**LoRa-E5 Mini Programozási Útmutató**

Ez a dokumentáció bemutatja, hogyan kell felprogramozni egy LoRa-E5 Mini modult. Az alábbiakban egy útmutató található a szükséges eszközökről, a firmware feltöltéséről, valamint a modul konfigurálásáról, programozásáról.

**Szükséges eszközök és szoftverek**

A LoRa-E5 Mini modul sikeres programozásához az alábbiakra lesz szükség:

Hardverek:  
- LoRa-E5 Mini modul  
- ST Link v2  
- Jumper kábelek és egyéb csatlakozók, szenzorok

Szoftverek:

- STM32CubeProgrammer (a firmware feltöltéséhez)

- STM32Cube IDE (a kódoláshoz)  
- Arduino IDE (opcionális a programozáshoz, ezzel is lehetséges)  
- Git (a forráskód klónozásához)  
- Seeed Studio által biztosított firmware a LoRa-WAN példákhoz

Illesztőprogram: Telepíteni kell a megfelelő USB-illesztőprogramot, amely lehetővé teszi a debugger kapcsolódását a számítógéphez.

**1. A LoRa-E5 Mini modul csatlakoztatása**

Először kössük össze a LoRa-E5 Mini modult és az ST-Link debuggerünket az alábbi módon:

ST-Link LoRa

VTREF => 3.3V

SWDIO => DIO

SWCLK => CLK

GND => GND

Abban az esetben ha a Reset lábat is bekötjük, akkor feltöltéskor nem kell nyomva tartani a Reset gombot a LoRa modulon. Ezután csatlakoztassuk az ST-Link debuggert a számítógéphez egy USB-kábellel, majd a LoRa-E5 Mini modult. Ellenőrizzük, hogy a debugger és a modul megfelelően csatlakoznak.

A debugger ellenőrzése:  
- Windows: Nyissuk meg az Eszközkezelőt, és keressük meg az USB-soros portot. Az ST-Linknek itt szerepelnie kell.

**2. Firmware feltöltése a STM32CubeProgrammer-rel**

1. Töltsük le és telepítsük az STM32CubeProgrammer és a STM32Cube IDE szoftvereket az STMicroelectronics weboldaláról.

2. Klónozzuk a Seeed Studio GitHub oldaláról a firmware forráskódját:  
 git clone <https://github.com/Seeed-Studio/LoRaWan-E5-Node.git>

3. Válasszuk ki a klónozott repó .project fájlját és nyissuk meg az STM32CubeIDE-vel. (LoraWan-E5-Node => Projects => Applications => LoRaWAN => LoRaWan\_End\_Node => STM32CubeIDE)

4. Nyissuk meg a project tulajdonságokat

A screenshot of a computer

Description automatically generated

5. A C/C++ Build => Settings => MCU Post build outputs alatt pipáljuk be a “Convert to Intel Hex file (-O hex)” opciót és válasszuk az Apply and Close gombot.

A screenshot of a computer

Description automatically generated

6. Fordítsunk egy Debug buildet, a projektnek gond nélkül kell fordulnia.

A screenshot of a computer

Description automatically generated

7. Mostmár csak a Device EUI-t, Application EUI-t, Application KEY-t és a LoRaWAN Region-t kell módosítanunk, annak érdekében, hogy a hálózathoz tudjunk csatlakozni.

Kicsit részletesebben:

A Device EUI egy egyedi azonosító, amelyet a LoRa eszközhöz rendelnek, és az eszköz egyedi azonosítására szolgál a LoRaWAN hálózatban.

Az Application EUI az alkalmazásszintű hálózat azonosítója, amely meghatározza, hogy az eszköz melyik alkalmazáshoz tartozik.

Az Application key egy titkos kulcs, amelyet az eszköz és a hálózati szerver oszt meg a kapcsolat hitelesítésére és a kommunikáció titkosítására.

A LoRawan Region a rádiós kommunikáció frekvenciatartományát határozza meg az adott földrajzi területhez igazítva. Például EU868 az Európában, US915 az Egyesült Államokban.

A LoRaWAN/App/se-identity.h alatt keressük meg a LORAWAN\_JOIN\_EUI amkrót és másoljuk be a mi Application EUI-nkat. Szintén ebben a header fájlban lesz a Device EUI és az Application KEY is. A Device EUI a LORAWAN\_DEVICE\_EUI makróba kell bemásolni, az Application EUI-t a LORAWAN\_NWK\_KEY makróba. A LoRaWAN Region-t a lora\_app.h headerben találjuk meg az ACTIVE\_REGION definenál, állítsuk az értéket LORAMAC\_REGION\_EU868-ra.

8. Buildeljük újra a projektet.

9. Nyissuk meg az STM32CubeProgrammer-t, csatlakoztassuk az eszközeinket amennyiben ezt még nem tettük meg, tartsuk nyomva a RESET gombot a LoRa modulon (ha nem kötöttük be a reset lábat is) és nyomjunk rá a Connect gombra.

A screenshot of a computer

Description automatically generated

10. Nézzük meg, hogy a Kiolvasásvédelem AA, ha BB-t mutat, válasszuk az AA-t, és kattintsunk az Apply gombra.A screenshot of a computer

Description automatically generated

11. Most lépjünk az Erasing & Programming oldalra, válasszuk ki a hex fájl elérési útvonalát (pl: C:\Users\Users\user\Downloads\LoRaWan-E5-Node\Projects\Applications\LoRaWAN\LoRaWAN\_End\_Node\STM32CubeIDE\Debug\LoRaWAN\_End\_Node.hex ), válasszuk ki a programozási lehetőségeket a képen látható módon, majd kattintsunk a Start Programming gombra!

A screenshot of a computer

Description automatically generated

12. Egy „Download verified successfully”üzenetet fogunk látni amennyiben a programozás megtörtént.

13. Nyissuk meg például az Arduino Soros Monitort, az alábbi üzenetet kell látnunk ha sikeres volt a csatlakozás.

A screenshot of a computer

Description automatically generated

14. Ezután el lehet kezdeni az egyedi kód fejlesztését a LoRa modulra, ha például egy DHT22-es szenzor hőmérséklet és páratartalom értékeit akarjuk továbbítani a LoRa hálózaton, akkor nyissuk meg a projekt .ioc fájlját, majd engedélyzzük az I2C kommunikációt (a szenzor miatt), majd módosítsuk a forráskódot ennek megfelelően és töltsük fel a programot a LoRa modulunkra.

**3. Arduino IDE használata (opcionális)**

Az Arduino IDE segítségével is fejleszthető kód a LoRa-E5 Mini modulhoz, de fontos megjegyezni, hogy az ST-Link használata ekkor is szükséges. Ha ST-Link nélkül akarunk kódot feltölteni a LoRa modulra, akkor nekünk kell megírni a booatloadert. Az alábbi lépések segítenek az STM32 board támogatásának hozzáadásában és a kód feltöltésében.

1. Töltsük le és telepítsük az Arduino IDE-t a hivatalos weboldalról.

2. Adjuk hozzá az STM32 támogatását az Arduino IDE-ben. Ehhez nyissuk meg a File => Preferences menüt, és adjuk hozzá a következő URL-t az “Additional boards manager URLS” mezőbe:  
https://github.com/stm32duino/BoardManagerFiles/raw/main/package\_stmicroelectronics\_index.json

3. Írjunk vagy töltsünk fel egyedi kódot az Arduino IDE segítségével.

**4. LoRa konfiguráció**

A LoRa modul konfigurációjához használjunk egy soros terminál programot (például az Arduino beépített soros monitorát). A parancsokat itt találhatunk: https://wiki.seeedstudio.com/LoRa\_E5\_mini/#basic-at-commands

**5. Hibaelhárítás**

Ha problémák adódnak a modulok használata során, az alábbi tippek segíthetnek a hibaelhárításban:

A modulok nem válaszolnak: Ellenőrizzük a csatlakozásokat és a megfelelő portokat. Próbáljunk például a soros monitoron egy AT parancsot küldeni a LoRának, hogy meggyőződjünk a helyes csatlakozásról és működésről.

A firmware feltöltése sikertelen: Győződjünk meg arról, hogy az STM32CubeProgrammer felismeri az eszközöket, és a megfelelő firmware verziót használjuk.

LoRa csatlakozási hibák: Ellenőrizzük a LoRaWAN paramétereket (DevEUI, AppEUI, AppKey).

No ST-Link target found hibaüzenet:

* Az STM32CubeProgrammerben álltsuk a Reset módot Hardware resetre, illetve Software resetre.
* A Mode-tNormal-ra állítani és nem Under Reset-re
* Alacsonyabb frekvencián csatlakozni
* SWD port helyett JTAG-gel csatlakozni
* Firmware upgrade
* Ellenőrizzük, hogy a LoRa modul megkapja-e a megfelelő tápfeszültséget (3.3V)
* Boot és RST-t gombokat egyszerre nyomvatartani a Connect gomb megnyomása előtt
* Az SCK lábat is bekötve feltölteni
* Győződjünk meg arról, hogy nem klónozott ST-Linkkel próbálkozunk

Eredeti ST-Link:

A usb cable connected to a cord

Description automatically generated

Klónozott:

A yellow and black usb stick

Description automatically generated