

갤럭시노트 디바이스를 이용한 무선 태블릿 시스템의 구현

윤동준*, 최병윤*

*동의대학교, 컴퓨터공학과

The implementation of the wireless tablet system using GalaxyNote device

Dong-June Yoon*, Byeong-Yoon Choi*

*Dept. of Computer Engineering, Dong-Eui University

E-mail : ydj414@naver.com

요 약

본 논문에서는 갤럭시노트를 사용하여 PC에 대한 무선 태블릿 장치의 효율적으로 구현하는 방안을 제시하였다. 무선 태블릿 장치는 갤럭시 노트, 스타일러스 펜과 블루투스 신호를 직렬 통신신호로 변환하는 장치로 구성된다. 무선 태블릿 장치는 스타일러스 펜이 가리키는 좌표정보를 PC로 전송하기 위해 블루투스 통신을 사용한다. 자체 개발한 윈도우 마우스 필터 드라이버는 전송받은 좌표를 X, Y값으로 분리한 후에 변환된 좌표값으로 윈도우 응용 프로그램이 동작하는 동안 마우스 포인터의 위치를 제어한다.

ABSTRACT

In this paper efficient design of wireless tablet system using GalaxyNote device for PC was proposed. The designed portable tablet consists of GalaxyNote device, Stylus Pen, and Bluetooth-to-serial converter. To transmit coordinate information of Stylus Pen on GalaxyNote device to PC, wireless portable tablet uses bluetooth wireless communications. After the custom mouse filter driver divides received coordinate into x-coordinate and y-coordinate, it controls position of mouse pointer using the converted coordinates while Windows application programs are running.

키워드

드라이버, 블루투스, 시리얼통신

1. 서 론

태블릿은 PC환경의 발달로 인해 그래픽 콘텐츠를 포함하고 있는 상용 프로그램에 있어 급부상하고 있는 장치이다.

태블릿은 사용자가 글이나 그림을 쓰게 되면 그것을 화면에 표현이 가능한 장치인데, 이러한 입력 장치가 수없이 개발되고 있다.

일반적으로 특정시스템에서 전문적인 사용목적에 가진 사용자들만 사용되었으나, 태블릿의 기능을 그대로 사용하면서 휴대가 간편화 하고자 하는 시도가 이루어지고 있다. 그 시도 중 하나로 태블릿의 무선화시도를 예로 들 수가 있고, 또 하나로는 태블릿의 스마트폰의 이식을 예로 들 수가 있다.

삼성전자의 갤럭시노트 시리즈가 태블릿의 스마트폰 이식화의 대표적인 예인데, 스마트폰의 장점인 휴대성에 필압 식별이 가능한 펜을 장착함으로써 단순히 스마트폰에 그치지 않고 스마트폰과 태블릿 사이의 영역을 개척함에 의의가 있다고 하겠다. 하지만 태블릿이 전문적인 사용자에게 필요성이 많은 입력장치이므로 일반적인 사용자가 태블릿이 필요한 상황이 왔을 때 고가의 기기를 사서 쓸 수 없는 상황이다[1].

본 논문은 사용자의 시중에 널리 보급되어있는 갤럭시노트 디바이스에 이식되어 있는 태블릿 시스템의 활용방안으로써 단순히 휴대폰에 그치지보다 블루투스통신과 윈도우 디바이스 드라이버를 통해 PC에서 간단한 태블릿 기능을 구현 해 보고자 한다.

II. 무선 태블릿 시스템의 구성과 윈도우 디바이스 드라이버의 구성

기본적으로 디바이스의 좌표를 전송받아 PC에서 마우스를 제어하기 위해 첫 번째로 좌표값의 통신 방법을 정해서 그 좌표값을 토대로 마우스 포인터를 제어하는 것인데, 본 논문에서는 블루투스 통신을 통하여 좌표값을 받은뒤에 그 좌표값을 토대로 필터드라이버로 전송한다. 구현된 시스템은 운영체제로 후킹된 마우스의 스캔코드를 수정하여, 운영체제로 다시 돌려주게 된다. 본 장에서는 디바이스의 좌표의 추출법과, 드라이버스택, 데이터구조에 대하여 설명한다.

2.1 안드로이드 디바이스상의 좌표추출

갤럭시노트 디바이스는 기본적으로 S노트라는 필기 어플이 탑재되어 출시된다. 이 어플리케이션에 사용된 API는 SPenSDK를 이용하여 제작되었는데 제작사에서 무료로 배포하고 있다. SPenSDK 사용하여 전용 어플리케이션을 제작한다.

SDK의 이벤트 처리기 중 TouchListener와 HoverListener가 있다. TouchListener를 이용하여 터치상태일 때(PC에서는 마우스 클릭상태)의 이벤트를 처리하게 되고, HoverListener를 이용하여 스타일러스펜(SPen)이 일정거리 떨어져 이동을 할 때의 이벤트를 처리하게 된다.

이 두 이벤트 처리 함수를 통해 현재 SPen의 좌표상태를 캐치하게되고 후에 블루투스 통신으로 이 좌표값을 전송을 하게 된다[2].

2.2 블루투스통신과 시리얼 통신

무선으로 구현하기 위해 블루투스통신을 사용한다 [7]. 블루투스 통신으로 전달받은 데이터를 PC로 전송하기 위해 시리얼 통신을 이용하여 데이터를 전송한다.

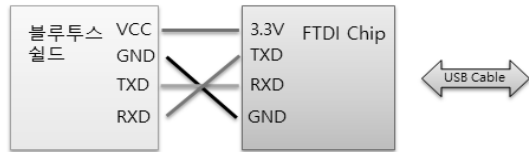


그림 1. 블루투스와 시리얼 통신 변환 하드웨어

어플리케이션상에서 블루투스를 켜진 상태에서 블루투스 모듈에 접속하여 실시간 좌표값을 바이트로 변환하여 계속 보내준다. 보낸 데이터를 FTDI(usb to serial)하드웨어를 이용하여 데이터를 처리할 PC로 전송을 해야 하는데 본 논문에서는 시리얼 통신을 이용하여 데이터를 전송하였다. 그림 1과 같이 출력신호(TX)와 입력신호(RX)를 서로 연결해줌으로

써 통신을 한다[3].

2.3 윈도우 응용 어플리케이션

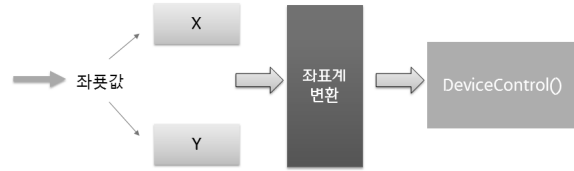


그림 2. 응용 어플리케이션에서의 좌표 처리

그림 2는 응용 어플리케이션에서 통신을 통해 받은 좌표값을 처리하는 과정이다. 먼저 전송받은 좌표값을 X축의 값과 Y축의 값으로 정확하게 분리해낸다. 그 뒤 마우스 드라이버의 좌표계로 변환을 해야 하는데, 마우스 드라이버의 절대 좌표계는 항상 65535 * 65535의 똑같은 값을 가진다.

이는 해상도에 상관없이 함수를 사용하기 위해서이다. 이 같은 사실을 단서로 삼아 모바일상의 좌표를 현재 사용하고 있는 모니터의 크기에 상관없이 비율에 맞게 변환을 해준다[4].

2.3 마우스 필터 드라이버

윈도우 응용프로그램에서 파싱과 좌표 변환을 거친 좌표는 마우스 필터드라이버를 통해 제어해야 한다. 이 때 응용프로그램은 드라이버에 적절한 I/O 명령을 통해 마우스를 제어해야 하는데 이를 제어하고 구분하는 신호를 IRP를 통해서 한다. IRP란 어플리케이션이 드라이버로 보내는 읽기, 쓰기, 생성등과 같은 I/O 관련 요청을 전달하는 역할이다.

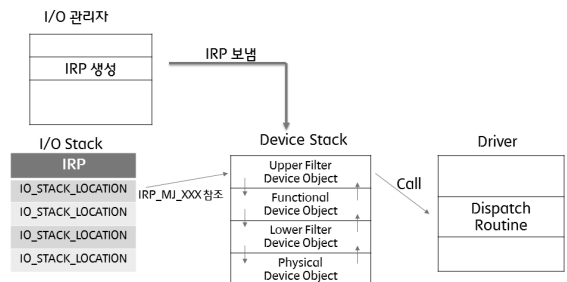


그림 3. IRP 생성과 전달 과정 [5]

윈도우는 커널 - 응용프로그램, 커널 - 커널 등 I/O 통신을 위해서 IRP라는 거대한 구조체를 가지고 관리한다.

그림 3과 같이 특정 디바이스에게 IRP를 보내면 그에 맞는 IO_STACK_LOCATION 구조체 멤버를 참조해 드라이버에서 등록된 Dispatch 루틴을 호출하게 된다.

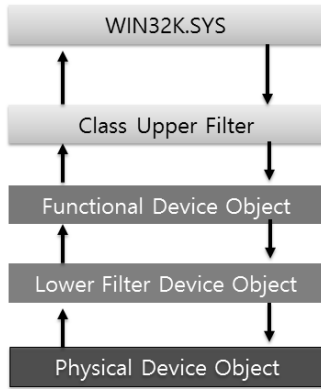


그림 4. IRP 흐름도와 디바이스 스택

응용 프로그램 상에서 전달하는 IRP는 Class Upper Filter를 거쳐 상태를 반환하게 되는데 마우스 필터 드라이버의 경우 IRP_MJ_READ를 받게 된다. 그러면 Class Upper Filter는 MOUSE_INPUT_DATA 형태로 작성된 구조체로 응답한다.

이 구조체는 마우스의 버튼 상태와 좌표상태를 담고 있다. 따라서 WIN32K.SYS는 마우스 상태정보를 얻을 때 마다 IRP_MJ_READ 신호를 보내게 된다. Class Upper Filter 디바이스 스택 부분에 마우스 필터 드라이버를 포함시켜 마우스 상태정보를 감시할 수 있도록 한다.

```

#define IOCTL_MOUSE_LDOWN
CTL_CODE(FILE_DEVICE_UNKNOWN, 0x801,
METHOD_OUT_DIRECT, FILE_ANY_ACCESS)

#define IOCTL_MOUSE_LUP
CTL_CODE(FILE_DEVICE_UNKNOWN, 0x802,
METHOD_OUT_DIRECT, FILE_ANY_ACCESS)

#define IOCTL_MOUSE_RDOWN
CTL_CODE(FILE_DEVICE_UNKNOWN, 0x803,
METHOD_OUT_DIRECT, FILE_ANY_ACCESS)

#define IOCTL_MOUSE_RUP
CTL_CODE(FILE_DEVICE_UNKNOWN, 0x804,
METHOD_OUT_DIRECT, FILE_ANY_ACCESS)

#define IOCTL_MOUSE_MOVE_X
CTL_CODE(FILE_DEVICE_UNKNOWN, 0x805,
METHOD_OUT_DIRECT, FILE_ANY_ACCESS)

#define IOCTL_MOUSE_MOVE_Y
CTL_CODE(FILE_DEVICE_UNKNOWN, 0x806,
METHOD_OUT_DIRECT, FILE_ANY_ACCESS)
    
```

그림 5. 드라이버를 사용하기 위한 매크로

그림 5는 필터드라이버의 동작을 정의하는 매크로 함수이다. 각 동작의 버퍼방식과 파일접근방식을 정의한다. IOCTL CODE를 이용하여 각 동작을 구

분하여 마우스를 제어하게 된다.[6][7]

```

struct DEVEXT
{
PDEVICE_OBJECT pNextDevObj;

USHORT Flags;
USHORT ButtonFlags;
USHORT ButtonData;

LONG LastX;
LONG LastY;

LONG OldX;
LONG OldY;
USHORT OldButtonFlags;
};
    
```

그림 6. 마우스 필터 드라이버 구조체

그림 6은 마우스 필터 드라이버를 사용하기 위한 구조체이다. 각 버튼의 상태와 버튼 데이터, 마우스의 좌표 등의 정보를 갖는 구조체이다.

이 외에도 필터드라이버를 설정하기 위한 DriverEntry, ReadHandler, PowerHandler, PnPHandler, AddDeviceHandler을 이용하여 구성한다[8][9][10].

III. 실험

안드로이드 디바이스 상에서 넘어오는 좌표값을 드라이버로 넘겨주어 마우스를 제어하기 위해서 DeviceIoControl() 함수를 사용하였다.

이 함수는 ReadFile()함수와 WriteFile()함수가 같이 데이터를 처리하는 함수인데, 각각 2개의 버퍼를 제공하는 특징을 가지고 있다.

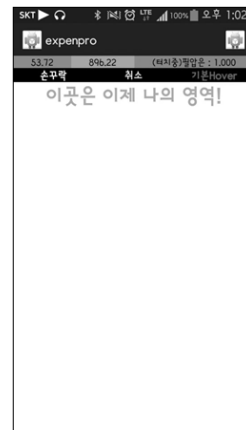


그림 7. 안드로이드 어플리케이션의 모습

그림 7은 안드로이드 어플리케이션의 모습이다. 화면 우측상단 액션바 부분의 버튼을 누르게 되면 블루투스 쉘드와 접속이 된다.

하단의 부분은 SPenSDK의 SCanvas로 이 영역에서 SPen을 touch하거나 hovering하게 되면 이벤트가 발생한다. 현재 상단에 이벤트가 발생할 때의 좌표상태와 모드정보를 나타내주고 있다.

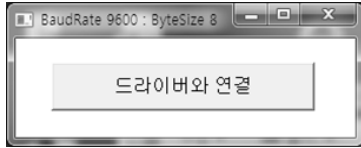


그림 8. 윈도우 응용어플리케이션

그림8은 드라이버와 디바이스간을 연결해주는 역할을 하는 윈도우 응용어플리케이션의 모습이다. 버튼을 누르게 되면 드라이버와 연결이 되어서 기능을 사용할 수 있는 상태가 된다.



그림 9. 갤럭시 노트 디바이로 필기 중인 모습

그림 9와 같이 갤럭시노트 디바이스 상에 필기를 하면 PC상에서도 필기가 가능하며, Hovering중일 때에는 마우스 이동이 가능하다.

IV. 결 론

본 논문에서는 갤럭시노트 디바이스에서상에서 만 가능한 태블릿 기능을 PC로 확장하여 사용할 수 있는 기능을 구현하였다. 이 시스템을 활용하여 일반 사용자들도 언제든지 소지하고 있는 갤럭시노트 디바이스를 활용하여 태블릿이 필요한 상황 또는 시급한 작업에 고가의 태블릿이 없이도 충분히 작업이 가능한 휴대용 태블릿으로 활용할 수 있다.

참고문헌

- [1] 홍동구, 유영기, “유선 태블릿의 좌표 측정 방법 연구”, 정보 및 제어 학술대회(CICS’ 08) 논문집, 2008, pp 438
- [2] <http://developer.samsung.com/s-pen-sdk/technical-docs/S-Pen-SDK-2-3-Tutorial>
- [3] http://www.sciencesoftware.co.kr/TALtech/Tutorials/intro_sc/intro_sc.htm
- [4] IT Expert 윈도우 디바이스 드라이버, 한빛미디어, 이봉석, 이재홍, 2009
- [5] <http://ezbeat.tistory.com/311>
- [6] 윈도우즈 API 정복, 한빛미디어, 김상형, 2011
- [7] 노우진, 김동현, 이동우, “키보드 필터 드라이버를 이용한 동적 기능키 구현”, 한국정보기술학회 하계학술대회 논문집, 2010, pp 280 ~ 283
- [8] WDF 윈도우를 위한 차세대 통합 드라이버 개발 모델, 에이콘, PENNY ORWICK, Guy Smith, CAROL BUCHMILLER, ANNIE PEARSON, 2008
- [9] SANGWOO HAN, SANGEUN HAN, “Recovering Device Drivers in Thread Context”, International Conference on Control, Automation and Systems, 2008
- [10] Young-Ohk Song, Duk-Jin Chang, Sung-Young Kim “The Proposal of the Driver Model for an efficient cognition of devices at TinyOS”, Scientific and Technical Information 2008