

11. Web szerver készítése az Arduino Ethernet shield segítségével

Ez a projekt egy kicsit eltér a többitől, itt főként szoftveres megjelenítések fognak végbe menni, mivel egy web szervert fogunk létrehozni melyen egy web oldalt is elfuttatunk. Ehhez egy Arduino Ethernet shieldet fogunk használni. Valamint egy potenciométer is segítségünkre lesz annak érdekében, hogy bebizonyítsuk az Arduino képes a bemeneteire érkező jeleket megjeleníteni egy webes felüleleten is keretein belül is.

Szükséges eszközök: Számítógép, Arduino IDE, Arduino Uno, egy db Arduino Ethernet Shield, egy db UTP kábel, egy db potenciométer, egy router internet hozzáféréssel, USB kábel és vezetékek.

Ahhoz, hogy az Arduino és az Ethernet shieldet használni tudjuk nem elég őket összekapcsolni. Mint tudjuk a shieldek vagy más szóval bővítőmodulok egyik fő tulajdonsága, az, hogy az Arduino-ra helyezhetők az erre a célra kialakított csatlakozó tűskéik segítségével. Az előző projektekből már tudhatjuk, hogy ebben az esetben is egy könyvtár hozzáadására biztosan szükségünk lesz. Ám fontos még megemlíteni, hogy a bővítőmodulunk az *SPI* buszt használja kommunikációra az eszközünkkel, ezért tanácsos először az *SPI.h* könyvtárat beszúrni. Majd ezt követően jöhet az Ethernet shield által igénybe vett *EthernetEncLib.h* fájl is. A könyvtárak importálásának a menete már az előzőekben ismertetésre került.

```
#include <SPI.h> //Az SPI kommunikációs protokoll használatához szükséges könyvtár
#include <EthernetLib.h> //Az Ethernet shield által használt könyvtár beillesztése
static unsigned char ipaddr[] = { 192, 168, 1, 147 }; //A szerver ip címének megadása
static unsigned char macaddr[] = { 0x00, 0x11, 0x22, 0x44, 0x00, 0x25 }; //A szerver MAC címének megadása
EthernetLib eLib(80); //A 80-as TCP port használata a Hypertext Transfer Protocol használatára

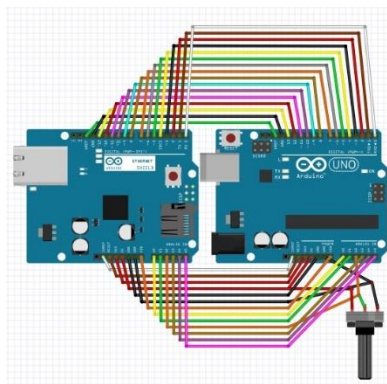
void setup()
{
  pinMode(10, OUTPUT); //Az SS-pin kimenetű alakítása
  Serial.begin(9600); //A soros porton történő kommunikáció bitrátája
  eLib.begin(ipaddr, macaddr); //Az eLib könyvtár használatának indítása valamin az ip és MAC cím átadása
  Serial.println("Szerver elindítva..."); //Megadott karakterlánc kiírása a soros portra
}

void loop() //ciklus
{
  if ( eLib.available() ) //Feltétel vizsgálat:
  {
    if (eLib.isIndexHtml) //Feltétel vizsgálat:
    {
      eLib.print("<HTML>"); //A HTML tag kiírása az eLib könyvtár segítségével
      eLib.print("<head> <meta charset='utf-8' /></head>"); //Karakterkészlet beállítása a Magyar nyelv megjelenítéséhez
      eLib.print("<body style='background-color:#00CED1;'>"); //A body tag kiírása és a weblap hátterének beállítása
      eLib.print("<h1 style='text-align:center;'>ARDUINO WEB SZERVER</h1>"); //A h1-es fejléc tag kiírása
      eLib.print("<h2 style='text-align:center;'>Ethernet shield-el</h2>"); //A h2-es fejléc tag kiírása
      eLib.print("<center>"); //Központozás
      eLib.print("<p>Ez a weblap az Arduino-ra csatlakoztatható bővítőmodulok egyikével az Ethernet shield-el készült.
      eLib.print("<p>Mivel a rendelkezésünkre álló shield nem rendelkezik SD-kártya olvasóval ezért az Arduino UNO R3 F
      eLib.print("<p>Emek ellenőrzé egy alap html oldal megjelenítését aludja végezni. Akár szenzorok által mért adatok
      eLib.print("<br>"); //Sortörés végzése
      eLib.print("<center>"); //Központozás
      eLib.print("<br>"); //Sortörés végzése
      eLib.print("<p>További tudnivalók az ethernet shield-ről és annak beállításairól az alábbi linkeken találhatók:</p>"); //Szöveg megjelenítés
      eLib.print("<p><a href='https://www.arduino.cc/en/Main/Arduino Ethernet Shield V1'>Arduino Ethernet Shield V1</a></p>"); //Híperhivatkozás
      eLib.print("<p><a href='https://store.arduino.cc/arduino-ethernet-shield-2'>Arduino Ethernet Shield V2</a></p>"); //Híperhivatkozás
      eLib.print("<p><a href='https://www.hiwarekita.com/zh/freze=Ethernet+Shield+W5100'>Bevezetéshez itt</a></p>"); //Híperhivatkozás
      eLib.print("</center>"); //Központozás végzése
      eLib.print("</body>"); //A form és body tag-ek vége
      eLib.print("</HTML>"); //A HTML tag vége
    }
    eLib.close(); //eLib könyvtárból való kilépés
  }
}
```

1. ábra Web szerver programkód
(forrás: saját szerkesztés)

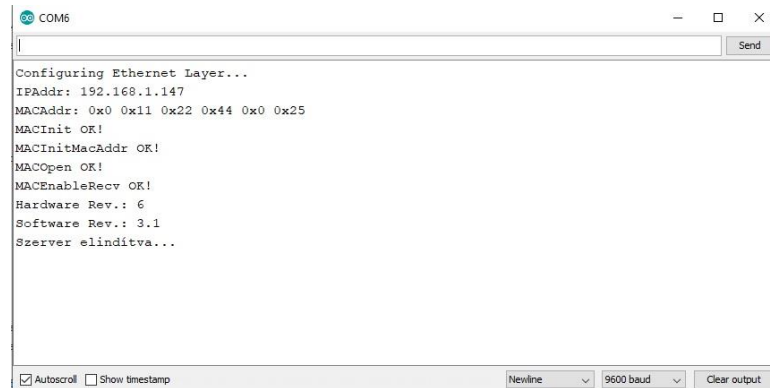
Az előző folyamat végeztével, létrehozunk kettő darab statikus 256 bit nagyságú, karaktereket tartalmazó tömb típust melyek az *ipaddr* és a *macaddr* nevet viselik az első a szerver ip címét tartalmazza a másik pedig az Ethernet shield fizikai MAC címét melyet bizonyos modelleknél megtalálhatunk maga a modulon esetleg hozzá mellékelt dokumentációban, viszont mi magunk is meghatározhatjuk. Ezt követően engedélyeznünk kell a *HTTP*-t vagyis *Hypertext Transfer Protocol*-t, ami egyenlő a 80-as *TCP* port megnyitásával, ezt a következő módon vihetjük véghez, *EthernetEncLib eLib(80)*. Következhet a programunk void setup szekciója, ahol először az Arduino *SPI* busza által használt *SS*-pint kell kimenetként megadnunk. Mivel szeretnénk használni a soros portunkat ezért szükséges azt is engedélyezni a baud érték változatlan, még mindig 9600-as érték. Az Ethernet shield által használt könyvtárunknak meg kell adnunk az ip címet és a MAC címet is egyaránt ez valósul meg a következő sorban, ezen művelet végrehajtása nélkül nem jöhet létre a kapcsolat. Végül tudassuk a felhasználóval, hogy a szerverünk elindult-e vagy sem, ezt egy egyszerű *Serial println* paranccsal oldjuk meg. Ami a program ciklus részét illeti itt rögtön egy feltétel vizsgálatot hozunk létre annak a vizsgálatára, hogy megtörtént-e a kapcsolódás. Amennyiben igen, megkezdődhet a webszerver tartalmának a megírása. Azt szeretnénk, hogy a szerverünk egy egyszerű *HTML* weblapot jelenítsen meg, sajnos sok lehetőségünk nincs mivel a rendelkezésünkre álló shield nem rendelkezik Micro SD-kártya befogadására alkalmas olvasóval, ezért az Arduino memóriáján osztozik mely csupán 32 Kb. Viszont egy egyszerű pár mondatnyi szöveget és egy analóg bemenet jelenlegi állapotának értékét megjelenítő weblapot képes futtatni. Mint tudjuk a weblapot *html* kód segítségével tudjuk megírni és *css* segítségével tudjuk formázni, vagyis nincs más dolgunk, mint a jelenlegi feltétel igaz ágában megkezdeni az oldal felépítését. A *html* kód megjelenítésére az *eLib.print(" ")* parancsot fogjuk használni. Elsőként a *<html>* és a *</html>* tag-eket kell létrehoznunk mivel e két tag között helyezkedik el az oldalunk tartalma. Ezt követően beállítjuk a karakterkészletet, mivel azt szeretnénk, hogy a weblapunk képes legyen a magyar nyelvű magánhangzók megjelenítésére. Ha ezt is megoldottuk akkor jöhet a weblap tartalmának a létrehozása és valamilyen szintű testre szabása. A *body* jelzésű tag-ek között létrehozunk kettő darab fejléct, mint tudjuk a *h1*-es jelzésű fejléc alapján viszonylag nagy méretű szöveget jelenít meg ezért itt adhatjuk meg a weblap tartalmát összegző címet mely a mi esetünkben az *ARDUINO WEB SZERVER* lesz. Ezután beillesztünk néhány sornyi szöveget melyet középre igazítunk. Természetesen sortöréseket is használhatunk a *
*-tag segítségével. Majd a már említett általunk kiválasztott analóg bemenet

aktuális értékét is megjelentjük. Ez a bemenet az A2-es sorszámu analóg bemenet az Arduino lapunkon, az ide csatlakoztatott potenciométer segítségével tudjuk a bemenet értékét szabályozni. Ahhoz, hogy ezt az értéket megjeleníthessük a weblapunk, nincs más dolgunk, mint kiíratni azt a következő módon, `eLib.print(analogRead(A2))`. Így tehát a weblapon megjelenik az analóg bemenetének aktuális értéke. Végül elhelyezzünk néhány hiperhivatkozást is a lapon melyekre, ha egerünkkel rakattintunk akkor egy általunk megadott weblapra ugorhatunk. A linkek elhelyezését követően be kell zárunk a weblap törzsét, ezt a `</body>` tag segítségével tudjuk meg tenni. Majd a `</html>`-tag jelöli a weblap tartalmának a végét. S legvégül bezárjuk az *eLib* könyvtárat.



2. ábra Web szerver bekötési rajz
(forrás: saját szerkesztés)

Első ránézésre bonyolultnak tűnhet a fenti 2.ábra mely szerint a bekötést kell véghez vinnünk, ám de tudjuk, hogy az Arduino bővítőmodulja ráhelyezhetők az Arduino eszközre. Ez az ábra csupán azt jelzi, hogy mely pinek hova csatlakoznak. Viszont a potenciométer csatlakoztatásának menetét itt figyelhetjük meg. Mint látható a vezető, vagyis a változtatható ellenállás középső csatlakozója az Arduino analóg bemenetei közül az A2-es sorszámuára csatlakozik. Az ellenállás bal szélső csatlakozójára kapcsoljunk 5 volt egyen feszültséget a jobb szélső csatlakozót pedig a GND-hez kapcsolunk. Amennyiben mindent az ábra szerint kapcsoltunk össze az Arduino soros portján a következő üzenet jelenik meg.



```
COM6
Configuring Ethernet Layer...
IPAddr: 192.168.1.147
MACAddr: 0x0 0x11 0x22 0x44 0x0 0x25
MACInit OK!
MACInitMacAddr OK!
MACOpen OK!
MACEnableRecv OK!
Hardware Rev.: 6
Software Rev.: 3.1
Szerver elindítva...
```

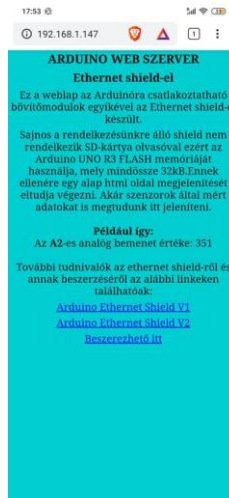
3. ábra Web szerver soros port
(forrás: saját szerkesztés)

Ha pedig egy böngészőprogram segítségével a megfelelő ip címre csatlakozunk láthatjuk az általunk elkészített weblap végső formáját.



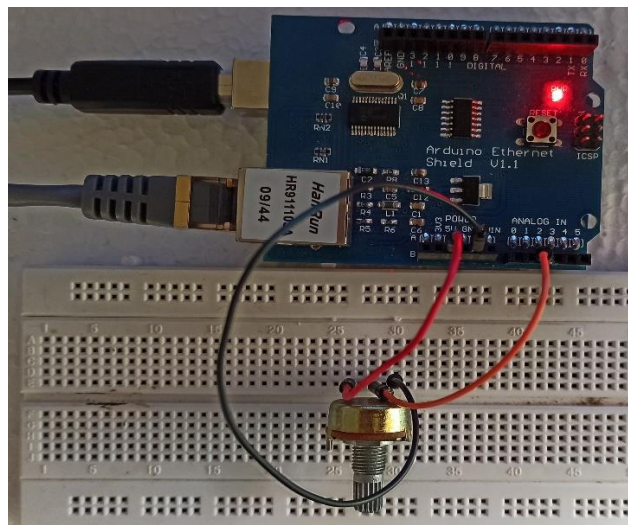
4. ábra weblap megjelenése
(forrás: saját szerkesztés)

Természetesen okostelefonosunkkal is leellenőrizhetjük a weblapunk megjelenésének módját (5.ábra). Mivel az oldal reszponzívan alkalmazkodik a képarányhoz ezért annak tartalma tökéletesen olvasható.



5. ábra weblap megjelenés okostelefonon
(forrás: saját szerkesztés)

Az áramkör megvalósítása fizikai formában pedig a következő 6. ábrán látható.



6. ábra Web szerver fizikai megvalósítása
(forrás: saját szerkesztés)