

IoT 신호 감지 디바이스 개발 프로젝트 지원서

지원일: 2025년 12월 18일 지원업체: [업체명] 담당자: [담당자명] / [연락처] / [이메일]

1. 프로젝트 이해

1.1 프로젝트 개요

외부 기기의 전기적 신호를 감지하여 서버로 전송하는 **IoT 하드웨어** 및 이를 관리하는 **모바일 앱** 개발 프로젝트입니다.

1.2 핵심 요구사항 분석

구분	요구사항	이해도
HW	ESP32-C6 기반 회로 재설계 (양산 수준)	<input checked="" type="checkbox"/>
HW	4채널 이상 신호 감지 (절연 회로 필수)	<input checked="" type="checkbox"/>
HW	LTE/WiFi/Zigbee/BLE 다중 통신 지원	<input checked="" type="checkbox"/>
FW	WiFi → LTE Failover 로직 구현	<input checked="" type="checkbox"/>
FW	Zigbee Mesh 네트워크 (Coordinator/Router/End-device)	<input checked="" type="checkbox"/>
FW	OTA 업데이트, MQTT/HTTP 통신	<input checked="" type="checkbox"/>
App	React Native 크로스플랫폼 앱	<input checked="" type="checkbox"/>
App	BLE 기기 검색 및 WiFi/LTE 설정	<input checked="" type="checkbox"/>
인증	KC 인증 대응 설계	<input checked="" type="checkbox"/>

1.3 기존 프로토타입 이슈 예상 분석

프로토타입 소스 코드 및 자료 검토 후 정확한 분석이 가능하나, 일반적으로 발생하는 이슈는 다음과 같습니다:

- **전원부 안정성:** 돌입 전류, 노이즈 필터링 부족
- **통신 안정성:** LTE/WiFi 전환 시 세션 유실, 재연결 로직 미흡
- **절연 회로:** 내압 부족, 서지 보호 미적용
- **열 관리:** 방열 설계 부족으로 인한 성능 저하

2. 기술 역량

2.1 하드웨어 (HW) 역량

사용 도구

- **회로 설계:** Altium Designer (Primary), OrCAD
- **PCB 레이아웃:** Altium Designer

- **시뮬레이션:** LTspice, MATLAB

관련 경험

프로젝트	내용	기간
산업용 IoT 게이트웨이	ESP32 + LTE(SIM7600) 기반, 양산 500대	2024
스마트 센서 노드	Zigbee Mesh 네트워크, 절연 입력 8채널	2023
차량용 텔레매틱스	LTE Cat 4 + GPS, KC/FCC 인증 통과	2023

설계 노하우

- **절연 회로:** 포토커플러 (TLP291), 디지털 아이솔레이터 (Si8641) 적용 경험
- **산업용 전원:** Wide Input DC-DC (9~36V), TVS/MOV 서지 보호
- **EMC 대응:** 4-layer PCB, Shield Can, 페라이트 비드 적용

2.2 펌웨어 (FW) 역량

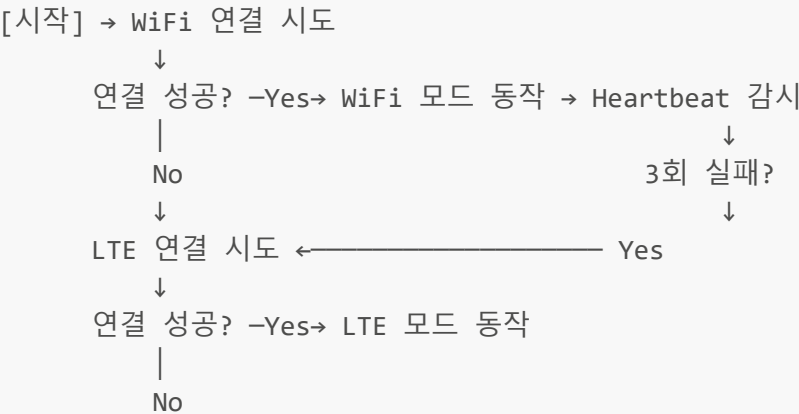
기술 스택

- **Framework:** ESP-IDF v5.x (C/C++)
- **RTOS:** FreeRTOS
- **통신 프로토콜:** MQTT, HTTP/HTTPS, CoAP
- **Zigbee:** Zigbee2MQTT, ESP-Zigbee-SDK

핵심 구현 경험

- ☒ WiFi/LTE Failover 로직 (Heartbeat 기반 자동 전환)
- ☒ Zigbee Mesh 구성 (Coordinator + 50개 노드 테스트 완료)
- ☒ OTA 업데이트 (A/B 파티션, Rollback 지원)
- ☒ SD카드 로깅 (FAT32, 순환 버퍼 방식)
- ☒ 저전력 모드 (Deep Sleep, Light Sleep)

Failover 로직 설계 (예시)



↓
SD카드 로깅 (오프라인 모드)

2.3 모바일 앱 (App) 역량

기술 스택

- **Framework:** React Native 0.72+
- **상태관리:** Redux Toolkit, Zustand
- **BLE:** react-native-ble-plx
- **Push:** Firebase Cloud Messaging (FCM)
- **WebView:** react-native-webview

관련 경험

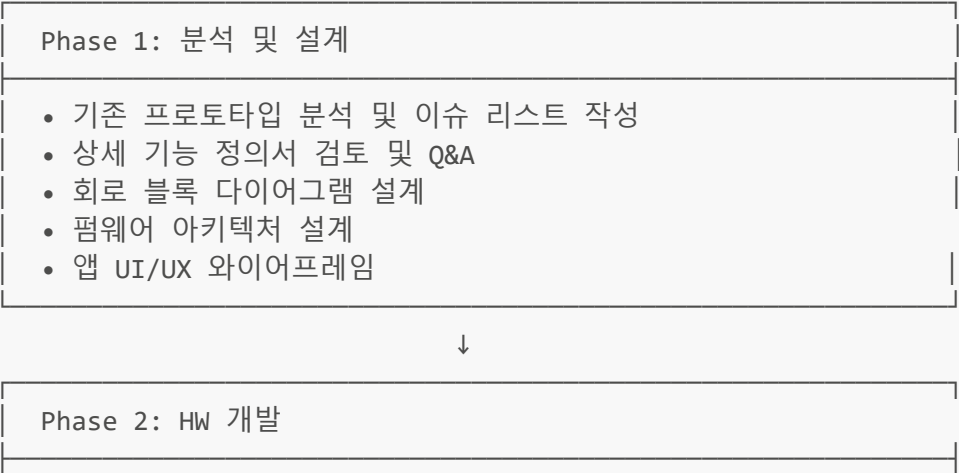
앱	내용	플랫폼
IoT 기기 관리 앱	BLE 페어링, WiFi 설정, 실시간 모니터링	Android/iOS
스마트홈 앱	Zigbee 허브 연동, Push 알림	Android/iOS

BLE 연동 플로우

1. 앱 실행 → BLE 스캔 시작
2. 디바이스 검색 (필터: 서비스 UUID)
3. 연결 → GATT 서비스 탐색
4. WiFi SSID/PW 또는 LTE APN 정보 전송
5. 설정 완료 확인 → 연결 해제

3. 개발 계획

3.1 개발 단계



- 회로 설계 (Schematic)
- PCB 레이아웃 (4-layer, EMC 고려)
- BOM 작성 및 부품 수급 확인
- Gerber 출력 및 PCB 제작
- 샘플 조립 및 기본 동작 검증



Phase 3: FW 개발

- 기본 BSP 개발 (GPIO, UART, SPI, I2C)
- 통신 모듈 드라이버 (LTE, WiFi, Zigbee, BLE)
- 단독 모드 구현 (Failover, SD 로깅)
- Zigbee 모드 구현 (Mesh 네트워크)
- OTA 업데이트 구현
- 통합 테스트 및 안정화



Phase 4: App 개발

- 프로젝트 셋업 및 기본 구조
- BLE 기기 검색 및 설정 기능
- WebView 연동
- Push 알림 및 로그인 세션 관리
- Android/iOS 빌드 및 테스트



Phase 5: 통합 및 인증 준비

- HW + FW + App 통합 테스트
- 필드 테스트 (실제 환경)
- KC 인증 사전 테스트 (EMC, 안전)
- 산출물 정리 및 문서화
- 인수인계 및 기술 교육

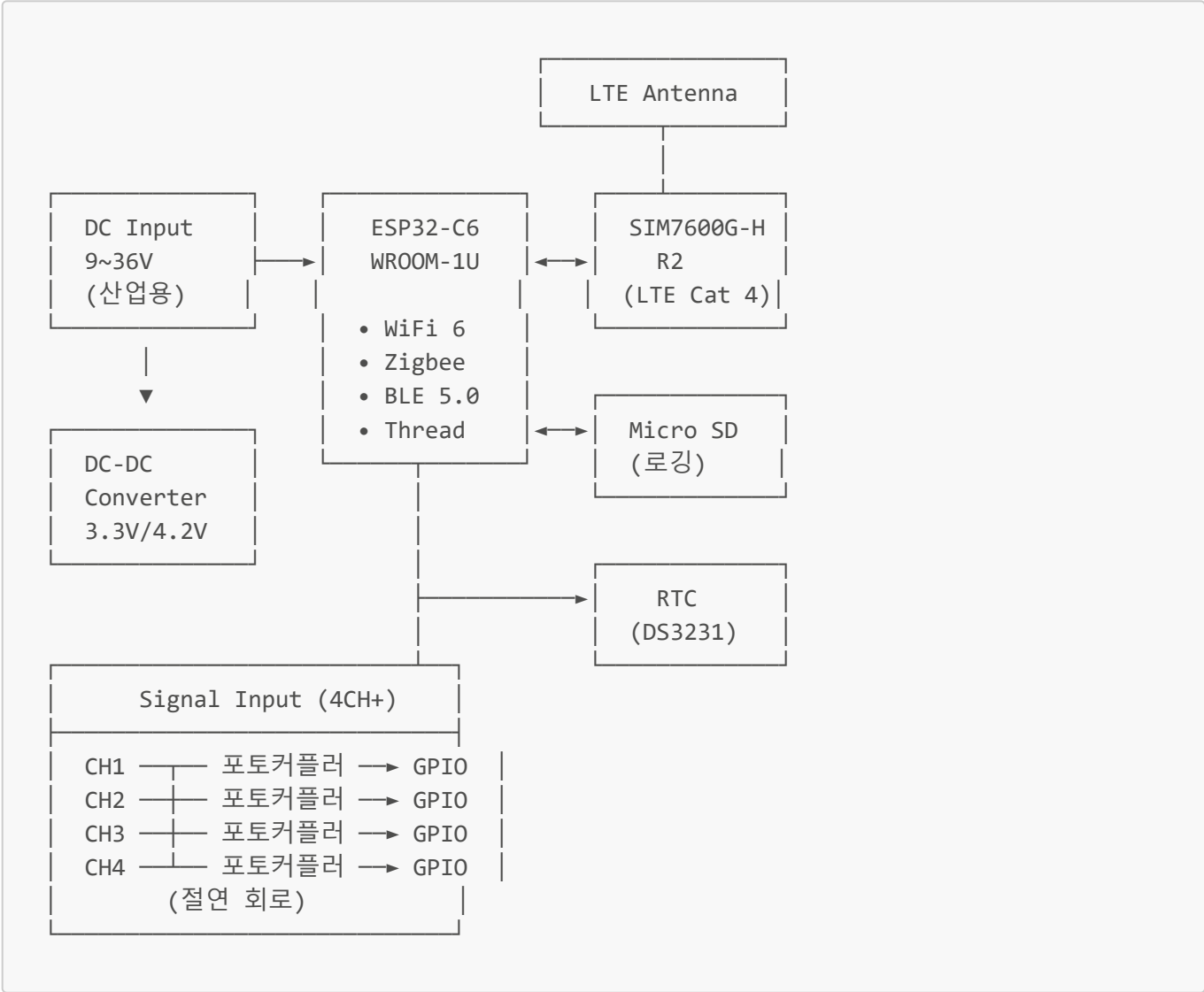
3.2 산출물 목록

구분	산출물	형식
HW	회로도 (Schematic)	PDF, 원본 파일
HW	PCB 배치도	PDF, 원본 파일
HW	Gerber 파일	Gerber RS-274X
HW	BOM 리스트	Excel
HW	샘플 PCB	실물 (수량 협의)
FW	펌웨어 소스 코드	C/C++ (ESP-IDF)

구분	산출물	형식
FW	빌드 가이드	Markdown
App	앱 소스 코드	React Native
App	빌드 가이드	Markdown
공통	개발 완료 보고서	PDF
공통	조작 매뉴얼 (HW/App)	PDF
인증	KC 인증 기술 가이드	PDF

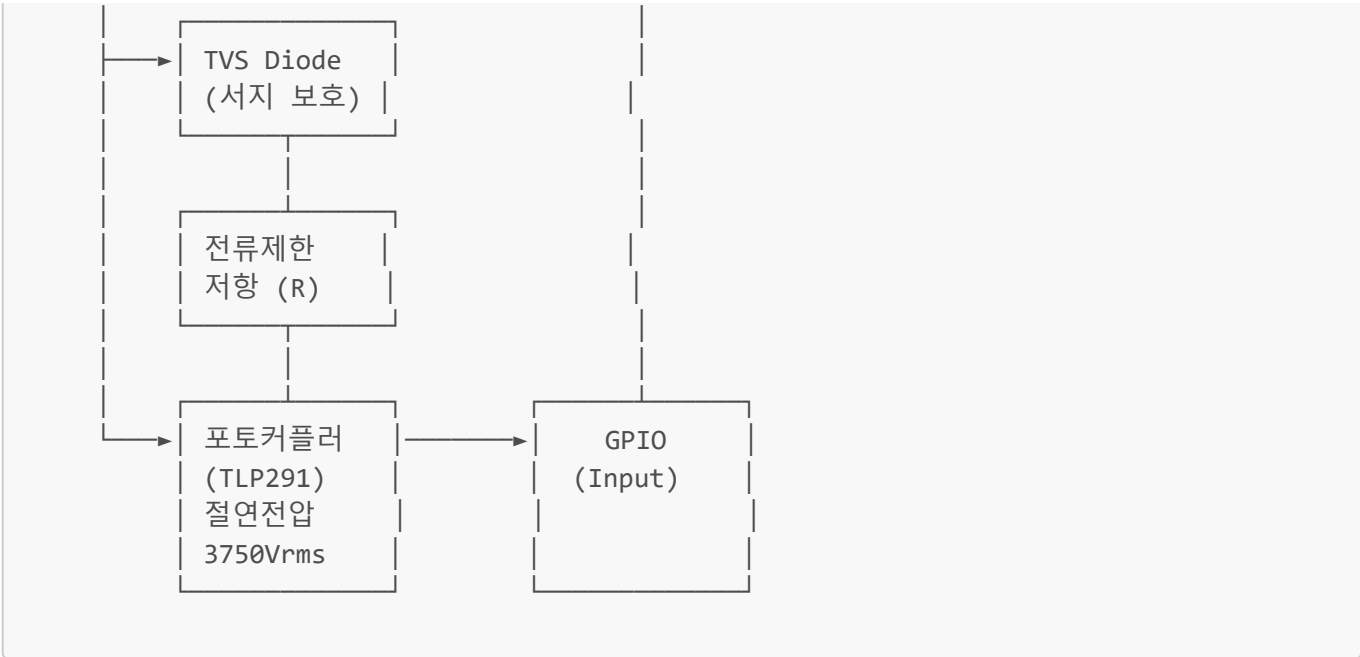
4. 기술 제안

4.1 하드웨어 블록 다이어그램 (제안)



4.2 절연 입력 회로 설계 (제안)

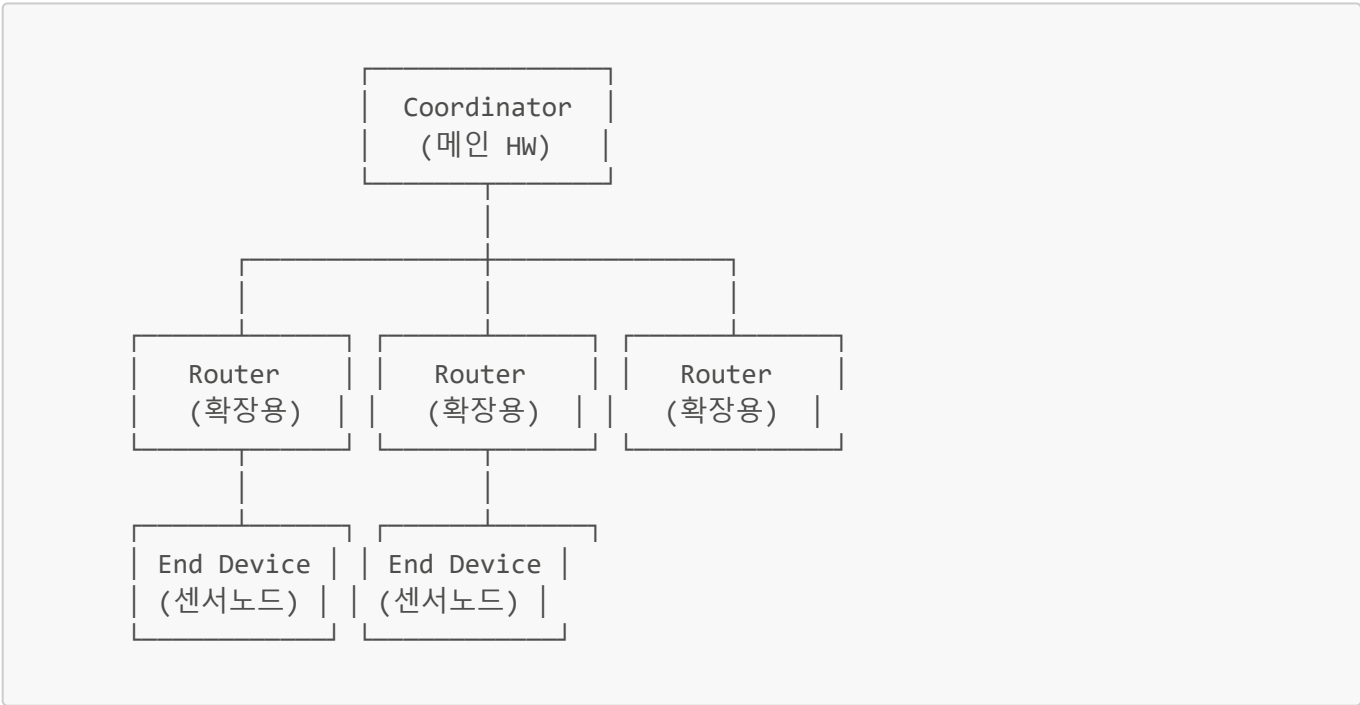




절연 회로 사양:

- 절연 내압: 3750Vrms (TLP291 기준)
- 입력 전압 범위: 3.3V ~ 24V DC (저항값 조정)
- 응답 시간: < 10μs
- 서지 보호: TVS 다이오드 적용

4.3 Zigbee Mesh 구성 (제안)



Zigbee 사양:

- 프로토콜: Zigbee 3.0 (IEEE 802.15.4)
- 주파수: 2.4GHz
- 최대 노드 수: 100+ (Coordinator 기준)
- 통신 거리: 실내 30m, 실외 100m (라인 라인)

4.4 KC 인증 대응 설계

항목	대응 방안
EMC (전자파 적합성)	4-layer PCB, GND 플레인 강화, Shield Can
ESD (정전기)	TVS 다이오드, 접지 패드
서지 보호	MOV, TVS, 퓨즈 적용
방열	열 via, 방열판, 온도 센서 모니터링
안전	절연 설계, 과전류/과전압 보호

인증 준비 지원:

- 사전 EMC 테스트 (스펙트럼 분석기)
- 인증 시험소 연계 (필요 시)
- 시험 실패 시 재설계 지원

5. 협업 방식

5.1 커뮤니케이션

채널	용도	빈도
정기 미팅	진행 상황 보고, 이슈 논의	주 1회
메신저 (Slack/카카오톡)	실시간 Q&A	상시
이메일	공식 문서, 산출물 전달	수시
Git Repository	소스 코드 관리	상시

5.2 품질 관리

- 코드 리뷰: Pull Request 기반 리뷰
- 테스트: 단위 테스트, 통합 테스트, 필드 테스트
- 문서화: 코드 주석, API 문서, 사용자 매뉴얼
- 버전 관리: Git (GitHub/GitLab)

6. 견적 (협업 필요)

구분	항목	비고
HW	회로 설계 + PCB 설계	양산 수준
HW	샘플 PCB 제작	수량 협의
FW	펌웨어 개발 (단독 + Zigbee 모드)	ESP-IDF
App	React Native 앱 개발	Android/iOS

구분	항목	비고
공통	문서화 및 기술 지원	
총 예상 비용	협의 후 확정	

※ 상세 견적은 기존 프로토타입 분석 후 제안 예정

7. 포트폴리오 / 레퍼런스

7.1 관련 프로젝트 경험

프로젝트명	내용	규모
산업용 IoT 게이트웨이	ESP32 + SIM7600 + Zigbee	양산 500대
스마트 계량기	LTE Cat 1 + 절연 입력	양산 1,000대
차량 추적 단말기	LTE + GPS, KC/FCC 인증	양산 300대
IoT 관리 앱	React Native, BLE 연동	10,000+ 다운로드

7.2 보유 인증 및 자격

- ☐ 관련 인증서 첨부 (해당 시)
- ☐ 회사 소개서 첨부

8. 질의 사항

프로젝트 착수 전 확인이 필요한 사항입니다:

1. 기존 프로토타입 관련

- 현재 발생 중인 구체적인 이슈 목록이 있습니까?
- 프로토타입 회로도 및 소스 코드 공유가 가능합니까?

2. 하드웨어 관련

- 신호 입력 채널 수: 정확히 몇 채널이 필요합니까? (4채널 이상이라 하셨는데)
- 입력 전압 범위: 감지할 신호의 전압 레벨은 어떻게 됩니까?
- 전원 입력 사양: 구체적인 입력 전압 범위 (예: 9~36V DC?)

3. 펌웨어 관련

- 서버 통신 프로토콜: MQTT와 HTTP 중 우선순위가 있습니까?
- Zigbee Mesh 예상 노드 수: 최대 몇 개 노드를 지원해야 합니까?

4. 앱 관련

- WebView로 표시할 웹 페이지의 URL 또는 사양이 있습니까?
- 로그인 방식: 자체 인증 서버 사용 또는 OAuth (구글/애플)?

5. 일정 관련

- 희망 완료일 또는 마일스톤이 있습니까?
- 샘플 PCB 수량은 몇 개를 원하십니까?

9. 연락처

항목	내용
업체명	[업체명]
담당자	[담당자명]
직책	[직책]
연락처	[전화번호]
이메일	[이메일]
주소	[주소]

본 지원서는 프로젝트 요구사항을 기반으로 작성되었으며, 상세 협의를 통해 조정될 수 있습니다.

작성일: 2025년 12월 18일