

Release Document

- WithBecon B2C -

- 1.7.1 -

2024. 3. 26

GADGET

Release Documents

History

년.월.일	관리 버전	내 용
2024.3.26	1.0	초기 버전
2024.4.24	1.1	가) Eyenix Firmware Fusing 방법 삭제, Eyenix 문서로 대체
		Transaction 변경 사항 정리
2024.5.9	1.2	Bluetooth LE Scan filtering UUID 내용 추가
2024.5.21	1.3	Wi-Fi Direct Transaction 내용 추가
2024.5.26	1.4	LED ON/OFF 관련 내용 변경
2024.6.1	1.5	부팅시 LED 동작 설명 내용 추가
		배터리 상태 표시 LED 동작 설명 내용 추가
2024.6.6	1.6	부팅후 BLE 대기 상태 LED 표시 내용 추가
2024.6.13	1.7	"res_hs" 내용 추가
2024.6.16	1.7.1	"res_hs" 빠진 내용 추가

목 차

- 가) Eyenix Firmware Fusing 방법
- 나) Block Diagram
- 다) Transaction 구조
- 라) Bluetooth LE Scan Filtering
- 마) Transaction 추가 / 변경 내용
- 바) Booting 시 LED 동작 설명
- 사) 배터리 상태 표시 LED 동작 설명

가) Eyenix Firmware Fusing 방법

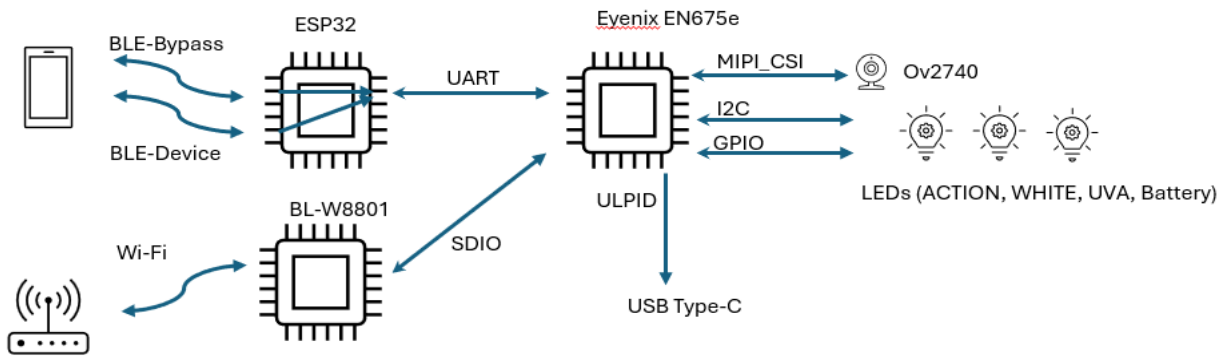
참조 1) EN675_ECMT_Manual.pdf

; Eyenix 공식 문서

참조 2) EYENIX_2022_EN675_GUIDE.pdf

; Eyenix 공식 문서

나) Block Diagram



다) Transaction 구조

Withbecon Device의 블루투스 통신 프로토콜을 정의한다. Mobile ↔ Device 간의 통신 프로토콜. (GATT)

Characteristic (Bypass)

ac920021-7237-468c-8193-10200d8f5fd5

Transaction 구조 : {"t":"...", "d":"..."}

t (type): String, d (data): String

Characteristic UUID : 'bypass' 와 'device' 로 구분

"bypass" : Apk 와 Eyenix EN675간 통신

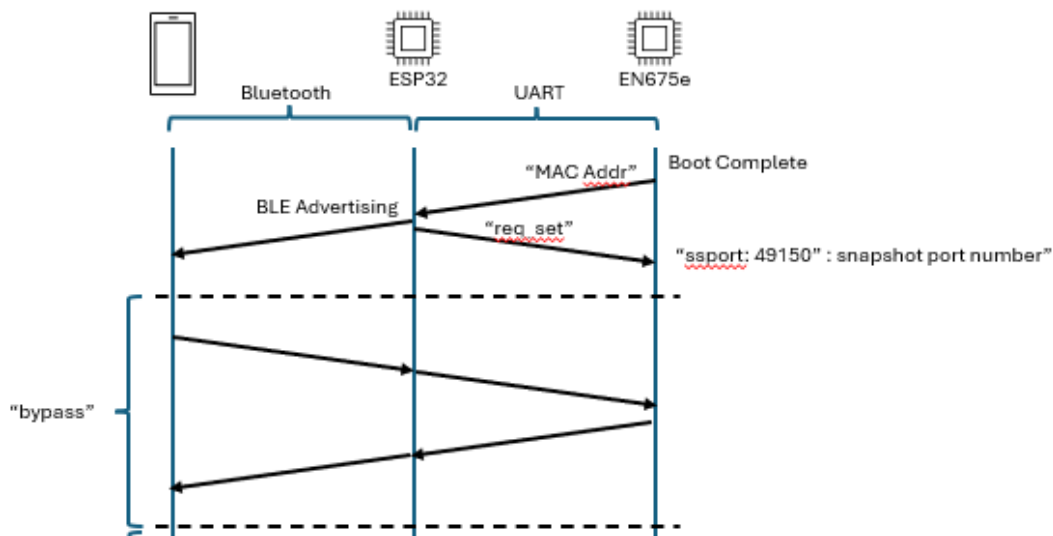
UUID : ac920021-7237-468c-8193-10200d8f5fd5

"device" : Apk 와 Bluetooth (ESP32) 간 통신

UUID : ac920021-7237-468c-8193-10200d8f5fd6

(이하 bypass 와 device 로 표기)

동작 구조



라) Bluetooth LE Scan Filtering

BLE Advertising 패킷의 최대 길이는 31 bytes 입니다.

Advertising 에 사용되고 있는 디바이스 이름은 "becon-xxxxxxxxxxx" 의 18 bytes 입니다.

Service UUID는 128bits 로 16 bytes 입니다.

따라서, 두 필드를 동시에 사용할 수는 없습니다. Advertising에 사용되는 UUID 를 16bits (0x5FD4)를 적용하였습니다.

```

val filters: MutableList<ScanFilter> = ArrayList()
val filteringUuid = ParcelUuid.fromString("00005FD4-0000-1000-8000-00805F9B34FB")
val scanFilter: ScanFilter = ScanFilter.Builder()
    .setServiceUuid(filteringUuid)
    .build()
filters.add(scanFilter)
val scanSettings = ScanSettings.Builder()
    .setScanMode(ScanSettings.SCAN_MODE_LOW_LATENCY)
    .build()
bluetoothLeScanner.startScan(filters, scanSettings, bleScanCallback)
  
```

Scanning filter 에서 16-bits를 128bits로 변환하여 사용 하도록 합니다.

Release Documents

마) Transaction 추가 / 변경 내용

Protocol Type	Type Value	Direction	Example	비고
RESPONSE_WIFI_STATUS (변경)	res_wcs	D → M	{ "t": "res_wcs", "d": { <기존 내용> "ip_addr": "192.168.1.4" } }	isConnected 가 false일 때, ssid, passwd = "none" (추가된 내용) ip_addr: <ip addr> or "none"
RESPONSE_WIFI_LIST (변경)	res_wl	D → M	"password": "withbecon"	Scan된 AP List들의 password 는 알 수 없음
RESPONSE_WIFI_CONNECT (변경)	res_wc	D → M	{ "t": "res_wc", "d": { <기존 내용> "ip_addr": "192.168.1.4" } }	isConnected 가 false일 때, ssid, passwd = "none" (추가된 내용) ip_addr: <ip addr> or "none"
REQUEST_ROOM_INFORMATION	req_ri	M → D		해당 사항 없음
RESPONSE_ROOM_INFORMATION	res_ri	D → M		해당 사항 없음
RESPONSE_DEVICE_STATUS (변경)	res_ds	D → M	{ "t": "res_ds", "d": <기존 내용> }	<변경내용> <connectStatus>
	DISCONNECTED: Wi-Fi가 연결되어 있지 않은 상태 INTERFACE_DISABLED: Wi-Fi 인터페이스가 비활성화되어 있는 상태 INACTIVE: Wi-Fi 인터페이스가 활성화되었지만 현재 연결을 시도하지 않고 있는 상태 SCANNING: Wi-Fi 스캐너가 현재 네트워크를 스캔하고 있는 상태 AUTHENTICATING: 인증 중인 상태 ASSOCIATING: 연결 중인 상태 4WAY_HANDSHAKE: 4-way handshake 과정을 진행 중인 상태 GROUP_HANDSHAKE: 그룹 키 핸드셰이크를 진행 중인 상태 COMPLETED: Wi-Fi 연결이 성공적으로 완료된 상태 CONNECTED: Wi-Fi 연결이 완료되어 있고, IP 주소를 얻었으며 통신이 활성화된 상태			
RESPONSE_DEVICE_CONNECT	res_dc	D → M		Transaction 은 BLE 연결상 태에서 동작하므로 해당 사항 없음
REQUEST_UPDATE_OS	req_uo	M → D		EN675 Storage 영역 부족 및 OTA 기능 미지원 으로 update 기능 안됨

Release Documents

Protocol Type	Type Value	Direction	Example	비고
RESPONSE_LED_CONTROL (추가)	res_l	D → M	{ "t": "res_l", "d": { "result": "ack" "code": 0 } }	result: "ack" or "nak" code: 0 = success -1 = fail
	"req_l", "req_ul", "req_lo", "req_wo", "req_wf", "req_uvo", "req_uvf" 에 대한 응답			
REQUEST_SENSORS	req_s	M → D		해당 사항 없음
RESPONSE_SENSORS	res_s	D → M		해당 사항 없음
REQUEST_CAPTURE_CAMERA	req_cc	D → M		Action Button을 짧게 눌렀을 때,
REQUEST_SEND_START	req_sst	M → D		해당 사항 없음
REQUEST_SEND_ING	req_ing	M → D		해당 사항 없음
REQUEST_SEND_DONE	req_dne	M → D		해당 사항 없음
REQUEST_RESTART	req_re	M → D	{ "t": "req_re", "d": "" }	video streaming 중지 wfd disconnect
REQUEST_REBOOT (변경)	req_rb	M → D		Board 회로 구성상 reboot 불가
	shutdown 기능으로 작용			
REQUEST_WIFIDIRECT	req_wd	M → D	{ "t": "req_wd", "d": { "action": "start" } }	action : start or stop
RESPONSE_WIFIDIRECT	res_wd	D → M	{ "t": "res_wd", "d": { "result": "ack" "name": "becon-xxxx" } }	result: "ack" or "nak" addr: D's BT Name
REQUEST_WIFIDIRECT_CALL	req_wdc	M → D	{ "t": "req_wdc", "d": { "name": "<BT Name>" } }	<BT Name> = Mobile의 BT Name
RESPONSE_WIFIDIRECT_CALL	res_wdc	M → D	{ "t": "res_wdc", "d": { "result": "ack" } }	result: "ack" or "nak"

Release Documents

Protocol Type	Type Value	Direction	Example	비고
REQUEST_WIFIDIRECT_FINISH	req_wdf	M → D	{ "t": "req_wdf", "d": "" }	Wi-Fi Direct Negotiation 절차 종료
RESPONSE_WIFIDIRECT_FINISH	res_wdf	M → D	{ "t": "res_wdc", "d": { "result": "ack" } }	result: "ack" or "nak"
REQUEST_SET (추가)	req_set	ESP32 → EN675	{ "t": "req_set", "d": { "ssport": 49150 } }	ssport: 1024 ~ 49151 default: 49150
EN675 초기 설정에 필요한 정보 전송				
REQUEST_LED_CONTROL (추가)	req_led	M → D	{ "t": "req_led", "d": { "action": "on", "white": "on" "uva": "on" } }	action; white: "on" or "1" ~ "255" "off" or "0" uva: "on" or "off"
RESPONSE_LED_CONTROL (추가)	res_led	M → D	{ "t": "res_led", "d": { "action": "on", "white": "on" "uva": "on" } }	LED 현재 상태 "on" or "off"
REQUEST_REPORT (추가)	req_rep	M → D	{ "t": "req_rep", "d": { "battery": <op> "wifisig": <op> } }	op: "start", "stop"
RESPONSE_REPORT (추가)	noti	D → M	{ "t": "noti", "d": { "battery": { "status": "Charging" "time": 45 "level": 1023 }, "wifi": { "rssi": -25 "speed": 26 "freq": 2442 } } }	time: min (부팅후) level : 0 ~ 1023 --> SOC (0 ~ 100)으로 추 후 변경 rssi : dBm speed: Mbps freq: MHz
battery: "Charging", "End Of Charge", "Suspend", "Fault"				

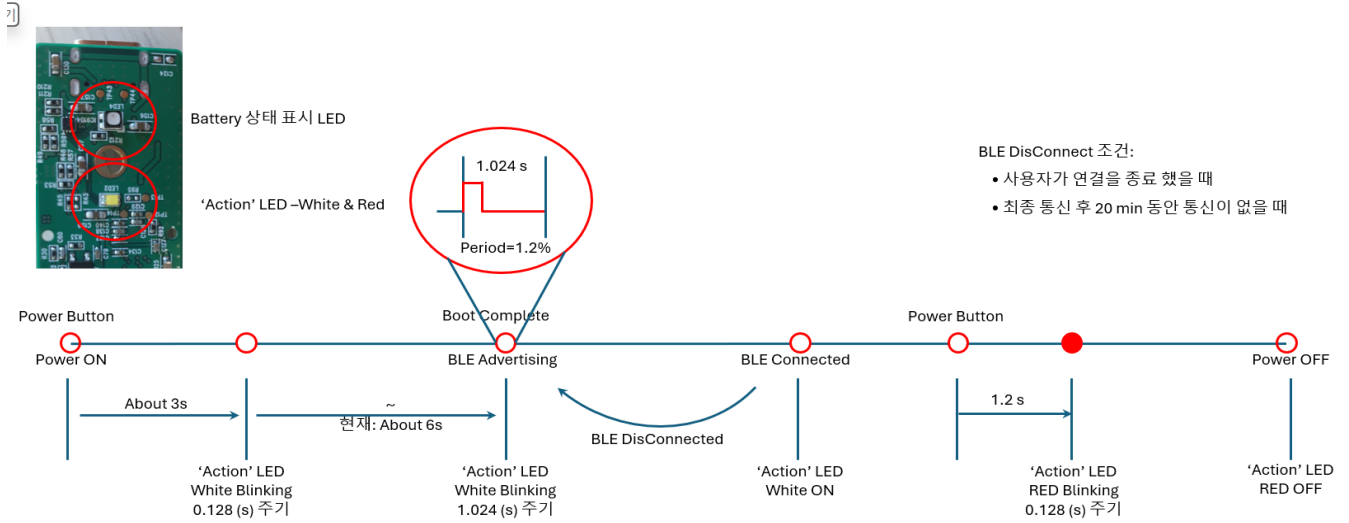
※ REQUEST_REPORT 는 Battery SOC 디버깅 및 SRT 스트리밍시 Wifi 신호 세기를 측정하기 위해 만든 transaction 임

Release Documents

Protocol Type	Type Value	Direction	Example	비고
REQUEST_HW_STATUS	req_hs	M → D	{ "t": "req_hs", "d": "" }	
RESPONSE_HW_STATUS	res_hs	D → M	{ "t": "res_hs", "d": { "charge": "Charging", "ssid": "iptime", "ip_addr": <ip address>, "ssport": <port number>, "srtport": <port number> "white": "on", "uva": "on", "streaming": "USB", "battery": 36, "os": "1.0.1" }	charge : <chrg sts> ssid: <ap_name> ip_addr: string ssport: [optional] srtport : port number uva : <led status> white : <led status> streaming : <path>
	<chrg> : "Charging", "End Of Charge", "Suspend", "Fault" <ssid> : <AP Name> or "none" ssport : snapshot port, srtport : streaming port <led status> : "off", "on", "failure" <path> : "UNKNOWN", "USB", "WIFI"			
REQUEST_STREAMING	req_str	M → D	{ "t": "req_str", "d": { "path": "usb" "control": "play" "srtport": 40052 } }	path : "usb", "wifi" control: "stop", "play"
RESPONSE_STREAMING	res_str	D → M	res_code: 0 : success, -1: fail, -2: invalid command/status -3: unknown command 3: UVC 동작 중 4: SRT 동작 중 5: UVC Stop 6: SRT Stop	
"device"				
READ		M → D	BLE READ	
		D → M	{ "t": "res_name", "d": { "name": "becon-744dbdc031e2" "version": "1.0.0" "status": "ready" } }	status: "ready", "fail"

바) 부팅시 LED 동작 설명

- 부팅시 LED 상태 표시



전원 버튼을 누르고, 소프트웨어에서 해당 GPIO를 ON 할 때까지 버튼이 눌러있어야 부팅 시퀀스가 진행됩니다.

```
U-Boot 2020.10 + EN675-Release + (May 31 2024 - 14:36:06 +0900)
LPDDR3(256MB) CLK: 796500000Hz (AXI CLK 398250000Hz), delay: 0x0
Model: eyenix,en675
DRAM: 256 MiB
MMC:
Loading Environment from SPIFlash... SF: Detected mx25l25635e with page size 256
Bytes, erase size 64 KiB, total 32 MiB
get CRC: 0x705fe70e, expected CRC:0xffffffff (ep->data:0000000083ee0a44, env_si
ze:0xffffc)
*** warning - bad CRC, using default environment
In: serial
Out: serial
Err: serial
Model: eyenix,en675
SF: Detected mx25l25635e with page size 256 Bytes, erase size 64 KiB, total 32 M
iB
NOR Flash lock (mask:0x3c)
Net: No ethernet found.
SF: Detected mx25l25635e with page size 256 Bytes, erase size 64 KiB, total 32 M
iB
en675 [ISP] Load start
--> DDR Base 80000000
--> DDR Area 100000
--> Flash offset 880000
--> user parameter for videocore is available
--> gzipped videocore uncompress start...--> uncompress success!
SF: Detected mx25l25635e with page size 256 Bytes, erase size 64 KiB, total 32 M
iB
en675 [kernel] Load start
--> DDR Base 80c00000
--> DDR Area 800000
--> Flash offset 80000
--> gzipped 'kernel' uncompress start... --> uncompress success!
EN675 BBL Loader
0.000000 OF: fdt: Ignoring memory range 0x80c00000 - 0x80e00000
0.000000 Linux version 4.20.0-Eyenix-EN675-Release (garamin@maru) (gcc ver
```

```
0.038616 NET: Registered protocol family 16
0.054441 en675-pad 45100800.pad: initialization complete
0.059308 GPIO line 16 (bt_reg_on) hogged as output/high
0.059573 GPIO line 19 (sys_pwr_on) hogged as output/high
0.112437 media: Linux media interface: v0.10
0.112790 videodev: Linux video capture interface: v2.00
```

[0.059573] 초에 GPIO가 ON 됩니다.
이때까지가 누적 1 sec 정도 됩니다.

```
1.922418 iio scheduler mq-usb:line registered (default)
1.922821 iio scheduler kyber registered
1.925692 i2c /dev entries driver
1.926237 en675-i2c0, irq disabled
1.927595 en675-i2c2, irq disabled
1.940701 usbcore: registered new interface driver uvcvideo
1.940956 USB video Class driver (1.1.1)
1.946091 ktd202x 2-0031: Driver loaded for a ktd2027.
2.018439 ktd202x 2-0030: Driver loaded for a ktd2027.
2.028147 44600000.serial: ttyFWDL at UART 0x44600000 (irq = 6, base baud
```

[1.946091] 초에 White LED가 점멸을 시작합니다.
이때까지가 누적 3 sec 정도 됩니다.

LED 드라이버 (KTD202x) 의 구동 순서는 다른 드라이버 보다 최대한 빠르게 앞당긴 상태 입니다.

사) 배터리 상태 표시 LED 동작 설명

- Battery 상태 표시 LED 동작 시나리오



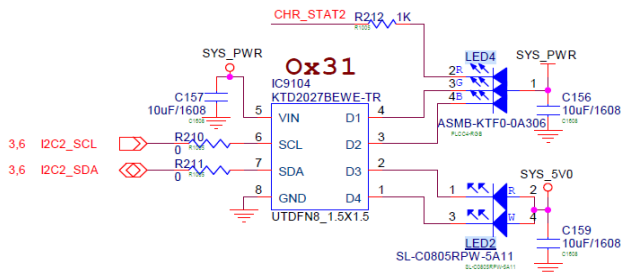
배터리 레벨을 한단계 더 두어서 녹 과 청을 blinking 하는 단계를 둘 수 있습니다.

ADC 값이 안정화 된 후 정리될 예정입니다.

RED LED의 경우 SW에서 제어 불가하므로, 밝기는 고정입니다.

원하는 색상이 아닐 경우, 녹, 청의 LED 밝기를 조정해서 맞추는 작업이 필요할 수 있습니다.

RED (적) LED 는 HW에서 직접 제어 하므로, SW에서 제어할 수 없습니다.



AP에서는 CHR_STAT2 는 배터리가 완충인지 아닌지의 판별(Read only) 용도로만 사용

RED LED는 충전중일 때만 ON 됩니다. (MP2607에서 제어)

: 충전 중인 상태는 전원케이블이 연결된 상태에서, 배터리가 완충되지 않은 상태를 의미 합니다.
즉, 전원케이블이 연결되어 있어도, 완충 상태이면, RED LED는 OFF 됩니다.