

Önálló laboratórium beszámoló

Dolgozat címe:

Flexibilisen alkalmazható ipari mérés adatgyűjtő rendszerek kialakítása TCP/IP hálózat használatával.

Konzulens neve: **Tihanyi Attila**

A Hallgató a kitűzött feladatot megfelelő színvonalon és a kiírásnak megfelelően

teljesítette

nem teljesítette

Konzulens aláírása

Hallgató neve: **Balogh Róbert Máté**

Képzés: **Mérnök Informatikus BSC**

Leadás dátuma:

Bevezetés

A mai világban ahol a technológia ugrásszerűen fejlődik , Moor törvény szerint, az integrált áramkörök összetettsége körülbelül 18 havonta megduplázódik. Egyre komplexebb, fejlettebb, több technológia egybeintegrálva elérhető egy helyen. Ezalatt az egyre fejlődő mikroprocesszorokra gondolok, melyek alkalmasak komplexebb feladatok megoldására is. Ez inspirált engem amikor úgy gondoltam, hogy belefogok és megismerkedek a mikrokontrollerek világával bővebben.

Rengeteg potenciál rejlik az olyan számítástechnikai eszközök területén, melyek kihasználják a TCP/IP hálózati protokoll család által nyújtott szolgáltatásokat és a mikrokontrollerek kompaktságát egészen az ipari felhasználástól kezdve az intelligens otthonok vezérléséig.

Azért gondolom, hogy az iparban van létjogosultsága ezen technológiák párosításának, mert például egy betont gyártó üzemben, ami mobilis, kézenfekvő lenne használni egy olyan mérőberendezést, ami megméri a hőmérsékletet, nedvességet, ugye ezek az adatok a legfontosabbak a megfelelő minőségű beton elkészítéséhez. Mivel mobilis üzembről volt szó, ami akár olyan környezetben is fel lehet állítva, ahol az elektromos áram korlátozottan elérhető, hasznos lehet egy olyan kompakt mérőberendezés használata mely egy kis elektromos fogyasztású mikrokontrollerből és a mérőkörökből épül fel, mely TCP/IP hálózaton keresztül kommunikálhat a vezérlő berendezéssel, tehát a tápellátás megoldható TCP kábelen keresztül (Power over Ethernet), vagy akár egy kis teljesítményű akkumulátorral is. Használhatunk a kommunikációhoz Wireless adatátvitelt, ha a kábelek kiépítése nem kézenfekvő. A lehetőségek tárháza szinte kimeríthetetlen, lényegében a legfontosabb a rugalmasság, kompaktság és az energiatakarékosság.

Akár az intelligens házak vezérlésére is alkalmas lehet ,mint már fent

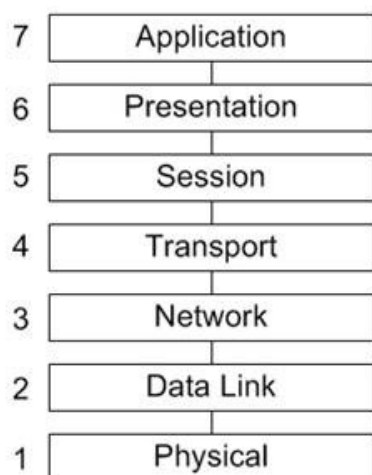
említettem, szerintem a legkézenfekvőbb az lehet, az automatizálást mikrokontrollerrel oldhatjuk meg, gondolok itt az érzékelőkre (hőmérséklet, páratartalom, fényviszonyok, stb.) és a vezérlőkre (ajtónyitás, világítás kapcsolás, klimatizáló berendezés szabályozása, web szerver, stb.) ezen rendszert TCP/IP hálózat felhasználásával kapcsolnám össze, a médium itt is lehet többféle (rádió hullám, réz) éppen mi a kézenfekvő. Ami számomra még érdekesebbé tette ezt a felvetést, az az hogy egy mikrokontrolleren implementált web szerver segítségével irányíthatnánk a interneten keresztül a kertünk öntöző rendszerét a világ túlsó feléről miközben nyaralunk.

Ide írja le a feladatot és a megoldás elő lépéseként következik a TCP/IP összefoglaló

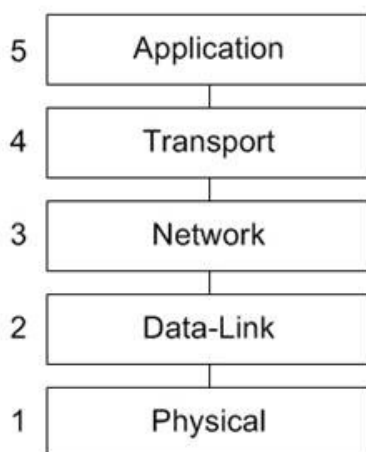
Rövid ismertető a TCP/IP protokoll családról

„Az Internet kialakulásához nagymértékben hozzájárult az Amerikai Védelmi Minisztérium egyik kutatási projektje, melynek célja egy nehezen megsemmisíthető, katonai célú számítógépes hálózat létrehozása volt. E kutatás során kifejlesztettek egy csomagkapcsolt hálózati protokollt mely felépítésével és redundáns útvonalaival biztosítani tudta a megfogalmazott követelmények maradéktalan teljesítését. A protokoll - amely TCP/IP néven vált ismertté - civil alkalmazására először a 70-es évek közepén került sor egy elsősorban az amerikai egyetemekhez kötődő számítógépes hálózatban, hogy később ebből a hálózathoz fejlődjön ki a mai Internet. Napjainkra a hálózat az egész világot átszövi, megváltoztatja az emberek üzletről, szórakozásról, információáramlásról alkotott képét, jelentősége napról napra nő és ezért rendkívül fontos az alapját képező technológia megismerése és megértése. (<http://avalon.aut.bme.hu/education/jegyzet/szeszk/tcpip/tcp.htm>)

A TCP/IP protokollt csomagkapcsolt hálózatok adatátviteli protokolljára hozták létre. Ez egy öt réteget tartalmazó hálózati protokoll rendszer, szemben az ISO OSI hét rétegű modelljével.



OSI Model



TCP/IP Model

Physical: Fizikai réteg, lehet többek között réz, üveg, telefon, rádió.

Data-Link: Ez a szint az OSI modell fizikai és adatkapcsolati rétegéhez áll közel, feladata a kapott byte-ok átvitele. A fölötte levő rétegek csak azt várják el tőle, hogy bájt folyamatokat tudjon fogadni, illetve átadni.

Network: Ez a réteg felelős az adatok átviteléért a hálózaton, a gépek közötti kapcsolat összeköttetés mentes és nem tudja biztosítani a réteg az adatok korrekt átvitelét sem. Ezen a szinten több protokoll is megtalálható, de ezek közül a legfontosabbak: IP(Internet Protocoll), ARP(Adress Resolution Protocol), ICMP(Internet Control Message Protocol).

Internet Protokoll: Ez a protokoll gondoskodik a csomagok átviteléről a hálózaton, az összes kommunikáció a host-ok között IP csomagok formájában történik. Ez egy kapcsolat nélküli protokoll, a kommunikációhoz nem szükséges előzetes kapcsolat felvétel. Adatátvitel szempontjából nem megbízható, a csomagokkal bármi megtörténhet: elveszhetnek, megsérülhetnek, sorrendjük összekeveredhet.

Transport: ez lehet TCP vagy UDP.

TCP: Felfelé megbízható adatátvitelt biztosít úgy, hogy alatta egy megbízhatatlan protokoll található. Felülről bájt folyamatosan lehet táplálni (stream orientált), full duplex átvitelt biztosít, kapcsolat orientált. Kliens- szerver kapcsolatok kialakítására képes, bárki lehet egyszerre kliens és szerver is. A kapcsolat felvétele mindkét oldalon nyugtázva történik. A kezdeményező küld egy kérés csomagot, a címzett válaszol a kérésre, majd a kezdeményező küld egy nyugtát arról, hogy vette a címzett válaszát. A kapcsolatok azonosítására szolgálnak a portok, a szerver egy adott porton érkező kérésekre figyel, és az ott érkező kéréseket kiszolgálja. A TCP portok közül az első 1024 foglalt a standard alkalmazások számára. A megbízhatóság érdekében a protokoll a teljes TCP csomagra számol ellenőrző összeget. A megbízható kapcsolat kialakítására pozitív nyugtázást használ. A küldő oldal nyugtát vár minden egyes elküldött TCP csomagról. Amennyiben egy később küldött csomagról előbb kap nyugtát, mint egy korábban küldött csomagról, akkor a korábban küldött csomagtól kezdve megismétli az adást. Egy TCP csomag szintén két részből áll egy header-ből, és az áthiendő adatokból.

UDP: Ez sokkal gyorsabb protokoll, mint a TCP protokoll, viszont nem megbízható adatátvitel szempontjából. Nem kapcsolat orientált, nincs hibajavítás, nincs nyugtázás. Tulajdonképpen az IP szint által biztosított szolgáltatásokat nyújtja felfelé. Akkor szokták használni, ha az adatátvitel sebessége a legfontosabb, minden többi feladatot a felette elhelyezkedő réteg lát el. Tipikusan a DNS-ek (Domain Name Server), real-time alkalmazások, játékok szokták használni (egy játékban vagy real-time hang átvitel esetén ha egy csomag rossz akkor ott legfeljebb döccen egyet, de ez még mindig kisebb baj, mintha az adott pontnál megállna és onnantól elkezdené újra adni a csomagokat). A szegényesebb szolgáltatásból adódóan sokkal egyszerűbb az UDP header.

Application: Ezen a szinten helyezkednek el az alkalmazások. Az adatok átvitelét TCP, vagy UDP porton keresztül történő hívásokkal valósítják meg. A portok tulajdonképpen a kommunikációs csatorna egy végpontjának az azonosítására szolgálnak a szabvány 65535 TCP és 65535 UDP portot engedélyez, ezek közül az első 1024 foglalt szabványosított protokollok számára (pl.: Telnet, SMTP, POP3, Rlogin, FTP, SNMP, STMP, HTTP). Az összeköttetés kliens szerver alapon valósul meg. Az egyes szerver alkalmazások az adott host gép operációs rendszerénél bejegyeztetik magukat, hogy egy adott TCP, vagy UDP portra érkező kérések kiszolgálásáért az adott alkalmazás a felelős. Az egyes kliens alkalmazások, amikor megszólítják a szerver gép adott partját akkor az ott futó operációs rendszer értesíti a szerver alkalmazást arról, hogy kérés érkezett az adott partra. Azaz amikor egy szolgálat azonosítására van szükség a hálózaton akkor nem elég a szerver gép IP címét megadni, hanem szükséges az adott szolgáltatáshoz kapcsolódóan megadni a szerver adott TCP, vagy UDP partját.

A TCP/IP nagy hibája, hogy prozentation réteget nem tartalmaz.

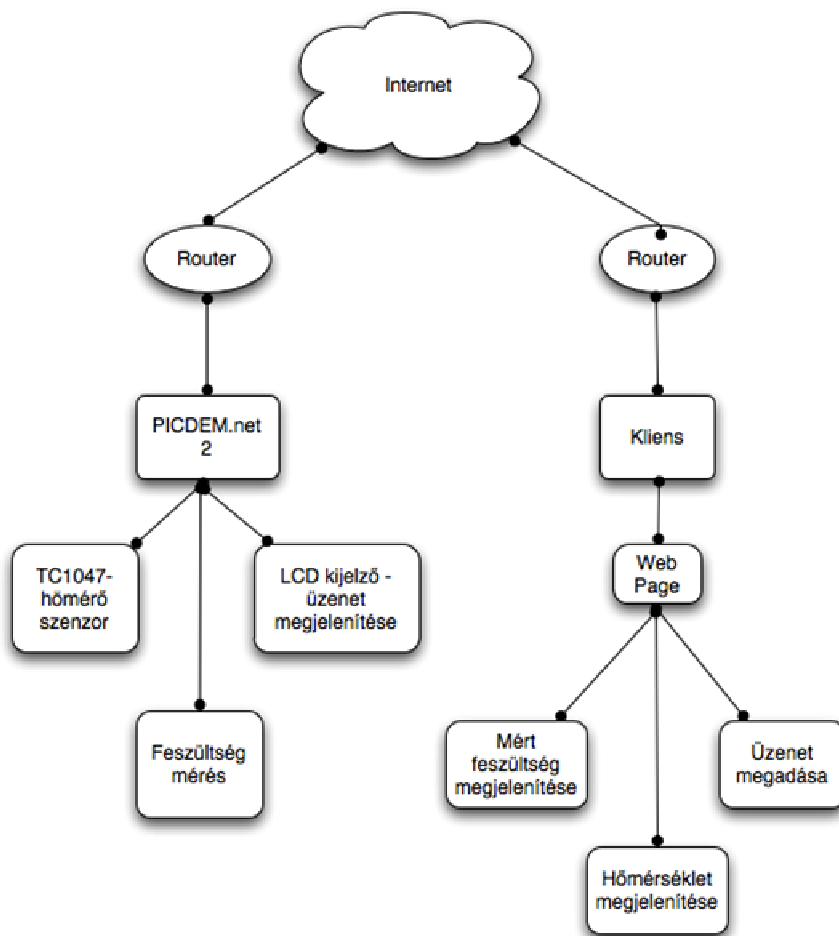
Rövid ismertetése a feladatnak

Az általam választott feladat a Flexibilisen alkalmazható ipari mérés adatgyűjtő rendszerek kialakítása TCP/IP hálózat felhasználásával című téma.

Célom ezt a témát tovább vinni diploma munkának, ennek első lépése az önálló laboratórium projekt elkészítése. A önálló laboratórium projekt keretein belül, el szeretnék végezni egy olyan mérést, ahol a mérő berendezést egy mikrokontrollerrel valósítom meg és ennek a mérésnek az eredményét TCP/IP hálózat és a mikrokontrolleren implementált web szerver segítségével megjeleníteni egy Web-es felületen. Tulajdonképpen két mérést végeznék el, hőmérséklet mérést és egy feszültség mérést rögzíteném, majd ezeket a valós idejű méréseket egy Web-es felületen jelenítem meg.

Az általam kiválasztott mikrokontrollert egy MICROCHIP PIC 18F97J60, 8 bit-es mikrokontroller, integrált Ethernet vezérlővel ami kompatibilis az IEEE 802.3 szabvánnyal.

Felhasznált eszközök



A fejlesztés során egy MICROCHIP PICDEM.Net 2 fejlesztő eszközt használtam és egy PicKit3 típusú programozót. MPLAB Ide fejlesztő környezettel dolgoztam és a 4.13-as verziójú MICROCHIP Ip Stack-et használtam a munkán során.

A kitűzött feladat egy web server implementálása a mikrokontrolleren, és a fejlesztő eszközön elhelyezett hőmérséklet szenzor és a potméter által mért feszültség érték megjelenítése egy webes felületen.

Kezdetek

Hogyan választottuk a microchip pic kontrollert és miért

Első lépésnek meg kellett ismerkednem a Pic 18F97J60-as mikrokontroller tulajdonságaival, szerencsére a MICROCHIP által nyújtott dokumentáció igen alapos, majd a fejlesztő eszköz áramkörének az tanulmányozása következett.

Következő lépésként össze kellett állítanom a fejlesztő környezetet, be kellett kalibrálnom a fordító programot, IP Stacket. Már ez a része a projektnek sem volt olyan könnyű feladat, mint számítottam rá, sok kompatibilitásból fakadó problémát kellett elhárítanom, amire le tudtam fordítani az első saját alkalmazásomat és le tudtam programozni a mikrokontrollert. Ezt a problémát a MICROCHIP hiányos dokumentációnak tudható be, és a legfrissebb kiadású PIC C18-as fordító program a PIC 18F mikrokontroller család számára hibás volt, sajnálatos módon az általam használt mikrokontrollert is érintette ez a hiba. A hiba miatt a mikrokontroller másodpercenként újraindult és ezt az állapotát nem lehetett megszakítani. Hosszas utánaolvasással megtaláltam a fordító hibás header fájl-t és ezt egy előző verziójú fordítóprogram header fileja segítségével kijavítottam. Sajnos a MICROCHIP meg nem javította ki a hibás fordító programot. Ezek után már csak a megfelelő típusú IP stack-et kellett installálnom. Itt is sokadik próbálkozásra sikerült eredményt elérnem, több verzióval is próbálkoztam, bizonyos funkciók nem működtek megfelelően az általam módosított példa programmal.

Másodsorban át tanulmányoztam a példa alkalmazásokat. A MICROCHIP példa alkalmazásai között ráleltem egy példa web szerver alkalmazásra. Ezt az alkalmazást tanulmányozva próbáltam megérteni a program működését. További fejlesztéseket kellett végeznem, hogy a hőmérséklet mérés algoritmusát implementálni tudjam és a web szervert fel tudjam készíteni a mérési eredmények megjelenítésére a web oldalon. Majd a potméteres feszültség mérést is implementálnom kellett, itt a mikrokontroller AD konverterének a működését kellett megvizsgálnom, természetesen a hőmérséklet mérés esetén is.

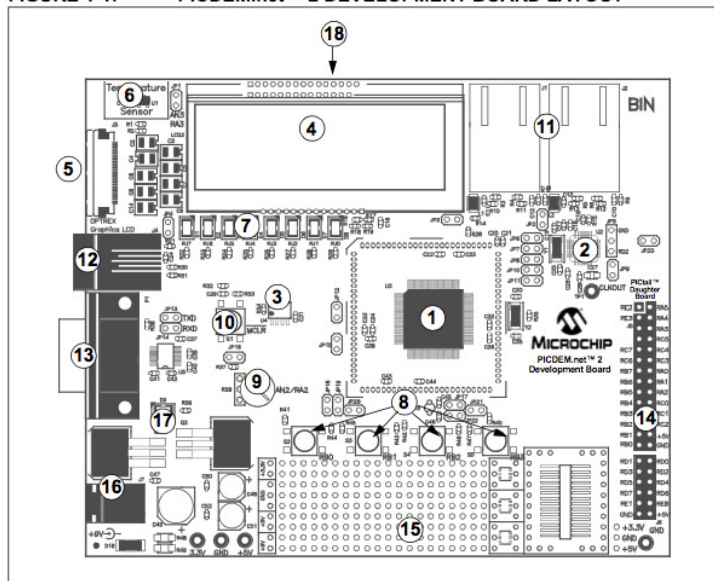
Harmadsorban áttekintettem a HTML nyelv szintaktikáját. El kellett készítenem egy egyszerű HTML oldalt ahol meg tudom jeleníteni a mért adatokat, tanulmányoznom kellett a HTML dinamikus változók tulajdonságait is. Ennek a részfeladatnak a megoldása során is egy példa alkalmazásból indultam ki, melyet áttanulmányozva el tudtam készíteni a saját alkalmazásomat.

A feladat részletes ismertetése

PICDEM.net 2 ismertetése:

Ez az eszköz a MICROCHIP által nyújtott egyik hálózati feljeleztő környezete. Azért esett a választásom erre a developer board-ra, mert két RJ45-ös hálózati ajzat is található rajta, az egyik vezérlését a lapon dedikált PIC 18F97J60-as mikrokontroller végzi, melyben egy beépített Ethernet vezérlő található. A másik csatlakozó vezérléséért egy külső Ethernet vezérlő felelős az ENC28J60 vezérlő. Azért előnyös számomra ez a felépítés, mert rálátást nyerhetek arra hogy, hogyan lehet használni egy külső Ethernet vezérlőt egy MICROCHIP PIC mikrokontrollerrel összehangolva, és tanulmányozhatom a belső Ethernet vezérlő működését is.

FIGURE 1-1: PICDEM.net™ 2 DEVELOPMENT BOARD LAYOUT



A fejlesztő eszközön installált TC1047 típusú hőmérő egység, mely kimenete feszültség és referencia feszültségként a mikrokontroller tápfeszültségét kapja ami 3.3V. Majd a mért feszültséget a mikrokontroller AD konverter bemenetére van kötve a AN3-as lábra.

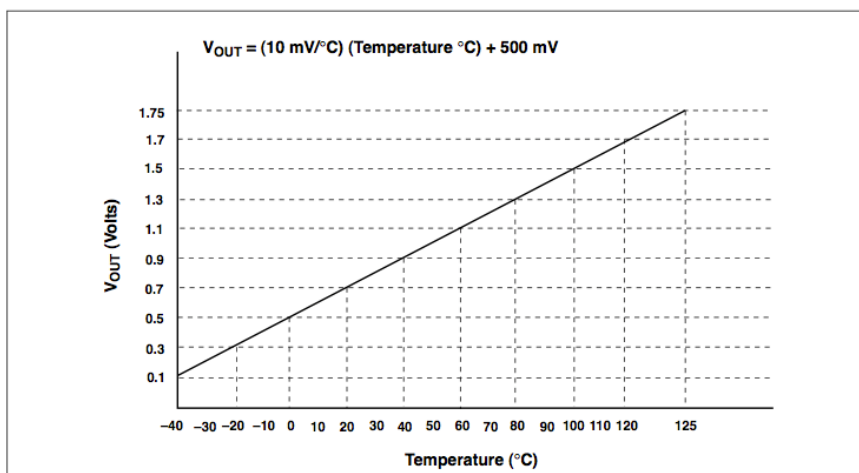


FIGURE 4-1: Output Voltage vs. Temperature.

A fenti ábrán látható függvény alapján lehet kiszámolni a mért feszültségből a hőmérsékletet ($V_{out} = (10\text{mV}/^\circ\text{C})(\text{Temperature } ^\circ\text{C}) + 500\text{mV}$). A

PIC 18F97J60 -as mikrokontrollerben 10bit-es AD konverter található, $2^{10}=1024$, tehát 1024 bit felel meg 3.3V-nak azaz 1024 bit a felbontás ebben az esetben, ami azt jelenti, hogy 1 bit = 3.22 mV-nak felel meg. Az AD konverter által mért érték az ADRES regiszterben tárolódik. Így könnyen meghatározható a Vout ($V_{out} = \text{AD konverter kimenet (bit)} * 3.22\text{mV}$), így már a fenti függvény alapján ki is tudjuk számolni a hőmérsékletet.

Az áramkörön installálva van egy potméter is, amely működési elve hasonlít az előbb említett hőmérséklet szenzor működési elvéhez. Referencia feszültségnek a potméter is 3.3V feszültséget kap. Majd a kimeneti feszültség a mikrokontroller AD konverter bemenetére van kötve a AN2-as lábra. 10 bit-es AD konverter esetén 1 bit = 3.22mV-nak felel meg, mint a fent említett esetben. Az AD konverter által mért érték az ADRES regiszterben tárolódik szintén.



Továbbá az áramkörön installálva van egy külső EEPROM tároló is, a mi a Web lapok tarolására szolgál. A Web lapok bináris formában tarolódnak, szerencsére a MICROCHIP rendelkezésre bocsájtott egy olyan konverter programot, ami a HTML oldalt átalakítja bináris formára és ezt a binárist lehet letölteni a külső EEPROM tárolóba. A web oldal feltöltésére több lehetőségünk is van FTP-n keresztül tudjuk feltölteni a fájlt, továbbá elérhető egy online feltöltő opció is a web böngészőbe begépeljük, hogy <http://192.168.1.1/mpfsupload>

ahol a 192.168.1.1 az eszköz IP címe és ekkor megjelenik egy online feltöltő form, ahol ki tudjuk választani a feltölteni kívánt fílet.

Az áramkörön Installálva van egy LCD kijelző is. Alap információk megjelenítése, IP cím, IP Stack firmware verzió megjelenítésére szolgál. Továbbá a web lap megjeleníti az LCD kijelzőn megjelenített karaktereket és van lehetőség arra, hogy üzenetet küldjünk az LCD kijelzőre a web lapon keresztül.

A PICDEM.net 2 eszközt egy PicKit 3 programozóval lehet programozni többekközött, mely egy RJ-11 típusú csatlakozón kapcsolódik az áramkörhöz. Csatlakoztatom a PicKit 3 programozó eszközt az áramkörrel az RJ-11 csatlakozón és USB porton keresztül a számítógéphez, majd a megfelelő hex kiterjesztésű állományt le lehet tölteni a mikrokontrollerbe. Azért esett a választásom erre a programozó eszközre, mert található benne Debug funkció is, mellyel a programon soronként végig tudok lépni, ezalatt vizsgálva a helyes működését, így az esetleges hibák könnyebben felderíthetők, illetve jobban nyomon követhető a program működése..

Az áramkörön található egy RS232-es interfész, mely segítségével bekonfigurálhatjuk a hálózati paramétereket (Hálózati maszk, Fix Ip cím, Defalut Getaway, Primary DNS, Secondary DNS, aktív DHCP engedélyezése,).

Használata:

A PICDEM.net 2 fejlesztő eszközt csatlakoztatni kell a helyi számítógépes hálózatba, akár egy router-en keresztül. Így az eszköz automatikusan megkapja az IP címét ami majd megjelenik az LCD kijelzőn is és már elérhetővé vált, vagy akár közvetlenül hozzá lehet csatlakoztatni a megfelelő kábellel a számítógépünk hálózati interfészéhez, de előbb ki kell kapcsolni az eszközben az active DHCP-t és be kell állítanunk egy fix IP címet, arra figyelve, hogy egy hálózati szegmensben legyen a számítógépünk hálózati interfészével. Azért van szükség az aktív DHCP kikapcsolására az eszközben, mert amíg él ez a beállítás,

addig az eszköz arra vár hogy kiösszák neki az IP címét, ami például egy router feladata lenne, de a mi számítógépünk ezt nem teszi meg.

Amint csatlakoztattuk az eszközt a hálózathoz, a fent említett bármely módon és az LCD kijelzőn leolvasható az eszköz IP címe, akkor már elérhetővé vált a helyi hálózatban. Begépelhetjük az eszköz IP címét egy web böngészőbe, ami támogatja a HTML protokollt, ekkor megjelenik a készülék kezdőlapja, ahol máris leolvashatjuk a pillanatnyi hőmérsékletet és a potméter által mért feszültség értékét és láthatjuk az LCD kijelzőn megjelenített karaktereket. Akár írhatunk egy üzenetet is a webes felületen, ami megjelenik az áramkör LCD kijelzőjén.

Akkor, ha az interneten keresztül szeretnénk megtekinteni a Web lapot, ahhoz át kell konfigurálnunk a helyi hálózatunkat. Elsősorban a routert-ben konfigurálni kell a Port Forwarding opciót a az áramkör IP címének a 80-as portját kell továbbítani, tehát ezzel azt érjük el, hogy az internet felől az eszközünk 80-as TCP portja fog látszani, ami mögött az eszközünkön implementált web szerver fogja a kéréseket kiszolgálni. Továbbá egy Domain nevet kell hozzárendelnünk az helyi hálózatunk külső internet felőli IP címéhez. Én a Dyndns.org ingyenes szolgáltatását vettem igénybe, azért esett erre a szolgáltatásra a választásom, mert az otthoni helyi hálózatomnak nincs fix külső internet felőli IP címe, ezért a dyndns.org elérhetővé tesz egy olyan alkalmazást (Dyndns Updater), ami frissíti a hálózat külső internet felőli IP címét az általunk lefoglalt Domain névhez, amit szintén ingyenesen megtehetünk a dyndns.org-nál. Így már csak annyi a dolgunk, hogy egy web böngészőbe begépeljük az általunk kiválasztott Domain nevet(<http://baloghrobert.dyndns.org>).

Konklúzió

A félév elején nagy energiával vágtam bele a feladat tanulmányozásába, kezdetben nagyon alábecsültem a feladat komplexitását, folyamatában döbbsentem rá, hogy igencsak nagy fába sikerült belevágni a fejszém. Önmagában is elég komplex feladat egy mikrokontrollert programozni, nekem sokkal nehezebbnek tűnik mit például Java nyelven elkészíteni egy GUI alkalmazást. Tovább fokozza a nehézségeket a TCP/IP hálózatos kommunikáció, az időm nagy része a ezen téma tanulmányozásával telt, ez is sokkal bonyolultabbnak tűnt mint elsőre gondoltam volna. A web szerver működésének megértése sem volt olyan egyszerű, és mindezek tetejében még a web fejlesztésbe is bele kellett folyni picit. Ennek a projektnek az egyik fő tanulsága számomra, hogy sokkal több időt kell szánni egy eddig idegen technológia megismerésére, hogy el tudjunk kezdeni dolgozni vele érdemben. Sok időt vett igénybe az irodalom tanulmányozása is .

Továbbra is szeretnék ezzel a témával foglalkozni a diplomamunka keretén belül is. Következő lépésként azt tervezem, hogy az eddig megszerzett tapasztalatok alapján, egy egyedi áramkört szeretnék megtervezni ami képes több (0-20 mA, 4-20 mA, 0-10 V) analóg valamint több digitális ki/bemenet kezelése. Az eszközhöz lehessen csatlakoztatni RS422, RS485, RS232 kapcsolaton keresztül további mérő és vezérlő intelligens eszközöket. Az internet felé továbbra is TCP/IP hálózati kapcsolatot használna a rendszer és egy központi adatbázisba gyűjthetné a mért értékeket és állapotokat. Továbbá megszeretném valósítani, hogy valamilyen mobil platformról, például iPhone, Andoid-ról kapcsolni lehessen a mikrokontroller egyes funkcióit, ezalatt a két állapotú kimenetek vezérlését értem. Amit még fontosnak tartanék megvalósítani az az, hogy nem csak UTP kábellel tudna az eszköz a hálózathoz kapcsolódni, hanem WiFi használatával is, de ez a része a projektnek sok kutatással jár amire implementálni tudom a mikrokontrolleren.

Sajnos nem sikerült minden elképzelésemet megvalósítani ebben a félévben amit szerettem volna, és viszonylag csak egy egyszerű rendszert tudtam

összeállítani, de a további munkához elengedhetetlen tapasztalatokat szereztem, viszont sokkal több kutatást kell végeznem a közeljövőben, ha meg szeretném valósítani az elképzeléseimet.

Ezúton szeretnék köszönetet mondani Tihanyi Attila tanár úrnak, aki rengeteget segített a feladat megoldása során, sok ötletet, instrukciót kaptam tőle.