Project Edison

Telepítési és beüzemelési dokumentáció

ET-Mechatronics

2015-2016

Tartalom

[1. Szükséges programok és fájlok 3](#_Toc454396972)

[2. Eszköz telepítése és beállítása (Yocto linux) 4](#_Toc454396973)

[2.1. Yocto linux feltelepítése az eszközre 4](#_Toc454396974)

[2.1.1. Windows 4](#_Toc454396975)

[2.1.2. Linux és Mac OS 4](#_Toc454396976)

[2.2. Kapcsolódás az Edisonhoz 5](#_Toc454396977)

[2.2.1. Windows 5](#_Toc454396978)

[2.2.2. Linux 6](#_Toc454396979)

[2.2.3. Mac OS 6](#_Toc454396980)

[2.3. Eszköz beállítása 7](#_Toc454396981)

[2.3.1. Eszköznév, jelszó és hálózat 7](#_Toc454396982)

[2.3.2. Csomagok 9](#_Toc454396983)

[3. Fejlesztőkörnyezet beállítása (Eclipse) 10](#_Toc454396984)

[3.1. Kapcsolódás 10](#_Toc454396985)

[3.2. Példa program 13](#_Toc454396986)

[4. ’Project Edison’ 15](#_Toc454396987)

[4.1. Beszerzés 15](#_Toc454396988)

[4.2. Fordítás 15](#_Toc454396989)

[4.3. Telepítés 15](#_Toc454396990)

# 1. Szükséges programok és fájlok

Az eszköz programozásához, kezeléséhez az eszközhöz csatlakozott számítógépre fel kell telepíteni a szükséges vezérlőprogramokat (driverek).

**Windows rendszer esetén:**

* VCP (Virtual COM Port) driver - *CDM v2.12.06 WHQL Certified.exe* (<http://www.ftdichip.com/Drivers/CDM/CDM%20v2.12.06%20WHQL%20Certified.exe>)
* Intel® IoT (Internet of Things) drivers - *w\_iot\_2015.0.028.exe* (<http://downloadmirror.intel.com/25384/eng/w_iot_2015.0.028.exe>)
* Serial/SSH (Secure Shell) kommunikációs program - PuTTY.exe (<http://the.earth.li/~sgtatham/putty/latest/x86/putty.exe>)

**Mac OS X esetén:**

* Intel® IoT drivers - *m\_iot\_dev\_kit\_2015.0.021.tar.gz*(<http://downloadmirror.intel.com/25384/eng/m_iot_dev_kit_2015.0.021.tar.gz>)

**Linux esetén:**

* Intel® IoT drivers - *l\_iot\_dev\_kit\_2015.0.012.tar* (<http://downloadmirror.intel.com/25384/eng/l_iot_dev_kit_2015.0.012.tar.gz>)

Az eszközre feltelepítendő Yocto linux fájlt is le kell tölteni, a dokumentum íráskor ez a 2.1-es verziójú képfájl:  *edison-iotdk-image-280915* (<http://downloadmirror.intel.com/25384/eng/edison-iotdk-image-280915.zip>)

Opcionálisan egy fejlesztőkörnyezet is ajánlott: *Eclipse IDE for C/C++ Developers Mars.1 Release* (4.5.1) (<http://www.eclipse.org/downloads/packages/eclipse-ide-cc-developers/mars1>)

Az esetleges verziófrissítések miatt melléklem az egyes komponensek letöltő oldalát:

<http://www.ftdichip.com/Drivers/VCP.htm>

<https://software.intel.com/en-us/iot/hardware/edison/downloads>

<http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>

# 2. Eszköz telepítése és beállítása (Yocto linux)

A friss eszközre fel kell telepíteni egy operációs rendszert, hogy használható legyen a későbbiekben, illetve ez a rendszer alkalmas legyen a programfejlesztésre is, vagyis támogassa a szükséges programcsomagokat.

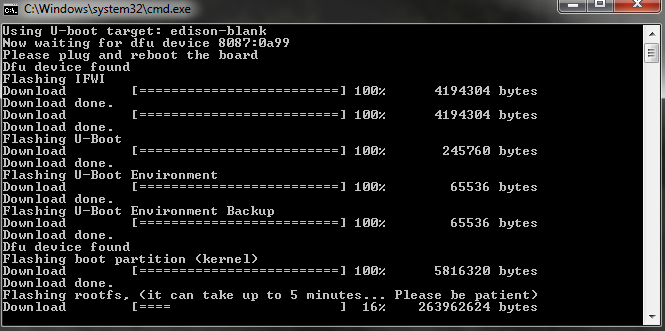
## 2.1. Yocto linux feltelepítése az eszközre

Első lépésben feltelepítjük a szükséges drivereket. Az IoT driverek (drivers, Phone Flash Tool) telepítése közben csatlakoztatni kell az Edisont a számítógéphez.

Ezt követően kicsomagoljuk a Yocto linux fájlt (*edison-iotdk-image-280915.zip*) egy külön mappába.

Mivel az eszköz már csatlakoztatva van a számítógéphez, ezért belépünk a kicsomagolt Yocto linux fájlokat tartalmazó mappába és elindítjuk a telepítést:

### 2.1.1. Windows

**1.** futtatjuk a *flashall.bat* állományt és követjük az utasításokat  


1. ábra: flashall.bat futás közben

**2.** az állomány lefutása után várunk 5-10 percet, ez alatt ez Edison befejezi a telepítést

**3.** áramtalanítjuk az eszközt, majd újra áram alá helyezzük, ezzel kényszerítjük az újraindulásra

### 2.1.2. Linux és Mac OS

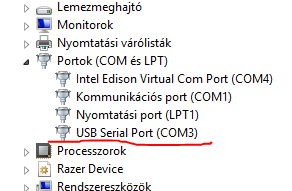
**1.** futtatjuk a *flashall.sh* álományt és követjük az utasításokat (ugyanaz a program kimenete, mint Windows esetén)  
 **2.** az állomány lefutása után várunk 5-10 percet, ez alatt ez Edison befejezi a telepítést

**3.** áramtalanítjuk az eszközt, majd újra áram alá helyezzük, ezzel kényszerítjük az újraindulásra

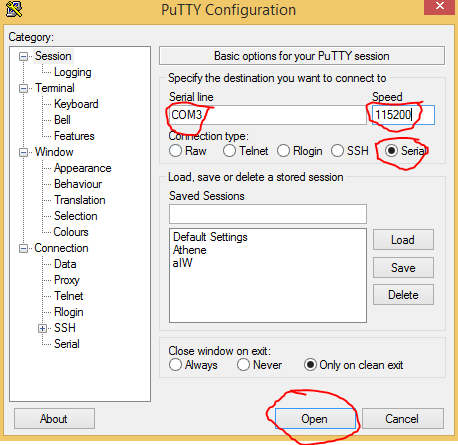
## 2.2. Kapcsolódás az Edisonhoz

A számítógép előkészítése után már kapcsolódhatunk az eszközhöz.

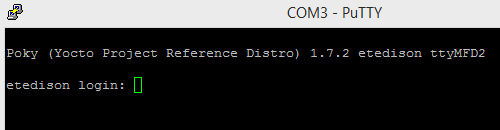
### 2.2.1. Windows

1. megnyitjuk az *Eszközkezelőt* és megkeressük az Edison kommunikációs portját a *Portok* almenüben  
   

2. ábra: Edison kommunikációs portja

1. megnyitjuk a *PuTTY.exe*-t, átállítjuk Serial kommunikációra, beállítjuk a portot és a sebességet átállítjuk *115200* bps-re, majd rányomunk az *Open* gombra  
   

3. ábra: Beállított PuTTY

1. a megnyitott ablakban nyomunk pár Entert, hogy felébresszük a kommunikációt és ha mindent jól csináltunk az Edison bejelentkező képernyője fogad  
   

4. ábra: Kapcsolódva PuTTY-n keresztül

### 2.2.2. Linux

Feltételezvén, hogy nincs másik USB kommunikációs eszköz csatlakoztatva a számítógéphez:

1. megnyitjuk a *Terminal*-t
2. beírjuk és futtatjuk a következő parancsot:   
   screen /dev/ttyUSB0 115200  
   (ha nem *root* felhasználóval léptünk be szükséges ilyen jogot szereznünk, tehát vagy a *sudo* vagy a *su* parancsot előtte kiadjuk)
3. nyomunk pár Entert, hogy felébresszük a kommunikációt és ha mindent jól csináltunk az Edison bejelentkező képernyője fogad

### 2.2.3. Mac OS

Feltételezvén, hogy nincs másik USB kommunikációs eszköz csatlakoztatva a számítógéphez:

1. megnyitjuk a *Terminal*-t
2. beírjuk és futtatjuk a következő parancsot:  
   ls /dev/cu.serialusb\*
3. leolvassuk a megjelenő állománynevet
4. beírjuk és futtatjuk a következő parancsot:   
   screen /dev/cu.usbserial-xxxxxx 115200  
   (az 'xxxxxx' helyére az előbb leolvasott érték kell)
5. nyomunk pár Entert, hogy felébresszük a kommunikációt és ha mindent jól csináltunk az Edison bejelentkező képernyője fogad

## 2.3. Eszköz beállítása

A következő lépések már operációs rendszer függetlenek.

A beírandó parancsokat az alábbi formátumban közlöm:

# parancs

Az esetleges program kimenetekre ezt a formátumot használom:

>> kimenet

### 2.3.1. Eszköznév, jelszó és hálózat

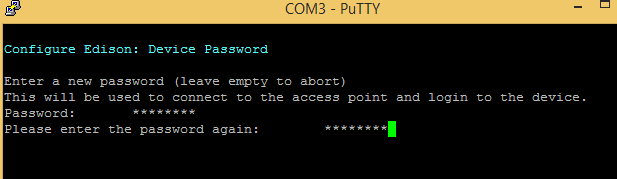
Bejelentkezünk az eszközbe:

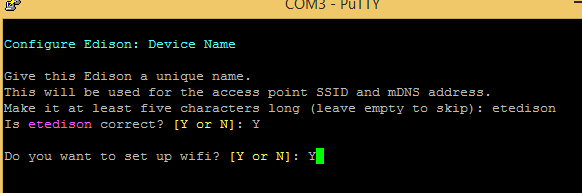
login: root

Jelszó nincs.

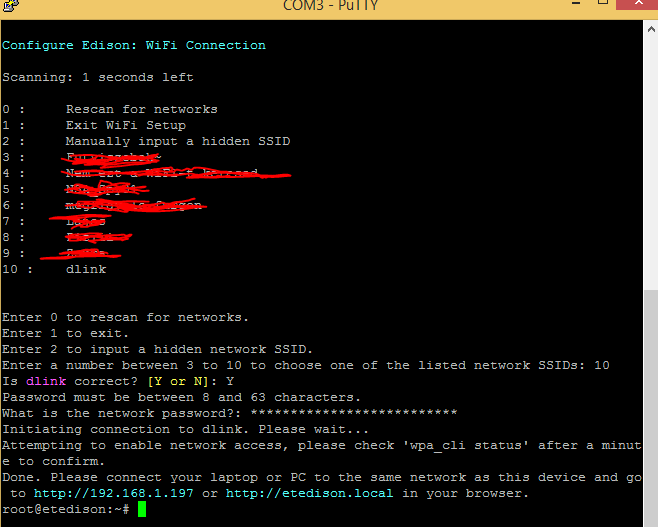
Elvégezzük az első beállításokat, ahol beállítjuk az eszköz nevét, a *root* felhasználó jelszavát és kapcsolódunk egy WiFi hálózathoz.

# configure\_edison --setup



5. ábra: Beállítás, első lépés - jelszó

6. ábra: Beállítás, második lépés - eszköznév



7. ábra: Beállítás, harmadik lépés - WiFi

Most leteszteljük az internetkapcsolatot, a Google DNS szerverének ping-elésével, ez nagy eséllyel online van.

# ping 8.8.8.8

Ha működik az internetkapcsolat, akkor valami hasonlót kell látnunk:

>> PING 8.8.8.8 (8.8.8.8): 56 data bytes  
64 bytes from 8.8.8.8: seq=0 ttl=48 time=52.314 ms  
64 bytes from 8.8.8.8: seq=1 ttl=48 time=55.288 ms  
64 bytes from 8.8.8.8: seq=3 ttl=48 time=69.878 ms

Ilyenkor a **Ctrl + C** billentyűkombinációval megszakítjuk az aktuális program futását.

Ha hibát ír a *ping*, akkor próbáljuk másik WiFi kapcsolatot beállítani, ehhez írjuk be:

# configure\_edison --wifi

### 2.3.2. Csomagok

Először is hozzá kell adni a csomagkezelőhöz a nem hivatalos forrásokat, ezek az Edisonra készült csomagok, csak az Intel nem vállal értük felelősséget.

# vi /etc/opkg/base-feeds.conf

A *vi* szövegszerkesztő megnyílása után, üssük le az **i** billentyűt (insert) és gépeljük be az alábbiakat:

src/gz all http://repo.opkg.net/edison/repo/all  
src/gz edison http://repo.opkg.net/edison/repo/edison  
src/gz core2-32 http://repo.opkg.net/edison/repo/core2-32

Ez után nyomjuk meg az **Esc** billentyűt és gépeljük ezt be:

# :wq

Ennek hatására a *vi* elmenti a változtatásokat és kilép.

Ellenőrizhetjük, hogy ez megtörtént:

# cat /etc/opkg/base-feeds.conf  
>> src/gz all http://repo.opkg.net/edison/repo/all  
src/gz edison http://repo.opkg.net/edison/repo/edison  
src/gz core2-32 http://repo.opkg.net/edison/repo/core2-32

Hozzáadjuk az Intel MRAA könyvtár hivatalos linkjét a csomaglistához:

# echo "src mraa-upm http://iotdk.intel.com/repos/3.0/intelgalactic-dev/opkg/i586" > /etc/opkg/mraa-upm.conf

Frissítsük a csomaglistát:

# opkg update

Töltsük le és telepítsük a szükséges csomagokat:

# opkg install opencv opencv-staticdev mraa git

# 3. Fejlesztőkörnyezet beállítása (Eclipse)

A kényelmes fejlesztés érdekében érdemes beállítani valamely fejlesztőkörnyezetet, természetesen *SSH*-n keresztül *vi*-ben is történhet a programírás, de az nem a legkényelmesebb megoldás.

## 3.1. Kapcsolódás

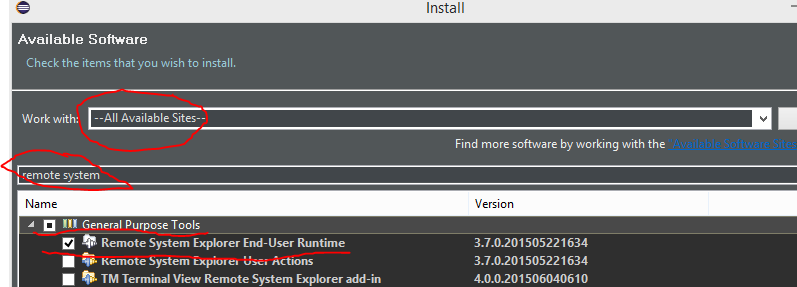
Bár soros porton vagy SSH-n keresztül szerkesztetünk fájlokat a *vi* segítségével, ez a módszer nagyobb forrásfájlok esetében igen kényelmetlen.

Az IoT driverek feltelepítésekor megjelenik a lehetőség az Eclipse fejlesztőkörnyezet feltelepítésére is, azonban ez csak az Edisonon, gyárilag található, linux-al működik együtt.

Az alábbi módszer egy frissen telepített Eclipse fejlesztőkörnyezet beállítását mutatja be, ami így képes lesz ez Edisonon lévő fájlok szerkesztésére és a program lefordítására is.

Megnyitjuk az Eclipse-t és a *Help* menüből kiválasztjuk az *Install New Software ..* menüpontot.

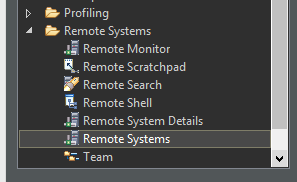
Beállítjuk, hogy az összes lehetséges forrásból keressen és a keresőbe beírjuk hogy "remote system", majd a találatok közül a *General Purpuse Tools* csoportból kiválasztjuk a *Remote System Exploter End-User Runtime* csomagtot.



8. ábra: RSE csomag

A *Next* és *Finish* gombok segítségével telepítjük.

Az Eclipse menüjéből a *Window -> Show View -> Other...* menüpontra kattintunk és a *Remote Systems* csoportból kiválasztjuk a *Remote Systems* elemet. Majd *OK* gombot nyomunk.



9. ábra: RSE megjelenítése

A megjelenő új ablakban csak a "Local" elem látszódik, erre kattintunk és az egér jobb gombját lenyomjuk, a megjelenő menüből a *New -> Connection ..* elemet kiválasztjuk.

A "General" csoportból kiválasztjuk a "Linux" elemet, majd *Next*.

A következő ablakban beállítjuk a kapcsolat címét, és nevét.

Az Edison IP címét az alábbi paranccsal deríthetjük ki az Edison parancssorából:

# ifconfig

>> ...

wlan0 Link encap:Ethernet HWaddr xx:xx:xx:xx:xx:xx

inet addr:192.168.1.197 Bcast:192.168.1.255 Mask:255.255.255.0

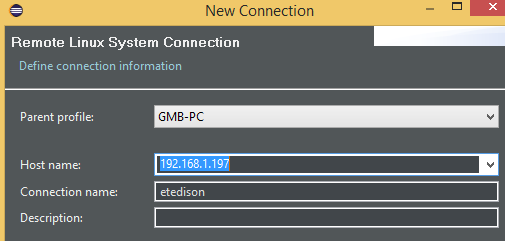
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1

RX packets:447 errors:0 dropped:0 overruns:0 frame:0

TX packets:142 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:1000

RX bytes:35170 (34.3 KiB) TX bytes:18543 (18.1 KiB)



10. ábra: RSE beállítása

Ez után megnyomjuk a *Next* gombot.

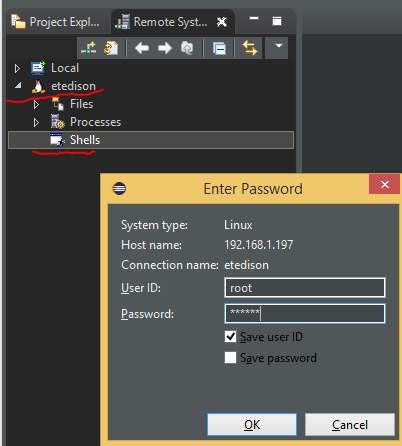
A "Configuration" menüből kiválasztjuk az "ssh.files" elemet, majd *Next.*

A "Configuration" menüből kiválasztjuk a "processes.shell.linux" elemet, majd *Next.*

A "Configuration" menüből kiválasztjuk a "ssh.shells" elemet, majd *Finish.*

Ekkor megjelenik az újonnan létrehozott kapcsolat a "Local" alatt. Ezt lenyitjuk és a "Shells" elemre kattintva jobb gomb lenyomása után kiválasztjuk a *Connect...* menüpontot.

A megjelenő ablakba beállítjuk a belépési adatainkat és megnyomjuk az *OK* gombot. (a felhasználónév "root" a jelszó pedig az ,amit beállítottunk a 2.3. pontban)

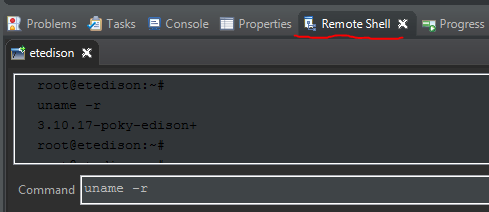


11. ábra: Kapcsolódás az Edisonhoz

Ha sikerült kapcsolódni a kapcsolat ikonján megjelenik egy zöld kis nyíl.

Újra rákattintunk a "Shells" menüpontra, majd jobb klikk és *Launch Shell*.

A megjelenő parancssort ki is próbálhatjuk, lekérdezzük a kernel verzióját.

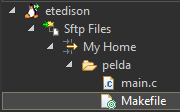


12. ábra: SSH tesztelése

## 3.2. Példa program

A kapcsolat neve alatt lenyitjuk az "Sftp Files" elemet majd a "My Home"-ot. Jobb klikket nyomunk és *New->Folder* paranccsal létrehozunk egy új mappát.

Az új mappára (*pelda*) kattintunk jobb klikkel és 2 új fájlt hozunk létre a *New->File* paranccsal: *Makefile* és *main.c*. Majd mindkét fájlra duplán kattintunk, hogy megnyíljanak a szerkesztőben.



13. ábra: pelda mappa elemei

*main.c* tartalma:

#include <stdio.h>

int

main(void){

printf("Hello World!\n");

return 0;

}

*Makefile* tartalma:

all:

gcc -o hello main.c

**Ctrl + Shift + S** lenyomásával elmentjük a fájlokat.

Majd beírjuk az Eclipse parancssorába:

# cd pelda

# make

# ./hello

>> Hello World!

Ha a fordítás közben hibát ír ki, vagy nem jelenik meg a "Hello World!" felirat, akkor ellenőrizzük az állományok tartalmát.

Ez a program már az Edisonon futott le.

# 4. ’Project Edison’

A vezérlőprogram beszerzése, fordítása és telepítése következik.

## 4.1. Beszerzés

*PuTTY* vagy *Eclispse* segítségével kapcsolódjunk és lépjünk be. Belépés után az aktuális felhasználó *home* könyvtárában találjuk magunkat. Ide töltsük le a *ProjectEdsion* forráskódot a *git* program segítségével:

# ls

>> ProjetEdsion

# git clone https://github.com/g0mb4/ProjectEdison

Ha lekészült a letöltése megjelent egy *ProjektEdison* mappa:

## 4.2. Fordítás

Belépünk a *ProjetEdsion* mappába:

# cd ProjectEdison

Majd elindítjuk a fordítást a *make* paranccsal:

# make

Ha minden csomagot megfelelően telepítettünk, a program sikeresen lefordul.

Működését ellenőrizhetjük:

# make run

A program indulásakor inicializálja a modulokat, hiba esetén egy hibaszámot ad, ami alapján visszakereshető a hiba oka.

A program futását a **Ctrl+C** billentyűkombinációval állíthatjuk le.

## 4.3. Telepítés

# make install

Sikeres fordítás és futás esetén a programot telepíthetjük:

Így a program a rendszer indulásával automatikusan indul majd.