

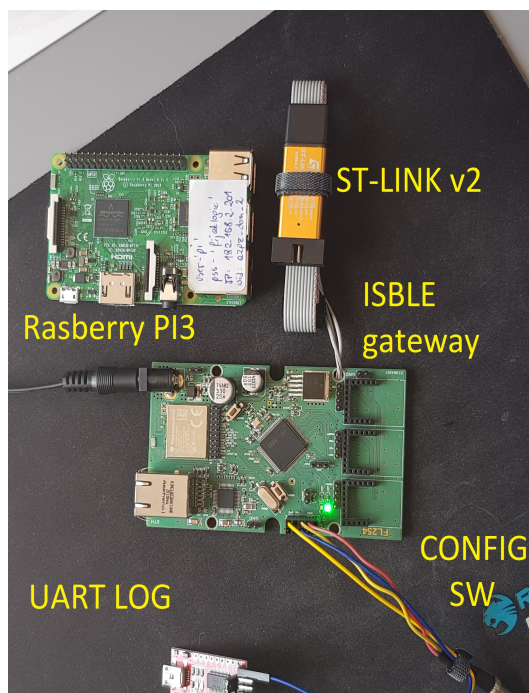
# **ISBLE Gateway User Manual**

Historia wersji:

Data	Autor	Wersja
12.06.2019		'v1.0

W skład zestawu ISBLE gateway wchodzi:

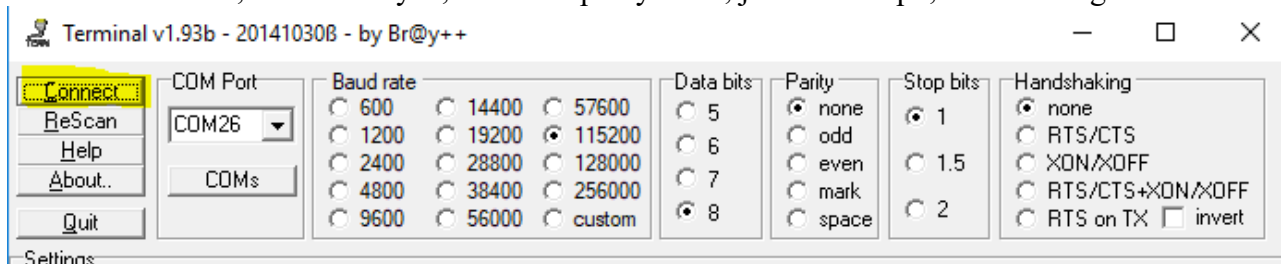
1. Układ ISBLE gateway + zasilacz,
2. Programator ST-LINK v2,
3. Konwerter UART<->USB,
4. Raspberry PI 3 + zasilacz,
5. Router D-Link + zasilacz
6. Pendrive z oprogramowaniem



W chwili pisania dokumentu urządzenie ISBLE gateway jest funkcjonalnie pełne. Komendy z rozdziałów „Konfiguracja sieci ISBLE – dodawanie urządzeń” oraz „Działanie sieci ISBLE – tryb normalny” są wspierane zgodnie z dokumentacją „Konfiguracja i działanie sieci ISBLE”. Odpowiedzi na zadane komendy są emulowane – podłączenie urządzenia gateway ISBLE do sieci węzłów nastąpi na późniejszym etapie. Projekt i niniejszy dokument mają charakter pomocniczy do rozpoczęcia pracy z warstwą GUI/interfejsem użytkownika.

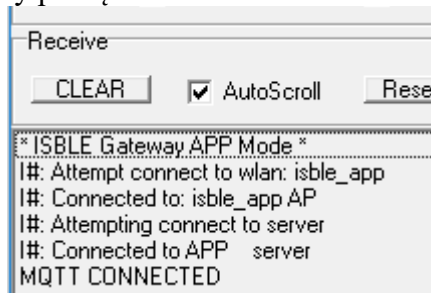
### Uruchomienie zestawu startowego – tryb normalny:

- Podłączyć router do zasilania. Zostanie stworzona sieć wlan o nazwie ssid: „isble\_app”, oraz hasło dostępowym „isble\_admin”, adres bramy: 192.168.2.1
- Podłączyć urządzenie Raspberry PI3 do zasilania, pełni ono rolę brokera mqtt, automatycznie podłączy się do istniejącej sieci, w której będzie widziany pod adresem IP: 192.168.2.201,
- Podłączyć konwerter UART<->USB do komputera PC, włączyć terminal (np. vTerminal 1.9b: [LINK DO POBRANIA](#)), wybrać odpowiedni port z listy dostępnych, wybrać prędość na 115200, 8bitów danych, brak bitu parzystości, jeden bit stopu, handshaking: none



Następnie wybrać przycisk „Connect”. Korzystanie z terminala nie jest wymagane, jednak są tam wyświetlane logi z działania urządzenia ISBLE gateway.

- Podłączyć urządzenie ISBLE gateway do zasilania, w trybie pracy normalnym, połączy się ono do istniejącej sieci wlan „isble\_app”, następnie podłączy się do serwera/brokera mqtt, Log z poprawnej procedury podłączenia:



Na tym etapie układ działa w trybie normalnym, obsługując komendy opisane w rozdziale „Tematy Węzłów – Komendy dla węzłów”, specyfikacji „Konfiguracja i działanie sieci ISBLE”.

Pomocniczo zostały stworzone dwa programy (PC) które emulują wysłanie komend odczytu temperatury oraz oświetlenia, oraz odczytują odpowiedzi. Oba programy łączą się z brokerem mqtt znajdującym się na raspberry pi. (**Ważne:** komputer na którym uruchamiane są opisane programy musi być podłączony do sieci isble\_app – spowodowane jest to faktem iż broker znajduje się lokalnie w tej sieci – nie jest wyniesiony na serwer zewnętrzny).

„isble\_datCollector” subskrybuje tematy:

- "546C0E83455C/LIGHT/RES"
- "546C0E83455C/TEMPERATURE/RES"
- "546C0E83455C/IO/RES"

Po otrzymaniu wiadomości na jeden z powyższych tematów (kierunek ISBLE gateway → broker

mqtt), parsuje odpowiedź i wyświetla ją w konsoli.

```
* ISBLE Gateway APP Mode *
I#: Attempt connect to wlan: isble_app
I#: Connected to: isble_app AP
I#: Attempting connect to server
I#: Connected to APP server
MQTT CONNECTED
Recv SUB_LIGHT_GET
546C0E83455C/LIGHT/RES, dest MAC: 54 6c e 83 45 1
Recv SUB_LIGHT_GET
546C0E83455C/LIGHT/RES, dest MAC: 54 6c e 83 45 1
```

„isbleApp\_msgTrig” służy do wysyłania wiadomości do modułu ISBLE gateway.

Do jego uruchomienia służą dwa skrypty (\*.bat) *TriggerLightRead.bat* oraz

*TriggerTemperatureRead.bat* które przekazują odpowiednie parametry wejściowe (oddzielone spacją) np.:

```
isbleApp_msgTrig 546C0E83455C/TEMPERATURE/GET 54 6C 0E 83 45 5C
```

Parametry są zgodne z dokumentacją tematu węzłów – odczyt temperatury. (Należy zwrócić uwagę że parametry wiadomości w skrypcie mają formatowanie ASCII, dla ułatwienia pracy.

Rzeczywiście są one transmitowane tak jak w dokumentacji, za pomocą wartości binarnych, wspomniana konwersja odbywa się w isbleApp\_msgTrig)

Po wykonaniu skryptu: *TriggerTemperatureRead.bat*, zostanie wyświetlony komunikat w Logu (terminal RS), oraz zostanie wyświetlona przeparsowana wiadomość w programie *isble\_datCollector*.

```
C:\Projects\KF\ISBLE\VS_emulator\emulator_exe\isble_datCollector.exe
Rx Mac: 54 6c e 83 45 5c Status: 0, Temperature value: 25
Rx Mac: 54 6c e 83 45 1 Status: 0, Light value: 33
```

**Oba programy mają charakter diagnostyczny, mają głównie ułatwić rozpoczęcie pracy projektem ISBLE gateway.**

### *Uruchomienie układu w trybie konfiguracji:*

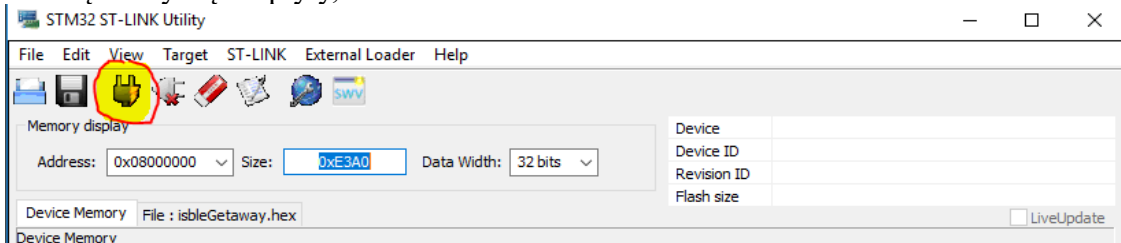
Układ przechodzi do trybu konfiguracji jeśli przy resetowaniu/włączeniu zasilania przytrzymany jest przycisk „config” (wyprowadzony w zestawie na przewodzie). Układ informuje stosownym komunikatem w terminalu, oraz sygnalizuje fakt wejścia w tryb konfiguracyjny zmieniając stan diody z okresem 100ms 25 razy.

W trybie tym układ łączy się do routera znajdującego się na telefonie komórkowym (system Android) o nazwie ssid: „isble\_config”, hasło: „isble\_admin”, następnie próbuje połączyć się z serwerem znajdującym się pod adresem IP: 192.168.43.1 (jest to standardowy adres gateway dla routera uruchomionego w systemie Android), numer portu 5000. W celu przetestowania pracy należy ściągnąć terminal TCP (google play), który umożliwia przesyłanie danych w trybie binarnym, lub napisać własny program pełniący funkcję serwera TCP i implementujący rozkazy **„Konfiguracja sieci ISBLE - dodawanie urządzeń,,**

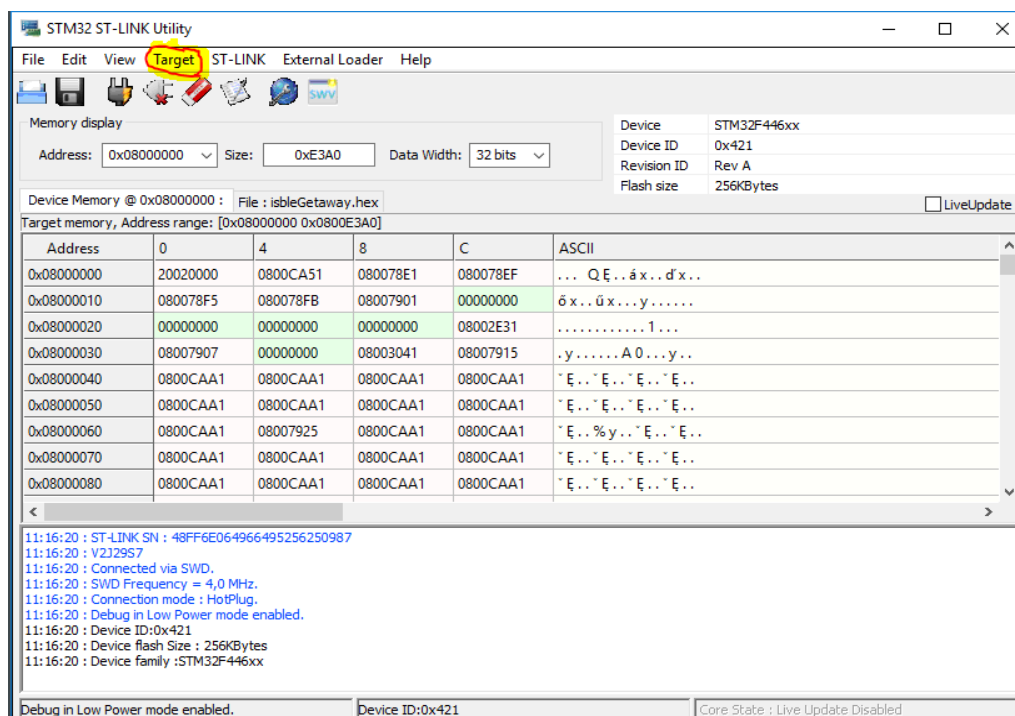
### Aktualizacja/zmiana oprogramowania układu ISBLE gateway.

Oprogramowanie ISBLE gateway jest w dalszym ciągu rozwijane, dlatego do zestawu został dołączony programator ST-LINK V2 który umożliwia wgranie nowego pliku wsadowego \*.hex. Po podłączeniu układu do PC, oraz zasileniu ISBLE gateway, uruchamiamy oprogramowanie „STM32 ST-LINK Utility” [LINK DO POBRANIA](#)

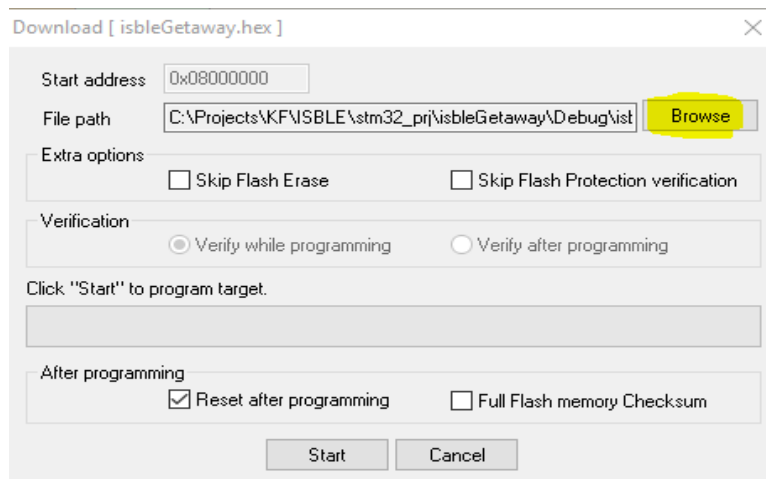
- Podłączamy się do płyty,



- Po poprawnym podłączeniu widzimy mapę pamięci,



- Wybieramy opcję: Target->Program, pojawi się okno w którym wybieramy plik wsadowy (\*.hex)



- Rozpoczynamy programowanie wybierając opcję „Start”, okno po procesie programowania zostanie zamknięte, a komunikat o zakończeniu programowania znajdować się będzie w konsoli:

```

11:10:20 : SWD Frequency = 4,0 Mhz.
11:16:20 : Connection mode : HotPlug.
11:16:20 : Debug in Low Power mode enabled.
11:16:20 : Device ID:0x421
11:16:20 : Device flash Size : 256KBytes
11:16:20 : Device family :STM32F446xx
11:16:48 : [isbleGetaway.hex] opened successfully.
          Address Ranges [0x08000000 0x080001C4] [0x080001D0 0x0800E3A4]
11:16:48 : [isbleGetaway.hex] checksum : 0x005A3A4F
11:19:00 : Memory programmed in 1s and 812ms.
Debug in Low Power mode enabled. Device ID:0x421

```