



2024 KBS 미디어창의기술전 대상 수상작 LTE와 MQTT를 이용한 Tally System 구현

글. 오원섭 KBS 제작기술국

중계현장에서 사용자가 간편하게 활용할 수 있는 무선 Tally 시스템을 만들기 위해 프로젝트를 시작하게 되었다. 이미 KBS 중계기술국에서는 Lora를 이용한 Tally를 개발해 사용하고 있었다. 현장에서 충분히 사용 가능했지만, 공연장처럼 중계차와 제작 현장의 거리가 멀거나 중계 현장이 다른 지역에 있는 상황을 고려하여 'LTE로 한번 만들어볼까?'란 아이디어 하나로 시작된 프로젝트가 거의 1년 남짓 시간이 흘러 'LTE를 활용한 Tally 시스템'이 만들어졌다.



그림 1. Lora를 이용한 Tally 장치

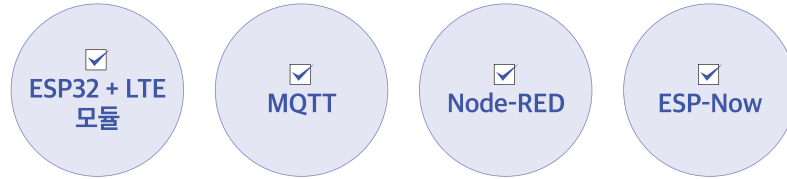
기본 구성

Tally 시스템은 비디오 스위처에서 특정 카메라의 영상 신호를 선택할 때, 해당 카메라를 운영하는 스태프에게 빨간 불빛으로 PGM 선택 상태를 알리는 시스템이다. 비디오 스위처에서 점점 신호를 받아 이를 전송하고, 수신된 신호를 카메라 운용자가 인식할 수 있도록 표시해주는 장치를 만드는 것이 이 프로젝트의 목적이었다.

장비가 설치와 운용이 복잡하면 점차 사용하지 않게 되므로, 이를 고려해 몇 가지 목표를 설정하고 작업을 시작했다. 송신기에 XLR 커넥터를 부착해 오디오 케이블로 간단히 연결할 수 있도록 설계하였다. 또한, Wi-Fi 대신 LTE 망을 우선적으로 사용해 네트워크 연결 작업을 간소화했다. 수신기도 전원 버튼만 누르면 별도 설정 없이 사용할 수 있도록 하고 최대한 지연시간을 줄여 방송제작에 문제가 없도록 하였다. 현재 포터블 1대와 랙 형태로 2대의 송신기를 만들어서 활용하고 있고 수신기는 상황에 맞게 필요한 만큼 세팅하여 사용하고 있다.

시스템 구성

ESP32 모듈 하나 가지고 시작한 작업이 장치를 만들고 테스트하면서 여러 기술들을 추가로 사용하게 되었다.



ESP32 모듈은 Tally 점점 신호를 받아서 송신하고 수신하는 역할을 한다. 송신기와 수신기는 입력을 받는 부분과 출력하는 I/O (input/output) 부분만 다르고 거의 동일하다. 문자메시지로 처리하려던 계획은 신호 지연 문제로 MQTT를 사용하게 되었다. MQTT 관리와 테스트 그리고 웹과의 연동을 위해 Node-RED를 사용하게 되었고 근거리에서 더 빠른 신호전달을 위해 ESP-Now(근거리 ESP 자체통신)를 추가로 적용하였다.

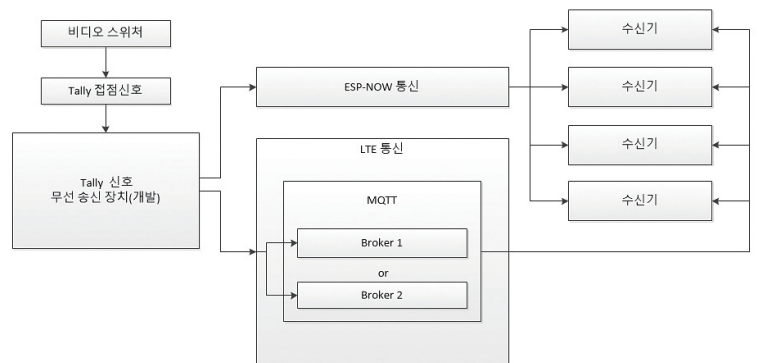


그림 2. Tally 시스템 구성도

ESP32 및 LTE 모듈

송수신기는 ESP32를 사용하였다. ESP32는 듀얼 코어 프로세서와 Wi-Fi, Bluetooth 기능을 갖춘 마이크로컨트롤러로, IoT 프로젝트에서 널리 사용된다. 저전력 소비와 여러 입출력 포트를 지원해 다양한 센서와 장치를 제어할 수 있다. 아두이노보다 기능이 많은 단일 프로세서라고 할 수 있다. 무엇보다 크기가 비교적 작고 프로그램이 업로드되면 긴 부팅 시간 없이 바로 사용이 가능하다.

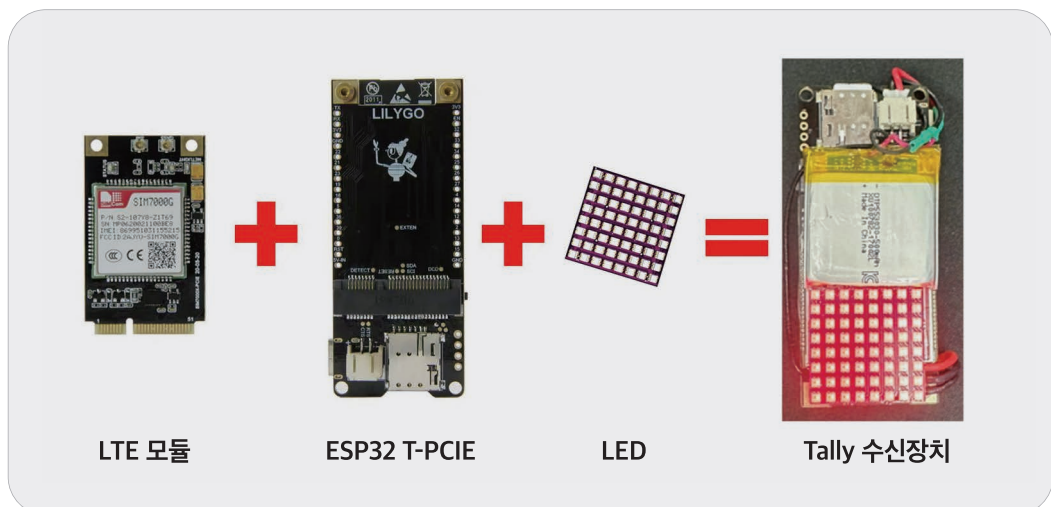


그림 3. Tally 수신장치 구성

ESP32도 버전이 여러 개 존재하는데 LTE 모듈을 장착하기 위해서 ESP32 T-PCIE를 선택하였다. LTE 모듈도 종류가 많은데 한국에서 사용할거면 Sim7600E 표시가 되어 있는 장비를 사용해야 한다. 7600E 모듈은 유럽, 남아시아, 남아메리카에서 사용할 수 있다.

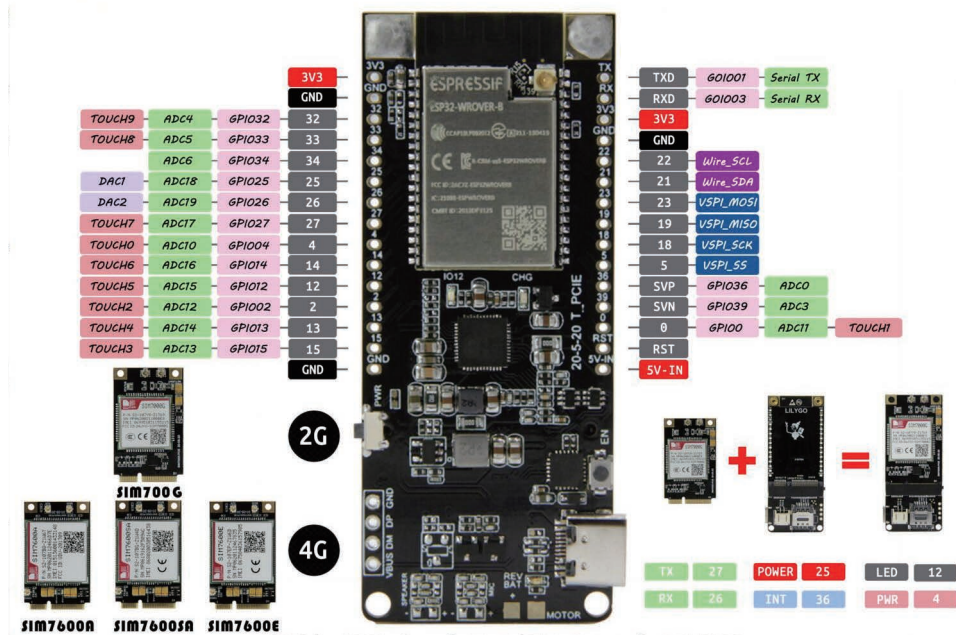


그림 4. ESP32 T-PCIE PINMAP

ESP32 프로그래밍은 아두이노 IDE를 사용했다. 초기 Wi-Fi 연결 코드의 참고 자료는 풍부하고 잘 동작했지만, LTE 모듈 관련 자료는 공개된 내용이 적어 제작 환경에 맞는 자료를 찾기가 어려웠다.

결국, 여러 시도를 통해 경험적으로 데이터를 확보해야 했다. ESP32에 LTE 모듈을 연결해 인터넷을 연결하는 데 많은 시간이 걸렸다. LTE 모듈은 ESP32에 연결하고 관련 IDE 프로그램을 업로드하여 커맨드 모드(Command mode)로 기본 설정과 동작 여부를 확인해야 사용이 가능하다.

Frequency Band Reference					
Product	REGION	FrequencyBands			
		LTE-FDD	LTE-TDD	WCDMA	GSM
SIM7600G	Global	B1/B2/B3/B4/B5/B7/B8/B12/B13/B18/B19/B20/B25/B26/B28/B66	B34/B38/B39/B40/B41	B1/B2/B4/B5/B6/B8/B19	850/900/1800/1900MHz
SIM7600E	Europe SoutheastAsia SouthAmerica	B1/B3/B5/B7/B8/B20	B38/B40/B41	B1/B5/B8	900/1800MHz
SIM7600A	NorthAmerica Canada	B2/B4/B12	-	B2/B5	-
SIM7600SA	SouthAmerica NewZealand Australia	B1/B2/B3/B4/B5/B7/B8/B28/B66	B40	B1/B2/B5/B8	850/900/1800/1900MHz
SIM7600NA	NorthAmerica Canada	B2/B4/B5/B12/B13/B14/B25/B26/B66/B71	B41	-	-
SIM7600JC	Japan	B1/B3/B8/B18/B19/B26	-	-	-
Data Transfer Differences					
Module	FrequencyBands				
	LTE(Mbps)	HSPA+(Mbps)	WCDMA(Kbps)	GPRS/EDGE(Kbps)	
SIM7600X	10(DL)/5(UL)	42(DL)/5.76(UL)	384(DL)/384(UL)	236.8(DL)/236.(UL)	
SIM7600X-H	👍 150(DL)/50(UL)	42(DL)/5.76(UL)	384(DL)/384(UL)	236.8(DL)/236.(UL)	

그림 5. LTE 모듈 종류와 주파수

아두이노 IDE 프로그램은 ESP32 보드에서 어떤 기능을 사용할지에 따라 관련 라이브러리를 설치해야 작동한다. 관련 라이브러리의 업데이트가 IDE 프로그램의 업데이트를 따라가지 못해 최신 IDE 프로그램에서는 동작이 잘 안되는 문제도 발생하니 라이브러리가 어느 버전까지 동작하는지 체크가 필요하다. 최신 프로그램으로 업데이트하는 것이 항상 최선은 아니다. 무심코 업데이트 버튼을 눌렀다가 며칠 동안 문제의 원인을 찾느라 애를 먹을 수도 있으니 주의가 필요하다.

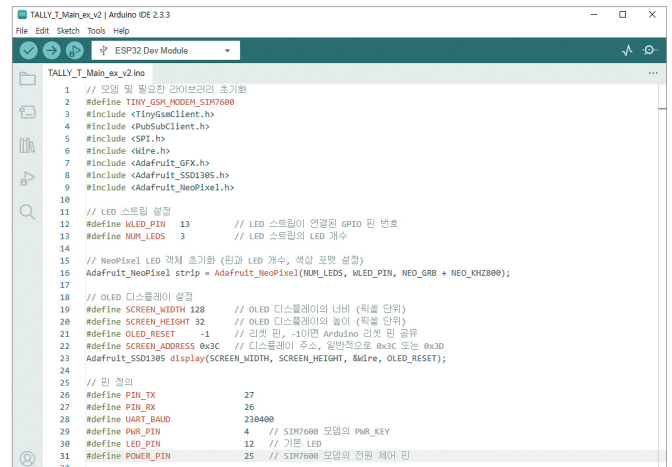


그림 6. 아두이노 IDE 프로그램

MQTT

MQTT(Message Queuing Telemetry Transport)는 IoT 기기 간의 통신을 위해 설계된 경량 메시지 프로토콜이다. 낮은 대역폭과 전력 소모 환경에서도 안정적으로 데이터를 전송할 수 있도록 설계되었으며, Publish/Subscribe 모델을 사용하여 데이터 송신자(Publisher)와 수신자(Subscriber) 간의 연결을 관리한다. 브로커(Broker)라는 중앙 서버가 있어, 모든 메시지가 브로커를 통해 전달되어 각 클라이언트는 직접 연결될 필요 없이 메시지를 주고받을 수 있다. 이로 인해 확장성이 뛰어나고 실시간성이 요구되는 IoT 환경에서 널리 사용된다. MQTT는 작은 데이터 패킷과 단순한 구조 덕분에 저전력 및 낮은 네트워크 속도에서도 효율적으로 작동하며, 주로 스마트 홈, 산업 자동화, 원격 모니터링 등의 IoT 애플리케이션에서 활용된다. 기본적인 동작은 Publisher가 특정 그룹(topic)으로 메시지를 보내면 Broker(서버)가 해당 topic을 구독하는 Subscriber에게 메시지를 전달하는 방식이다.

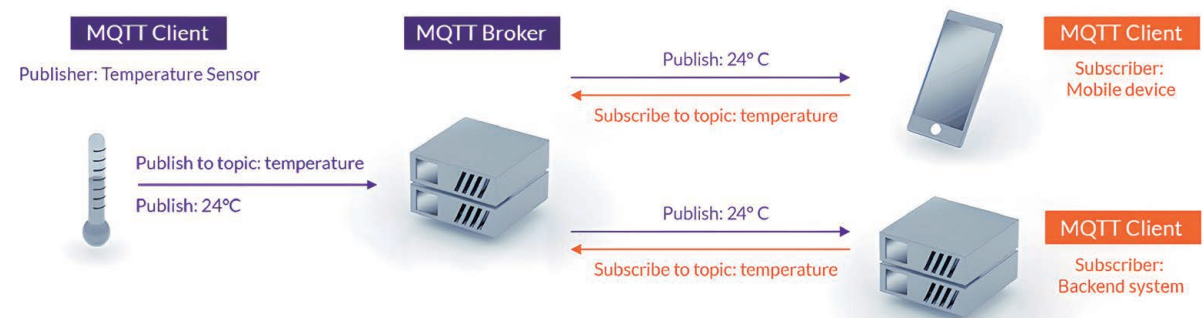


그림 7. MQTT 발행/구독 구조

ESP32의 인터페이스에 비디오 스위치의 GPO의 신호를 연결하고 점점 신호가 들어오면 MQTT 서버로 특정 문자를 보내고 MQTT 서버(브로커)는 수신기(구독자)에 해당 문자를 보내면 수신기는 그 문자를 판단해서 Tally On인지 Off인지 판단하여 LED에 적색 램프를 On/Off 하도록 프로그래밍을 하였다.

구독 관리는 토픽(Topic)으로 구분된다. MQTT에 연결할 때 어느 토픽을 구독할지 브로커에게 전달하도록 되어 있다. 여러 토픽을 구독해도 되고 하나의 토픽을 구독해도 된다. 브로커는 해당 토픽으로 발행된 메시지가 있으면 해당 토픽을 구독하는 모든 장비에 메시지를 보내도록 되어 있다.

MQTT는 일단 연결이 되면 설정에 따라 접속여부를 지속적으로 체크하게 되어 있어 접속이 끊기게 되면 재접속을 하도록 프로그램을 하였다.

테스트 과정에서 MQTT Broker가 필요해 Mosquitto를 설치하여 자체적으로 구축했다. 하지만 이를 항상 운영해야 한다는 부담이 있어 Public server를 제공하는 MQTT 서비스 업체의 공용 MQTT Broker(서버)를 이용하기로 결정을 하고 HiveMQ와 EMQX를 테스트하였다. EMQX에 접속을 시도하고 접속이 안 될 경우 HiveMQ에 접속하도록 하고 송신기와 수신기에 어떤 서버에 접속했는지 표시하도록 코딩하였다. 우선순위는 현재 EMQX 먼저 접속을 시도하도록 했고 MQTT 접속이 끊기면 3번 정도 다시 시도하고 HiveMQ에 접속하도록 설정하였다.



그림 8. 수신기의 단계별 표시 내용과 의미

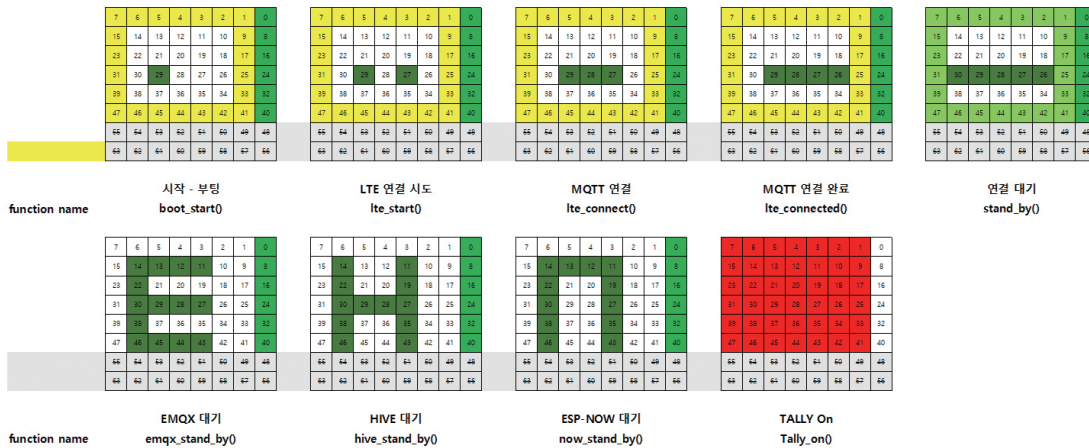
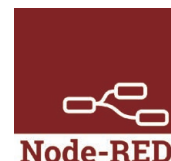


그림 9. 수신기의 단계별 표시 내용과 의미

실제 방송 현장에서 2024년 3월부터 사용하기 시작했으며, 이후 꾸준히 업그레이드를 진행했다. 수신기의 LED를 교체하여 수신기의 상태를 표시할 수 있도록 보강하였다. 최근에는 배터리 충전 용량을 표시할 수 있도록 저항을 이용해 전압을 측정하고 이 값을 아날로그 단자에 연결하여 수치를 입력받아 실제 전압과 비교하여 보정값을 적용해 배터리 상태를 표시하는 기능을 추가했다.

Node-RED

장비 운용에는 특별히 문제는 없었는데 테스트 과정과 현재 MQTT 상태를 모니터링하고 싶은 생각이 들어 Node-RED라는 프로그램을 이용하게 되었다. Node-RED는 오픈소스 비주얼 프로그래밍



그래밍 도구로, 주로 사물인터넷(IoT), API 통합, 자동화 작업에 사용된다. 웹 브라우저 기반의 흐름 기반 개발 환경을 제공하며, 간단한 드래그 앤드 드롭 방식으로 프로그래밍을 할 수 있다.

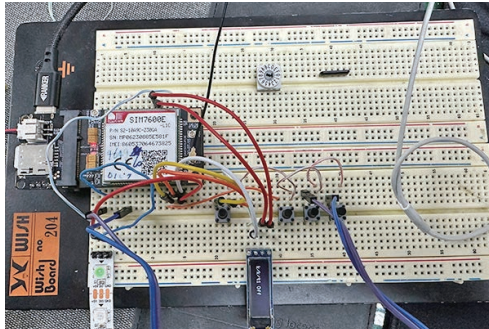


그림 10. 브레드보드에서 테스트

그림처럼 직접 스위치를 만들어서 점점 신호를 보내야 테스트가 가능했지만, Node-RED를 이용하면 웹상에서 MQTT 신호를 발생시키고 수신되는 신호 확인도 가능하다. 신호 송수신 유무를 바로 확인할 수 있으며 방송현장에서 잘 작동하고 있는지 모니터링도 가능하다. Node-RED는 모바일 앱도 제공하지만, 추가 요금이 발생하기 때문에 사용하지 않고 있다.



그림 11. Node-RED 프로그래밍

드래그 앤드 드롭 방식으로 기능을 배치하고 선으로 연결해서 흐름을 작성하고 처리 관련 코드를 작성하여 사용한다. MQTT와 대시보드를 사용하기 위해서는 해당 라이브러리를 설치해야 한다. Node-RED에서는 팔레트 창에 가서 노드를 추가해야 한다.

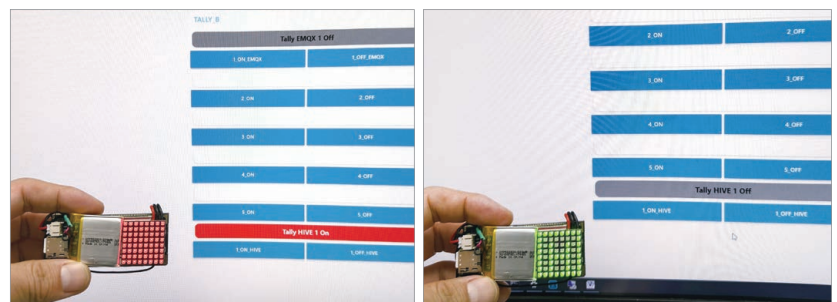


그림 12. Node-RED Dashboard 화면

On/Off 버튼을 만들어서 버튼을 누르면 MQTT의 해당 토픽으로 On일 때는 1, Off일 때는 0을 보내고 해당 토픽에 수신된 신호가 있으면 그 신호가 1인지 0인지 함수에서 판단하여 템플릿에서 색으로 표시하도록 설정한 화면이다.



그림 13. Vmix를 활용하여 리턴신호로 송신

Node-RED를 사용하게 되면서 수신기가 없는 경우에도 웹 화면으로 신호를 확인할 수 있어 Vmix와 연동해서 리턴신호에 Tally 신호 표시도 가능하겠다는 생각이 들어 2024 파리올림픽에서 현장 커멘터리(펜싱, 수영, 양궁) 리턴 화면에 Tally 신호를 적용하여 커멘터리석에 전송할 수 있었다.

Node-RED를 실행하면 해당 PC의 IP 주소로 웹 개발화면과 대시보드 화면에 접속이 가능하다. 대시보드 웹화면 주소를 Vmix web 페이지로 불러와서 해당 부분을 적당히 조절해서 PGM 화면에 Overlay 하여 리턴 신호로 전송하였다.

ESP-NOW

LTE는 거리 제약이 없고 별도의 설정 없이 사용할 수 있는 장점이 있지만, LTE 망과 MQTT 서버를 사용하기 때문에 신호가 VMU → ESP32 → MQTT → ESP32로 전달되는 과정에서 약간의 지연이 발생한다. 해외에서도 사용이 가능한 장점이 있지만 중계차 근처에서 제작할 경우 굳이 LTE 망을 사용하지 않고 ESP 내부에 ESP-Now라는 자체 Wi-Fi 기반 통신방식을 사용하면 MQTT로 신호를 송수신하는 시간을 줄일 수 있어 송신기와 수신기 간 ESP-Now 통신이 가능할 경우에는 ESP-Now 통신으로 동작하고, ESP-Now 신호가 연결이 안 될 경우에는 LTE로 작동하도록 프로그래밍을 수정하였다. 그리고 수신기 프로그램을 변경하여 중계기 역할을 하는 장치를 만들어 전송거리를 증가시켜 중계현장에서 적용할 수 있도록 제작하였다.

중계기는 송신기에서 발생한 신호를 받아 수신기에 ESP-NOW 신호(A1, A0와 같은 신호)를 수신기로 전달하는 방식으로 동작한다. Lora를 사용하면 전송거리가 늘어나겠지만 수신기의 크기와 안테나 등을 설치하지 않기 위해 내장된 ESP-NOW를 활용하는 방향으로 추가 기능을 넣었다. ESP-NOW는 ESP32 장치의 맥어드레스를 이용해 통신한다. 송신기에서 수신기와 중계장치로 신호를 보내고 수신기는 이 신호가 수신되면 MQTT 수신 관련 동작을 일정 시간 처리하지 않도록 코딩하였다. 최근에는 ESP32에 내장된 안테나 출력을 높이기 위해 외장 안테나를 부착할 수 있는 장치를 구매하여 테스트하였고 송신기에 부착해서 MQTT와 ESP-NOW 신호 전송을 분리해서 전송하도록 업그레이드할 예정이다.

유심 관리

LTE 통신을 하려면 유심을 구매해야 했다. Hologram 유심을 구매해 사용했다. MQTT로 전송하는 데이터는 A1, A0, B1, B0처럼 최대한 간단한 문자 정보만 보내기 때문에 데이터 사용량이 많지 않았다. 데이터 사용량도 각각 모니터링이 가능하고 잔고가 부족하면 자동으로 충전하는 기능도 있어 유심 관리가 용이했다. 요금제는 여러 가지 요금제가 있어 환경에 맞게 선택이 가능하다. 하지만 영상이나 음성 데이터같이 지속해서 많은 데이터 전송은 주의가 필요하다.

DEVICE NAME	STATUS	LAST ACTIVE	DATA USAGE	DATA PLAN	IMEI
09_KBS_M_Tally6 (83933)	Ready	a month ago	—	Global Pay as you Go	
08_KBS_M_Tally5 (83230)	Ready	a month ago	—	Global Pay as you Go	
06_KBS_M_Tally3 (57459)	Ready	a month ago	—	Global Pay as you Go	
03_KBS_M_Tally_Main (82527)	Ready	4 days ago	234 KB	Global Pay as you Go	
05_KBS_M_Tally2 (82618)	Ready	7 days ago	19.42 KB	Global Pay as you Go	
07_KBS_M_Tally4 (82840)	Ready	18 days ago	15.88 KB	Global Pay as you Go	
02_KBS_Tally1 (68307) park	Ready	7 hours ago	293.46 KB	Global Pay as you Go	
01_KBS_Tally Main (68315)	Ready	a month ago	—	Global Pay as you Go	

그림 14. Hologram 유심 관리 화면

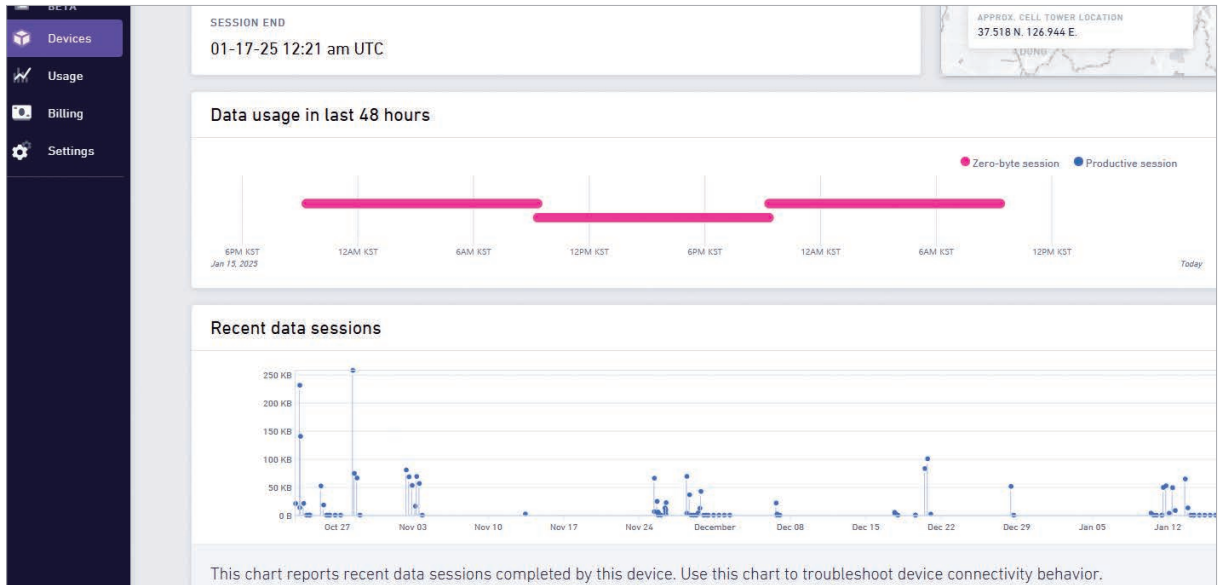


그림 15. 유심 사용량 관리

Hologram 사이트에서 유심 요금, 사용 내역, 활성화 등 조회가 가능하다. 비용은 유심 구입비용 약 1만 원이고 여러 요금제가 있는데 매월 0.7\$에 1Mbyte 당 0.19\$ 요금제를 사용하고 있고 데이터가 많지 않기 때문에 저렴한 비용으로 사용 가능하다. 일단 유심은 활성화를 하면 매달 요금(0.7\$/month)이 나간다. 유심은 정지 상태에서도 요금이 부과된다. 해지하면 요금이 안 나가지만 그 유심은 다시 사용할 수 없다. 유심 구매 비용이 약 1만 원 정도 하기에 요금을 잘 고려해서 운영해야 한다.

현장 활용

W/L 카메라, 와이어캠, 드론과 같은 특수 카메라가 설치되는 슈퍼레이스, 축구, 배구, 야구 중계방송에 활용하고 있고 랙형 장치는 중계차에 장착 예정이다. 2024 프랑스 파리올림픽에서는 방송센터에 설치하여 중계현장 커멘터리의 리턴영상에 Tally 신호를 합성하여 전송하였다.



그림 16. 2024 경주마라톤 RC-CAR 조종기에 Tally 수신기 부착

Tally System



그림 17. 마라톤 바이크 카메라에 수신기 장착, 초기 버전



그림 18. 짐벌 카메라에 수신기 장착

송신기, 수신기 변화 과정

1년 넘게 장치를 만들다 보니 송신기, 수신기의 디자인 변화가 많이 있었다. 초기에는 기능 위주로 만들다 보니 초기 버전은 HDD 케이스에 케이블 타이와 글루건으로 고정을 해서 만들었으나 현재는 3D 프린터를 이용해서 만든 포터블용 송신기와 랙 형태로 만든 2종류의 송신기가 만들어졌다. 1RU 랙을 구매해서 커넥터와 디스플레이 사이즈에 맞게 구멍을 뚫어 제작하였다. 중계차에서 인터넷을 사용할 수 있을 경우를 대비해서 유선 인터넷과 Wi-Fi를 사용할 수 있도록 기능을 추가하는 중이다. 유선 → Wi-Fi → LTE 순으로 부팅 시 망을 확인하면 좀 더 효율적인 운영이 가능하다.

송신기



HDD 케이스 버전



보조배터리 케이스 버전



3D 프린트 버전



현재 - 1RU Rack 버전

그림 19. 송신기 변화 과정



그림 20. 포터블 송신기 세트

수신기

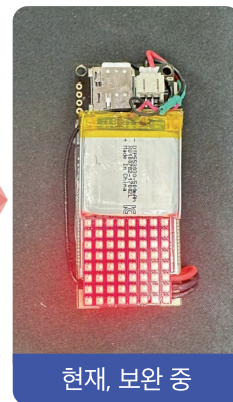
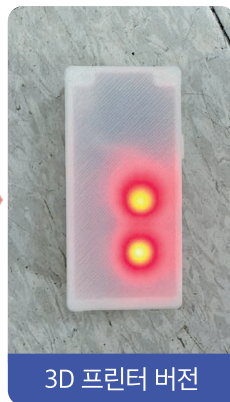


그림 21. 수신기 변화 과정

마무리

LTE 망을 활용해 Tally 신호를 전송하는 기술에서 시작해 관련 기술을 적용한 결과, LTE 기반 IoT 장치가 개발되었다. 이 장치는 점점 신호, 아날로그 신호, 디지털 신호나 문자데이터를 입력받아 전송하거나, 수신된 신호를 처리할 수 있고 Node-RED를 적용해 웹에서도 접근이 가능하다. 추후에 중계차에 장착해서 LTE 모듈에 내장된 GPS 수신장치를 활용해 위치 추적과 중계차의 전력, 온도, 습도, CO₂ 농도 등을 모니터링하고 제어하는 IoT 기능도 구현할 수 있을 거 같아 기능을 추가할 예정이다.

참고

- mqtt.org
- nodered.org
- www.espressif.com
- www.hologram.io