
Chapter 238. 지그비 (ZigBee)

1. 무선 통신 (Wireless Communication)

무선 통신이란 둘 이상의 지점 사이에 선의 연결 없이 정보를 전송하는 것을 말한다. 선의 연결이 없으므로 공기 중으로 신호를 전송하여야 하는 것은 당연하며 일상생활에서의 대화 역시 공기 중으로 목소리가 전달되고 있다. 하지만 멀리 있는 친구를 불러보자. 거리가 조금만 멀어진다면 전달 과정에서 반사, 산란, 감쇄 등으로 인해 목소리가 전달되지 못한다. 따라서 멀리 떨어져 있는 두 지점 사이에 정보를 전달하기 위해서는 높은 주파수의 신호에 낮은 주파수의 데이터 신호를 실어서 보내는 변조(modulation) 과정이 필요하다. 높은 주파수의 신호는 직진성이 좋고 간섭에 강하므로 먼 거리까지 전송이 가능하다. 반대로 수신된 고주파 신호에서 저주파의 데이터 신호를 분리해 내는 것을 복조(demodulation)라고 한다.

변조에서 고려해야 하는 사항은 여러 가지가 있으며 그 중 하나가 신호 전달을 위해 사용하는, 흔히 캐리어(carrier)라고 불리는 고주파 신호의 주파수다. 주파수가 높을수록 정보는 멀리까지 전달될 수 있지만 주파수가 높을수록 장애물을 피해가기는 어렵다. 따라서 정보를 전달하고자 하는 지점 사이의 특징을 고려하여 캐리어 주파수가 선택된다. 아주 먼 거리에서의 통신이 필요하며 장애물이 거의 없는 우주 통신의 경우 기가헤르츠(GHz) 단위의 캐리어 주파수를 사용하는 반면 TV나 라디오와 같이 산이나 건물의 영향을 받는 경우에는 메가헤르츠(MHz) 단위의 캐리어 주파수를 사용한다.

캐리어 주파수가 선택된 경우에도 한 가지 더 고려해야 할 사항으로 대역폭(bandwidth)이 있다. 그림 1은 주파수 변조(Frequency Modulation, FM) 방식에 의해 변조된 신호를 보여주고 있다.

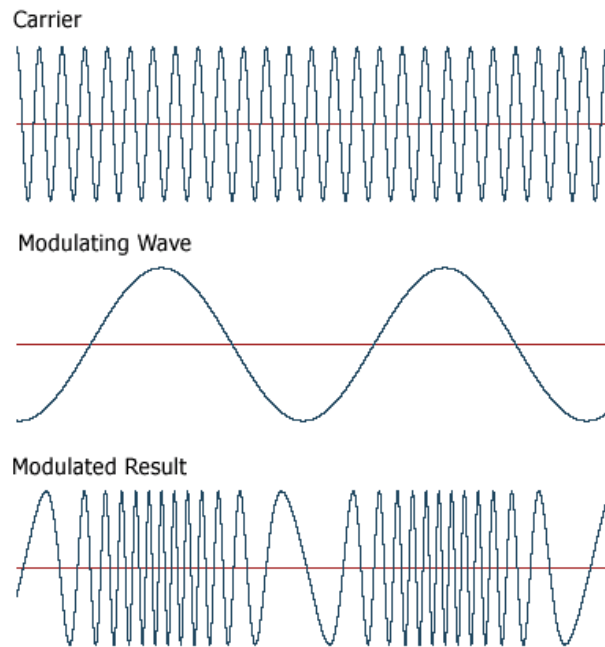


그림 1. 주파수 변조

그림 1에서 알 수 있듯이 캐리어 주파수에 전달하고자 하는 신호를 실어서 변조된 신호는 하나의 정현파로는 설명할 수 없으며 일정한 범위의 주파수에 있는 정현파의 합으로 표시된다. 대역이란 변조에 사용되는 ‘일정한 범위’의 주파수를 가리키는 말로, 대역폭이 넓을수록 (즉, 변조된 신호를 구성하기 위해 사용하는 정현파의 종류가 많을수록) 더 많은 정보를 전송할 수 있다. (즉, 더 복잡한 신호를 캐리어에 실어 보낼 수 있다.) 대역폭은 종종 수도관의 굵기와도 비교된다. 수도관이 굵을수록 더 많은 물이 한꺼번에 흘러갈 수 있는 것과 마찬가지로 대역폭이 넓을수록 더 많은 정보를 한꺼번에 전송할 수 있다. 따라서 흔히 무선통신에서 말해지는 ‘대역(band)’이란 일정 범위의 주파수를 가리키는 말이며 그 넓이가 ‘대역폭’에 해당한다. 대역은 다시 몇 개의 채널(channel)로 나뉘어져 있다. FM 방송을 위해 사용하는 대역은 88.1~107.9MHz이며, 이는 100개의 채널로 나뉘어져 있어 100개의 서로 다른 라디오 방송이 가능하다.

지그비는 블루투스나 마찬가지로 ISM (Industrial, Scientific and Medical) 대역이라 불리는 2.4GHz 대역을 사용한다. 이외에도 ISM 대역을 사용하는 대표적인 통신 방식에는 WiFi가 있다. 이처럼 다양한 통신 방식에서 동일한 주파수 대역을 사용하므로 상호 간섭에 의한 통신 장애가 발생할 수 있다. 특히 최근 블루투스 기기 사용의 증가로 인해 블루투스를 사용하는 키보드나 마우스의 연결이 끊어지는 현상을 발견할 수 있다.

2. ZigBee

지그비는 개인 근거리 저전력 무선 통신을 위한 프로토콜을 말한다. 비록 지그비가 저전력의 근거리 통신이지만 지그비 노드들 사이의 데이터 중계를 통한 메시 네트워크(mesh network)를 구성함으로써 데이터 전송 거리를 늘리는 것이 가능하다. 지그비를 통해 전송할 수 있는 데이터는 2.4GHz 대역에서 250Kbit/s로 적어서 주기적으로 또는 간헐적으로 센서나 입력 장치에서 데이터를 전송하는 경우에 적합하다. 동일한 대역을 사용하는 블루투스 2.0의 경우 이론상 최고 3.0Mbit/s 속도가 가능하며, WiFi 802.11g의 경우 최고 54Mbit/s의 전송이 가능한 등 지그비와 달리 많은 양의 데이터를 전송하는 것이 가능하다.

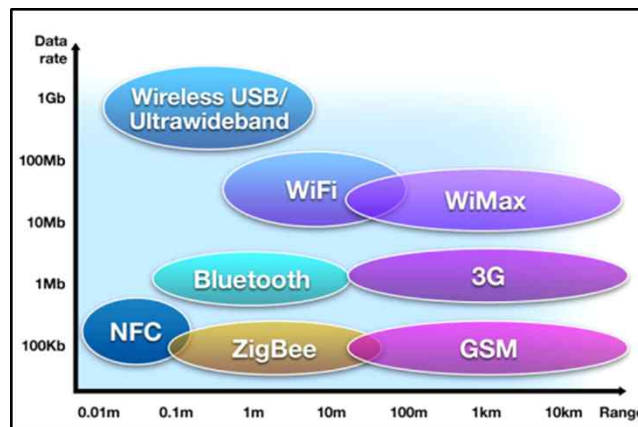


그림 2. 무선 통신 방식의 통신 거리와 데이터 전송 속도

블루투스나 와이파이의 전송 속도면에서 지그비보다 우수하지만 상대적으로 가격이 비싸므로 지그비는 블루투스나 와이파이를 대체하여 간단하고 저렴한 무선 네트워크를 구성하기 위해 고안되었으며, 주기적으로 적은 양의 데이터를 수집하는 센서 네트워크 구성에 흔히 사용된다. 지그비라는 이름은 Zig-Zag와 Bee의 합성어로 꿀벌의 정보 전달을 위한 비행 패턴에서 그 이름을 따왔다고 한다. 지그비의 전송 거리는 실외에서는 100m까지 가능하지만 실내에서는 30m 정도로 짧은 범위에서만 가능하다.

지그비는 IEEE 802.15.4를 바탕으로 하고 있다. IEEE 802.15.4는 저속의 개인용 근거리 무선 네트워크를 위한 표준 중 하나이다. 지그비의 하위 프로토콜이 IEEE 802.15.4라면 상위 프로토콜은 지그비 연합(ZigBee Alliance)에 의한 지그비 명세(ZigBee Specification)에 의해 관리되고 있으며 이 상위 프로토콜이 흔히 지그비 프로토콜이라 한다.

IEEE 802.15.4는 스타 또는 P2P (Point to Point) 형태의 네트워크만을 지원하지만 지그비 프로토콜은 트리나 메시 형태의 네트워크도 지원한다. 즉, 지그비 프로토콜은 IEEE 802.15.4

를 기반으로 다양한 형태의 네트워크를 지원하여 활용성을 높이고 있다.

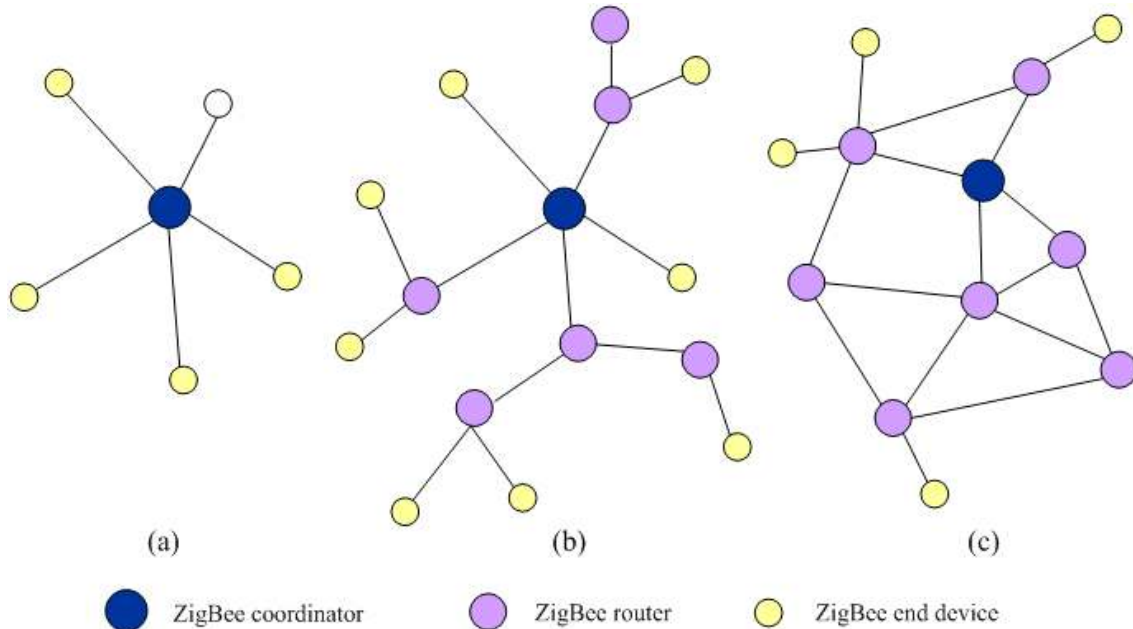


그림 3. 지그비 네트워크

3. 지그비 모듈

이 장에서 사용하는 지그비 모듈은 DiGi International¹⁾에서 제작하여 판매하는 지그비 모듈이다. DiGi International에서 다양한 종류의 지그비 모듈을 판매하고 있다. DiGi International에서 판매하는 모듈들은 'Xbee'라는 이름으로 판매되고 있다. 지그비는 저속 근거리 개인용 무선 네트워크를 위한 표준을 가리킨다면, Xbee는 지그비 호환 모듈에 붙여진 DiGi International의 상표 이름에 해당한다. 또한 Xbee는 지그비 프로토콜을 바탕으로 보다 쉬운 사용을 위해 Xbee 전용의 프로토콜을 제공하고 있으며, Xbee 프로토콜을 사용하는 경우 일반 지그비 모듈과 통신이 불가능할 수도 있다는 점을 염두에 두어야 한다.

Xbee 모듈에는 시리즈 1과 시리즈 2의 두 종류가 있다. 두 시리즈의 제품들 사이에는 여러 가지 차이가 있지만 가장 큰 차이는 사용하는 프로토콜에 있다. 시리즈 1은 IEEE 802.15.4 펌웨어가 기본으로 설치되어 있으므로 P2P 또는 스타 네트워크만을 사용할 수 있으며 메시 네트워크는 사용할 수 없다. 메시 네트워크를 사용하기 위해서는 DiGi International에서 제공하는 DigiMesh 프로토콜을 지원하는 펌웨어를 교체하여야 한다. DigiMesh 프로토콜은 지

1) <http://www.digi.com/>

그비 프로토콜과 마찬가지로 메시 네트워크를 지원하기는 하지만 DiGi International 자체 프로토콜로 지그비 프로토콜과는 동작 방식에서 약간의 차이가 있다. 시리즈 2는 지그비 프로토콜이 기본으로 설치되어 있어서 지그비 표준 메시 네트워크를 지원한다. 지그비 프로토콜에 비해 DigiMesh 프로토콜이 메시 네트워크 구성이 간편하고 전력 소비 면에서도 우수하므로 DigiMesh 프로토콜이 흔히 사용되고 있다. 하지만 DigiMesh 프로토콜은 DiGi International의 제품에서만 사용할 수 있다는 점을 명심하여야 한다.

Xbee 모듈은 일반 모듈과 프로 모듈로도 구분된다. 일반 모듈과 프로 모듈의 차이는 출력의 차이로, 일반 모듈은 최대 1mW 출력을 가지고 최대 90m 정도 통신이 가능하지만 프로 모듈은 최대 100mW의 출력을 사용할 수 있어 최대 1.6Km 까지 통신이 가능하다. Xbee 모듈은 사용하는 안테나의 종류에 따라서도 여러 가지 모듈이 판매되고 있으므로 사용 환경에 따라 선택하여 사용할 수 있다. 이 책에서는 와이어 안테나를 사용하고 1mW의 출력을 가지는 시리즈 1 Xbee 모듈을 사용한다.

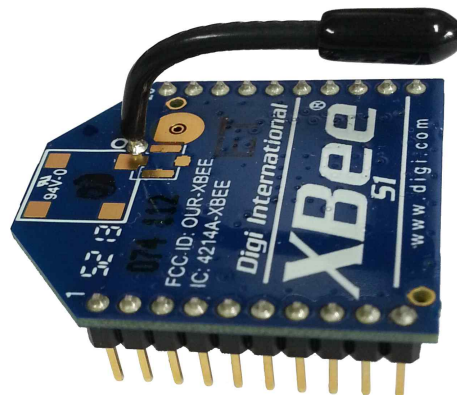


그림 4. Xbee 모듈

Xbee 모듈은 20개의 핀을 가지고 있다. 20개의 핀 간격은 일반적인 브레드보드의 핀 간격인 2.54mm보다 좁은 2.0mm이므로 브레드보드에 꽂아 사용하기 위해서는 별도의 변환 모듈이 필요하다. 또한 Xbee 모듈은 3.3V 전원을 사용하므로 전원 연결에 주의하여야 한다.

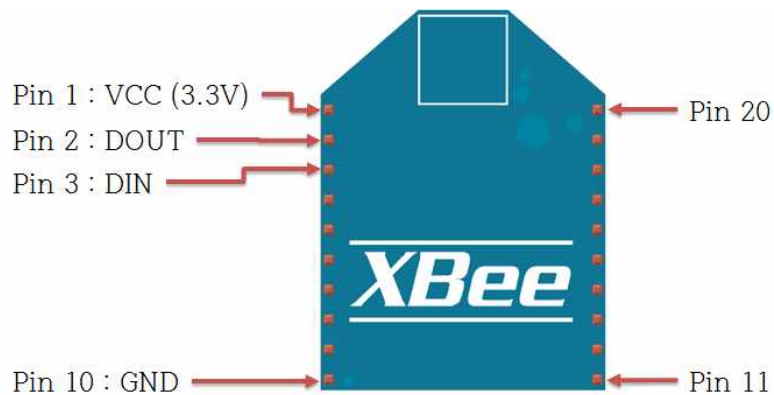


그림 5. Xbee 모듈 핀 번호

Xbee 모듈의 핀 번호는 그림 5와 같으며, 이 중 1번 핀은 VCC를 연결하는 핀으로 3.3V 전원을 연결하여야 하고 10번 핀이 GND에 해당한다. 2번과 3번 핀은 UART 시리얼 통신에 사용되는 핀으로 2번 핀이 DOUT(TX), 3번 핀이 DIN(RX)에 해당한다.

핀 번호	기능	비고
1	VCC	
2	DOUT	Xbee 모듈의 데이터 출력
3	DIN	Xbee 모듈로의 데이터 입력
10	GND	

표 1. Xbee 모듈 핀

Xbee 모듈을 컴퓨터와 연결하기 위해서는 USB 익스플로러가 필요하다. USB 익스플로러는 FTDI 칩을 사용하여 USB로 연결을 가상의 COM 포트로 인식되도록 해주는 기능을 하므로 Xbee 모듈을 통해 수신된 데이터를 컴퓨터에서 터미널 프로그램을 통해 확인할 수 있도록 해준다.

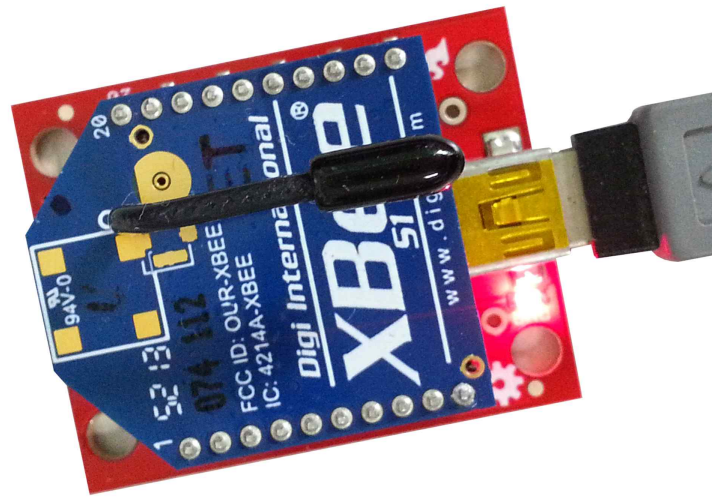


그림 6. USB 익스플로러

USB 익스플로러에 Xbee 모듈을 연결하고 컴퓨터에 연결해 보자. USB 익스플로러는 장치 관리자에서 시리얼 포트로 나타나는 것을 확인할 수 있다.

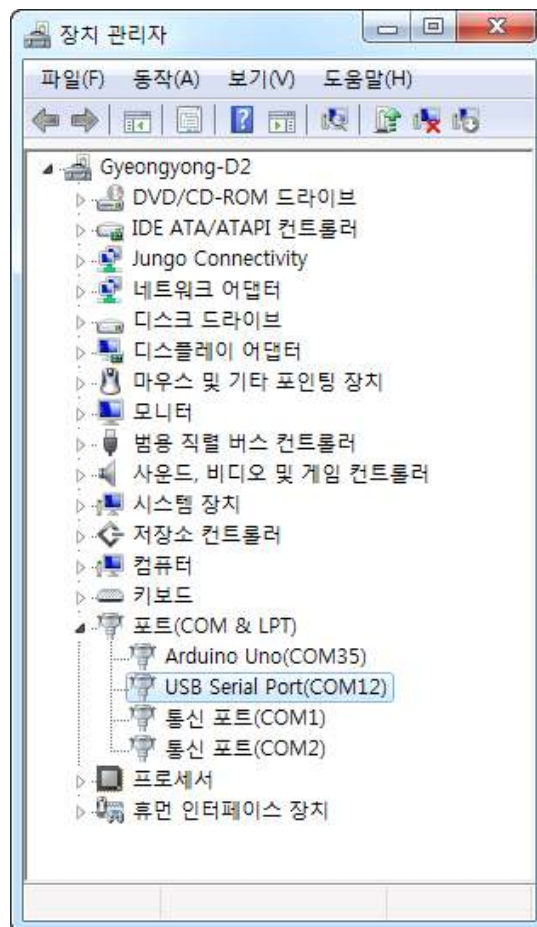


그림 7. USB 익스플로러 연결 후 시리얼 포트

또 다른 Xbee 모듈에 전원을 연결하고 DOUT과 DIN 핀을 점퍼선으로 연결하자. DOUT과 DIN을 연결하면 Xbee 모듈로 수신되는 데이터가 다시 재전송되므로 Xbee 모듈의 동작 여부를 확인할 수 있다. DOUT과 DIN을 연결하여 전송을 테스트하는 것을 루프백(loop back) 테스트하고 한다.

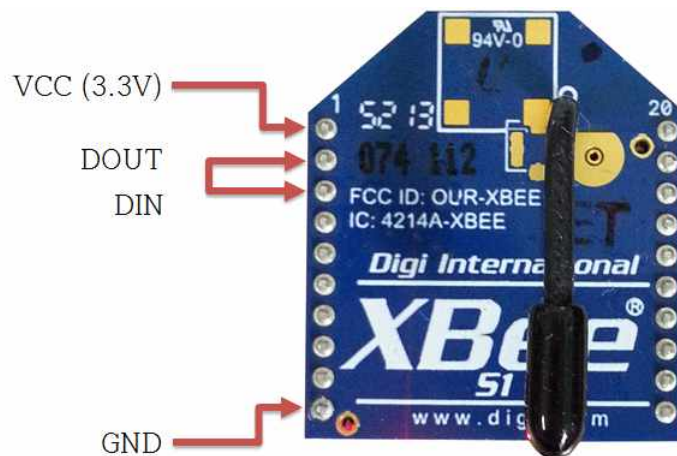


그림 8. 루프백 테스트를 위한 Xbee 모듈 연결

USB 익스플로러를 컴퓨터에 연결하고 또 다른 Xbee 모듈을 루프백 테스트를 위해 연결하였다면 RealTerm 프로그램을 실행시켜서 USB 익스플로러와 연결된 포트를 열어 문자열을 전송해 보자. 전송한 문자가 그대로 수신되어 화면에 나타난다면 두 개의 Xbee 모듈은 정상적으로 통신하고 있다고 생각할 수 있다. Xbee 모듈은 디폴트로 9600보율로 설정되어 있으므로 정상적으로 데이터가 수신되지 않는 경우 확인하도록 한다.

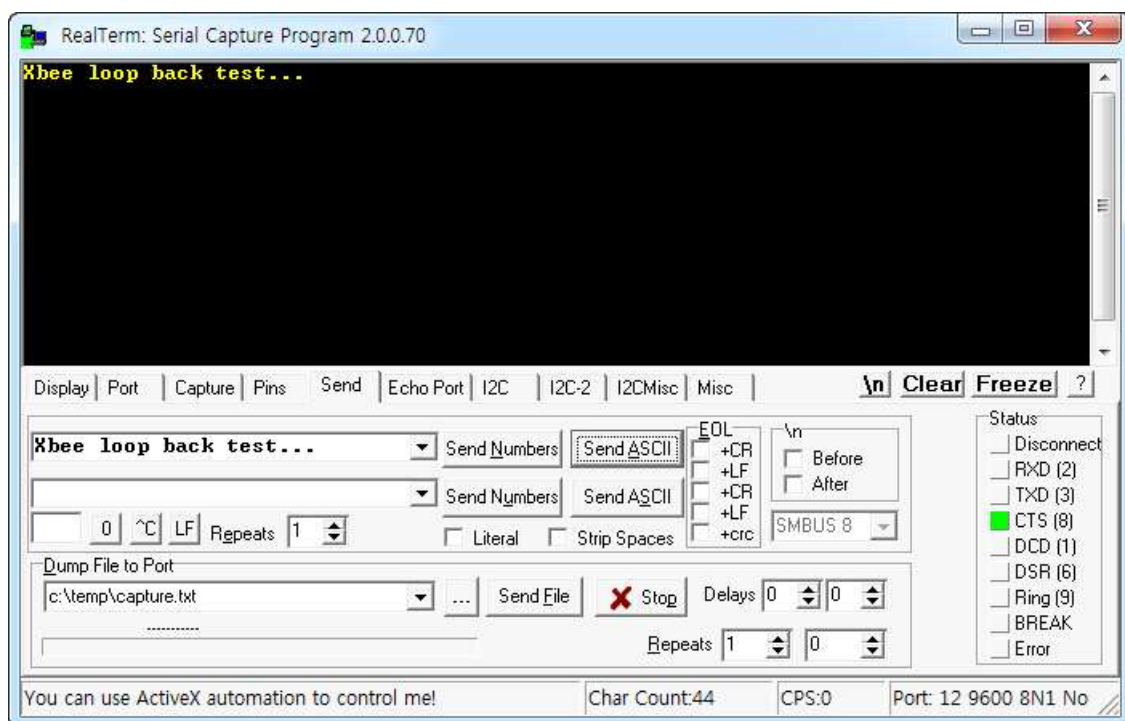


그림 9. 루프백 테스트

DiGi International에서는 모듈 테스트와 더불어 다양한 설정 기능을 제공하는 XCTU(X Control and Test Utility)라는 전용 프로그램을 제공하고 있으므로 이를 다운로드하여²⁾ 설치해 보자. 설치가 완료되면 프로그램을 실행시켜 보자.

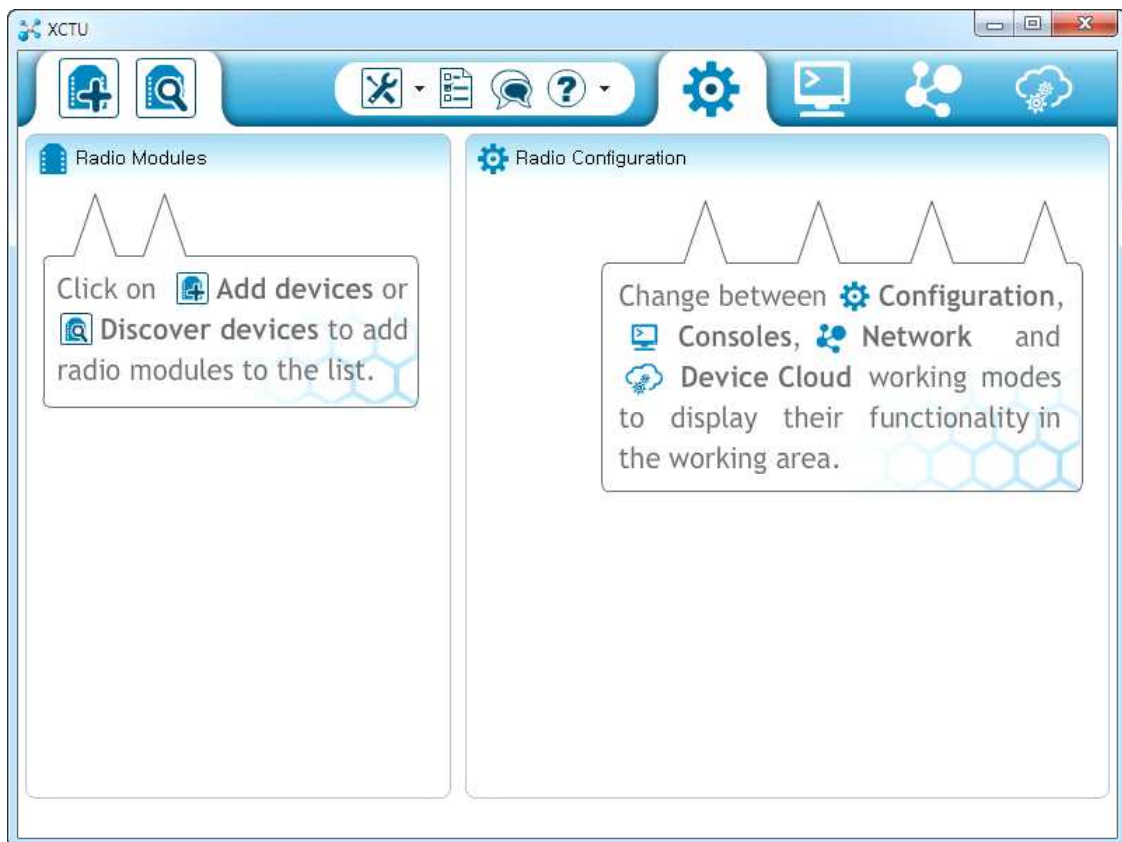


그림 10. XCUT 실행 화면

왼쪽 위의 Add 버튼()을 누르고 USB 익스플로러가 연결되어 있는 시리얼 포트를 선택하자.

2) <http://www.digi.com/products/wireless-wired-embedded-solutions/zigbee-rf-modules/xctu>

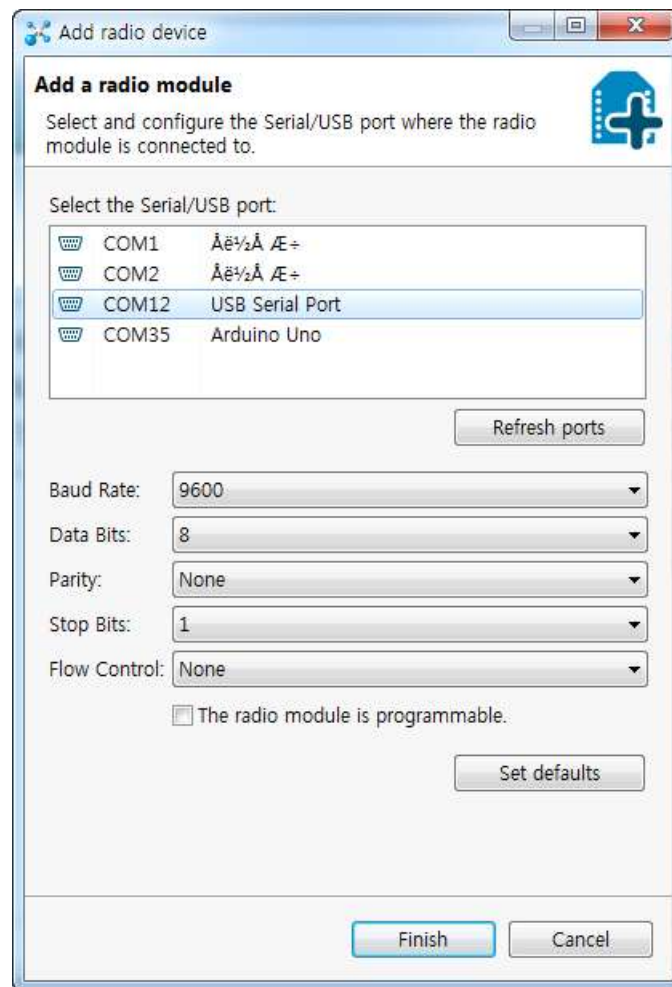


그림 11. 장치 추가

XCUT는 지정한 포트를 검사하여 Xbee 모듈을 찾아내고 관련 정보를 표시해 준다. 추가된 Xbee 모듈을 선택하면 오른쪽에 자세한 정보가 표시된다.

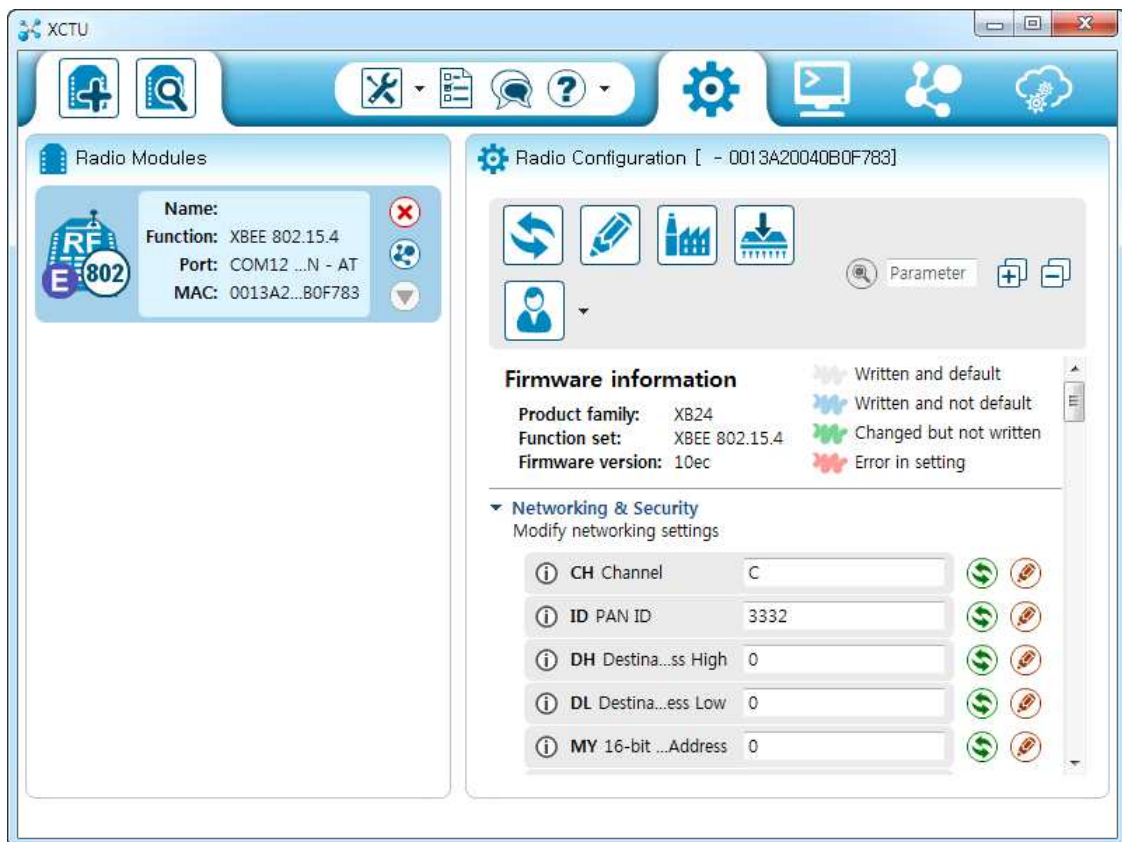


그림 12. Xbee 모듈 정보

콘솔 모드()를 열고 시리얼 연결()을 선택하면 데이터를 입력할 수 있는 창이 활성화된다. 창에 'Test'라고 입력해 보면 각 글자가 두 번씩 나타나는 것을 확인할 수 있다. 파란색은 송신한 데이터에 해당하며, 빨간색은 루프백에 의해 수신된 데이터를 나타낸다.

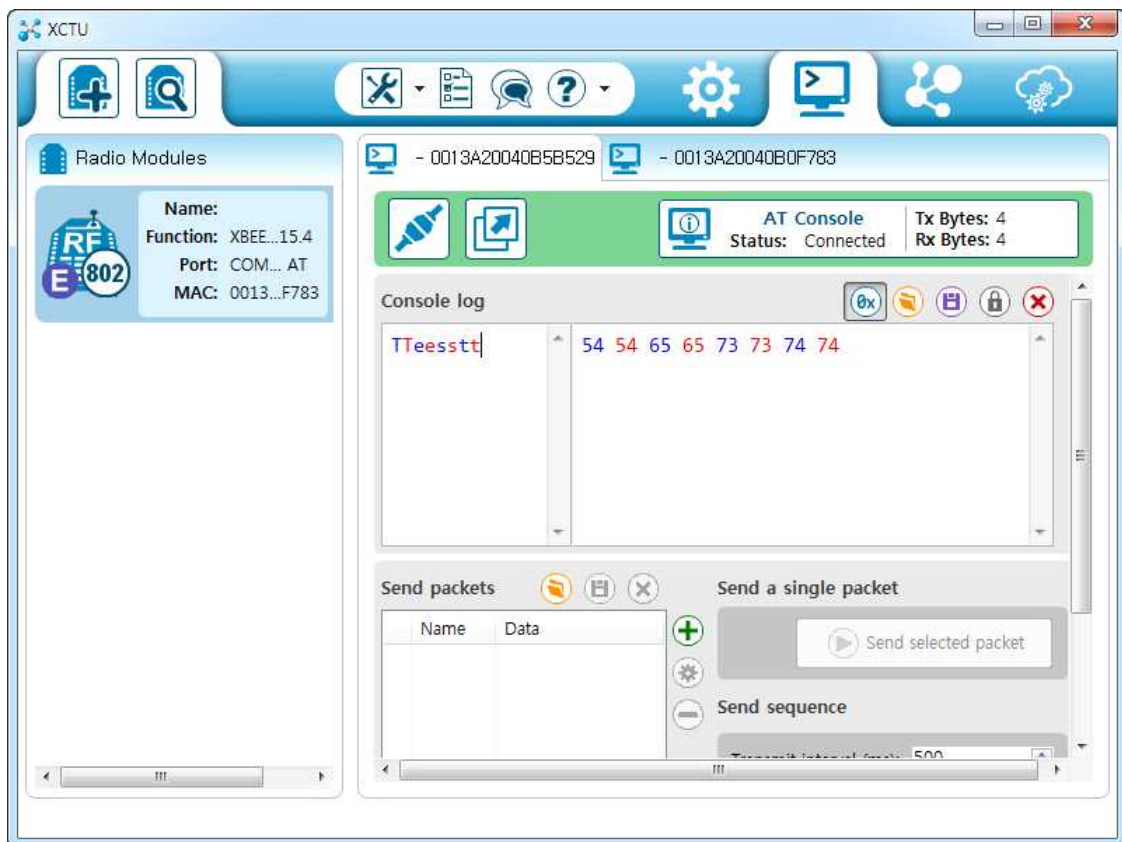


그림 13. 루프백 테스트

설정 탭으로 돌아와서 화면 왼쪽의 Xbee 모듈 옆에 있는 무선 노드 검색 버튼()을 누르면 현재 동일한 네트워크상에 있는 Xbee 모듈이 검색되어 나타난다.

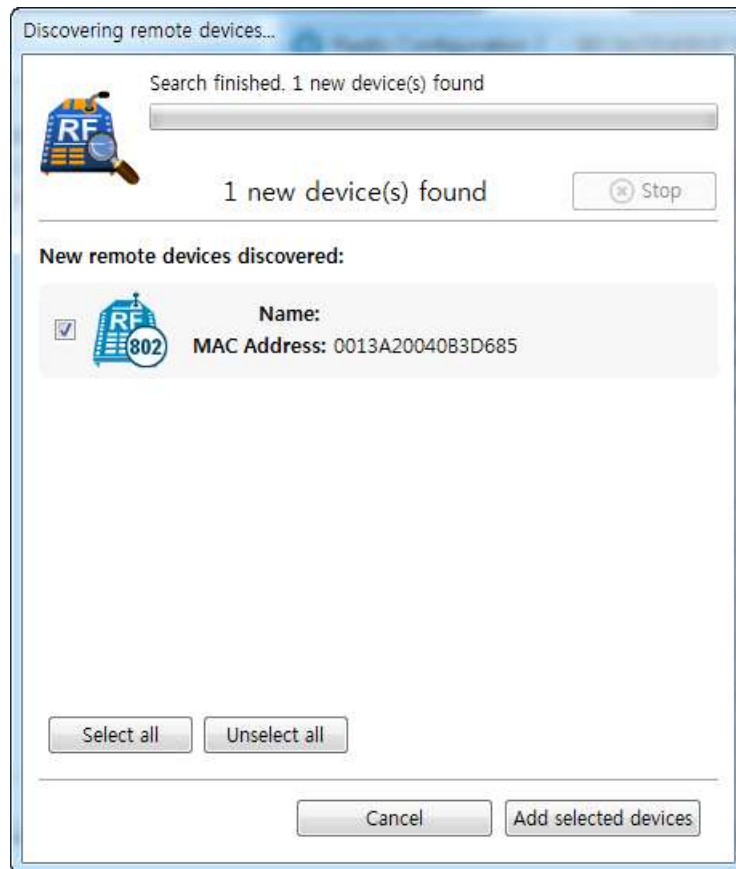


그림 14. 무선 노드 검색

추가 버튼을 누르면 USB 익스플로러의 Xbee 모듈 아래에 원격 모듈로 추가된다.

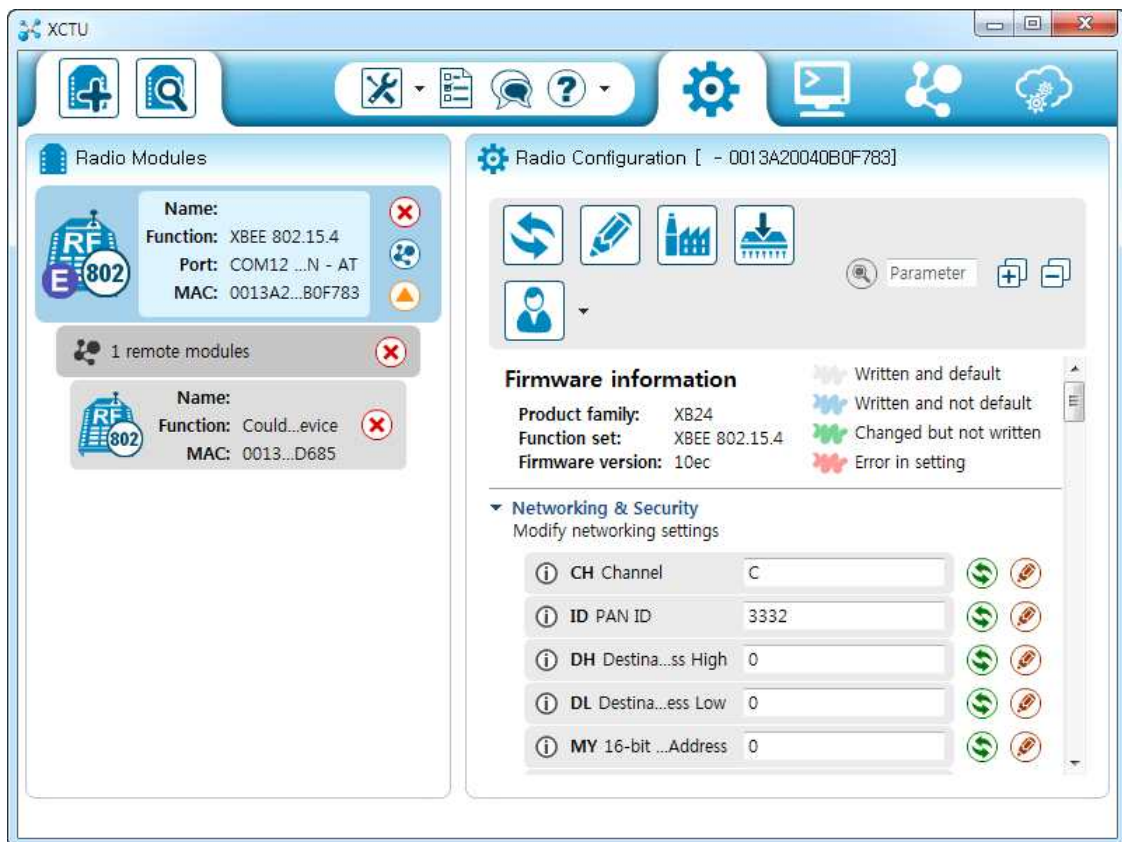


그림 15. 원격 모듈 추가

Xbee 모듈을 하나 더 루프백 테스트를 위해 DIN과 DOUT을 연결하고 전원을 연결하자. 콘솔 모드를 열고 시리얼 연결을 수행한 후 문자를 입력해 보자. 어떤 일이 발생하는가?

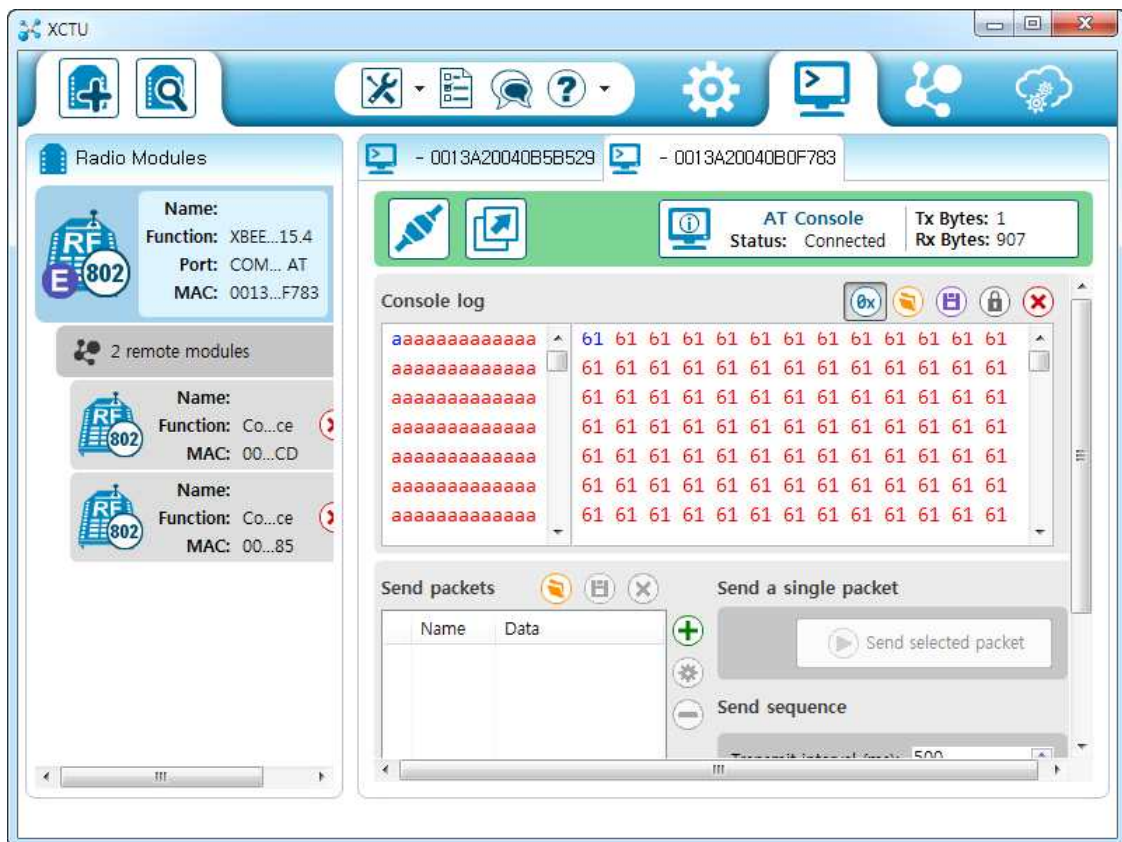


그림 16. 3개 모듈로 구성되는 Xbee 네트워크에서의 루프백 테스트

그림 16에 나타난 것처럼 한 번 전송된 데이터가 계속해서 수신되는 것을 확인할 수 있다. 이는 2개의 Xbee 모듈이 수신된 데이터를 다른 두 Xbee 모듈로 재전송함으로써 계속해서 동일한 데이터가 반복해서 수신되기 때문이다.

Xbee 모듈은 디폴트로 동일한 네트워크에 소속되어 있고 동일한 주소를 가지는 것으로 설정되어 있으므로 별다른 설정 없이도 네트워크를 구성하는 3개의 노드, 즉, Xbee 모듈이 서로 연결되고 데이터를 주고받을 수 있음을 확인하였다. 하지만 특정 노드에만 데이터를 전달하고 싶다면 어떻게 해야 할까?

네트워크상에서 노드를 식별하기 위해서 지그비는 주소를 사용한다. USB 익스플로러에 연결된 Xbee 모듈을 선택하고 오른쪽에 나타나는 정보 중에서 'Networking & Security' 부분을 살펴보면 노드의 주소와 관련된 부분을 확인할 수 있다.

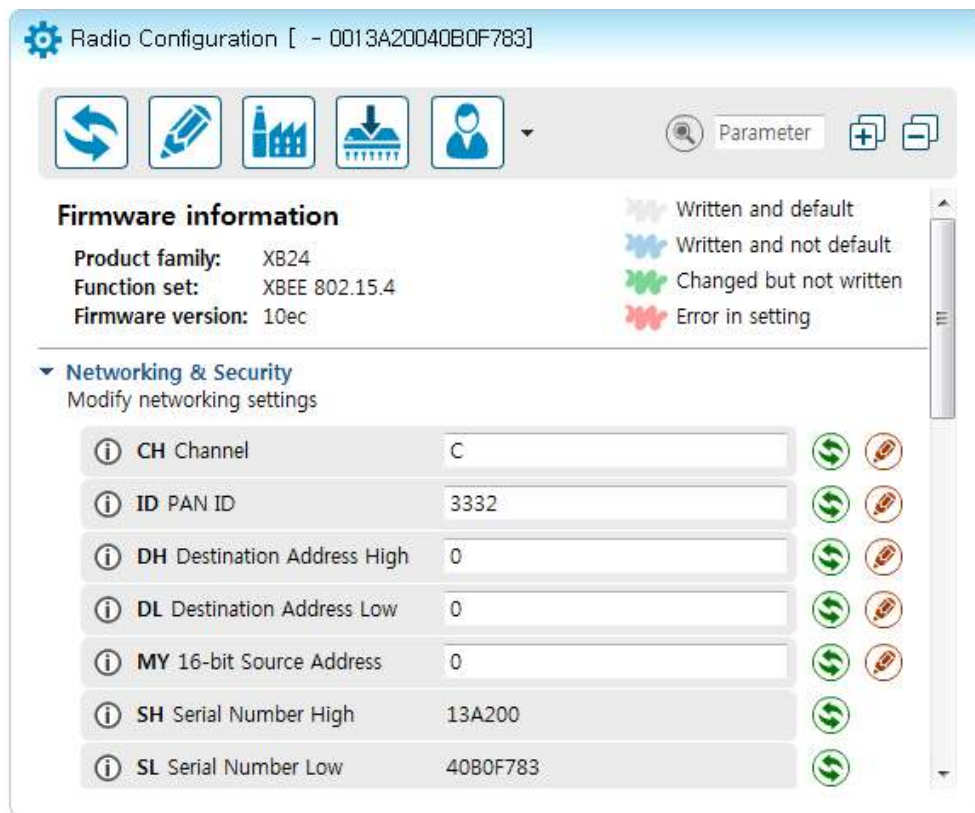


그림 17. 노드의 주소 설정

첫 번째 항목인 채널(CH, Channel)은 Xbee가 사용하는 2.4GHz 대역 중에서 사용할 채널을 나타낸다. 노드들이 서로 통신하기 위해서는 동일한 채널로 설정되어 있어야 한다. PAN ID는 Personal Area Network ID의 약어로 개인용 네트워크를 식별하기 위해 사용된다. 네트워크 ID는 0x0에서 0xFFFF 사이의 값을 가지며 동일한 네트워크에 소속된 노드들 사이에서만 통신이 가능하다. 위의 두 가지 항목은 네트워크를 특징짓기 위한 파라미터에 속한다. 다음 두 가지는 네트워크 내에서 노드를 특징짓기 위한 파라미터이다.

동일 네트워크상에서 모든 노드는 0x0에서 0xFFFF 사이의 16비트 주소를 가지며 이를 MY 주소 또는 소스 주소(source address)라고 한다. 반면 목적지 주소(destination address)는 노드가 데이터를 보낼 수 있는 주소를 가리킨다. 예를 들어 노드 1의 MY 주소가 0x1234인 경우, 노드 2의 목적지 주소가 0x1234로 설정되어 있어야만 노드 2는 노드 1로 데이터를 전송할 수 있다. 그림 17에서 알 수 있듯이 목적지 주소는 2개의 16비트 값으로 구성된다. 데이터를 수신할 노드의 MY 주소로 목적지를 지정하는 경우 DH 값은 0으로, DL 값은 MY 주소로 설정하면 된다. 또는 목적지 노드의 시리얼 번호(Serial Number)를 DH와 DL 값으로 지정해주어도 목적지 노드를 설정할 수 있지만 시리얼 번호는 수정이 불가능하므로 MY 주소를

사용하는 것이 편리하며 더 많이 사용된다.

Xbee 모듈의 디폴트값은 그림 17에 나타난 것처럼 모든 주소가 0으로 동일하게 설정되어 있어 모든 노드로 메시지가 전달되었다. 서로 다른 소스 주소를 가지는 노드로 구성된 네트워크에서 모든 노드로 메시지를 전달하기 위해서는 브로드캐스트(broadcast)를 사용할 수 있으며, 브로드캐스트를 사용하기 위해서는 목적지 주소를 DH = 0x0, DL = 0xFFFF로 설정하여야 한다.

세 개의 Xbee 모듈이 서로 통신할 수 있도록 표 2와 같이 주소를 설정해 보자. IEEE 802.15.4 프로토콜은 메시 네트워크를 지원하지 않으므로 하나의 노드를 코디네이터(coordinator)로 두고 다른 모든 노드들은 코디네이터로 데이터를 전송하도록 설정한 예가 표 2이다.

파라미터	노드 1	노드 2	노드 3	비고
CH	C			통신 채널
ID	0x1234			네트워크 ID
DH	0	0	0	
DL	0xFFFF	1	1	
MY	1	2	3	노드 주소
비고	코디네이터 PC와 연결 브로드캐스트	노드 1로 데이터 전송	노드 1로 데이터 전송	

표 2. 노드 주소 설정

표 2와 같이 주소를 설정하고 루프백 테스트를 해보자. 그림 16에서와는 달리 노드 1은 모든 노드로 데이터를 전송하지만, 노드 2와 3은 노드 1로만 데이터를 전송하므로 수신이 2번 일어남을 알 수 있다.

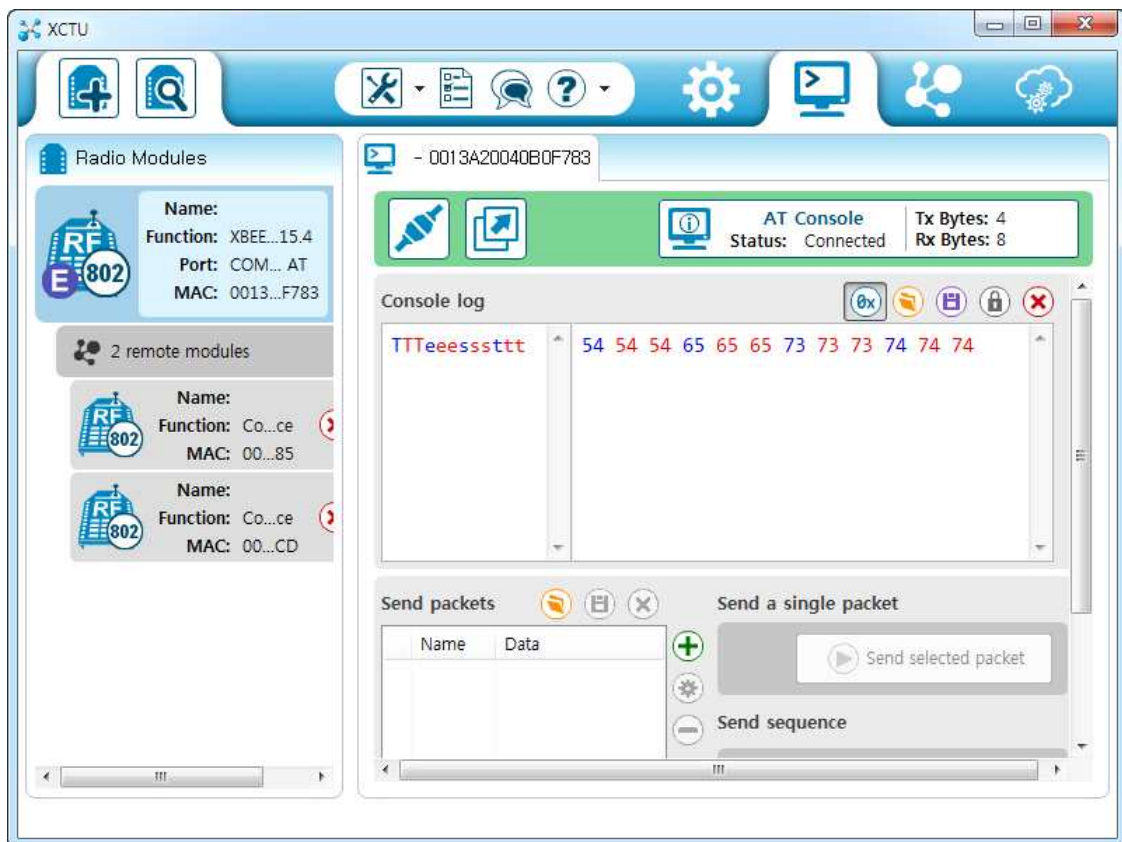


그림 18. 루프백 테스트

4. 아두이노와 컴퓨터의 통신 테스트

지그비 모듈을 사용하는 방법은 여러 가지가 있지만 여기서는 블루투스 통신을 대체하는 간단한 무선 통신 모듈로서의 사용 방법에 대해 알아보자. 표 2에서 노드 1과 노드 2를 사용하며 노드 1은 컴퓨터에 USB 익스플로러를 통해 연결하고 노드 2는 그림 19와 같이 아두이노 우노와 연결한다.

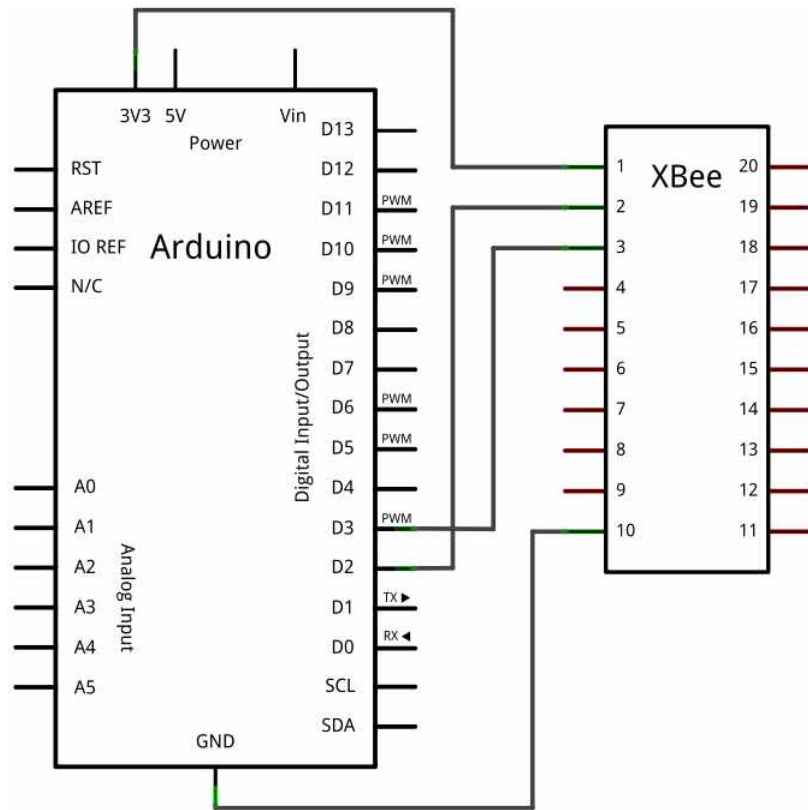


그림 19. 지그비 모듈 연결 회로도

아두이노에는 1초에 1씩 카운터를 증가시키고 이를 전송하도록 스케치 1을 업로드 한다. 스케치 1은 컴퓨터에서 전송된 데이터를 수신하여 이를 시리얼 모니터로 출력하는 기능 역시 포함 되어 있다.

스케치 1. 카운터 값 전송

```
#include <SoftwareSerial.h>

SoftwareSerial xbee(2, 3);           // (RX, TX)
byte count = 0;
unsigned long time_previous, time_current;

void setup() {
  Serial.begin(9600);
  xbee.begin(9600);                  // Xbee 모듈 연결 초기화
```

```

    time_previous = millis();
}

void loop() {
    time_current = millis();
    if(time_current - time_previous > 1000){          // 1초에 한 번 실행
        time_previous = time_current;
        count++;                                     // 카운터 증가
        xbee.println(count);                         // 데이터 전송
    }

    if(xbee.available()){                            // 수신 데이터를 시리얼 모니터로 출력
        Serial.write(xbee.read());
    }
}

```

그림 20은 스케치 1을 실행시키고 컴퓨터에서 RealTerm을 통해 USB 익스플로러가 연결된 시리얼 포트 데이터를 출력한 것이다.

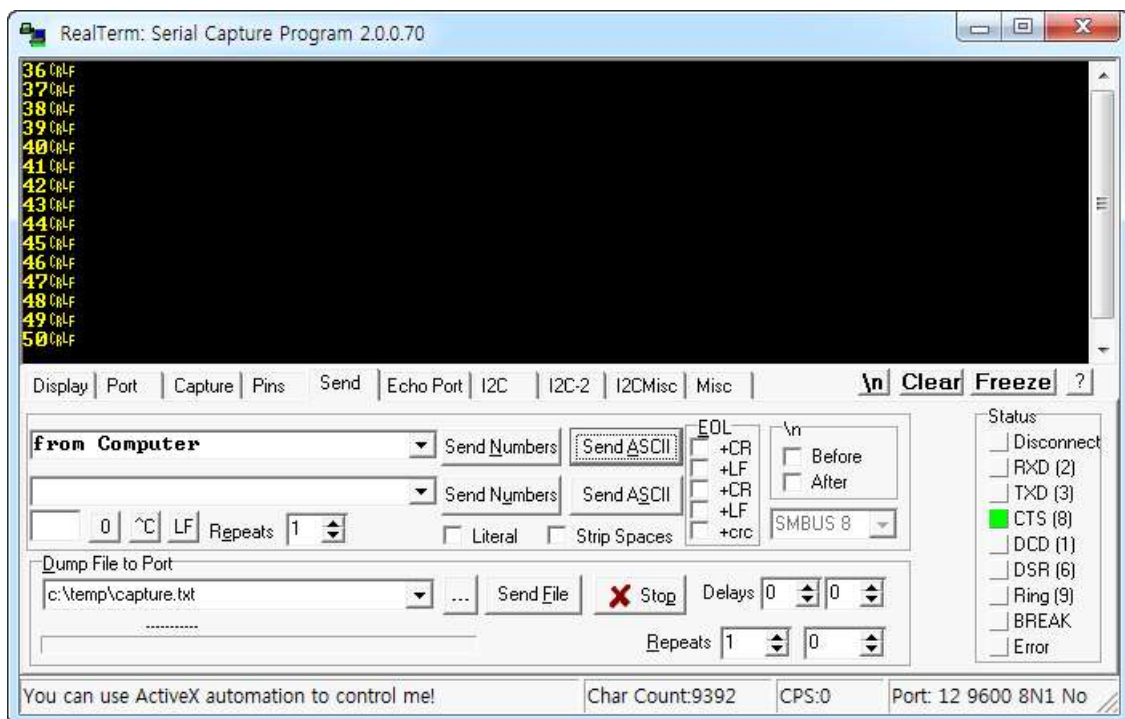


그림 20. 스케치 1 실행 결과

RealTerm에서 데이터를 입력하면 아두이노와 연결된 시리얼 모니터에 표시되는 것 역시 확인할 수 있다.

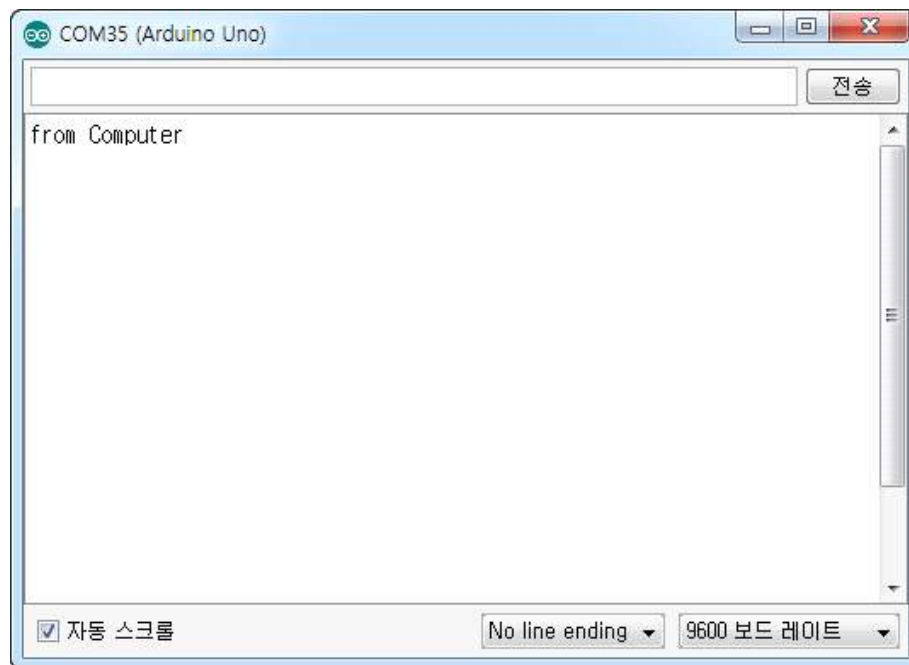


그림 21. 스케치 1 실행 결과