

유무선 기반 멀티미디어 임베디드 시스템을 위한 네트워크 동기화 모듈 연구 및 구현

A Study and Implementation of Network Synchronization Module for Wired and Wireless based Multimedia Embedded Systems

김 홍 규*, 문 승 진
(Hong-Kyu Kim and Seung-Jin Moon)

Abstract : It is common to use PC or Digital picture frame in stand-alone fashion to view images, movies, or to listen MP3 musics which are considered as multimedia contents, However, such existing methods have weakness for expanding network requirements or augmenting extra data, in such case inevitably requiring external devices. With keeping in mind for such expansion, in this study, we have suggested a new concepts of network module which may utilize an image server for data transmission, in the proposed module, data alarm packet was defined for alerting incoming data from the image server and it became possible to synchronize between the image server and device not only in wired but also in wireless environments through UART. The method consists of a control module for an image server and a synchronization module between the server and the device. We have also tested the feasibility for future commercial usages such as advertisements through performance evaluations.

Keywords : embedded system, digital picture frame, network module, network synchronization, multimedia server

I 서론

대표적인 멀티미디어 임베디드 시스템은 디빅스 플레이어, MP3플레이어, 디지털 전자책자 등으로 나뉘며 특히 멀티미디어 임베디드 디스플레이 장치는 전자책자의 기능과 동영상 재생 및 음악파일 감상이 가능하며 추가적으로 정지영상과 함께 음악파일 감상이 가능한 장치로서 디지털 전자책자로 주로 사용된다. 디지털 전자책자는 일반 액자에 LCD 디스플레이 장치와 임베디드 시스템을 융합한 장치로 초소형 이미지 처리기에서 촬영된 이미지 및 동영상들을 감상할 수 있다. 이미지, 동영상, 음악파일(MP3)은 디지털 데이터로 이야기 하고, 디지털 데이터를 감상하기 위해 외부 저장장치를 이용하여 멀티미디어 디스플레이 장치(이하: 디지털 전자책자)에서 감상할 수 있는 방식이다. 이에 기존의 디지털 전자책자는 디지털 데이터를 외부 저장장치에서 복사하여 사용하는 등의 불편함과, 네트워크 지원이 불가능하여 원거리에서 촬영된 데이터를 사용자가 원하는 디지털 전자책자로 전송하여 감상 할 수 없는 단점이 있다.

디지털 전자책자의 네트워크 기능 지원은 전자책자의 가격 경쟁력을 위해 임베디드 프로세서(MCU)는 저렴한 칩을 사용하는데 프로세서 자체에 네트워크 지원이 불가능한 경우가 많다. 따라서 기존의 전자책자를 소유 하고 있으며 멀티미디어 서버를 통한 디지털 데이터와 소프트웨어 펌웨어 업그레이드는 호스트PC를 이용해야 가능하다.

이에 본 논문에서 유·무선 네트워크 기반에서 디지털

전자책자와 네트워크 통신 모듈이 연결되어 새로운 디지털 데이터를 다운받아 디스플레이 하거나 디지털 전자책자의 새로운 소프트웨어 펌웨어를 자동으로 업데이트 할 수 있도록 네트워크 동기화 모듈을 제안한다.

디스플레이 장치를 이용하여 초소형 이미지 처리기에서 촬영된 이미지들을 감상할 수 있는 장치이다. 이미지를 감상하기 위해 촬영된 이미지나 동영상과 같은 디지털 데이터를 외부 저장장치를 이용하여 디지털 액자에서 디스플레이 하는 방식으로 외부 저장장치에서 이미지를 옮기거나 삭제하는 등의 불편함이 있는데, 이러한 제한을 보완하기 위해서는 네트워크의 지원이 필수적이다. 이러한 전자책자는 디스플레이 장치의 가격이 비싸지만, 가격이 저렴해 진다면 여러 분야의 많은 사용자들이 이용할 것으로 예측된다.

본 논문에서는 유·무선 네트워크를 통하여 기존의 전자책자의 이미지 처리 기술에 이미지서버 연동을 통하여 새로운 이미지를 받아 디스플레이 하거나 전자책자의 새로운 펌웨어를 자동으로 업데이트 할 수 있도록 네트워크 모듈을 제안한다. 제안된 방법은 네트워크 프로토콜과 파일 관리 및 네트워크와 디지털 액자 간 동기화 방법으로 특정 주기마다 이미지 서버의 새로운 파일을 체크하여 어디서든 지 이미지를 목표 디지털 액자로 전송할 수 있다. 이처럼 제안된 네트워크와 디지털 액자 간 동기화 방법은 디지털 액자와 통신 채널을 형성하고 디지털 이미지 처리 장치에 의해 촬영된 디지털 데이터를 디지털 액자로 전송하여 소정 기간별로 세팅이 가능하다. 또한 키 입력에 대한 디지털 액자 제어 명령과 서버명령 방식으로 디지털 액자로 전송하여 소정 기간별로 세팅된 시간 경과를 판단하여 이미지 처리를 자동으로 표시하여 감상할 수 있다.

다음과 같이 2장에서 디지털 전자책자와 임베디드 이미

* 책임저자(Corresponding Author)

논문접수 : 2007. 9. 29., 채택확정 : 2007. 10. 26.

김홍규, 문승진 : 수원대학교 컴퓨터학과

(beeniis@suwon.ac.kr/sjmoon@suwon.ac.kr)

※ 본 논문은 2007년 산학 협력실 2차 지원 사업에서 지원하였음.

지 서버의 연동 기술을 소개하고, 3장에서 기술된 기법을 이용하여 모듈별 구조 및 구현을 설명하였다. 4장에서 구현된 기술을 바탕으로 성능평가를 수행하여 마지막 5장에서 본 논문의 결론과 앞으로 해결해야 할 과제들을 제시하였다.

II. 디지털 전자액자 네트워크 모듈 통신

초기의 전자액자는 간단한 이미지를 보여주는 기능에 특화된 장치이며, 외부의 디지털 데이터를 저장장치를 이용하여 복사하는 기능에 만족하였다. 이러한 특화된 기술이 점차 확대되어 디지털 TV, 디지털 냉장고등과 같은 생활 필수 가전제품에도 디지털 전자액자 기능을 추가 하는 추세이다. 이러한 디지털 전자제품의 초기 모델은 특정기능에 충실한 임베디드 장치이므로, 네트워크와 같은 통신 모듈을 따로 설치하지 않아 디지털 전자제품에서 디스플레이 될 수 있는 이미지는 한정적일 수밖에 없다. 초기의 디지털 전자제품과 본 논문에서 초점이 되는 디지털 전자액자에 네트워크 기능을 추가하기 위해 여러 통신 방법을 이용하여 네트워크를 최대한 활용할 수 있도록 본 절에서는 모듈별 통신 방법에 관하여 논한다.

1. 모듈별 통신

임베디드 장치에서 특화된 기술의 하나인 디스플레이 기술에 네트워크 기능을 포함하여 디지털 데이터 또는 장치의 펌웨어 모듈과 같은 소프트웨어등을 특정 디지털 서버에서 다운로드 받아 자동으로 처리 할 수 있는 모듈을 구현하였다.

따라서 디지털 서버와 네트워크 모듈 그리고 전자액자의 모듈이 서로 연동하려면 표 1과 같이 내외부적으로 여러 통신방법을 요구한다.

임베디드 네트워크 모듈은 독립된 프로세서를 가지고 있어 디지털서버와 디지털 전자액자 사이에서 데이터를 중계하는 기능이다. 이 모듈은 디지털서버의 새로운 데이터를 30초 단위의 폴링주기 동안 새로운 디지털 데이터 또는 소프트웨어 펌웨어 모듈 체크를 하거나, 디지털 전자액자로부터 디지털 데이터 삭제를 디지털서버의 디지털 데이터 리스트에서 삭제할 수 있도록 설계 하였다. 따라서 네트워크 모듈은 디지털서버와 네트워크로 연결되어 있고 디지털 전자액자와 USB와 시리얼(RS-232C)로 연결되어 데이터 통신을 한다. 네트워크 모듈과 디지털 전자액자의 두 가지 통신 방법은 현재의 디지털 전자액자는 MCU의 특성상 멀티

표 1. 네트워크 연동 모듈별 통신 방법.

Table 1. Network SYNC. module communication the table.

| 구분 | 통신방법 | 설명 |
|---------|-----------------------|-------------------------------------|
| 디지털서버 | Network | 네트워크모듈과 유·무선 연결 설정 및 데이터 송·수신 |
| 네트워크 모듈 | Network, RS-232C, USB | 이미지서버와 데이터 통신과 전자액자와 터널링 및 데이터 송·수신 |
| 전자액자 | RS-232C, USB | 네트워크 모듈과의 터널링 작업 및 데이터 송·수신 |

태스킹이 불가능한 제품이 다수 이므로 시리얼 인터럽트 기능을 이용하여 현재 진행중인 작업을 잠시 멈추고 USB를 이용하여 디지털 데이터를 주고받을 수 있는 터널링 하기 위해 사용된다.

시리얼 통신은 디지털 전자액자와 네트워크 모듈의 시스템 구성파일등을 항상 같게 유지 하거나 디지털 데이터 또는 소프트웨어 펌웨어를 다운 받기 위한 USB 마운트 포인트를 설정하기도 한다. 데이터의 전송은 USB를 사용하여 한꺼번에 많은 디지털 데이터나 업데이트 소프트웨어를 빠르게 다운 받을 수 있다.

2. 시리얼 인터럽트 패킷 정의

시리얼 통신은 네트워크 모듈과 디지털 전자액자의 데이터를 전송하는데 중요한 USB 마운트 또는 언 마운트 시점을 포인팅 시켜주는 중요한 역할을 한다. 네트워크 모듈과 디지털 전자액자가 서로 주고받는 패킷은 그림 1과 같으며 패킷의 총 크기는 10byte로 구성되어 있다.

그림 1의 기능(function)은 1byte로 표 2와 같이 구성된다.

10byte의 패킷은 헤더, 인덱스, 명령, 체크섬 비트 값, 패킷 종단, 체크섬 값들로 구성되어 있으며 각 패킷의 길이와 비트 값을 참조하여 패킷의 손실 체크를 할 수 있다. 패킷 헤더와 인덱스 값으로 각 모듈의 송신자와 수신자를 구분하고 기능키(표 2)를 참조하여 수행한다.

시리얼 데이터 패킷의 정의는 네트워크 모듈이 디지털 서버에서 폴링 주기 때 생성한 데이터 체크 리스트를 참조하여 새로운 데이터가 존재 할 경우 디지털 전자액자의 동작은 중단되고 USB를 마운트 하거나 디지털 전자액자에 존재하는 디지털 데이터를 삭제하여 데이터 체크 리스트가 변경되어 디지털 서버의 데이터를 삭제하기 위해 사용된다.

디지털 서버와 디지털 전자액자의 체크 리스트 비교를 이용한 디지털 데이터 다운로드 및 삭제 방법은 디지털 서버의 데이터와 디지털 전자액자의 저장 데이터를 항상 같은 내용으로 유지하기 위해 사용된다.

USB 마운트 요청은 디지털 전자액자에서 네트워크 모듈

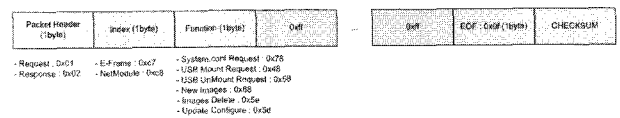


그림 1. 네트워크 모듈의 전자액자 패킷 정의.

Fig. 1. Digital picture frame of network module the figure.

표 2. 기능키 파일 인덱스.

Table2. File index of function the table.

| 기능명 | 파일 인덱스 |
|---------------------|--------|
| system.conf Request | 0x78 |
| USB Mount Request | 0x48 |
| USB Unmount Request | 0x58 |
| New Images | 0x68 |
| Images Delete | 0x5e |
| Update Configure | 0x5d |

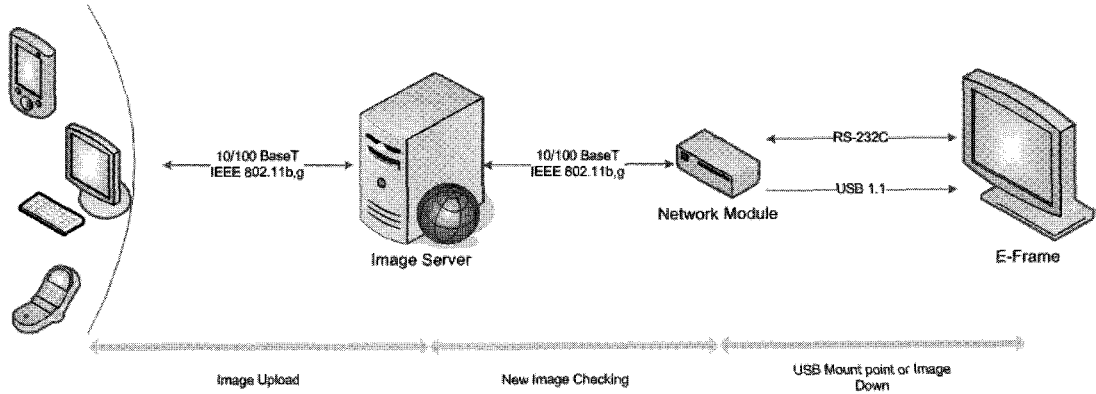


그림 2. 네트워크 모듈 시스템 흐름도.
Fig. 2. Network module system the figure.

로 요청할 수 있으나 마이크로프로세서와 하드웨어 구조의 특성상 디지털 전자액자 스스로 마운트 상태로 변경하지 못하며 네트워크 모듈에서 마운트 요청 패킷(0x48)이 수신 되어야만 마운트 모드 대기상태로 변경할 수 있다.

앞에서 설명된 시리얼과 USB를 이용한 디지털 데이터 전송방법은 네트워크 기능을 갖추지 못한 디지털 전자액자에 이용자가 네트워크 모듈을 장착하여 디지털 전자액자의 소프트웨어 펌웨어 업그레이드만으로 별도의 시스템 구성 없이 편리하게 이용할 수 있다.

3. 시스템 구성 시나리오

기존 디지털 전자액자와 네트워크 모듈이 연결된 시스템 구성을 설계하기 위하여 시스템 시나리오를 다음과 같이 구성하였다.

step 1: 이용자는 휴대폰이나 디지털카메라 또는 PDA와 같이 외부기기에서 작성된 이미지나 동영상을 디지털 서버에 웹페이지나 개인 블로그)를 이용하여 저장한다.

step 2: 저장된 이미지들을 관리하여 개인 디지털 전자액자로 다운로드 하는 디지털 서버는 디지털 전자액자의 초기화 정보 패킷에 포함된 액자 ID로 이미지 리스트를 생성하여 저장한다.

step 3: 이미지가 디지털 서버에 새로 등록되거나 디지털 전자액자로부터 삭제 요청이 있을 경우 디지털 리스트는 항상 갱신되고 디지털 전자액자의 네트워크 모듈과 30 초 주기로 비교한다.

step 4: 디지털 전자액자는 초기 전원이 켜지면 네트워크 모듈과 시스템 설정파일을 복사하여 항상 같은 파일을 유지한다. 이러한 시스템 구성 파일의 네트워크 정보와 디지털 전자액자 정보로 네트워크 모듈은 다음과 같은 작업을 수행 한다.

- 네트워크 유선설정(IP, subnet, gateway) 또는 무선설정 (SSID, webkey 등).
- 디지털 전자액자 고유 ID 및 제조사, 시리얼번호 정보 확인
- 디지털 전자액자 사용중인 메모리 카드 용량 확인
- 소프트웨어 펌웨어 정보 확인

```
typedef struct {
    char pk_header;           /* 헤더구분필드*/
    char lo_index;           /* 명령 인덱스*/
    char ap_function;        /* 응용 명령*/
    char ap_check_field;     /* 패킷 체크 섬 필드 */
    char pk_end_of;         /* 패킷 중단*/
    char ap_checksum;        /* 체크섬 구분자*/
}
```

그림 3. 시리얼 패킷헤더 기본 구조체.
Fig. 3. Serial(RS-232C) packet header basic struct the table.

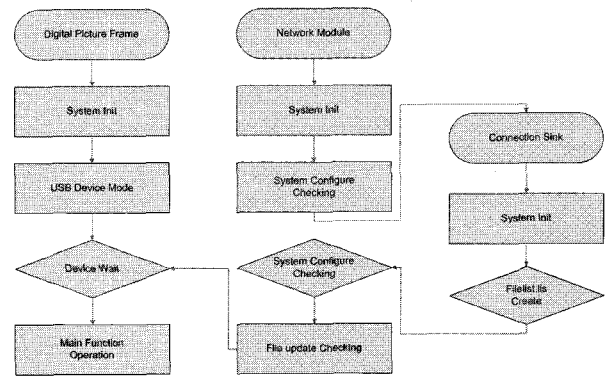


그림 4. 디지털 전자액자, 네트워크 모듈 서비스 흐름도.
Fig. 4. Digital picture frame and network module service flow the figure.

- 디지털 서버와 초기화 통신(고유 ID, 제조사, 시리얼번호-디지털 서버에 등록된 전자액자만 통신이 가능)

step 5: 시스템 구성(system configure)의 이미지 서버 IP 주소에 접속하여 전자액자 통신 승인 요청 및 확인

step 6: 30초 폴링 주기로 이미지 서버의 이미지 및 에어 업데이트 정보를 확인하여 다음 폴링 때 이미지를 다운 받아 네트워크 모듈의 저장장치에 임시 저장한다.

step 7: 네트워크 모듈은 다운받은 이미지를 디지털 전자액자의 저장장치에 복사하기 위해 시리얼통신을 이용하여 USB 마운트 요청한다.

1) 본 논문에서는 구현하지 않았음.

step 8: 네트워크 모듈은 step 7의 요청 후 디지털 전자액자와 다음과 같은 작업을 수행 후 종료한다.

- 새로운 디지털 이미지 디지털 전자액자로 복사
- 네트워크 모듈의 디지털 리스트 갱신
- 디지털 전자액자의 사용중인 저장장치 갱신
- 디지털 전자액자의 소프트웨어 펌웨어 정보 확인

본 논문에서 구현된 시스템 모듈은 크게 디지털 전자액자의 콘텐츠 관리자(contents agent management; DF_CAM)와 네트워크 모듈의 통신 관리자(communication agent management; NET_CAM), 디지털 데이터 동기화 관리자(digital data sync. agent management; NET_DSAM)로 구성되고, 디지털 서버의 디지털 데이터 관리자(digital data agent management; SER_DAM)는 이미 구현되어 있는 것(SKT의 리뷰-lovev website)을 사용하였다.

III. 임베디드 기반 네트워크 모듈 설계 및 구현

본 논문에서의 임베디드 모듈은 안정적인 디지털 파일 동기화와 데이터 송수신을 위해 두 가지 이상의 통신 방법을 지원해야 하며 그중 네트워크 모듈과 디지털 서버와의 연계 통신에서는 연동 프로토콜을 필요로 한다. 이 때 디지털 서버와 네트워크 모듈은 TCP/IP 프로토콜을 기준으로 하여 일정 30초 폴링 주기 동안에 PULL 방식으로 데이터를 송수신하며, 하나의 패킷은 여러 필드로 구분되어 지는데 이러한 필드 구분을 ';'로 구분하며 메시지의 끝은 '|'로 정의 하였다.²⁾

1. 패킷식별자

본 논문에서 구현된 네트워크 기능은 디지털 전자액자에서 디지털 서버와 통신 및 데이터의 송수신을 하기 위하여 패킷을 사용하는데 이러한 패킷이 어떠한 기능을 하게 되는 것에 대한 구분이 필요하며 이것을 패킷식별자라 한다.

디지털 서버와 네트워크 모듈의 패킷 구분 식별자는 표 3과 같고 패킷 구분 식별자는 TCP/IP 기반의 프로토콜로 구성되어 있다. 이에 본 논문의 모든 패킷 통신은 표 3의 형식을 기준으로 네트워크 모듈과 디지털 서버가 전자액자에 서비스를 제공하기 위해 공통 헤더를 사용하여 해당 ID를 가지고 있는 디지털 전자액자와 통신한다. 이러한 패킷정보 메시지는 17byte의 크기로 생성되며 length("LENGTH")+10의 길이를 스트링(string) 형태로 함께 가지고 있으며 헤더 타입에는 공통 헤더를 포함하고 있다. 즉, LENGTH Filed를 제외한 전체 길이의 LENGTH value는 고정길이 10byte를 사용하고 패킷을 수신할 때 처음 17byte를 먼저 읽고 길이 값을 파싱하여 전체 패킷을 수신하게 되는 방식으로 LENGTH filed에 설정된 길이와 실제 수신한 길이가 다르면 오류로 간주한다.

그림 5의 "MmiKeyAvailable()" 제작 함수를 이용하여 패킷의 시리얼 인터럽트 상태를 실시간 모니터링 하며 패킷 시작 열이 검출되면 "SerialDataAvail()"에서 인터럽트 명령을 확인한다. 키 값의 인터럽트 명령 수행 부분에서 인터럽트 명령 수행이 존재할 경우 "SerialGetByte()"로 키를 입력

표 3. 패킷식별자.

Table3. Packet identifier the table.

| field | type |
|---------------|---------|
| 초기화 메시지 | INIT |
| 사진 동기화 메시지 | SYNC |
| 사진 삭제 요청 메시지 | DEL |
| 사진 전송(pull) | PULL |
| update 요청 메시지 | UPDATE |
| upgrade 메시지 | UPGRADE |
| file_ok 메시지 | FILE_OK |

```

BOOL MmiKeyAvailable(void) {
... 중략 ...
#ifdef NETWORKKTYPE
if(key!="") {
TimerWaitMs(5);
if(SerialDataAvail()) {
key = SerialGetByte();
switch(key) {
case 'INIT':
초기화 실행;
return(retval);
break;
case 'SYNC':
이미지 동기화 실행;
return(retval);
break;
... 중략 ...
}
}
}
}
    
```

그림 5. 패킷 필터링 및 수행.

Fig. 5. Packet filtering and execution the table.

받아 변수 키(key)에 저장하여 7개의 지정된 인터럽트를 수행한다.

2. 전체모듈 초기화

디지털 전자액자의 DF_CAM과 네트워크 모듈의 NET_CAM, NET_DSAM을 구동하여 디지털 전자액자와 네트워크 모듈의 이상상태를 체크 하고 네트워크 연결에 필요한 정보 등을 확인하여 네트워크를 통한 디지털 서버의 디지털 파일 리스트, 소프트웨어 펌웨어 업데이트 정보를 갱신하며 그림 6과 같이 메시지 흐름을 나타낸다.

그림 6의 DF_CAM은 NET_DSAM의 디지털 파일 리스트 프로세스에 의한 파일 목록 체크 후 파일리스트 큐의 메시지들을 수신하여 큐에 저장하고 있다. NET_DASM의 메시지 수신 후 디지털 파일 동기화와 DF_CAM의 설정파일(configure file) 체크 여부와 디지털 전자액자의 시스템 설정정보 확인, 변경, 적용 후 빈파일(zero size file)을 제거 하여 핸들러 이벤트를 DF_CAM로 전송한다. DF_CAM은 메시지 프로세스와 핸들러에 의한 초기화, 동기화, 펌웨어 업

2) STK Loview 연동규격서 V1.1

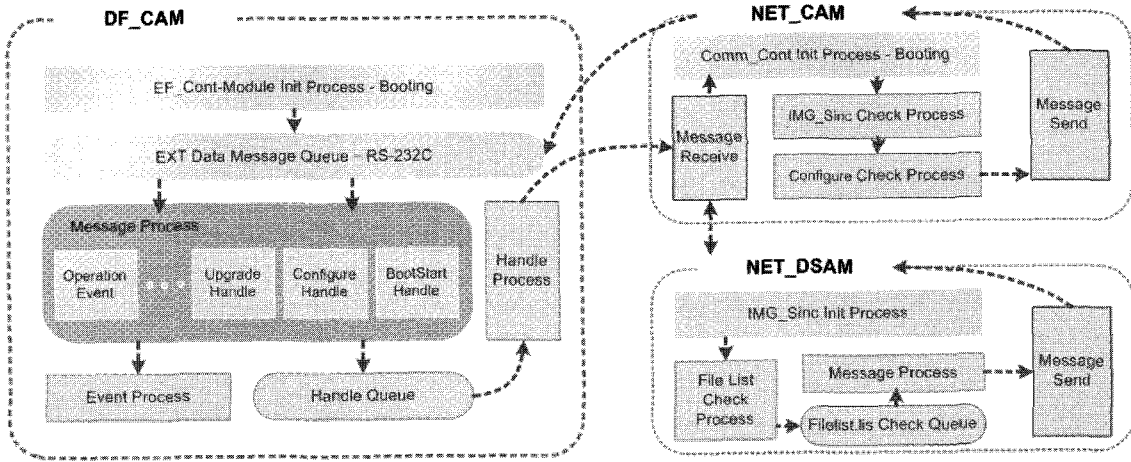


그림 6. 모듈 초기화 메시지 흐름도.

Fig. 6. Module init message flow the figure.

```

BOOL AppCheckAllDisks(UNIT32 AutoCheck) {
... 중략 ...
Forced_checking:
*** 현재 사용중인 메모리 슬롯 체크 ***
#ifdef NETWORKTYPE
if(UpdateFile()) { /* 업데이트 후 펌웨어 삭제 */
DiskSetReadOnly(FALSE);
DosDeleteFile("UPF_Check.dat");
DosDeleteFile("Binari.bin");
DiskSetReadOnly(TRUE);
}
/* 새로운 펌웨어 업데이트 */
if(CheckForFirmwareUpdateFile_Nor("Binari.bin",
FALSE == TRUE))
}
    
```

그림 7. 초기화 펌웨어 업데이트 체크.

Fig. 7. Init firmware update checking the table.

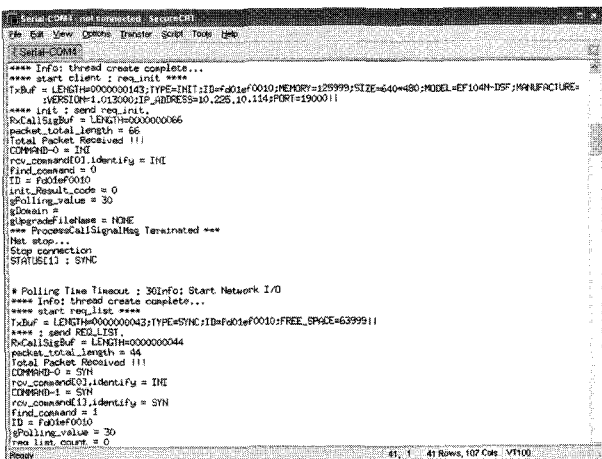


그림 8. 네트워크 모듈과 디지털 서버 초기화 과정.

Fig. 8. Network module and digital server init the figure.

그레이드, 설정변경(configure) 초기화 완료 이벤트 등을 수행하게 되고 최종적으로 이벤트 프로세스가 수행된다.

네트워크 모듈의 req_init는 system.conf 파일에 포함된 정보를 담고 있으며 디지털 전자액자의 고유한 ID, memory, LCD size, model, manufacture, version information, ip address, server address, port 등이 포함되어 있다. 고정 IP를 이용하여 푸쉬(push) 서비스가 가능할 수 있도록 IP와 포트 정보를 포함한다. 디지털 서버는 res_init 메시지를 전송하며 이 메시지는 디지털 전자액자의 ID와 리절트(result)로 디지털 서버 등록여부와 함께 폴링 주기가 초 단위로 전송하고 업데이트 정보를 포함한다.

그림 8과 같이 네트워크 모듈의 NET_DSAM은 req_init를 시작으로 Txbuf에 여러 정보를 포함하여 req_init를 전송하고 RxCallSigBuf의 길이를 응답 받아 확인한다.

3. 디지털 데이터 동기화 및 다운로드

디지털 서버의 디지털 데이터 리스트와 디지털 전자액자 네트워크 모듈의 데이터 리스트를 동일하게 유지 하는 것을 동기화라고 하며, 동기화는 디지털 데이터의 다운로드나 삭제를 뜻한다. 따라서 새로운 디지털 데이터를 다운받기 위해서는 반드시 데이터 동기화가 선행 되어야 하며, 디지털 데이터 다운로드 후 디지털 데이터 리스트를 갱신한다. 구현된 풀(pull)과 푸쉬(push)방식의 디지털 데이터 체크 및 다운로드는 디지털 서버와 res_list 메시지에 있는 아이템(item) 파일에 신규, 삭제 등의 정보 패킷을 만든다. 디지털 전자액자 네트워크 모듈은 디지털 서버와 TCP 연결(connect)을 통해 req_list를 송신하고 디지털 서버는 네트워크 모듈러 res_list를 소신하여 새로운 디지털 데이터 체크를 하고 TCP 연결을 끊고(disconnect) 디지털 서버로 송신함으로써 한 번의 디지털 데이터 동기화 작업이 종료 된다.

그림 9와 같이 DF CAM은 디지털 전자액자에 저장된 데이터 들을 디스플레이와 동시에 NET_CAM의 시리얼 패킷 수신대기하고 있으며 이벤트 핸들러에 의해 메시지를 구분하여 디지털 데이터 리스트를 구성한다. 이러한 목록을 NET_CAM에서 표 4와 같은 형태의 메시지를 req_list로 풀

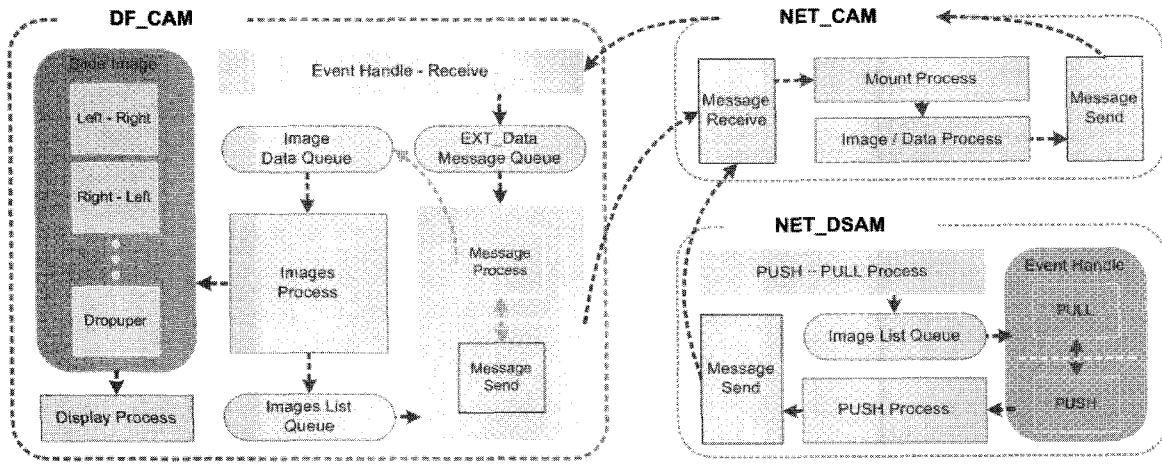


그림 9. 디지털 데이터 동기화 및 다운로드 흐름도.
Fig. 9. Digital data SYNC and download flow the figure.

방식으로 디지털 데이터 리스트를 다운받아 NET_DSAM에서 표 4와 같은 res_list를 송신하는데 req_list와 res_list는 모두 디지털 전자액자의 고유 ID로 구분되고, req_list는 디지털 전자액자에 사용가능한 메모리 용량을 kbyte 단위로 전송한다.

디지털 서버의 res_list는 디지털 전자액자의 고유 ID와 폴링 주기의 새로운 디지털 데이터 총 개수를 표 8의 "COUNT"로 보내며 "COUNT"가 "0"이면 새로운 데이터가 없고, "COUNT"가 "0"이 아니면 데이터의 상태(신규, 삭제)인 "FLAG"와 "FLAG"가 정의하는 디지털 데이터 이름인 "ITEM"을 하나의 패킷으로 송신한다.

디지털 데이터는 HTTP/GET으로 풀 방식으로 해당 데이터를 한 번에 한 개의 데이터만 수신 받을 수 있다. 디지털 전자액자는 디지털 서버로 TCP_connect와 req_digital을 송신하고, res_digital과 디지털 데이터를 수신 받는다. 이에 네트워크 모듈의 NET_CAM은 디지털 전자액자의 DF_CAM과

표 4. REQ_LIST 메시지 형식.

Table4. REQ_LIST message the table.

| field | type | value | 비고 |
|------------|--------|------------|-----------|
| ID | String | E-Frame ID | Unique ID |
| FREE_SPACE | String | Digit | Kbytes |

표 5. RES_LIST 메시지형식.

Table5. RES_LIST message the table.

| field | type | value | 비고 |
|---------|--------|-----------|---------------|
| ID | String | E-FrameID | Unique ID |
| POLLING | Int | 20 | Polling ; Sec |
| COUNT | Int | 숫자 | ITEM No |
| FLAG | String | 데이터구분자 | N, D |
| ITEM | String | 데이터이름 | 데이터이름 |
| FLAG | String | 데이터분자 | N, D |
| ITEM | String | 데이터이름 | 데이터이름 |

```

Serial-COM4 - not connected - SecureCRT
File Edit View Options Transfer Script Tools Help

Serial-COM4
ID = fd01ef0010
init_result_code = 0
gPolling_value = 30
gDomain =
gUpgradeFileName = NONE
*** ProcessCallSignalMsg Terminated ***
Net stop...
Stop connection
STATUS(1) : SYNC
Polling Time Timeout : 30Info: Start Network I/O
*** Info: thread create complete...
**** start req_list ****
TxBuf-1 = LENGTH=0000000000;TYPE=SYNC
TxBuf-2 = LENGTH=0000000000;TYPE=SYNC
TxBuf-3 = LENGTH=0000000000;TYPE=SYNC;ID=fd01ef0010
Entering search Disk Size :
str = 1 /mnt/usb

TxBuf-5 = LENGTH=0000000000;TYPE=SYNC;ID=fd01ef0010;FREE_SPACE=63399
Total Length = 43
rxbuf = LENGTH=00000000043;TYPE=SYNC;ID=fd01ef0010;FREE_SPACE=63399
**** : send REQ_LIST
RxCallSigBuf = LENGTH=000000000082
packet_total_length = 82
Total Packet Received !!!
COMMAND=0 = SYN
rcv_command[0].identify = INI
COMMAND=1 = SYN
rcv_command[1].identify = SYN
first_command = 1
ID = fd01ef0010
gPolling_value = 30
req_list_count = 1
flag = N
*****New File Lists*****
bapFilename = 200609041107000057990.jpg
newname = 00057990.jpg
Truly New File = 200609041107000057990.jpg
*** ProcessCallSignalMsg Terminated ***
Net stop...
Stop connection
STATUS(2) : REQ_PICTURE
**** MOUNT PACKET SEND for DATA DOWNLOAD ****
Data Download[REQ_PICTURE] : Check Disk Mount Status
    
```

그림 10. 신규 디지털 데이터 다운로드.

Fig. 10. New digital data download the figure.

USB를 통한 데이터를 송수신하기 위한 마운트 모드로 대기한다. 디지털 데이터 다운로드 종료 후 NET_DSAM은 새로운 데이터 리스트를 갱신하여 NET_CAM으로 전달함과 동시에 프로세스에 의해 새로운 데이터부터 디스플레이 프로세스로 이동하여 화면에 표시한다. 모든 작업이 종료되면 NET_DSAM은 데이터 서버로 FILE_OK 패킷 식별자를 송신한 후 TCP_disconnect를 수행한다.

4. 디지털 전자액자 업데이트

디지털 전자액자의 데이터 삭제로 인한 데이터 리스트 수정, 시스템 설정파일의 수정과 같은 작업을 수행하여 디지털 서버에 적용시킬 우에 사용한다. 디지털 전자액자의 업데이트 요청 메시지는 디지털 서버에서 TCP_connect와 함께 req_update를 송신하게 되면 res_update 메시지를 통해 업데이트 내용을 디지털 서버에 통보한 후 TCP_disconnect

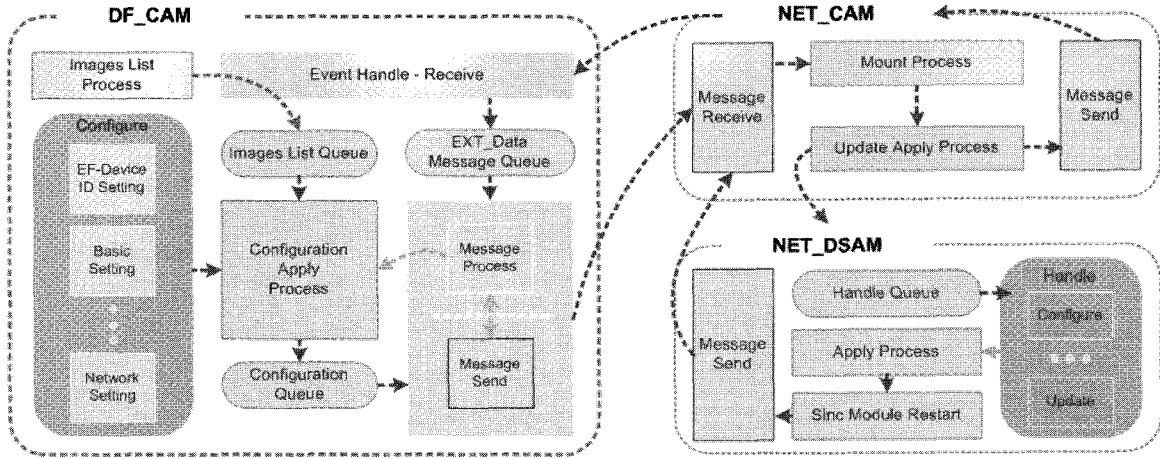


그림 11. 업데이트 메시지 흐름도.
Fig. 11. Update message flow the figure.

```

void MainFunction(void) {
    ... 중략 ...
    DCFCacheResetBuffer();
    if((DCFCacheInitialize(JPEG)==TRUE) &&
    !DCFCacheFileAvailableInDisk(JPG_OR_MOVIE_ONLY)) {
        FileCheck=FALSE;
        IsUsbConnect=0;
        MenuMode(START_MENU); }
    else {
        /** 이미지가 있을 경우 자동 재시작 **/
        FileCheck=TRUE;
        IsUsbConnect=0;
        AutoPlay &= AutoPlay_flag;
        if(AutoPlay==TRUE){
            ThemeSetupMenuCreate(&themeInfo);
            CurrentThemeCheck();
            Viewer(JPEG);
            ... 중략 ...
        }
    }
}
    
```

그림 12. 초기화 펌웨어 업데이트 체크.
Fig. 12. Init firmware update checking the table.

함으로서 업데이트 작업을 종료한다. 하지만, 이미지 삭제 요청 메시지는 업데이트와 다른 방법으로 메시지 통신을 하며 디지털 전자액자에서 TCP_connect와 함께 req_del을 송신하고 디지털 서버에서 res_del을 수신하면 TCP_disconnect를 함으로서 종료된다.

본 논문에서 구현된 업데이트와 삭제는 그림 11과 같은 방법으로 진행하게 된다.

그림 11의 DF_CAM은 이벤트 핸들러의 메시지 수신을 위해 대기 중이고, 이와 함께 데이터 리스트 프로세스는 큐에 보관하고 있다. 따라서 업데이트는 디지털 전자액자에서 설정하고자 하는 각 시스템 변수들을 적용시켜 프로세

스에서 처리되며 이와 함께 데이터 리스트 큐에 저장된 내용과 함께 환경 설정 큐에 저장되어 메시지를 NET_CAM으로 보내게 된다. NET_CAM은 DF_CAM에서 수신한 데이터를 확인하여 업데이트 어플라이 프로세스를 실행하며 DF_DSAM을 호출하여 핸들러 큐에 저장한다. 핸들러는 시스템 설정, 업데이트, 디지털 데이터 삭제 등과 같은 작업을 구분하여 어플라이 프로세스에 전달하며 변경된 내용을 적용하기 위해서 NET_DSAM을 재시작 한다.

디지털 전자액자의 데이터 동기화 메시지에 있는 데이터 리스트를 디지털 서버의 리스트와 비교하여 일치 되지 않은 디지털 데이터에 대해 삭제 또는 TCP/IP를 이용하여 디지털 데이터를 전송하는 방법으로 수동으로 데이터를 관리할 수 있다.

IV. 모듈 유용성 검증

네트워크 모듈을 이용한 디지털 서버의 콘텐츠 다운로드 유용성을 검증하기 위해 폴링(polling) 주기 30초, 시리얼 패킷 전송속도 20kbps, USB 콘텐츠 데이터 이동속도 1.5Mb로 규정하였다. 디지털 서버에 저장되어 있는 콘텐츠는 이미지(JPEG), 음악파일(MP3), 동영상(MPEG), 펌웨어(update.wrm)를 다운 받는다.

디지털 서버와 네트워크 모듈은 동일한 네트워크상에 위치하여 각 콘텐츠별 단일, 다중, 펌웨어 다운로드 시간과 적용시간 등을 측정하였다. 디지털 서버에서 네트워크 모듈로 다운받아 전자액자에서 확인하기까지 걸리는 시간 측

표 6. 디지털 서버 콘텐츠.

Table 6. Digital server content's the table.

| 콘텐츠 | 용량(kbyte) | 기타 |
|-----|-----------|------------------|
| 이미지 | 650 | 1280×1024 |
| 음악 | 5120 | 5분 |
| 동영상 | 15360 | 5분 |
| 펌웨어 | 2048 | 네트워크모듈, 디지털 전자액자 |

표 7. 콘텐츠 다운로드 및 적용과정.

Table 7. Content's download and apply the table.

| 분류 | 설명 |
|-------------------|---------------------------------|
| polling | 네트워크 모듈과 디지털 서버 통신 |
| content checking | 새로운 콘텐츠 검색 후 다운로드 설정 |
| USB mount | 네트워크 모듈과 디지털 전자액자 시리얼, USB 연결설정 |
| polling | 새로운 콘텐츠 다운로드 연결설정 |
| content download | 콘텐츠 다운로드 |
| content list edit | 다운로드 콘텐츠 리스트 갱신 및 펌웨어 버전 수정 |
| unmount | 네트워크 모듈과 디지털 전자액자 USB 연결 해제 |
| restart(apply) | 디지털 전자액자 리셋 후 신규 데이터 뷰어 시작 |

정을 “polling→image checking→USB mount→polling→image download→digital list edit→unmount→viewer”의 8가지 과정으로 분류하였다.

표 7의 과정은 디지털 서버와 네트워크 모듈, 디지털 전자액자와의 통신 과정을 요약하여 진행된다. 따라서 본 논문에서 구현한 네트워크 모듈의 콘텐츠 다운로드 유용성을 검증하기 위하여 표 7을 기준으로 단일 콘텐츠 및 다중 콘텐츠 다운로드와 펌웨어 업데이트 성능 평가 등을 실험하였다.

각 콘텐츠는 동일한 네트워크에서 속도를 측정하여 커다란 차이를 보이고 있지 않다. 적용시간 그림은 단일 콘텐츠와 동시에 여러 콘텐츠의 전송 및 적용 시간을 나타내고 있지만 이들 콘텐츠 간의 시간 간격은 커다란 차이를 보이고 있지 않다. 하지만, 콘텐츠를 다운받기 위해 필요한 두 번째 폴링 시간이 첫 번째 폴링 시간보다 긴 것을 확인할 수 있으며, 콘텐츠의 양이 증가 할수록 폴링 대기 시간도 증가 하는 것을 확인하였다. 이는 한꺼번에 많은 콘텐츠를 확인하여 디지털 서버의 콘텐츠 대기 리스트를 확인 하는 과정이 상당히 오래 걸리는 것을 반증한다. 즉, 본 시스템의 특성상 하나의 콘텐츠를 다운받아 네트워크 모듈에 저장하고 저장된 콘텐츠를 다운받기 위해 디지털 서버에 재접속하고 다운 받은 콘텐츠의 플래그를 "N"에서 "Y"로 변경하기 위함이다. 따라서 두 번째 이상의 콘텐츠를 받는 시간 때문에 폴링 주기 시간이 늘어난다. 이러한 방식은 다운로드를 위한 폴링 시간은 길어지지만 디지털 전자액자의 메모리 블록을 줄일 수 있는 장점이 있다.

다운받은 이미지 콘텐츠를 재생하기 위하여 디지털 전자액자의 뷰어 모듈을 재 시작하여 적용하게 되는데 적용시간은 비슷하다. 따라서 그림에서의 이미지 콘텐츠 다운로드 시간은 폴링 주기와 디지털 전자액자의 메모리 블록으로 인하여 길어지지만 재생시간은 모두 비슷하다.

디지털 서버에서 디지털 전자액자로 소프트웨어 펌웨어를 다운받아 적용하는 시점은 다른 디지털 데이터와 동일

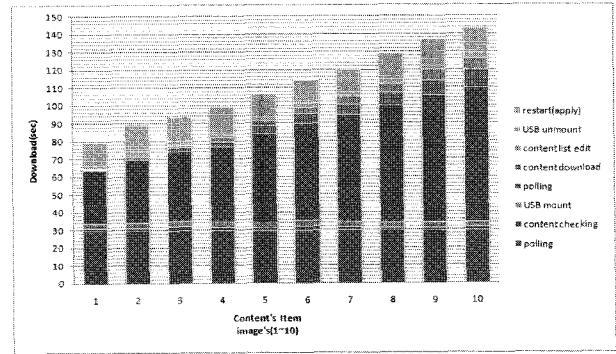


그림 13. 이미지 콘텐츠 다운로드 및 적용시간.

Fig. 13. Image content download and apply time the figure.

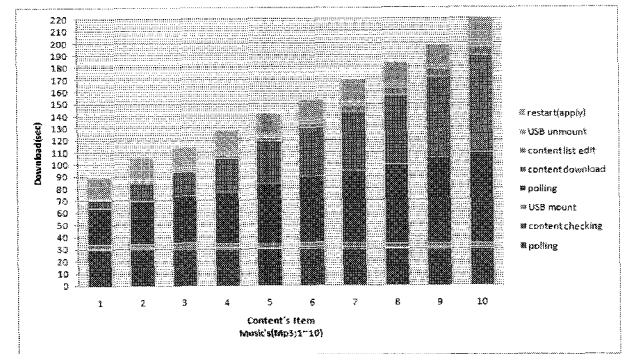


그림 14. 음악 콘텐츠 다운로드 및 적용시간.

Fig. 14. Music(MP3) content download and apply time the figure.

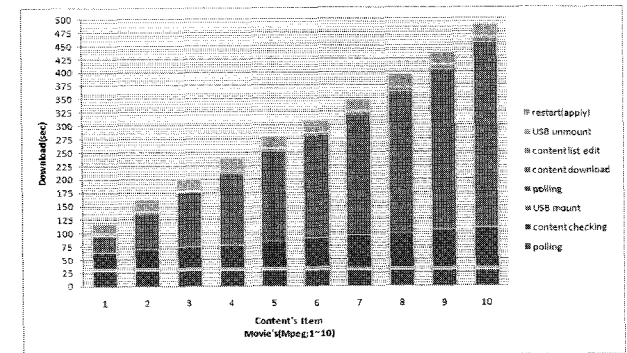


그림 15. 동영상 콘텐츠 다운로드 및 적용시간.

Fig. 15. Movie content download and apply time the figure.

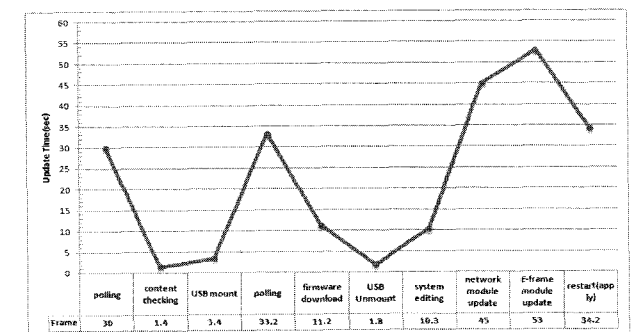


그림 16. 펌웨어 다운로드 및 업데이트 적용시간.

Fig. 16. Firmware download and update apply time the figure.

하나 콘텐츠 체크 시간과 다운로드 폴링 시간 그리고 시스템 재시작 적용시간은 현저하게 길다. 펌웨어는 각 모듈에 프로세서가 하나씩 있으며 따라서 운영 모듈이 각기 존재하기 때문에 네트워크 모듈 펌웨어와 디지털 전자액자 펌웨어 두 가지이다. 따라서 펌웨어 모듈의 적용시간이 긴 이유는 각 펌웨어를 적용시키고 재시작 하는데 걸리는 시간이 길다는 것을 보여 준다. 두 모듈의 펌웨어 적용시간이 이론적으로 약 22.35초 걸리지만 각 모듈은 동시에 펌웨어를 업데이트 하므로 약 178.5초 정도 소요되는 것을 그림 16에서 확인할 수 있다.

V. 결론

본 논문에서 기존 단순이 이미지 슬라이더로 사용하여 네트워크 기능이 없는 디지털 전자액자에 네트워크 모듈을 이용하여 새로운 디지털 데이터나 펌웨어를 보다 편리하게 업데이트 할 수 있도록 개선하였다. 현재의 전자액자는 이미지 슬라이드 쇼와 MP3 플레이, 동영상 플레이 기능과 간단한 테마기능이 전부이고 네트워크 기능이 없어 새로운 디지털 데이터를 저장 하고자 할 때 사용중인 메모리에 복사하여 사용하는 불편함이 있다. 이에 본 논문에서 제한한 네트워크 모듈을 이용하여 디지털 카메라 또는 휴대폰 카메라에서 촬영된 이미지나 동영상을 해당 디지털 서버에 업로드 하거나 개인 블로그의 이미지 앨범등과 연동하여 자신의 디지털 전자액자로 디지털 데이터를 전송 받을 수 있으며, 이러한 방법은 단순히 디지털 전자액자의 이미지 보기 기능에서 벗어나 기업에서의 업무 흐름도의 분석표기 기능 또는 상하관계 조직도 표시 기능, 광고 시장에서의 많은 광고비용으로 종이 인쇄에서 벗어난 디지털 전자액자를 이용하여 보다 역동적인 광고를 원거리에서 무선 네트워크를 통하여 제어 할 수 있으며, 광고 주체의 내용까지도 변경이 가능하다. 또한 기존의 네트워크 기능이 제외된 디지털 전자액자에 네트워크 모듈만을 추가 하여 새로운 디지털 전자액자를 다시 구입하지 않더라도 네트워크 디지털 전자액자의 모든 기능을 그대로 구현할 수 있는 장점이 있다. 향후 콘텐츠 다운로드에 소요되는 시간을 단축시키기 위한 방법으로 분산된 디지털 미디어 서버 운영 및 효율적인 네트워크 프로토콜 개선 방안 에 관한 연구를 수행할 계획이다.

참고문헌

[1] TiVo Inc, "TiVo home media optin: music and photos

server protocol specification," 2003.
 [2] T. B.-Lee, "RFC 1738, uniform resource locators (URL)," Xerox corporation, 1994.
 [3] K. Wittenburg, "A high performance multi-protocol network-based application for video beam diagnostics," 2000.
 [4] T. G. Lane, "Independent JPEG Group's software," www.ijg.org
 [5] T. Tojo and T. Enokido, "Fully distributed communication protocol for exchanging real-time multimedia data in a group," 17th AINA'03, p.427, 2003.
 [6] H.-S. Shim and T.-H. Kim, "Communication protocol for the URC robot and server," *NRC Symposium*, 2004.
 [7] K. D. Moore, "Digital frame averaging and Dark Mapping for a Video-Based Underwater Imaging Spectrometer System," *Appl. Opt.* 42, 4793-1801, 2003.
 [8] D. Lu and H. Sheng, "Effects and Implications of File Size/Service Time Correlation on Web Server Scheduling Policies," 13th IEEE International Symposium on Modeling SCTS, pp.258-270, 2005.
 [9] Yoko Onoe and Nagatsugu Yamanouchi, "Network Information Based Rate Controls on Multimedia Streaming Servers," *ICDCSW'03*, pp.543, May 2003.
 [10] 이상호, "OSGi기반 UPnP를 이용한 전자액자 서비스의 설계 및 구현," 한국정보과학회, 2006.
 [11] 이연정, "전자출판산업의 전략적 육성방안," 한국문화정책개발원, 2000.
 [12] 손영성, 김희정 외 2인, "P2P 네트워크에서 Agent 기법을 이용한 파일 전송 방법," 2004.
 [13] 손영성, 김희정 외 2인, "인터넷 환경을 고려한 P2P 파일 저장 프로토콜," *JCCI*, 2003.
 [14] 김희식, 남철 외 3인, "임베디드 리눅스 상에서 USB Digital Camera를 이용한 실시간 동영상 방송시스템 구현," 2004.
 [15] 김태현, 조진성 외, "확장가능한 다목적 주문형 멀티미디어 시스템의 설계 및 구현," 한국정보과학회 제 24 권 제 2 호, pp. 519-522, 1997, 10.
 [16] 최영도, 심재홍 외 2인, "멀티미디어 서비스를 위한 차세대 프로토콜의 요구사항에 관한 제안," 한국정보과학회 제 29 권 제 2 호, pp. 511-513, 2002, 10.



김 홍 규
 2004년 평택대학교 컴퓨터학과 졸업.
 2006년 수원대학교 컴퓨터학과 석사.
 2006년~현재 수원대학교 컴퓨터학과 박사과정. 관심분야는 센서네트워크 위치추적, 센서네트워크 헬스케어(BSN), 실시간 센서네트워크 운영체제, 실시간 데이터 베이스.

간 데이터 베이스.



문 승 진
 1976년 The University Of Texas At Austin 졸업. 1991년 The Florida State University 석사. 1997년 동 대학 박사. 1997년~현재 수원대학교 IT대학 학장. 관심분야는 실시간 멀티미디어 리눅스, 실시간 모바일 데이터베이스, 실시간 임베디드 시스템, 실시간 센서 네트워크 시스템.

임베디드 시스템, 실시간 센서 네트워크 시스템.