

야외 작업 환경에서의 무선LAN 서비스 제공을 위한 SUB-1GHz 무선 인터넷 중계 장치

양지혁, 천지민, 김영탁

정보통신공학과, 기계IT대학, 영남대학교
214-1. 대동, 경산시, 경상북도, 712-749, 대한민국

flash0223@ynu.ac.kr, koka1000@naver.com, ytkim@yu.ac.kr

요약문

본 논문에서는 시골의 논과 밭, 건설 현장 등과 같은 반경 100m 이내에 유선 랜 접속이 불가능한 야외 작업 환경에서는 Egg나 스마트폰LTE의 값 비싼 데이터를 사용하여 인터넷에 접속을 해야 한다. 혹은 2.4GHz 대역의 WiFi 중계기를 사용해야하지만 주파수 특성상 회절이 적어 운용거리가 짧다. 본 논문에서는 반경 1km의 범위를 가진 900MHz 대역의 RF 모듈을 이용하여 인터넷 중계 기능을 제안한다. 1:1 TDD Communication을 기반으로 한 TI CC1310 Device Driver, Packet의 segmentation & reassembly기능과 Flow Control기능이 포함된 CC1310과의 SPI Communication을 기반으로 한 Linux Network Device Driver를 구현하였다. 이를 통해 기존에 짧은 운용 거리를 가져서 실내의 음영 지역 보안을 해주는 WiFi 중계기에 비해 500m~1Km의 운용거리를 가져 시골의 논과 밭, 건설 현장 등에 종사하시는 분들을 위해 손쉽게 인터넷 사용 환경을 구축함으로써 더 나은 환경에서 일을 할 수 있도록 해준다.

I. 서론

유선 LAN 접속이 불가능한 환경 (즉, 100m 이내에 랜선을 사용할 수 없는 곳) 에서의 인터넷에 접속 한다는 것은 스마트폰의 LTE 데이터를 이용하여 스마트폰으로 접속을 하거나 스마트폰의 LTE 데이터를 이용한 핫스팟을 통해서 노트북과 같은 다른 기기를 인터넷에 접속 시키는 것을 말한다. 아니면 통신사에서 가입할 수 있는 값 비싼 요금의 EGG를 사용하여 인터넷에 접속을 하는 방법 밖에 없다. 하지만 이 모든 것들은 값 비싼 데이터 요금으로 인해서 야외 작업 종사자에게는 비 적합한 방법이다. 흔히 생각 할 수 있는 유선 LAN 접속이 불가능한 환경 야외 작업 환경이라면 시골의 논과 밭이나 건설 현장 등을 들 수가 있는데 이와 같은 야외 작업 환경 종사자들은 인터넷 접속을 하기 위해 자신의 사비를 통해 EGG

의 데이터나 자신의 스마트폰의 데이터를 요금을 지불해야하면서 사용을 해야 한다. 특히, 이러한 환경에서 장시간동안 통신을 하게 되면 이용자들은 높은 이용 금액을 지불해야 할 것이고, 시골의 논과 밭의 경우에는 EGG나 스마트폰이 사용하는 LTE 통신을 사용할 환경이 구축이 안 되어 있는 곳도 있다.

이에 100m이내의 유선 LAN 접속이 불가능한 야외 작업 환경에서 900MHz 대역 RF 통신을 활용하여 중계기와 AP를 구성하여 인터넷 접속을 할 수 있게 된다면, 이용자에게 부담이 될 수 있는 금전적인 문제를 해결할 하나의 새로운 대안으로 제시될 수 있다.

현재 나와 있는 비슷한 기술의 사례로는 가장 먼저, 공유기 한 대를 클라이언트 모드로 사용을 하여 인터넷과의 연결을 공유기와 공유기 사이의 통신을 통해 이루어지게 하는 **Wi-Fi Client mode**[7] 기술이 있다. 하지만 이 기술을 운용거리가 너무 짧아 효율이 떨어지며, 무선 연결을 할 수 없다는 단점이 있다. 건물 내 Wi-Fi 음영 지역을 보완하기 위하여 사용하는 **Wi-Fi 중계기(Extender)**[8]도 기존의 연구 사례로 볼 수 있다. 하지만 와이파이 중계기 또한 2.4GHz 대역을 사용하여 운용거리가 짧기에 효율이 떨어진다는 단점이 있다.

이러한 기존 연구 사례들이 가지는 단점들을 보완하고 유선 LAN 접속이 불가능한 환경에서의 인터넷 접속을 가능하게 만드는 기술로 제시 될 수 있는 것이 반경 1Km의 범위를 가진 900MHz 대역의 RF 모듈을 이용하여 인터넷 중계 기능을 만드는 기술이다.

시골의 논과 밭, 건설 현장 등에서의 이용자들에게 안정적인 통신이 가능하도록 하며, 스마트폰 데이터 요금을 절감하도록 돕는데 본 논문의 목적이 있다.

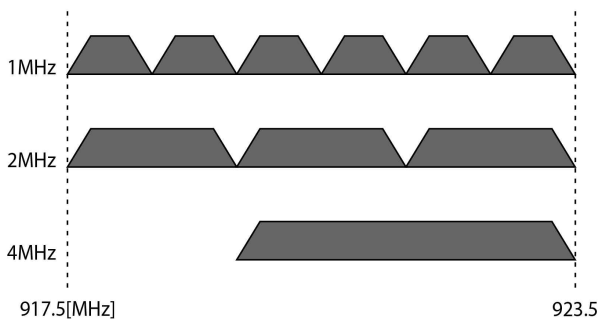
본 논문은 다음과 같이 구성되어있다. 제 2장에서는 900MHz 대역을 통한 인터넷 접속을 도모하는 프로토콜인 802.11ah, 900MHz RF 통신에 대해서 언급을 한다. 제 3장에서는 900MHz RF module인 CC1310을 이용한 패킷 송수신기를 구성하는 상세 기능 구조와 Linux에서 구성된 송수신기를 통해 패킷을 주고받는 Network Device Driver에 대해 설명한

다. 제 4장에서는 개발된 900MHz RF Module의 송수신기와 직접 구성한 Network Device Driver를 통해 중계기를 구성하여 성능 및 평가 결과를 분석한다. 제 5장에서는 본 논문의 결론을 맺는다.

II. 관련 연구

2.1 IEEE 802.11ah HaLow

IEEE 802.11ah는 2016년도에 나온 Sub 1GHz (900MHz) 밴드에서 동작하는 새로운 Physical Layer와 MAC Layer의 디자인이다[9]. 대부분 와이파이는 2.4GHz 혹은 5GHz에서 동작하지만 이 802.11ah는 900MHz 대역의 저전력 와이파이 표준이다. WIFI Alliance에서 표준으로 채택하면서 HaLow라고 불리기도 한다. 802.11ah는 확장된 범위의 Wi-Fi와 IoE (Internet-of-Everything)를 지원한다. 802.11ah의 900MHz 무선 파장은 2.4GHz와 5GHz보다 더 나은 회절 특성을 통해 하나의 AP로 하나의 주택을 커버할 수 있다. 최소 150Kbps 데이터 전송속도는 센서류들이 전력 소모를 최소화하면서도 짧은 순간의 데이터 패킷을 전송할 수 있게 해준다. 따라서 저전력 센서 네트워크, IoT (Internet of Things)에 최적화된 표준이다. 802.11ah의 한국에서의 주파수 할당 내역은 아래의 (그림 1)과 같다.



(그림 1) 한국 802.11ah 채널 할당

2.2 WiFi Extender

와이파이 확장기 (WiFi Extender)란 유무선 공유기의 와이파이 신호를 확장 시켜주어 운용거리를 늘리는 장치이다. 일반적인 가정에서 공유기가 설치되어있는 방의 방문을 닫거나, 공유기가 설치된 방과 좀 멀리 떨어진 부근이거나 화장실과 같은 장소에서는 2.4GHz 주파수 대역을 사용하는 WiFi 공유기의 운용거리가 50m인 반면 회절이 잘 안되는 고 주파수 대역의 특성 때문에 (그림 2)와 같이 실내에서도 WiFi를 원활하게 사용을 할 수 없다. 이러한 WiFi의 운용거리가 짧은점을 해결 하기 위해서 WiFi Extender라는 기술이 도입되었다. WiFi Extender를

사용한다면 (그림 3)과 같이 운용거리가 다소 넓어져 집에서 WiFi 신호가 잡히지 않는 음영지역을 해결해준다.



(그림 2) Wi Fi 확장기를 사용하지 않았을 경우 Wi Fi 운용 범위[8]



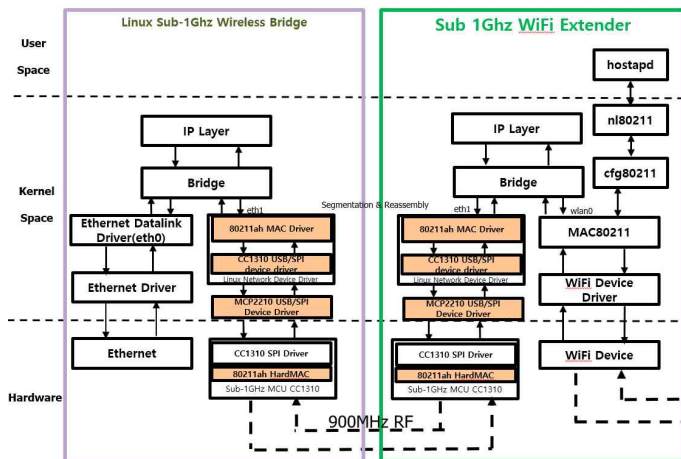
(그림 3) Wi Fi 확장기를 사용하였을 경우 운용거리 [8]

III. IEEE 802.11ah 기반 Sub-1GHz WiFi 중계 장치

3.1 IEEE 802.11ah기반 Sub-1GHz WiFi 중계기 System Structure

이 3.1절에서는 IEEE 802.11ah 기반 Sub-1GHz 무선 인터넷 중계기에 대한 시스템 구조를 설명한다. 아래의 (그림 4)가 전체 시스템 구조도이다. 먼저 Linux Sub-1GHz Wireless Bridge 블록은 인터넷에 연결되어있어 인터넷망에 직접적으로 연결하여 인터넷 패킷을 리눅스 브릿지 시스템을 구성하여 인터넷 프레임을 CC1310 네트워크 인터페이스로 넘겨주게 되며 네트워크 디바이스에서 데이터 크기에 따라 Segmentation을 한 후 SPI 통신을 통해 CC1310으로 넘겨주게 된다. 그리고 Bridge Side CC1310은 중계기 역할을 하는 Sub-1GHz WiFi Extender의 CC1310으로 전달하게 된다. 패킷을 전달 받은 Extender Side의 CC1310은 SPI 통신을 통해

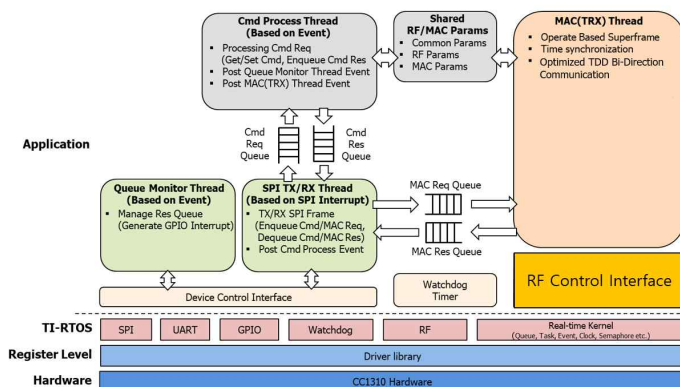
Raspberry Pi로 전달하게 되고 Raspberry Pi는 전달 받은 패킷을 Reassembly를 통해서 원래의 이더넷 프레임으로 합쳐서 AP를 구성하는데 사용하는 user daemon인 hostapd가 구성한 CC1310 Network Interface - Wireless LAN Network Interface간의 브릿지를 통해 Wireless LAN Network Interface로 전달하게 되고 그 전달된 패킷은 중계기 AP에 접속된 Station에게 전달되게 된다.



(그림 4) System Block Diagram

3.2 CC1310 RF Packet Transceiver

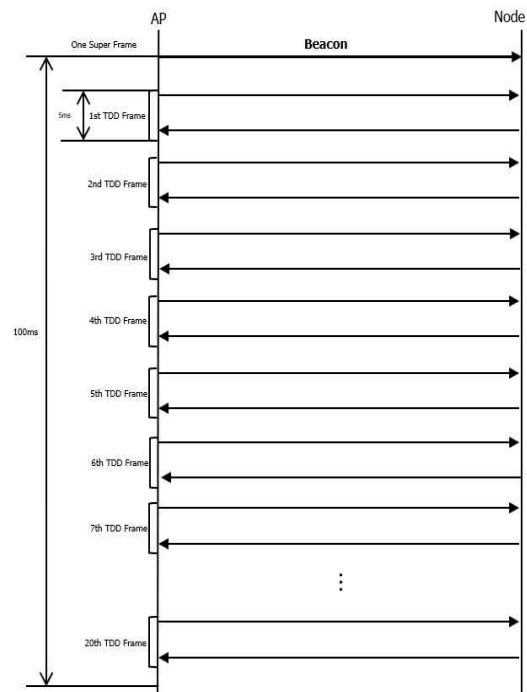
CC1310 RF Packet Transceiver는 Raspberry Pi로부터 Packet을 전달받고, RF통신을 통해 Packet을 Raspberry Pi로 전달하기 위해 SPI 통신과 RF 통신을 포함해야하며 (그림 5)와 같은 구조를 가진다.



(그림 5) CC1310 Software Block Diagram

구조를 살펴보면 (그림 5)에서 MAC Thread는 CC1310간의 실시간 무선 통신을 담당 하는 Thread이며 받은 프레임은 MAC Response Queue에 enqueue하고 보낼 프레임은 MAC Request Queue로

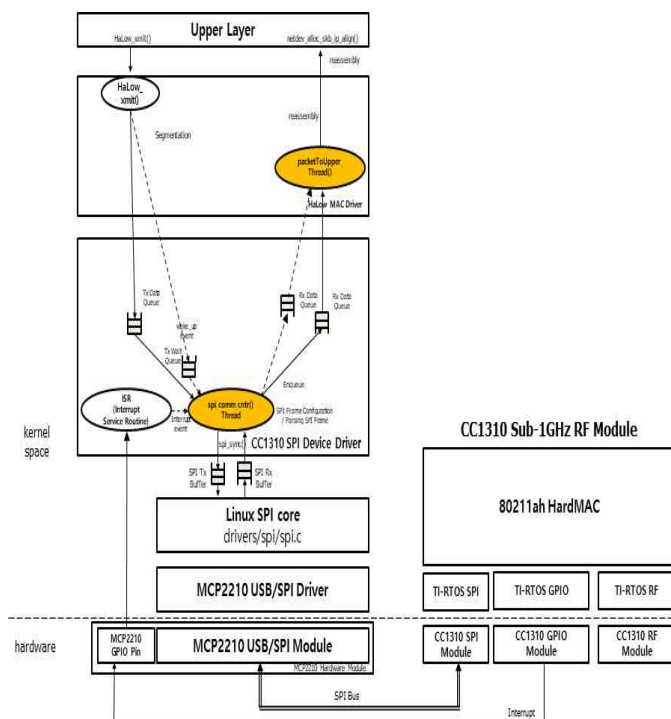
부터 Dequeue하여 가져온다. TDD(Time-Division Duplexing) bi-direction MAC protocol을 기반으로 구성되어있다. (그림 6)과 같이 구성되어 있으며 하나의 Super frame은 100ms로 구성되어 있으며 하나의 Time Slot은 5ms 동안 AP(Ethernet에 접속된 Raspberry pi와 연결된 CC1310)과 Node(Wi-Fi역할을 할 CC1310)이 각각 프레임 하나를 주고받도록 구성되어있다. Queue Monitor Thread는 CC1310에서 Raspberry Pi쪽으로 보낼 데이터가 있을 때 SPI 통신 상에서 Master인 라즈베리 파이에 데이터를 보내기 위해 GPIO interrupt를 발생시킨다. SPI TX/RX Thread는 SPI 통신을 담당하는 Thread이다. Raspberry Pi로부터 받은 Frame은 CC1310의 RF/MAC Parameter를 설정하는 Frame인지 Data Frame인지에 따라 Cmd Request Queue 또는 MAC Request Queue에 enqueue해준다. 그리고 다른 CC1310으로부터 RF 통신을 통해 받은 데이터를 관리하는 MAC Response Queue나 Command ACK를 관리하는 CMD Response Queue로부터 dequeue하여 Raspberry Pi쪽으로 데이터를 전송할 준비를 한다. 각 response queue로부터 dequeue를 하였을 때 GPIO Interrupt를 발생시키기 위해 Sleep 상태의 Queue Monitor Thread를 깨운다.



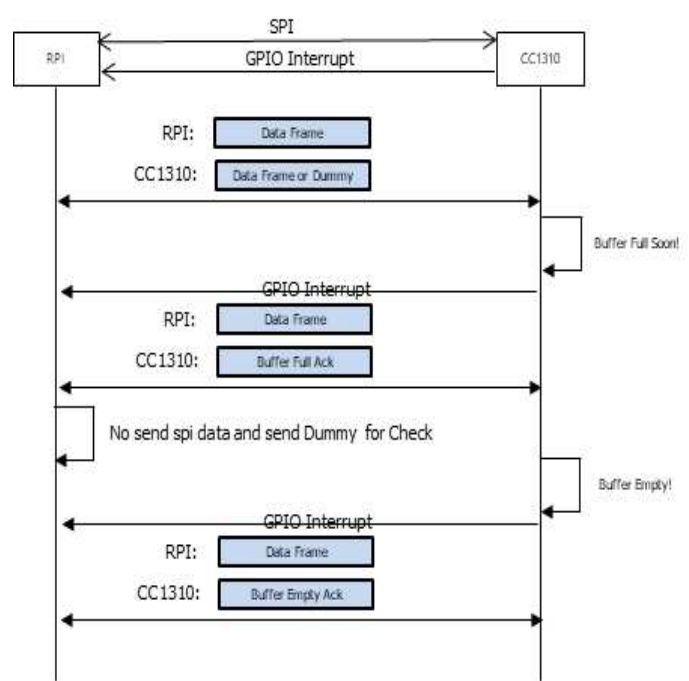
(그림 6) TDD (Time-Division Duplex)기반의 bi-directional MAC Protocol

3.3 CC1310 Linux Network Device Driver

CC1310 RF Packet Transceiver는 Raspberry Pi에 구성된 Network Device Driver에 의해 컨트롤 된다. (그림 7)는 구성된 Network Device Driver의 내부가 어떤 구조로 되어있는지에 대한 Software Block Diagram이다. Upper Layer에서 Packet을 가지고 HaLow_xmit()을 호출하게 되면 HaLow_xmit()은 받은 패킷을 segmentation하여 spi tx data queue에 enqueue한다. 그 후 spi통신을 담당하는 spi_comm_cntr thread가 spi tx data queue에서 데이터를 dequeue하여 spi 통신을 통해 cc1310으로 보낸다. 이 때 spi 통신 후에 CC1310으로 부터 받은 데이터가 있다면 Rx Data Queue에 enqueue하고 reassembly를 담당하는 PacketToUpper thread가 dequeue하고 reassembly후에 upper layer로 패킷을 보내주게 된다. 만약 네트워크의 데이터를 들어오는 대로 spi 통신으로 cc1310쪽으로 보내버리면 상대적으로 버퍼의 사이즈가 작은 임베디드 보드인 cc1310이 동작이 멈추는 경우가 발생하므로 SPI 통신시 Flow control이 필요하였고, (그림 8)과 같이 버퍼가 준비 되었을 때 한번 꽉찰을 때 한번 ACK를 보내고 그에따라 raspberry pi쪽에서는 spi data를 보내지 않도록 구성하였다.



(그림 7) CC1310 Linux Network Device Driver Block Diagram



(그림 8) SPI Control flow control sequence diagram

3.4 MAC Frame Structure

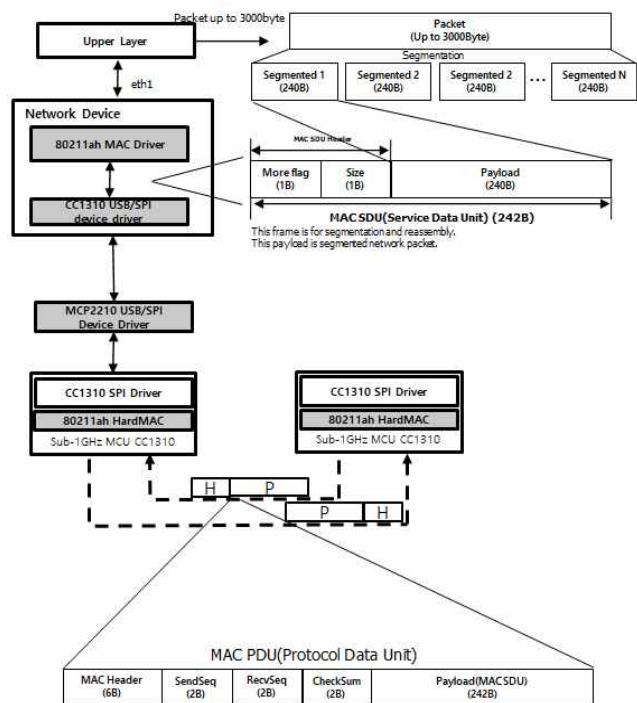
Sub-1GHz 무선 인터넷 중계기를 구현하기 위해선 전체적으로 두가지의 MAC Frame이 존재한다. 첫 번째는 (그림 9)에서 Upper Layer 블록을 통해 들어오는 패킷을 Segmentation & Reassembly를 하기 위해서 사용 되는 Frame인 MAC SDU(Service Data Unit)이고, 또 다른 프레임은 CC1310간의 RF TDD 통신에서 요구되는 MAC PDU(Protocol Data Unit)이다. MAC SDU(Service Data Unit)의 구조는 (표 1)과 같다. 1byte의 more flag는 reassembly시 어디서부터 어디까지가 하나의 패킷인지를 알려주는 요소이고, 1byte의 size는 payload의 사이즈를 나타낸다. 240byte Payload는 segmentation후의 조각난 packet을 담은 실제 데이터 부분을 뜻하는 요소이다. 그리고 MAC PDU(Protocol Data Unit)의 구조는 (표 2)와 같다. (표 2)에서 6byte MAC Header는 주소 정보와 같은 MAC프로토콜을 구성하기 위한 기본적인 정보들이 들어있는 요소이고, 2byte의 Send Sequence Number와 Receive Sequence Number는 Piggy Back ARQ를 위한 송수신 정보를 담은 요소이다. 2byte의 CheckSum은 error control을 위한 checksum정보를 담은 요소이다. 242 byte의 Payload는 MAC SDU 정보를 담은 요소이다. MAC SDU의 헤더 정보에 해당하는 more flag와 size 정보는 Reassembly시 필요한 정보이기 때문에 MAC PDU에 포함이 되어야 한다.

크기(byte)	요소	내용
1	More Flag	Reassembly를 위한 more flag
1	Size	Payload Size
240	Payload	segmented packet

(표 1) MAC SDU Structure

크기 (byte)	요소	내용
6	MAC Header	Include address information
2	Send Sequence Number	Send Sequence Number
2	Receive Sequence Number	Receive Sequence Number
2	CheckSum	check sum for error control
242	PayLoad	MAC SDU

(표 2) MAC PDU Structure



(그림 9) Device Driver Detail Software Block Diagram

IV. 성능 평가

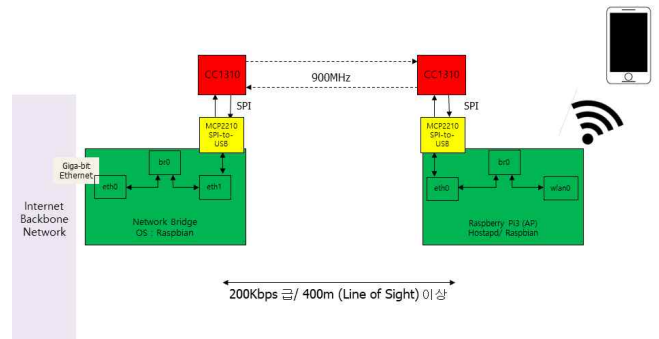
4.1 Raspberry Pi 기반 Sub-1GHz 무선 인터넷 중계기 성능측정

무선 인터넷 중계기의 성능측정을 위해 Raspberry Pi 와 RF module을 각각 두 개씩 준비하고 유선랜 접속이 불가능한 지역에 노드를 구성한다. 그리고 노드에는 hostapd로 만든 WIFI AP에 접속할 Station 으로 사용할 스마트폰을 준비한다.

RF module인 CC1310은 SPI Interface를 이용해 Raspberry Pi 와 연결하였다. CC1310칩의 중심주파수는 917MHz, 전송 전력은 26dBm로 설정하였다. 그리고 RF 채널의 전송속도는 500Kbps로 설정을 하였다.

각각의 Raspberry Pi의 Network Interface에 대한 설정은 Bridge side Raspberrry Pi에서는 제작한 network interface와 ethernet network interface를 bridge util을 사용해 소프트웨어적으로 이어준다. 그리고 Extender side Raspberry Pi에는 제작한 network interface와 wireless LAN network interface를 hostapd설정을 통해 소프트웨어적으로 이어줌으로써 실험 환경을 구성한다.

최종 구성도는 다음의 (그림 10)과 같다.



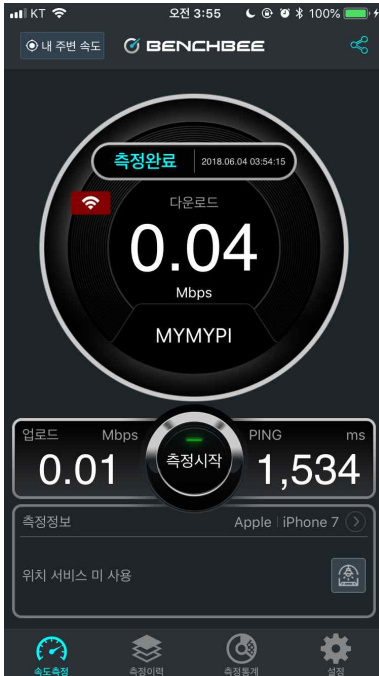
(그림 10) 최종 구성도

4.2 성능평가 및 결과분석

본 논문에서 구성한 시스템은 유선랜 접속이 불가능한 지역에 900MHz 통신을 이용해 WiFi AP를 만드는 것이다. 따라서 보다 정확한 측정을 위해 WiFi 성능 측정 애플리케이션인 벤치비를 이용하여 측정하였다. 측정 결과는 (그림 11)와 같다.

RF 채널의 전송속도는 이론적으로 500Kbps까지이지만 벤치비 애플리케이션을 사용해 실제 구현결과물의 성능평가를 통해 얻은 최대 전송속도는 다운로드 속도는 40kbps ~ 70kbps정도로 측정됨을 확인 할 수 있었고 업로드 속도는 10kbps 정도로 확인 할 수 있었다. 이론적 속도에 비해 굉장한 성능 저하가 있는데 이는 smart phone과 주고받는 data packet만

RF통신을 통해 송수신을 하는 것이 아니고 network control packet도 RF 통신을 통해 송수신을 하기 때문에 심각하게 속도가 저하 될 수 밖에 없다는 이유를 제시할 수 있다.



(그림 11) 중계기 성능 분석

V. 결 론

본 논문에서는 시골의 논과 밭, 건설 현장 등과 같은 반경 100m 이내에 유선 랜 접속이 불가능한 야외 작업 환경에서 반경 1km의 범위를 가진 900MHz 대역의 RF 모듈을 이용하여 인터넷 중계 기능을 제안했다. 기존의 야외 작업 환경에서는 유선랜, Wi-Fi 등을 사용해도 100m내외의 거리에서만 통신이 가능했지만, 본 논문의 900MHz RF 채널을 이용하면 1Km의 통신가능반경을 가질 수 있어 야외 작업 환경 종사자 분들이 좀 더 나은 환경에서 일을 할 수 있도록 해준다.

앞으로 추가적인 연구 사항은 1Km 이상의 거리에서 통신이 가능한지 확인하는 것, 여러 AP가 하나의 브릿지에 연결할 수 있는 서비스를 제공하는 것이다. 그리고 속도 향상을 위해 몇몇 Network Control Packet을 필터링해 주는 기능을 넣어야 할 것이다.

감사문

본 연구는 대한민국 영남대학교 기계IT 대학 정보통신공학과와 LINC 사업의 CDP (Capstone Design Project) 지원 아래 수행하였습니다.

참고문헌

[1] MCP2210 DataSheet, Microchip.

[2] CC1310 DataSheet, TI.

[3] 김기태, *HardMAC-based Smart Control of Transmission Speed and Power for Sub-1GHz IoT Networking*, 영남대학교 일반대학원 정보통신공학과 석사학위논문, 2018 4.

[4] 정용환, *Linux Device Driver of CSMA_CA-TDMA Hybrid MAC for IoT Networking based on IEEE 802.11ah*, 영남대학교 일반대학원 정보통신공학과 석사학위논문, 2018 4.

[5] mcp2210-linux (mcp2210 library), Daniel-santos, 2017.

[6] Christian Benvenuti, *Understanding Linux Network Internals*, O'REILLY, 2006

[7] openWRT doxygen, "Wi Fi client mode", <https://wiki.openwrt.org/doc/howto/clientmode>

[8] doxygen, "Wi Fi Extender", https://www.100mb.kr/bbs/board.php?bo_table=information&wr_id=4002

[9] WIKIPEDIA, "802.11ah" https://en.wikipedia.org/wiki/IEEE_802.11ah