

논문 2009-4-6

# 모바일 애드-혹 망에서 Bluetooth 기반 MMORPG의 설계

## Design of Bluetooth based MMORPG Game in MANETs

오선진\*

Sun-Jin Oh

요 약 최근 무선 모바일 컴퓨팅 응용 기술과 휴대용 모바일 단말장치 개발 기술의 급속한 발전과 더불어, 요즈음 가장 각광을 받는 이 분야의 이슈는 무선 모바일 애드 혹 네트워크 환경에서의 온라인 게임의 설계에 관한 것이다. 모바일 컴퓨팅 환경에서의 온라인 게임은 이동 단말들이 갖는 제약들: 즉, 낮은 성능의 프로세서, 극히 제한적인 메모리 공간, 무선 기반의 적은 통신 대역폭과 한정된 배터리 파워 등의 제한으로 인해 본격 온라인 게임 개발에 많은 제약이 따른다. 따라서 지금까지의 대부분의 모바일 게임들은 온라인이나 멀티플레이 기능에 매우 제한적이다. 본 논문에서는 많은 제약을 갖는 이동 단말을 기반으로 모바일 컴퓨팅 환경에서 다수의 사용자가 멀티플레이가 가능한 온라인 MMORPG를 설계하고 구현하였다. 제안한 온라인 게임은 Bluetooth를 이용하여 국부적으로 무선 모바일 애드 혹 네트워크를 클라이언트들 간에 일시적으로 구축하고 MMORPG 게임을 온라인으로 수행할 수 있도록 설계하였으며, 또한 이들 간의 멀티플레이를 지원한다.

**Abstract** With the rapid growth of recent wireless mobile computing application technology and handheld mobile terminal device development technology, one of the big issue in these fields is to design online games in wireless mobile ad hoc network environment. Online games in mobile computing environment have lots of constraints for developing online games because mobile terminals have many limitations such as low performance of processor, limited memory space, small bandwidth of wireless communication, and confined life of battery power. Therefore, most of mobile games are restrictive in the function of online and multi-play up to date. In this paper, the online MMORPG game, capable of multi-play with many other mobile users using mobile terminals in wireless mobile computing environment, is designed and implemented. Proposed mobile online game uses bluetooth to construct temporary wireless mobile ad hoc network with other mobile clients, and designed to carry out online MMORPG game with these clients. It also supports multi-play among them.

**Key Words :** MANETs, MMORPG Game, Bluetooth

### I. 서 론

무선 인터넷과 휴대용 단말장치 기술의 급속한 발전과 더불어, 모바일 컴퓨팅 환경에서 시간과 장소의 구애 없이 자유롭게 이동하면서 무선으로 서로 접속하여 통신

하는 유비쿼터스 환경을 위한 다양한 응용들이 속속 제안되고 있다. 또한 기존의 인터넷과 같은 고정된 유선의 인프라 구조를 이용하지 않고 무선을 기반으로 하는 일시적인 통신망을 이용한 다양한 응용들에 대한 연구가 절실히 요구되고 있다. 모바일 애드 혹 망 (MANETs)은 고정된 인프라 구조가 없는 무선 기반 환경에서 모바일 노드들로 구성된 자기 조직화 네트워크를 말하며, 이렇

\*정회원, 세종대학교 정보통신학부  
접수일자 2009.6.23, 수정완료 2009.8.3.

게 형성된 망은 모바일 노드들의 이동성에 따라 망의 토폴로지가 유동적이며, 쉽게 단절될 수 있는 취약성을 내포하고 있어, 모바일 애드 혹 망을 기반으로 하는 응용에 대한 개발에 많은 어려움이 있다. 지금까지의 이에 관한 연구는 이동성을 가진 모바일 호스트들 간에 높은 통신 효율성을 제공하기 위한 라우팅 프로토콜과 모바일 호스트의 위치 기반 서비스에 관한 연구가 주류를 이루고 있다. 하지만 최근에는 모바일 애드 혹 망 환경에 적합한 응용모델에 대한 관심이 높아지면서 모바일 온라인 게임 등 다양한 응용에 대한 연구가 활발하게 진행되고 있다. 그러나 이러한 모바일 응용을 살펴보면 지금까지는 간단한 오피스 관련 프로그램에서부터 단순한 개인 정보관리, 인터넷 접속과 브라우징, 이메일 검색, 그리고 혼자서 즐기는 싱글 게임 수준의 응용들이 주류를 이루고 있으며, 이들은 대체로 단순하고 너무 일률 단편적이어서 모바일 응용 마니아들의 욕구를 충족시키기에는 절대 부족한 실정이다.<sup>[1]</sup>

본 논문에서는 이러한 최신 정보 인프라와 더불어, 자유롭게 이동하면서 시간과 장소에 구애 없이 언제 어디서나 컴퓨팅 할 수 있는 유비쿼터스 환경에서의 Bluetooth를 이용한 무선 모바일 애드 혹 망을 기반으로 클라이언트들 간의 멀티플레이가 가능한 온라인 MMORPG를 설계하고 구현하였다. 본 논문에서 설계하고 구현한 온라인 MMORPG 게임은 이동 망에서의 대표적인 이동 단말인 PDA를 기반으로, Bluetooth를 이용한 일시적인 국부 무선 애드 혹 망을 구축하고, 온라인 상태에서 다수의 게임자가 PDA를 통해 실시간으로 게임을 할 수 있도록 설계하였으며, 이들 클라이언트들 사이에서 무선으로 게임 정보들이 실시간으로 송수신 되도록 구현하였다.

본 논문의 구성은 다음과 같다. 2장에서는 MMORPG를 위한 시스템 모델을 소개하고, 3장에서는 Bluetooth 기반 온라인 MMORPG 롤플레이 게임의 기본 구조와 구현 알고리즘을 서술하였으며, 4장에서는 본 논문에서 구현한 MMORPG의 구현 결과와 실행 화면 등을 소개하였고, 마지막으로 5장에서 결론을 맺는다.

## II. 시스템 모델

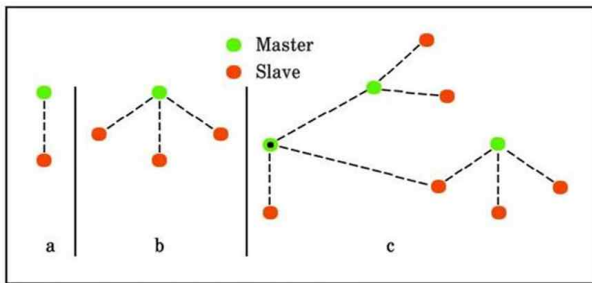
모바일 컴퓨팅 환경에서의 PDA나 모바일 폰과 같은

이동 단말장치의 프로세스 처리 능력은 데스크 탑 컴퓨터에 비해 매우 성능이 떨어지고, 저장 장치 용량 역시도 매우 제한적이다. 또한 이동성을 보장하기 위해 전원을 배터리로 사용하기 때문에 이동 단말장치의 전원 공급에 대한 많은 제약이 따르며, 통신은 무선을 기반으로 이루어지므로 무선 통신의 낮은 대역폭을 이용하여 데이터를 송수신해야 하기 때문에 전송속도가 떨어지고 전송 데이터의 양이 제한적이며 또한 데이터에 대한 신뢰성도 낮다. 따라서 모바일 컴퓨팅 환경에서의 모바일 게임의 설계는 이러한 이동 단말이 갖는 제약과 무선 통신으로 인한 비효율성을 고려해야만 한다.<sup>[2,3]</sup>

Bluetooth를 기반으로 클라이언트들 간의 일시적인 모바일 애드 혹 망을 구축하고, 온라인 게임이 실행 운용되며, 중간 중간의 클라이언트들에 의해 실시간으로 수행되는 게임 정보들이 서로 무선통신으로 교환되는데 이때 Bluetooth를 사용한다. Bluetooth는 0.5평방인치 이내로 작고, 5 달러 정도의 저렴한 가격, 100mW의 적은 전력 소모로 모바일 폰, 휴대용 PC 등과 같은 휴대 단말장치, 네트워크 액세스 포인트, 기타 주변장치들 간의 100미터 이내의 작은 구역 내의 2.4GHz 대역에서의 무선 연결을 위한 기술적인 규격 사양이다.<sup>[4]</sup> Bluetooth의 기원은 1994년 에릭슨의 이동통신 그룹 모바일 폰과 주변 기기들 간의 소비전력이 적고 가격이 싼 무선 인터페이스를 연구하기 시작해서. 마침내 1998년 5월 에릭슨, 노키아, IBM, 도시바, 인텔로 구성된 Bluetooth SIG가 발족되면서 시작되었고, 현재 모토로라, 마이크로소프트, 루슨트 테크놀로지, 3COM 등 4개사가 가세해 확보부동한 세계적 규격으로 자리 잡고 있다. Bluetooth 기술은 근거리 내에서 하나의 무선 연결을 통해서 장치들 간에 필요한 여러 케이블 연결을 대신하게 해준다. 블루투스 무선 기술은 기존의 데이터 망과 주변 장치들 간의 인터페이스, 그리고 고정된 네트워크 하부 구조로부터 멀리 떨어진 장치들 간에 특별한 그룹을 형성시켜주는 보편적인 다리 역할을 제공한다. 잡음이 많은 라디오 주파수 환경에서 작동하도록 고안되었기 때문에, 블루투스는 빠른 인식과 주파수 호핑 방식을 사용하여 연결을 튼튼하게 한다. 블루투스 모듈은 패킷을 전송받거나 보낸 후에 새로운 주파수 호핑을 함으로써 다른 신호들과의 간섭을 피하며, 같은 주파수대에서 작동하는 다른 시스템들과 비교하여 블루투스는 특별히 빠르고 짧은 패킷을 사용한다.<sup>[5]</sup>

블루투스 무선시스템은 사용자가 필요치 않은 2.4GHz의 ISM 주파수대에서 작동한다. 주파수 호핑 송수신기는 간섭과 페이딩에 저항하도록 고안되었으며, 이진 FM 변조 방식은 송수신기의 복잡함을 최소화하도록 고안되었다. 최대 데이터 전송속도는 1Mb/s이고, 폴 듀플렉스 전송을 위해서는 시간분할 다중 방식 (Time-Division Duplex scheme)이 사용된다.<sup>[6]</sup>

블루투스 시스템은 일-대-일과 일-대-다의 연결을 지원한다. 여러 개의 피코 넷들이 함께 조직되고 연결될 수 있으며, 각각의 피코 넷들은 서로 다른 주파수 호핑 순서에 의해 구분된다. 같은 피코 넷에 속한 모든 사용자들은 동일한 호핑 순서로 동기화 된다. 위상은 하나의 다중 피코 넷 구조로 묘사되는 것이 가장 적절하다. 열 개의 완전히 적재된 독립된 피코 넷들을 가진 하나의 다중 피코 넷 구조 안에서 폴 듀플렉스 데이터 전송률은 6Mb/s 이상이다.<sup>[6,7]</sup> 그림 1은 Bluetooth의 링크 구조를 싱글 슬레이브 피코 넷, 멀티 슬레이브 피코 넷, 그리고 스캐터 넷 등으로 확장되는 모습을 보여 준다.



▲ (a) 싱글 슬레이브 피코넷 (b) 멀티 슬레이브 피코넷 (c) 스캐터넷

그림 1 Bluetooth 링크 구조  
Fig. 1. Bluetooth Link Structure

표 1은 본 논문에서 제안한 모바일 컴퓨팅 환경에서의 Bluetooth기반 온라인 MMORPG 구현을 위한 시스템 모델을 보여준다.

표 1. 시스템 모델  
Table 1. System Model

온라인 MMORPG 게임	
개발 언어	Windows CE API와 Game API
개발 Tool	Embedded Visual Studio 4.0
Client 단말 사양	HP iPaq hx4700 PDA
운영체제	Microsoft Windows Mobile 2003 - Second Edition

Pocket PC Professional 한글판		
프로세서	400MHz Samsung S3C2440	
디스플레이	컬러 수	16-bit 65,536 컬러
	터치스크린	지원
	해상도	240 X 320
	크기	3.5 inch (96mm)
메모리	SDRAM	128MB 플래쉬 RAM (iPAQ File Store 83MB사용가능)

표에서와 같이 온라인 게임을 하는 클라이언트들은 Bluetooth가 장착된 PDA를 사용하여 국부적으로 모바일 애드 혹 망을 구축하도록 하였고, 이들 클라이언트들 중 한 명이 온라인 게임의 주 모듈을 구동하게 하여, 다른 클라이언트들이 이 주 모듈에 접속하여 인증을 받은 후 주 모듈로부터 게임 정보를 다운로드 받아 MMORPG 애드 혹 망이 일시적으로 구성되도록 하였다. 이때 모든 클라이언트들은 게임 배경 지도, 캐릭터와 몬스터 정보 등 동일한 MMORPG의 기본 정보를 가진 게임엔진이 설치되어 있고, 주 모듈을 구동하는 클라이언트는 온라인 게임의 주 서버 역할을 하며, 전반적인 게임의 제어정보나 동작 메시지를 제어하거나 중계하는 역할을 담당하고, 전체적인 온라인 게임의 수행과 관리를 담당하게 된다. 게임과 관련된 정보나 메시지들은 Bluetooth를 통해 각 클라이언트들에게 실시간으로 전달되어 운영된다.<sup>[8,9]</sup> Bluetooth를 통해 클라이언트들 간의 데이터 전송을 최소화하기 위해 모든 게임의 제어권과 운영권을 주 모듈을 가진 클라이언트가 가지며, 게임이 진행되는 동안 발생하는 게임 정보의 변동이 있을 때만 일정 시간 간격으로 패킷을 구성하여 Bluetooth를 통해 전송하게 된다. 이때 발생하는 게임 정보는 주 모듈을 가진 클라이언트가 일괄 저장 관리 하며, 다른 클라이언트들은 현재의 자신에 대한 위치 정보만을 저장 관리하게 하여 저장 장치의 사용과 전송 데이터의 양을 최소화 한다.<sup>[10,11]</sup>

각 클라이언트들의 소프트웨어 플랫폼은 컴포넌트를 기반으로 Windows CE를 포팅한 모바일 장치로 확장하여 시간 활용도를 극대화하는 PDA 포켓 PC를 사용하였고, Bluetooth를 통해 무선 환경에서 쉽게 접속할 수 있도록 설계하였으며, 클라이언트들에는 게임을 위한 게임엔진 어플리케이션과 Pocket Internet Explorer를 통해 무선으로 다른 클라이언트들과 접속할 수 있도록 설계하였

다.<sup>[12,13]</sup> 본 논문에서 제안한 온라인 MMORPG의 구현은 Embedded Visual Studio 4.0을 이용하여 Windows CE API와 Game API를 사용하여 PC상에서 구현하여 PDA에 포팅하였다. 그림 2는 본 논문에서 제안한 Bluetooth 기반 온라인 MMORPG의 시스템 구성도를 보여준다.

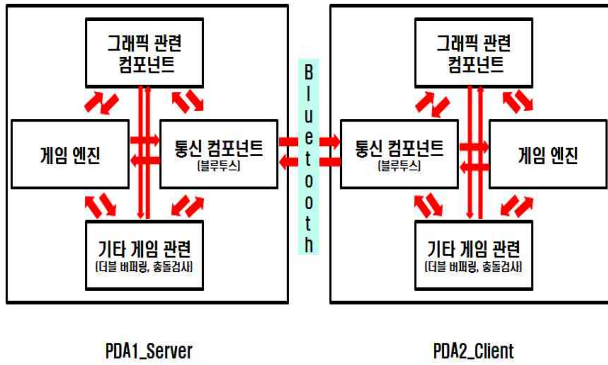


그림 2 시스템 구성도  
Fig. 2. System Structure

며, 수신 패킷에 대한 정보는 Agent에서 해독하여 해당 작업에 넘겨주게 된다. 게임엔진 컴포넌트는 캐릭터나 몬스터의 이동 방향을 결정하고 이동 위치의 변화와 몬스터 공격 등의 게임 정보부분 등 게임 전반의 동작을 제어하고 처리하는 주요 부분이다. 그림 3은 게임엔진의 생성 주기를 보여준다.

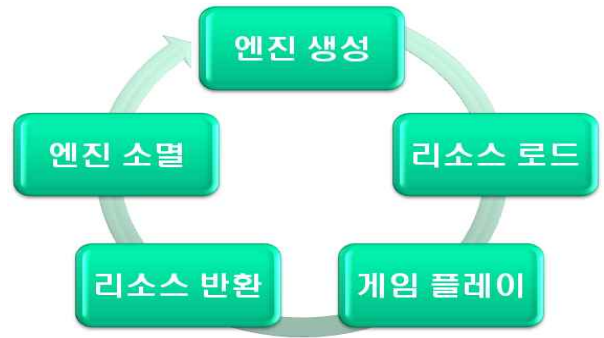


그림 3 게임엔진의 생명주기  
Fig. 3. Life Cycle of Game Engine

### III. 온라인 MMORPG 설계

#### 1. MMORPG 컴포넌트 구성

본 논문에서 제안하는 온라인 게임 “MMORPG”에 대한 정의는 “Massively Multi-Payer Online Role Playing Game”으로 다수의 클라이언트들이 동시에 게임 할 수 있는 리니지1이나 바람의 나라와 같은 대형 롤플레이팅 게임을 말하며, RPG와 TRPG에 MUD와 MUG의 개념을 더한 것이다. 기존의 롤플레이팅 게임과는 달리 게임에서 제공하는 그래픽은 2D에서 3D로 전환된 본격 온라인 롤플레이팅 게임이다. 제안한 MMORPG의 각 클라이언트들이 갖는 주요 컴포넌트를 살펴보면 그림 2에서와 같이 그래픽 처리 관련 컴포넌트, 소켓 처리 관련 통신 컴포넌트, 이동/공격 등 게임 전반의 동작을 관리하는 게임엔진, 그리고 더블 버퍼링과 충돌검사 등을 수행하는 기타 게임 관련 컴포넌트 등으로 구성된다. 그래픽 처리 컴포넌트는 캐릭터나 몬스터 그리고 공격 등의 그래픽 부분을 담당하는 곳으로 여기엔 스프라이트 및 각 그림을 프레임 별로 순차적으로 보여 연속된 동작으로 보이게 하는 부분이 있다. 통신 컴포넌트의 Bluetooth 소켓은 게임서버로 패킷을 보내거나 또는 각 클라이언트들이 게임서버로부터 패킷을 받는 작업을 하는 부분으로 이루어져 있

그림에서 보는 바와 같이 게임엔진은 엔진 생성-리소스 로드-게임 플레이-리소스 반환-엔진 소멸 등의 생명 주기를 반복하며 주기적으로 게임이 이루어진다. 이 때 발생하는 게임 정보는 변경된 부분만 패킷으로 구성되어 주기적으로 주 모듈이 동작하는 서버의 통신 컴포넌트를 통해 전송된다. 기타 게임관련 컴포넌트는 이들 컴포넌트들을 관리하여 상호 유기적으로 동작하게 하는 부분으로, 특히 게임의 원활한 진행과 화면 깜빡임 등의 문제가 발생 될 수 있는데 이를 제어하고 관리하는 일을 수행하는 곳으로 클라이언트의 총체적인 관리를 담당한다. 무선통신 과정에서 클라이언트 부분은 소켓의 제한으로 인해 블로킹 모드로 구현해야 하고, 이로 인해 게임 진행이 지연되는 문제가 발생하게 되는데, 이 문제를 극복하기 위해 본 논문에서는 송수신 소켓 프로세스를 스레드로 나누고 시간에 따라 강제로 종료하고 재시작하게 하는 방법으로 문제를 해결하였다. 또한 Windows CE 개발 환경에서의 repaint 문제로 인해 게임이 진행되는 동안 화면 깜빡임 문제가 발생되는데 이 문제를 해결하기 위해 배경이미지와 캐릭터를 분리한 더블 버퍼링 기법을 통해 이 문제를 해결하였다.

```

// StylusDown() PDA 단말의 스크린을 터치하는 경우 좌표값 반환
void StylusDown(int X, int Y)
{
    switch(Game_Mode)
    {
        case Game_Title:
            // 게임시작을 클릭하였을 경우
            if(RectStart.left <= X && RectStart.right >= X)
            {
                if(RectStart.top <= Y && RectStart.bottom >= Y)
                {
                    ResetGame();
                }
            }
            break;
    }
}

```

그림 4. 스크린 위치 반환 알고리즘  
Fig. 4. Screen Location Aware Algorithm

## 2. 구현 알고리즘

본 논문에서 제안한 온라인 MMORPG의 주요 알고리즘은 다음과 같다. 그림 4는 이동 단말인 PDA 스크린을 스타일러스 펜으로 터치했을 때 스크린 상의 위치 좌표값을 반환하는 알고리즘을 보여준다. 이것으로 캐릭터의 이동 방향을 설정하고 몬스터에 대한 공격을 구현할 수 있다. 그림 5는 MMORPG에 등장하는 캐릭터들의 움직임을 구현하는 Animate() 알고리즘으로 캐릭터의 움직임과 이동을 표현하기 위해 우선 캐릭터의 이전 위치의 비트맵을 제거하고, 캐릭터를 새로운 위치로 이동시킨 다음, 그 캐릭터가 있던 배경 이미지를 저장하고, 이들을 비트맵 이중 버퍼에 출력하여서 게임 내의 캐릭터들의 이동과 움직임을 표현한다. 이때 배경 이미지에 캐릭터의 이미지로 인해 색의 차이가 나타나는 문제가 발생하는데 이 문제를 해결하기 위해 본 논문에서는 TransBlit() API 함수를 사용하여 캐릭터나 몬스터가 갖는 배경색을 투명하게 처리하여 캐릭터가 위치한 곳의 배경 이미지를 투과시켜 색 차이를 해소한다.

```

// Animate() 캐릭터의 움직임을 구현
void Animate(HDC hdc)
{
    static BOOL bFirst = TRUE;
    int n;
    if(bFirst)
    { // 이미지 저장
        csChar->SaveUnder(hdc);
        csMonster[0]->SaveUnder(hdc);
        csMonster[1]->SaveUnder(hdc);
        csMonster[2]->SaveUnder(hdc);
        bFirst = FALSE;
    }
    // 이미지 복구
    csChar->RestoreUnder(hdc);
    for(n=0; n<3; n++)
    {
        csMonster[n]->RestoreUnder(hdc);
    }
}

```

그림 5. 캐릭터 동작 구현 알고리즘  
Fig. 5. Character Animation Algorithm

그림 6은 주 모듈을 갖는 클라이언트 서버가 블루투스를 이용하여 소켓을 주고받는 통신 알고리즘을 보여준다. 클라이언트 서버는 소켓을 초기화하여 생성하고 Bind 후 Listen 상태에서 읽기와 쓰기 쓰레드를 생성하고 다른 클라이언트들의 접속 시도를 기다린다. 다른 클라이언트가 접속을 요청하면 둘 사이에 연결이 이루어지고 정보 교환을 위해 read와 write 쓰레드를 사용해서 게임 정보를 송수신하게 된다. 이러한 게임 정보 교환은 주기적으로 일어나며 변경된 정보만을 교환하게 된다. 클라이언트들 간의 통신이 종료되면 점유했던 리소스들을 해제하고 접속을 종료한다. 클라이언트 측 통신 알고리즘은 서버 측 통신알고리즘과 유사하다. 각 클라이언트들은 자신의 소켓을 초기화하고 스트림 기반의 연결 중심으로 소켓을 생성한 후 장치를 검색한 다음 통신을 개시한다. 통신은 쓰레드 단위로 주기적으로 이루어지며 서버와의 통신이 종료되면 점유했던 리소스들을 해제하고 접속을 종료한다.

```

// 클라이언트 서버측 통신 알고리즘
DWORD WINAPI Bluetooth_Server(LPVOID arg)
{
    ...
    if(ServSock ==socket(AF_Bluetooth,
        SOCK_STREAM, 0)) == INVALID_SOCKET)
        Err_quit(_T("Socket")); // SOCK_DGRAM 사용불가
    //linger option 설정
    ling_l_onoff = 1;
    ling_l_linger = 10; // 10초 대기
    if(setsockopt(ServSock, SOL_SOCKET, SO_LINGER, (char*)&ling,
        sizeof(ling)) == SOCKET_ERROR) Err_quit(_T("Linger Setting"));
    // Bind
    ...
    // Listen
    ...
    //Read
    if((hThread[0] = CreateThread(NULL, 0, ReadThread,
        (LPVOID)CIntSock, 0, &ThreadID[0])) == NULL)
    {
        MessageBox(chat_hWnd, _T("쓰레드 시작 불가"), _T("오류"),
            MB_ICONERROR);
        State.MyState = S_NOT;
        EndDialog(chat_hWnd, 0);
    }
    //Write
    if((hThread[0] = CreateThread(NULL, 0, WriteThread,
        (LPVOID)CIntSock, 0, &ThreadID[0])) == NULL)
    {
        MessageBox(chat_hWnd, _T("쓰레드 시작 불가"), _T("오류"),
            MB_ICONERROR);
        State.MyState = S_NOT;
        EndDialog(chat_hWnd, 0);
    }
    // 쓰레드 종료 대기
    ...
    //closesocket
    closesocket(ServSock);
    return 0;
}

```

그림 6. 클라이언트 서버 측 통신 알고리즘  
Fig. 6. Communication Algorithm in Game Server

#### IV. 구현 결과

이 장에서는 제안한 Bluetooth 기반 MMORPG의 구현결과를 보여준다. 그림 7은 게임 중 데이터 전송을 위해 사용한 패킷 구조를 보여준다. 무선통신 과정에서 교환하는 데이터의 양을 최소화해야 전송 지연 문제를 해결할 수 있다. 사용하는 패킷의 총 길이는 전송 속도를 효율적으로 하고 전송 지연을 줄이기 위해 60 바이트로 제한하였다. 패킷의 효율적인 처리를 위해 패킷검사를 하

는데, 이 패킷이 게임 작업 처리 데이터인지 게임 정보 데이터인지를 판별 한 후 해당 작업을 하게 된다. 그림 8은 본 논문에서 제안한 모바일 컴퓨팅 환경에서의 Bluetooth 기반 온라인 MMOPRG의 실제 실행 화면을 보여준다.



그림 7. 패킷 구조  
Fig. 7. Packet Structure

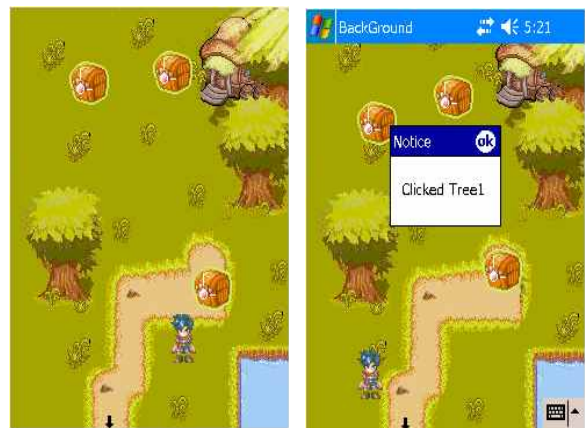


그림 8. 게임 실행 예  
Fig. 8. Example of Game Execution

#### V. 결론

최근 모바일 컴퓨팅 기술과 무선 통신 기술의 급속한 발전과 더불어 이동 단말도 빠르게 보급되고 있다. 그러나 이를 위한 응용이나 게임은 많은 제약으로 인해 제한적이다. 특히 롤플레이 게임과 같이 방대한 양의 이미지 데이터와 다양한 시나리오, 복잡한 컨트롤을 가진 게임의 경우 무선 이동 단말이 갖는 제약, 즉 저장장치 크기의 제약, 처리기의 성능 한계, 배터리 기반의 전원 공급 제약, 무선통신으로 인한 대역과 전송속도 제약 등으로 인해 구현이 매우 어려운 실정이다. 본 논문에서는 이러한 문제를 해결하는 모바일 컴퓨팅 환경에서의 블루투스 기반 온라인 MMORPG를 설계하고 구현하였다. 클라이언트들은 PDA를 사용하여 블루투스로 일시적인 국부 모바일 애드 혹 망을 구축하고, 무선으로 다수의 클라이언트들과 멀티플레이 할 수 있도록 설계하였다.

이동 단말의 제한된 리소스와 무선 기반 애드 혹 망상에서의 통신으로 인해 게임 진행 지연이나 화면 깜빡임 등의 문제가 발생할 수 있다. 무선통신 과정에서 클라이언트 부분은 소켓의 제한으로 인해 블로킹 모드로 구현해야 하고, 이로 인해 게임 진행이 지연되는 문제가 발생하는데 본 논문에서는 송신과 수신 소켓 프로세스를 스레드로 나누고 시간에 따라 강제로 종료하고 재시작하게 하는 방법으로 이 문제를 해결하였다. 또한 Windows CE 개발 환경에서의 repaint 문제로 인해 게임이 진행되는 동안 화면 깜빡임 문제가 발생하는데 이 문제를 해결하기 위해 배경이미지와 캐릭터를 분리한 더블 버퍼링 기법을 통해 화면 깜빡임 등 repaint 문제를 해결하였다.

### 참 고 문 헌

- [1] John W. Fisher II, "Methods and Considerations in Online Game Design," Master's Thesis at Michigan State University, East Lansing, MI USA, 2003.
- [2] Jim Adams, Programming Role Playing Games, Course Technology PTR, pp. 974, 2002.
- [3] Anthony Jones, Jim Ohlund : Network Programming For Microsoft Windows, Microsoft Press, 2003.
- [4] <http://www.bluetooth.com> (Bluetooth White Paper).
- [5] Brent A. Miller, Chatschik B., Bluetooth Revealed, 2001.
- [6] Jennifer B. and Charles F. Sturman, Bluetooth Connect without Cables, McGraw-Hill, 2001.
- [7] Nathan J. Muller, Bluetooth Demystified, McGraw-Hill, 2001.
- [8] Jonathan S. Harbour, Pocket PC Game Programming: Using The Windows CE Game API, Microsoft Press, 2003.
- [9] Charles Petzold, Programming Windows, 5th ed. Microsoft Press, pp. 1536, 2005.
- [10] 오선진, 임베디드시스템 소프트웨어 개발방법론, 한울출판사, pp. 461, 2007.
- [11] 백창우, TCP/IP 소켓 프로그래밍, 한빛미디어, pp. 744, 2005.
- [12] E. L. Lamie, 실시간 임베디드 멀티스레딩, 에이콘출판, 2005.
- [13] Alex J. Champandard, 인공지능 게임 프로그래밍 실전 가이드, 에이콘출판, 2004.

### 저자 소개

오 선 진(정회원)  
제 6권 2호 참조