스가 여러 가지 일을 하는 멀티스레드 방법 보다 안정적이라는 것을 확인 할 수 있었다.

3.3 다른 작업과 병행 시 멀티태스킹 방법

임상검사장비가 연결되어 있는 컴퓨터에 다른 작업이 있을 것을 예상하여, 0에서 499까지 각각 변하는 두수를 곱하여 배열에 저장하는 것을, 입력하는 값(500)의 5배를 반복 실행 후 이때까지 걸린 시간을 출력하고 위의 내용을 무한 반복하도록 하는 프로그램을 그림 6에 나타내었다.

```

...
setenv(ENVIRONMENT, "HITACHI")
...
main()
{
    int i;
    for(i=0; i<10; i++)
    {
        printf("HITACHI\n");
        sleep(1);
    }
}
...

```

그림 6. 부하를 위한 프로그램
Fig. 6. Program for load

부하를 위한 프로그램을 2개 실행하였을 경우 CPU 사용이 이미 100%가 되었으며, 부하를 위한 프로그램을 4개를 실행시킨 결과를 그림 7에 나타내었다.

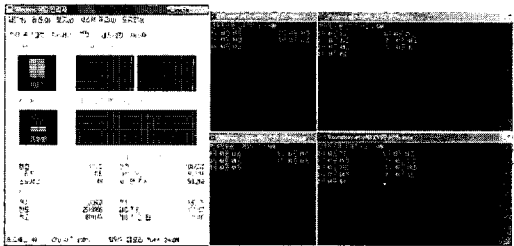


그림 7. 부하 프로그램 실행 결과
Fig. 7. Execution result of load program

부하를 위한 프로그램을 1개씩 늘려갔을 때의 데이터 처리 결과를 표2에 나타내었다. 부하를 위한 프로그램을 3개까지 실행했을 경우는 데이터를 제대로 처리하였으나, 4개로 하였을 경우 클라이언트에서 처리율이 HITACHI 경우는 100%이었으나 URISCAN의 경우 90%이었다.

표 2. 부하 수에 따른 처리율
Table 2. Processing rate at load number

부하수	장비	송신	수신	처리율(%)
3	HITACHI	90	90	100
	URISCAN	300	300	100
4	HITACHI	90	90	100
	URISCAN	300	270	90

각각의 임상검사장비에 대한 프로토콜을 분석하고 설계에서 제시한 내용을 수렴하여 각 임상검사장비에 대한 프로그램을 구현하였으며, HITACHI의 경우 장비를 선택하고 시작을 누르게 되면 OnCommHITACHI가 호출되는 것을 시작으로 하여 검사결과 값을 화면에 보여주고, 설정에 따라 로그 보관 및 MDB에 저장하도록 한다. 이를 그림 8에 나타내었다.

환경설정에서는 통신방법, 로그, MDB, 조회 저장의 메인 DB 접속정보 등을 설정할 수 있도록 하였으며, 설정값은 INI 파일과 MDB에 저장하도록 하였다 이를 구현한 결과를 그림 9에 나타내었다.

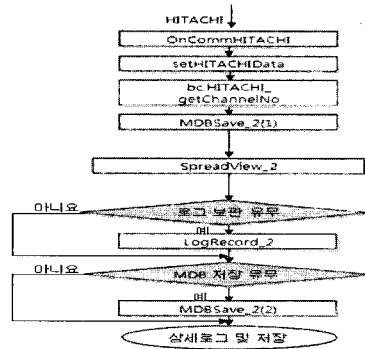


그림 8. HITACHI 흐름도
Fig. 8. Flowchart for HITACHI

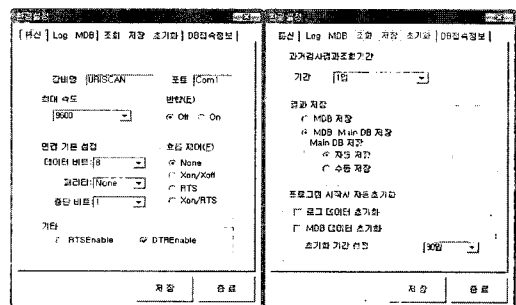


그림 9. 환경설정 화면
Fig. 9. window for environment configuration

설계에서 제시한 화면디자인을 포함한 각종 요구사항들을 고려하여 인터페이스 프로그램을 구현하였으며, 이를 병원 현장에 적용한 결과 안정적으로 동작함을 알 수 있었다. 그림 10에 메인 프로그램 화면을 나타내었다.

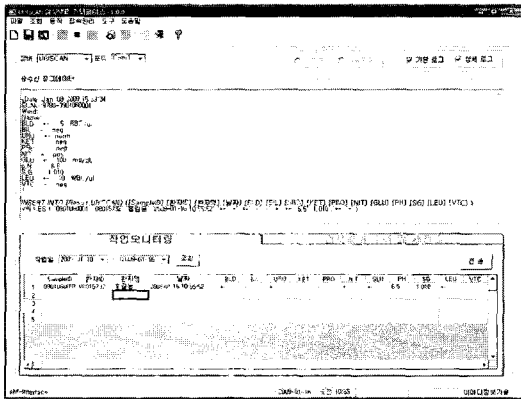


그림 10. 메인 프로그램 화면
Fig. 10. Main Program window

IV. 결론

영상검사장비의 인터페이스 프로그램은 검사장비로부터 검사결과를 취합하고 얻은 검사정보를 효율적으로 관리하고 운영하는 핵심적인 작업이며, 인터페이스 프로그램에서 영상 검사 장비를 지원하는 방법은 둘 이상의 스레드를 동시에 실행할 수 멀티스레드 방법과 같은 시간에 여러 개의 프로그램을 띄우는 멀티태스킹 방법이 있다.

본 연구에서는 위에서 제시한 두 방법 중 어느 방법이 안정적인지 시뮬레이션을 통하여 비교 분석한 결과, 멀티태스킹 방법은 하나의 프로세스가 한 가지 일만 하므로 하나의 프로세스가 여러 가지 일을 하는 멀티스레드 방법보다 안정적이라는 것을 확인 할 수 있었다.

부하를 위한 프로그램을 2개 실행하였을 경우 CPU 사용이 이미 100%가 되었으며, 3개까지 실행했을 경우 검사결과를 제대로 처리할 수 있으므로, 영상검사 장비가 연결되어 있는 컴퓨터에 사용자가 다른 작업들을 하면서도 검사결과를 잘 처리 할 수 있을 것으로 생각한다.

설계에서 제시한 화면디자인을 포함한 각종 요구사항들을 만족하며, 멀티태스킹 방법으로 영상검사장비 인터페이스 프로그램을 구현하였으며, 제한한 영상검사 장비의 인터페이스 프로그램을 병원 현장에 적용한 결과 안정적으로 동작함을 알 수 있었다.

참고문헌

- [1] 박광석, “디지털 병원의 발전동향“, 대한전자공학회지, 33권, 1호, pp. 1193~1198, 2006.
- [2] 박종성, “영상 검사정보시스템의 최근 동향“, 보건과학연구논집, 제 7권, 제 1호, pp. 1~8, 1998.
- [3] 양현택, 김원중, “병원 처방전달시스템 구축을 위한 의료장비 통합인터페이스 구현“, 한국정보과학회 가을 학술발표논문집, Vol. 25, No. 2, pp. 493~495, 1998.
- [4] 고신관, 양현준, 한동균, 김옥동, 강병삼. “PACS 도입에 의한 현상시스템 폐수 감소효과에 관한 연구“, 방사선기술과학, Vol. 30, No. 2, pp. 167~175, 2007.
- [5] 송준현, 김일곤, 이성현, 도형호, 예정훈, “검사실 정보 교류를 위한 임상문서표준규격의 적용“, 한국컴퓨터종합학술대회 논문집, Vol. 34, No. 1(B), pp. 21~26, 2007.
- [6] 조익성, 권혁승, “영상 정보교환을 위한 HL7-CDA 기반의 전자의무기록 시스템의 설계 및 구현“, 한국통신학회논문지, Vol. 33, No. 5, pp. 379~385, 2008.
- [7] 유시원, “보건의료분야 정보통신기술 활용 동향분석“, 보건복지포럼, pp. 62~71, 2004.
- [8] <http://ko.wikipedia.org/>

저자소개



이종혁(Jong-Hyeok Lee)

1975년 부산대학교 전자공학과 학사
1980년 부산대학교 전자공학과 석사

1991년 부산대학교 전자공학과 박사
1990년~현재 경성대학교 컴퓨터공학과 교수
1998년 7월~1999년 6월 미국 Beckman Institute, Univ. of Illinois, 객원연구원

※ 주관심분야: 인공지능, 컴퓨터시스템, 음성인식