

SZAKDOLGOZAT

Simon Béla
Pásztor Gábor

Debrecen
2010

Debreceni Egyetem

Informatikai Kar

**Mikrokontrolleren alapuló időjárás előrejelző
hálózat**

Szakdolgozat

Témavezető:

Dr. Tóth László

egyetemi adjunktus

Nagy Gábor

NI Hungary

Készítette:

Simon Béla

Pásztor Gábor

Mérnök informatikus BSc

Debrecen

2010

Tartalomjegyzék

1. Bevezetés	5
2. Technikai fejlődés hatása a meteorológiai előrejelzésre	7
2.1. A meteorológia kialakulása	7
2.2. A technológia kapcsolata a meteorológiával	9
3. Az Időjárásról	15
3.1. A légkörről	16
3.2. A légkör szerkezete	16
3.3. A nyugalomban lévő vígőzzel teli levegő fizikai állapotjelzői	17
3.4. A légköri mozgások	18
4. A mikrokontrollerek	21
4.1. PIC 18F család	21
5. PIC Perifériák	24
5.1. Kommunikációs perifériák	24
5.2. Univerzális aszinkron adóvevő	26
5.3. I2C	28
5.4. Ethernet	32
6. Felhasznált IC-k	36
6.1. TMP101	36
6.2. ENC28J60	39
7. Eredmények	43

8. Összefoglalás	49
9. Köszönetnyilvánítás	51
10. Mellékletek	54

1. fejezet

Bevezetés

Miért pont meteorológia és miért pont mikrokontrollerek? Jogosan adódik a kérdés a szakdolgozat címe láttán. A válasz abban rejlik, hogy a meteorológia fejlődése minden ember számára fontos. Egy pontosabb méréssel vagy ötlettel, esetleg egy gyorsabb előrejelzéssel katasztrófák vagy akár az egész földre kiható időjárási változások lennének megelőzhetőek. A témaválasztást egyrészt a mikrokontrollerek utáni érdeklődés és az összes bennük rejlő lehetőség kiaknázása ihlette, másrészt a különböző szenzorok alkalmazása megteremtette annak a lehetőségét, hogy komolyabb méréseket tudjunk megvalósítani.

Rövid bevezetés után a dolgozat második fejezete elénk tárja a meteorológia kialakulásának történetét, kezdve az időjárással és matematikával foglalkozó tudósok felfedezéseivel egészen az elektronika terén elért sikereken át napjainkig. A természeti és technológiai szálak összefűzéséhez szükségünk van az egyes időjárási fogalmak tisztázására és megértésére.

A harmadik fejezet az időjárásról, időjárási tényezőkről, ezek kölcsönhatásairól, légkörről, különböző légmozgásokról (vízszintes-függőleges) adnak képet.

Az időjárási fogalmak és ismeretek tárgyalása után a következő szakaszban a mikrokontrollerekről, működésükről, felépítésükről esik szó.

A ötödik fejezetben a mérőszenzorok és mikrokontrollerek kapcsolatát biztosító perifériák tárgyalása, az egyes kommunikációs szabványok bemutatása történik.

Az hatodik fejezet betekintést ad a felhasznált (pontos mérésre alkalmas) szenzorról, illetve a közte és a mikrokontroller között zajló kommunikáció problémájának megoldásáról. Itt kellett megismernem és megértenem az I2C és az SPI megoldásokat természetesen szenzor és más IC specifikusan. Ezek teszik lehetővé, hogy skálázható időközönként adatok áramoljanak bitek formájában a mikroprocesszor felé. A felhasznált IC-k adatlapjainak beszerzése, tanulmányozása, az egyes kommunikációs szabványokkal való egyeztetése után elkezdődhetett a legnagyobb nehézséget jelentő programozás fázisa. Ebbe értem mind az önálló programok létrehozását illetve a Microchip Technology Inc. által kiadott általános célú több ezer soros források megértését, majd módosítását. Ezen részben kerül tárgyalásra az Ethernet vezérlő és működtetése is.

A hetedik fejezet a gyakorlatban felmerült problémákról, a munkafázisokról és eredmények bemutatásáról szól.

2. fejezet

Technikai fejlődés hatása a meteorológiai előrejelzésre¹

2.1. A meteorológia kialakulása

A meteorológia (régies nevén légtüneménytan) az időjárás folyamatokkal és előrejelzésekkel foglalkozó tudományág, a megfigyelhető időjárás eseményeket magyarázza meg. Ezek a Föld légkörében található változóktól függenek, például a hőmérséklet, légnyomás, a páratartalom, valamint ezek kölcsönhatása egy adott időn belül. A Földön megfigyelt időjárás események nagy része a troposzférában játszódik le.

A meteorológia tudományával már az idősámításunk előtti 4. században Arisztotelész is foglalkozott, a kifejezés is *Meteorológia* c. művéből ered.

Mivel ekkor még nem voltak precíz mérő berendezések ezért főként tapasztalatokra és megfigyelésekre épült az akkori meteorológia. A változásokra általában valami misztikus illetve földönkívüliekkel kapcsolatos magyarázat volt. Szellemeknek, isteneknek, démonoknak a befolyását hangoztatták. A 18. században a tudósok különböző kezdetleges műszerek segítségével már méréseket tudtak végezni. A méréseket felhasználva felfedezték a különböző légköri áramlásokat valamint a különféle képződményeket, főleg ciklonokat és anticiklonokat.

¹A fejezet forrásai: <http://www.idokep.hu>
, <http://wapedia.mobi/hu/Meteorologia#1>.
, http://www.met.hu/omsz.php?almenu_id=misc&pid=metsuli&mpx=0&pri=0&sm0=&dti=3&tfi=0
, http://hu.wikipedia.org/wiki/A_számítógép_története

1607-ben Galileo Galileinek sikerült egy olyan eszközt építenie, ami képes volt a testek hőmérsékletváltozását jelezni (**termoszóp**). Torricelli 1643-ban a róla elnevezett csőben lévő higany magasságát figyelve felfedezte, hogy a légnyomás állandóan változik, így az első **barométer** is az ő nevéhez fűződik. Torricelli eredményeit Pascal pontosította, felfedezése szerint a légnyomás csökken ahogy nő a tengerszint feletti magasság, ebből pedig az következik, hogy a légkör felett vákuum van. Az első **szélmérő** Robert Hooke találmánya 1667-ben. Benjamin Franklin megfigyelése szerint az észak-amerikai időjárás rendszerek nyugatról keletre mozognak, a villámlást elektromos jelenségnek tulajdonította, továbbá térképpel szolgált a Golf-áramlatról. A páratartalom mérésére Horace de Saussure megépítette az első **hygrometert**. Az időjárás megfigyelések összefoglalását még mindig akadályozta az egyes időjárás sajátosságok leírásának nehézsége (felhők, és szelek). Ezt a problémát oldotta meg Luke Howard és Francis Beaufort amikor bemutatták rendszerüket a felhők (1802) illetve a szelek sebességének (1806) osztályozásáról. A Krím-félsziget mellett [1] 1854. november 14-ére virradó éjszaka a Balaklavai-öbölben néhány óra alatt elpusztult az egyesült angol-francia hadiflotta, mert az öbölbe hirtelen pusztító erejű vihar érkezett. A francia kormány Le Verrier csillagászt (2.1. ábra) azzal bízta meg, hogy az akkor Európában és az észak-atlanti térségben már működő 45 meteorológiai állomás adatainak vizsgálatával utólag derítse ki, látható volt-e előre a vihar közeledte. A megelőző napok megfigyelési adatainak térképre vitelével (*szinoptikus térkép: az aktuális időjárás helyzet komplex analízise, és előrejelzés készítése a szinoptikus elvek és a numerikus modellek segítségével*), felismerhető és nyomon követhető volt a ciklon, amelyhez a vihar kapcsolódott.



2.1. ábra. Le Verrier²

A 20. század elején egyre gyakoribbak voltak az úgynevezett "szabad-léggömb", majd mérőműszerekkel felszerelt **ballon-szonda** felszállások. A meteorológia a közlekedés egy

²http://www.wired.com/images/index/2008/01/le_verrier_500px.jpg

másik ágával, a repüléssel került szoros kapcsolatba. Nem csak a magas-légkörről szerzett ismeretek járultak hozzá az első naptól kezdve a repülés biztonságához, hanem a repülés is sokat adott a meteorológiának a légkör jobb megismerését szolgáló erőfeszítésben.

Az első világháború idején az előrejelzés tudománya (**szinoptika**) forradalmi változásokon ment keresztül. Az elméletek továbbgondolásának székhelye Norvégia lett. A szinoptikában V. Bjerknes és J. Bjerknes tették a mai napig is talán az egyik legnagyobb felfedezést, miszerint az időjárás változásának legfőbb okai az **időjárás frontok**. Továbbá kimondták, hogy a frontoknak meghatározott szerepe van a légkör illetve a ciklonok szerkezetében. Erre később egy új tudományág épült a frontológia.

Lehetővé vált a légkör fizikai állapotának és a benne lejátszódó mozgások, áramlások fizikai törvényeinek megismerése. A szinoptikus térképek egymásra következése az időjárás törvényszerűségeinek fokozatos megismerését segítette elő. Az akkori szakember az előrejelzés módszerként a térképek sorozatából megismert folyamatot, vagyis a már bekövetkezett időjárást a jövőre extrapolálta.

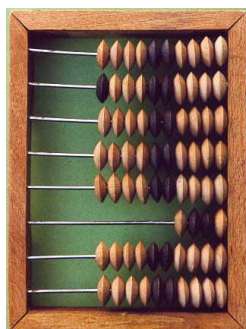
2.2. A technológia kapcsolata a meteorológiával

Az emberi történelem során bekövetkező ipari forradalmak olyan változásokat eredményeztek, melyek hatással voltak mind a társadalmi, gazdasági és technológiai életre. A gyáripar térhódítása szükségessé tette az olyan új találmányok alkalmazását, amelyek csökkentik az emberi beavatkozás mértékét, felgyorsítják a számolási feladatokat, automatizálják az egyes folyamatokat.

Egyre sürgetőbb volt a technikai és szervezeti feltételek megteremtése, azért hogy a meteorológiai állomások adatai nagyon rövid idő alatt kerüljenek a feldolgozási helyekre, így elemzésük, következtetések levonása, előrejelzés készítése minél hamarabb és pontosabban történjen. Nemzetközi egyezmények születtek az adatok kölcsönös cseréje érdekében, a csere eszköze a távíró volt. Magyarországon 1870-ben alakult meg az első önálló nemzeti meteorológiai szervezet, az Országos Meteorológiai és Földdelejtési Intézet, ami elsősorban az állomáshálózat szervezésével, a mérések beindításával, a mérési eredmények éghajlati feldolgozásával foglalkozott. A század első felében kezdtek foglalkozni a dinamikus meteorológia

eszközeivel, azokat elméleti síkon, matematikai egyenletek segítségével írták le. Ekkor már napirenden volt az egyes meteorológiai elemek légköri paramétereinek (mezőinek) számszerű előrejelzése is.

A matematika, a számolás tudománya már az ókorban foglalkoztatta az emberiséget, legjobb példa erre az **abakusz**³ (2.2. ábra), ami rudakon, drótokon vagy hornyokban ide-oda mozgatható golyókat tartalmaz. Az egy-egy rúdon lévő golyók helyzete egy-egy számjegyet, a rudak egy-egy helyiértéket jelentenek. Így egy hatsoros (hat rudat tartalmazó) abakuszon a legnagyobb ábrázolható szám a 999 999. Az összeadás és a kivonás igen egyszerűen és gyorsan elvégezhető abakusszal, a szorzás és az osztás sokkal körülményesebb.



2.2. ábra. Abakusz⁴

Az első ismert mechanikus számológép megalkotója **Wilhelm Schickard**. Az ő gépében az átvitelt egy tízfogú és egy egyfogú fogaskerék végezte, ez a gép már mind a négy alpműveletet el tudta végezni, 1643-ben **Pascal** épített egy tízes számrendszerre épülő fogaskerékes összeadó-kivonó gépet, amelyet 1647-ben szabadalmaztatott. A gép a szorzást és osztást nem volt képes elvégezni. Pascal gépét **Gottfried Wilhelm Leibniz** fejlesztette tovább, így mind a négy alpművelet elvégzésére képes volt. Leibniz vetette fel először a kettes számrendszer használatát, de a számok hossza miatt ezt nem tudta megvalósítani. Az áttörést **Charles Xavier Thomas de Colmar** mechanikus számológépe jelentette (1820), ez már kereskedelmi forgalomba is került. Mind a négy alpműveletre képes volt, a feltaláló már ekkor felvetette egy programvezérelt gép építésének gondolatát is.

Johan Müller fogalmazta meg azt, hogy szükség van a részeredmények tárolására, ez jelentette az első lépést a programozás feltalálásához. A tárolókat regisztereknek nevezte el.

³<http://www.ttk.pte.hu/ami/phare/tortenet/abakusz.html>

⁴<http://www.informatik-vollmer.de/software/latex2man.php>

Kevesebb, mint fél évszázaddal később Joseph Marie Jacquard olyan mechanikus szövőgépet épített, mely automatikusan, külső programozás révén szőtt mintákat. A gépet lyukkártyákkal vezérelte, ezek tartalmazták a mintákat.

Az 1930-as évek végén John Vincent Atanasoff és asszisztense, Clifford Berry megterveznek egy csak elektronikus egységekből álló gépet, ez volt az első elektronikus digitális számítógép, az *Atanasoff - Berry Computer* (ABC).

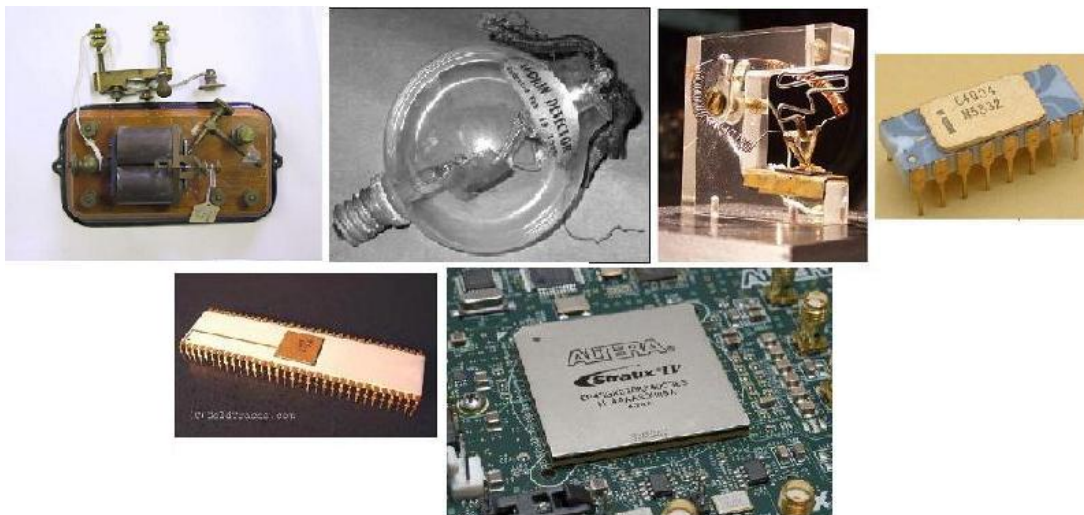
A második teljesen elektronikus számítógép az *Electronic Numerical Integrator and Calculator* (ENIAC) a Pennsylvania Egyetemen készült, ez már első generációs számítógép. A számításához szükséges adatokat tárolta, a programot kapcsolótáblákon kellett beállítani. Elektroncsővel működött (2.3. ábra), a programozása gépi nyelven történt, a meghibásodási ráta magas volt. Az elektroncső az elektronikában használt aktív eszköz, amely elektronok vákuumban vagy gázzal töltött térben való áramlásán alapul. Az elektroncső⁵ egy többnyire hengeres védőburában rögzített, elektródákat tartalmazó, légmentesen lezárt elektronikai eszköz. Az elektronikus számítógépek logikai tervezésében kiemelkedő érdemeket szerzett a magyar származású **Neumann János**. Alapvető gondolatait a kettes számrendszer alkalmazása, memória, programtárolás, utasításrendszer Neumann-elvekként tartjuk számon. Neumann János irányította az EDVAC megépítését is 1944-ben, amelyet 1952-ben helyeztek üzembe. Ez volt az első olyan számítógép, amely a memóriában tárolja a programot is. Ennek a számítógépnek a terve és a továbbfejlesztett Neumann-elvek alapján készülnek a mai számítógépek is.

A második generációs számítógépek már tranzisztorokat (2.3. ábra) (Walter Brattain és John Bardeen) tartalmaztak elektroncsövek helyett. A tranzisztorban⁶ három különböző szennyezettégű, rétegszerűen egymáshoz csatlakozó tartományt van. A kapcsolások jellegétől függően feszültség, áram és teljesítmény erősítésére egyaránt felhasználható. Ennek hatására a számítógépek megbízhatósága rohamosan javult, és az áramszükségletük is a töredéke lett a korábbiaknak. A tranzisztor a tömeges elterjedését a szórakoztató elektronikának köszönhette. A második generációs gépeknél jelenik meg a megszakításrendszer, amelyekkel a hardveres jelzéseket a számítógépek kezelni tudják. Ekkor jelentek meg az operációs rendszerek, valamint a magas szintű programozási nyelvek. Memóriaként mágnesrákat használtak, a háttértár mág-

⁵<http://hu.wikipedia.org/wiki/Elektroncső>

⁶<http://www.sulinet.hu/fizika/tranyo/tranyo2.htm>

nesszalag, majd mágneslemez volt.



2.3. ábra. Az első relay, elektroncső, tranzisztor, mikroprocesszor, mikrokontroller és FPGA napjainkból⁸

A harmadik generációs számítógépek abban tértek el legfőképpen az előzőektől, hogy már integrált áramköröket használnak (IC), amiket 1965-ben találtak fel. Ezek képesek voltak arra, hogy egy időben több feladatot is végrehajtsanak a multiprogramozásnak és a párhuzamos működtetésnek köszönhetően. Megjelent a grafikus monitor, és a programozási nyelv is közértetőbbé vált (BASIC). Fejlődésnek indult az adatátvitel is. 1971 hozza a nagy fordulatot: John Blankenbaker megépíti az első személyi számítógépet a Kenbak I-t.

A negyedik generáció kezdetének a világ első mikroprocesszorának megjelenését tekintjük (2.3. ábra). Nincsenek alapvető változások a számítógépek szervezésében, csupán a korábbi megoldásokat tökéletesítik. Ezek már nagy integráltságú áramköröket használnak. Erre a korszakra jellemző, hogy a szoftvergyártás óriási méretűvé válik. A szoftverek árai elérik, egyes

⁸http://research.microsoft.com/en-us/um/people/gbell/cybermuseum_files/bell_artifacts_files/images/B68_Telegraph_Receiver_and_Relay.JPG
<http://www.antiqeradio.com/images/May04-Bonhams-Audion.jpg>
http://www.hobbielektronika.hu/cikkek/files/etan_tranyo/Elso_tranzisztor.jpg
<http://retrothing.typepad.com/photos/uncategorized/2008/11/16/intel4004.jpg>
<http://www.goldtraces.com/pics/TexasInstru/SE1099.jpg>
http://theasicguy.com/blog/wp-content/misc/Altera_FPGA.jpg

esetekben meg is haladhatják a hardverét.

Az 1980-as évek elején olyan félvezető eszközök fejlesztése kezdődött (2.3. ábra) (**Field-Programmable Gate Array**⁹), amely "logikai blokk"-oknak nevezett programozható komponenseket és programozható összeköttetéseket tartalmaznak.

Az 1950-es évek elején lehetőségessé váltak a számítógépes számítások és kísérletek. Az első időjárás-előrejelzések a *barotropikus* (olyan légkör, amiben a levegő sűrűsége csak a légnyomástól függ) karakterisztikákból származnak, amelyekből sikeresen modellezhető a középmagas légkörben elhelyezkedő nagy mozgású *Rossby hullámok*, amik az alacsony, illetve a magas légköri nyomást rajzolják ki. Az 1960-as években ismerte fel Edward Lorenz az időjárás kaotikus természetét, megtalálva a káoszelmélet alapját. Ez a fejlődés tette lehetővé a jelentősebb előrejelző központokban készített jelenlegi globális időjárás-előrejelzést, amely figyelembe vette a kaotikus természetnek köszönhető bizonytalanságot.

A **TIROS-1** időjárás-előrejelző műhold 1960-as fellövése óta lehetőség van az űrből történő megfigyelésekre, kiterjesztve ezzel a megfigyelhető területek méretét. Az időjárás műholdak a különböző magasságokban keringő általános célú megfigyelőműholdakkal együtt nélkülözhetetlen eszközzé váltak a kiterjedt jelenségek tanulmányozásának, az erdőtüzeztől az El Nino-ig.

Napjainkban szuperszámítógépek (**Föld-szimulátor**, Japán) fejlesztésével a légkör matematikai leírása egyre pontosabbá válik. Ezen nagyteljesítményű számítógépek képesek a Földet egységes éghajlatrendszerként modellezni, ahol az atmoszféra, az óceán, a növényzet, és az emberi tényező függ egymástól. Az éghajlatváltozás előrejelzése alapján véve ezeknek a rendszereknek köszönhető.

Ha az elmúlt három évszázadot egy-egy technológiával jellemeznénk, elmondhatnánk, hogy a 18. századot az ipari forradalom során megjelenő nagy mechanikai rendszerek, a 19. századot a gőzgép, a 20. századot pedig az információgyűjtés, az információfeldolgozás és az információterjesztés jellemzi. A számítógépek és a távközlés egybeolvadása alapvetően befolyásolta a számítógépes rendszerek szervezését. A régi modell az volt, hogy egy intézmény teljes számítástechnikai igényét egyetlen gép szolgálta ki, azonban ezt a modellt felváltotta az, hogy a feladatokat sok-sok különálló, de egymással összekapcsolt számítógép látja el. Két

⁹[http://hu.wikipedia.org/wiki/FPGA_\(Field-programmable_gate_array\)](http://hu.wikipedia.org/wiki/FPGA_(Field-programmable_gate_array))

számítógépről akkor mondjuk, hogy összeköttetésben állnak, ha képesek információt cserélni egymással. A kapcsolatnak nem feltétlenül rézvezetéken keresztül kell megvalósulnia; fényvezető szálak, mikrohullámok, infravörös fényt vagy kommunikációs műholdakat is használhatunk. A hálózatoknak sokféle alakjuk, méretük és formájuk lehet. Az Internet azonban nem számítógéphálózat, hanem hálózatok hálózata.

A világon számos hálózat létezik, és ezek hardvere, valamint szoftvere sok esetben eltér egymástól. Azok a felhasználók, akik egy adott hálózathoz kapcsolódnak, gyakran szeretnének más hálózatokhoz kapcsolódó felhasználókkal is kommunikálni. Ez az igény váltotta ki a különböző, egymással sokszor nem kompatibilis hálózatok összekapcsolását, amit általában egy átjárónak nevezett számítógép segítségével valósítanak meg. Az ily módon összekapcsolt hálózatokat együttesen internetworknek, vagy röviden **internetnek** nevezzük. Tudósok és fejlesztők éveken át párhuzamosan dolgoztak anélkül, hogy tudtak volna egymásról. Ezt a korlátot kezdetben a hadsereg ívelte át a központok összekapcsolásával, majd egyetemek, intézmények is hasonló módon szerveződtek. A végeredmény az ez egész világ behálózása az információ megosztása, áramlása érdekében.

3. fejezet

Az Időjárásról¹

Mielőtt még meteorológiáról beszélnék először is meg kell említenem néhány fontos fogalmat, amely nélkülözhetetlen az időjárás pontos definíciójához. Ezek meghatározása előtt tisztázni kell, hogy időjárásunkat rengeteg tényező befolyásolja. Itt említhetjük meg a földfelszínre jutó napsugárzás mennyiségét vagy a domborzatot is. Az időjárás a légkör folyamatainak, és fizikai tulajdonságainak egy adott környezettel való kölcsönhatása. Környezetünket a növénytakaró és az emberi tevékenység egyaránt befolyásolja, a növénytakaróban zajló energiaátalakulások kihatnak a légkör állapotára. Az előbb említettek alapján bizonyos állapotjelzők tetszőleges értéket nem vehetnek fel, például Magyarországon a legalacsonyabb hőmérséklet nem érheti el a -50°C -ot. Mindezzel azt akartam megmutatni, hogy a légkör változásai meghatározott határokon belül maradnak. Ezek az állapotjelzők jól körülhatárolhatóak, így meghatározható egy rendszer, amely ezek változásait tartja valamilyen keretek között, valamint meghadható egy egyensúlyi állapot is, mely mentén az állapotokat leíró tényezők jó közelítéssel behatárolhatóak. Az éghajlati vizsgálatoknál főképp a földfelszínnel érintkező néhány méteres légréteget mondhatnánk a legmeghatározóbbnak. Persze nem azt mondjuk hogy pl. 15 km-es magasságban vagy feljebb nem következnek be fontos változásokat előidéző kölcsönhatások, de ha a megfigyelésekre és a "pontos" mérésekre támaszkodunk, akkor azt mondhatjuk, hogy a légkör nem ezen része a meghatározó.

Az éghajlatot illetve az éghajlattant gyakran a "klíma" szóval is illetjük, ami valójában a görög "klinein" (hajlani) szóból ered. A fogalmat Arisztotelészhez köthetjük.

¹ A fejezet forrása [2] és [13]

3.1. A légkörről

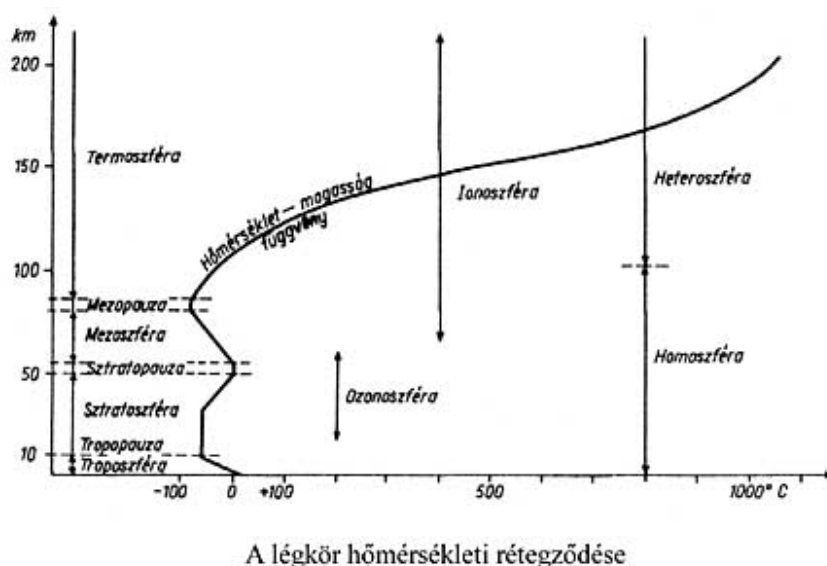
Részletesebben tanulmányozva a Föld légkörét, azt vehetjük észre, hogy számos gáz keverékből áll.

- Nitrogén: 78,084%
- Oxigén: 20,946%
- Argon: 0,934%
- Szén-dioxid: 0,032%

Megfigyelhetők olyan gárzszecskék, molekulák, amelyek állandóan jelen vannak légkörünkben, és vannak természetesen olyanok is, amelyek folyamatosan változnak, vagy több különböző kölcsönhatás eredményeképpen jönnek létre. A levegőben található továbbá úgynevezett **aeroszol** részecskéket is, amelyek "diszperz állapotú szilárd és cseppfolyós anyagok".

3.2. A légkör szerkezete

A légkört szerkezetileg több részre oszthatjuk, minél feljebb haladunk a földfelszíntől annál jobban változik a légkör kémiai és fizikai sajátossága. A legelső réteg 87 km magasságig terjed, **homoszférának** nevezzük. Feljebb haladva a 90 km felett már a légkör összetétele is megváltozik, ez **heteroszféra**. Az előbb említett két réteg után már a hőmérséklet változása szerint 5 részre oszthatjuk a légkör tartományát. A felszín felett elhelyezkedő körülbelül 12 km-es réteget az jellemzi, hogy 100 méterenként a hőmérséklet $0,65^{\circ}\text{C}$ -al csökken. Ebben a rétegben képződik a csapadék, itt helyezkedik el a vízgőz nagy része. Ezt a réteget **troposzférának** hívjuk. Tovább haladva elérünk a **sztratoszférához**, ahol a hőmérséklet nagy mértékben megnövekszik az ózon jelenléte miatt. A sztratoszférát elhagyva 50-90 km-es magasságban a **mezoszférába** jutunk, ahol a hőmérséklet ismét óriási mértékben lecsökken akár -92°C -ra. E felett pedig a **termoszféra** helyezkedik el (3.1. ábra). A légkör 1000 km feletti rétegét **exoszférának** nevezzük, ahol a hőmérséklet kb. 1000°C .



3.1. ábra. A légkör szerkezete [2]

3.3. A nyugalomban lévő vígőzzel teli levegő fizikai állapotjelzői

A légkörben található vízmennyiség majdnem teljes része a **troposzférában** található, ez Föld teljes vízkészletének körülbelül 0,0009 %-a. A légkörben lévő vízmennyiség elsősorban vízgőz formájában található meg. Ha a vízgőz részecskék összeállnak, csapadék keletkezik, amely megjelenhet esőcsepp vagy hó formájában is.

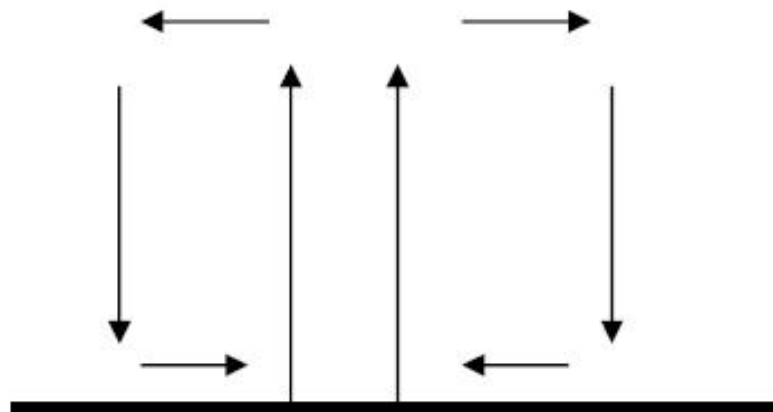
Itt meg kell említenünk két fontos fogalmat:

- **Harmatpont:** Harmatpontnak nevezzük azt a pontot, ahol a relatív nedvesség eléri a 100 %-ot, azaz ahol a levegő gőznyomása megegyezik az ugyanahhoz a hőmérséklethez tartozó telítettségi gőznyomással. Másképpen fogalmazva: az a hőmérséklet, amelyre le kell hűteni a vízgőzt tartalmazó telítetlen levegőt, hogy elérje a telítettséget.
- **Relatív nedvesség:** A relatív nedvesség nem más, mint a nedves levegő vízgőznyomásának, illetve telítési vízgőznyomásának hányadosa.

3.4. A légköri mozgások

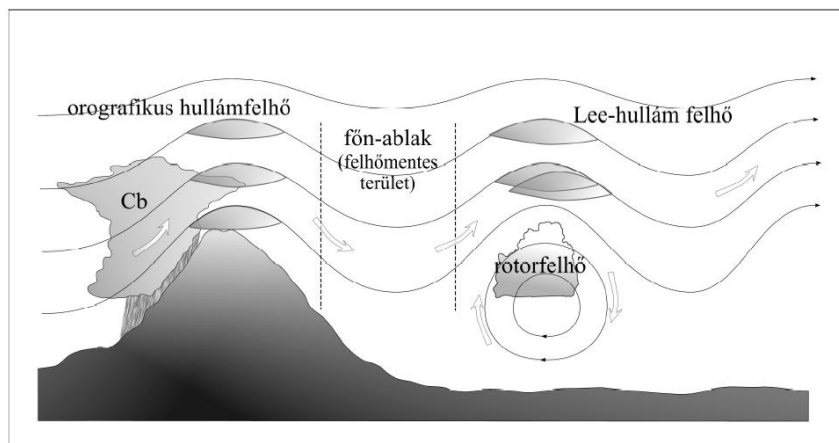
Eddig csak a nyugalomban lévő légkörről beszéltünk, de a légkörben illetve a levegőben lévő részecskék soha sincsenek nyugalomban, állandó mozgásban vannak. Itt meg tudunk különböztetni két csoportot a mozgás iránya alapján:

- **A függőleges légmozgások kialakulása:**A függőleges légmozgások keletkezésének több okát is megkülönböztethetjük, mégis a legmeghatározóbb tényező a földfelszín felmelegedésére vezethető vissza. A földfelszín nem egyenlő mértékben melegszik fel a változó domborzat, illetve a felszínen található növényzet miatt, ezért a levegő sem. Mivel a meleg levegő kiterjed és a sűrűsége a környezetéhez képest csökken, emiatt felemelkedik, ezt konvektív feláramlásnak nevezzük. Miközben ez a feláramlás végbemegy közben vannak leszálló részek, és úgynevezett **konvekciós cellák** (3.2. ábra) alakulnak ki.



3.2. ábra. Konvekciós cella [2]

A 3.3. ábra jól szemlélteti, hogy a feláramlást tovább "szabályozzák" a különböző **orográfiai akadályok**, mint a domb illetve hegyvonulatok.

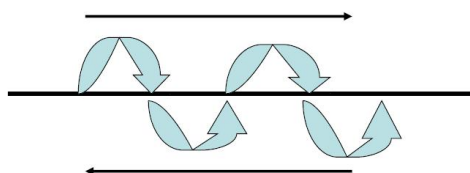


3.3. ábra. Orográfiai akadályok [13]

Itt meg kell említenünk a **turbulencia** fogalmát, ami akkor keletkezik, amikor a vízszintes levegőről talajközeli sűrűdés esetén a változó hőmérsékleti eloszlás hatására különböző lég-részek válnak le örvényes mozgás közben.

Végül de nem utolsó sorban a függőleges légmozgásokat okozhatják a különböző hőmérsékletű és sűrűségű légtömegek találkozásai. Amikor a sűrűbb hideg és a kisebb sűrűségű meleg levegő találkozik akkor a meleg légtömeg a hideg **lejtős** határfelületén feláramlik. Továbbá lehet olyan változat is, hogy két eltérő sűrűségű légréteg egymás felett ellentétes irányú légmozgást végez(a határfelületen pedig hullámmozgás figyelhető meg).

A 3.4. ábra ezt jól szemlélteti:

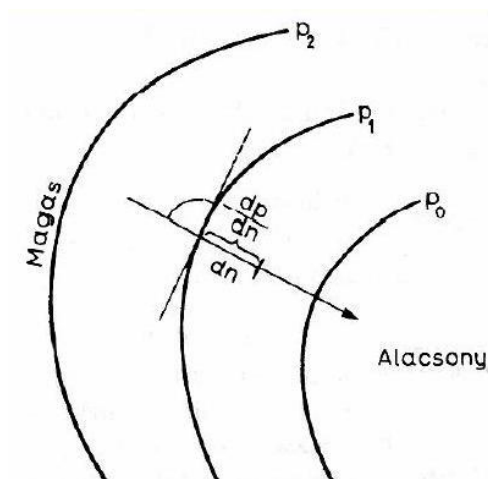


3.4. ábra. Határfelületi hullámmozgás [13]

- **A vízszintes légmozgások kialakulása:**

A légkörben fellépő vízszintes irányú mozgásokat **szélnek** nevezzük. A szél egy vektormennyiség, amit irányával és nagyságával (sebességével) adhatunk meg. A bárius mező

(légnomás térbeli eloszlása) fontos jellemzője a Bárikus gradiens (3.5. ábra, vagyis légnyomási gradiens. E vektor nagysága a nyomáskülönbséggel arányos, iránya a legerősebb nyomásváltozás (megállapodás szerint a legnagyobb nyomásesés) irányába mutat. Ez az irány a nyomási felülethez húzott normális irány.



3.5. ábra. Bárikus gradiens és gradiens erő [2]

Fontosnak tartottam ismertetni az előbbieken említett tudományág és az elektronika tudományának kapcsolatát, hiszen mindössze 39 éve 1971-ben jelent meg az első mikroprocesszor (2.3. ábra), 10 évvel később az első IBM személyi számítógép, és innentől kezdve már számítógépek vesznek részt az újabb számítógépgenerációk létrehozásában, 2004-ben a magyar háztartások 35%-ban volt legalább egy személyi számítógép, sőt szinte minden hétköznapi elektromos készülékben megtalálhatók a mikrokontrollerek. A számítógépek megjelenése minden korábbi technikát és technológiát átalakított, beleértve a meteorológiát is.

4. fejezet

A mikrokontrollerek¹

Az első mikroszámítógép (2.3. ábra) 1976-ban jelent meg, TMS 1000 néven a Texas Instruments termékeként. Ahogy a 4.1. ábra szemlélteti, szemben a mikroprocesszorokkal, egyetlen tokba épített adatmemória, programmemória és I/O egység jellemezte. Alig 50 utasítást ismert, közel 2300 tranzisztort tartalmazott és 16 lábú DIP tokba (Dual In-Line Package) szerelték. 1977-ben az Intel is piacra dobta a saját mikrovezérlőjét a 8048-at. Ez is Harvard architektúra volt, bár a programmemóriát kívülről kellett csatlakoztatni ugyanarra a címvonalra. A későbbi mikrovezérlői is nagy népszerűsége tettek szert, 1988-ig több mint egy billiót adtak el belőlük.

A Harvard-architektúrát kezdetben a Signetics 8x300 mikrokontrollerben használták, majd a General Instruments adaptálta és periféria illesztő vezérlőiben (**P**eripheral **I**nterface **C**ontroller) alkalmazta. A gyártás később a Microchip Technology-hoz került, s a PIC lett a cég fő gyártmánya.

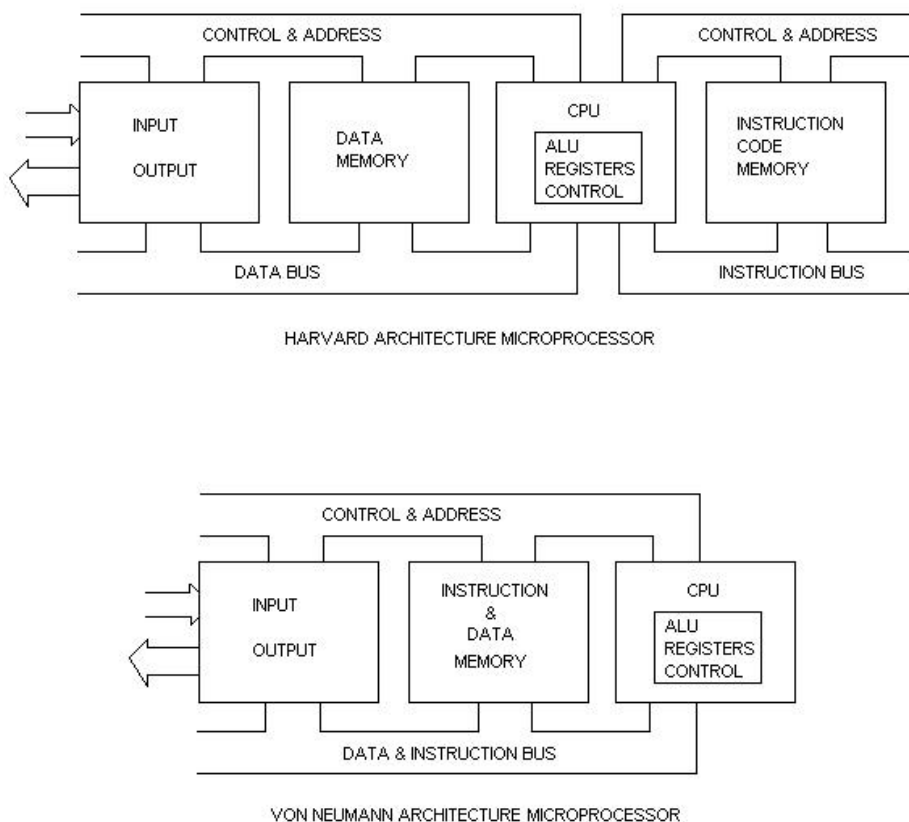
Mikroszámítógépnek vagy mikrogépnek az integrált áramkörökből (félvezető lapkán, esetleg lapkákon kialakított nagyon kis méretű áramkör) felépített programot végrehajtani tudó rendszert nevezünk.

4.1. PIC 18F család

Mivel a Microchip Technology PIC-jeit használtam fel a szakdolgozathoz, ezen mikrokontrollerek tárgyalása következik.

¹A fejezet forrása [3] és <http://plc.mechatronika.hu/mikrovez/tortenet.htm>

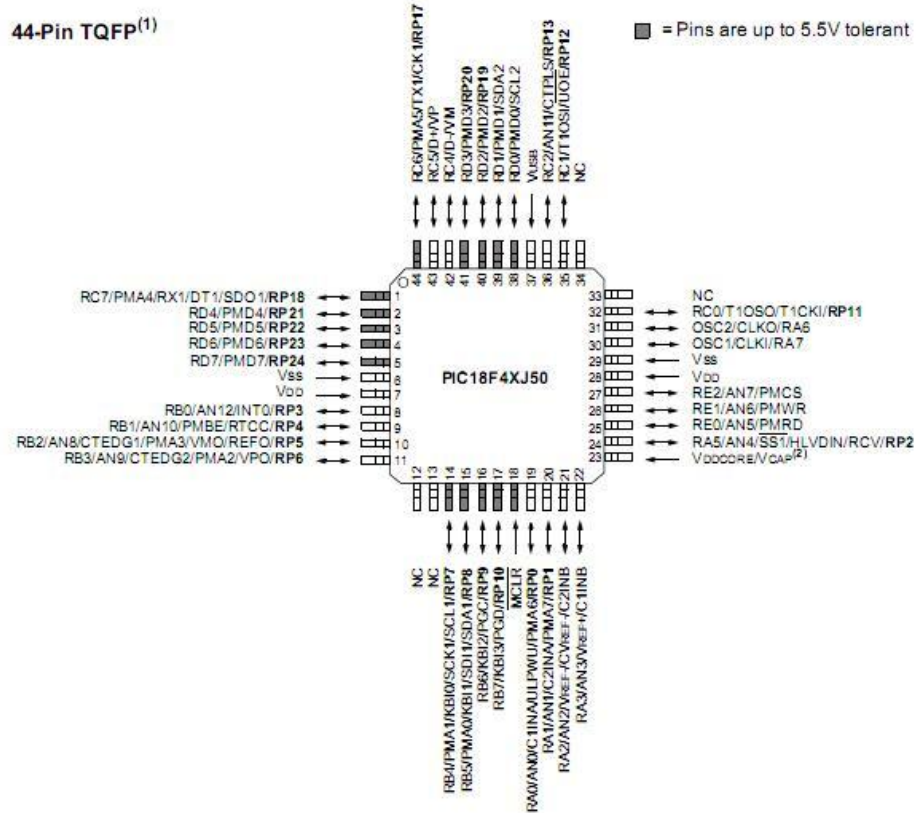
²http://www.electronics.dit.ie/staff/tscarff/architecture/neumann_harvard.gif



4.1. ábra. RISC-CISC architektúra²

A PIC családokat az egyes PIC-ek utasításainak hossza alapján csoportosíthatjuk. A névben az "F" a felhasznált (flash) memóriát jelöli.

A PIC18F termékcsaládra jellemző, hogy tápfeszültsége 2,0 - 3,6 V között lehet, órajele maximum 40/48MHz, teljesítménye 12 MIPS 3 V-os tápfeszültségnél, a programmemória újraírhatósága 1000-10000 közötti. Ha több mint 32kb-ot programmemóriát igényel az alkalmazás, akkor ez a legolcsóbb megoldás. A dolgozathoz PIC18F45K20 és PIC18F46J50 mikrokontrollereket használtuk fel, az utóbbi lábkiosztása a 4.2. ábrán látható. Azért erre a controllerre esett választásunk, mert megfelelő számú lábbal rendelkezik, támogat USB, három vezetékes SPI, és I2C kapcsolódási lehetőséget, maximum 13 bites Analóg/Digitális átalakítót, 64 Kbyte programkód tárhelyet.



4.2. ábra. PIC18f4xj50 [4]

5. fejezet

PIC Perifériák¹

A periféria nem más, mint egy olyan hardver, amely segítségével egy gazdaszámítógéphez további eszközök illeszthetők, lehetővé téve a külvilággal történő kapcsolattartást. Léteznek **analóg**, **digitális** és **kommunikációs** perifériák.

5.1. Kommunikációs perifériák

Ilyen perifériák felhasználásával kapcsolatot létesíthetünk és információt cserélhetünk más eszközökkel.

A PIC-hez csatlakoztatott perifériával kapcsolatot tartani különböző parancs-és státuszbitok segítségével van lehetőségünk. A **parancsbitekkel** a periféria működésmódját szabályozhatjuk, példa erre az I/O lábak adatarányát meghatározó TRIS regiszterek értékei, a **státuszbitok** a perifériától érkeznek, valamilyen állapotát jelezve a gazdagépnek. Továbbá beszélhetünk **be-meneti** és **kimeneti** adatbitekről attól függően, hogy a külvilág felől érkeznek-e a perifériába a bitek, vagy éppen a perifériából a gazdagépbe.

Megkülönböztethetünk **soros** és **párhuzamos** kommunikációt attól függően, hogy egy időpillanatban hány bitet továbbítunk. Hatékonyabbnak tűnhet a párhuzamos kommunikáció, azonban fizikai jelenségek miatt bekövetkező, a jelek egymásra való romboló hatása elveti ezt a feltételezést. A végeredményben gyorsabb és biztonságosabb soros kommunikációról a később-

¹A fejezet forrása [5]

biekben esik szó.

Tekinthetjük még az adatáramlás irányát aszerint, hogy egy időpillanatban csak egy irányban folyhat-e (half-duplex), vagy mindkét irányba (full-duplex).

A kommunikációs csatorna **két vonalat** tartalmaz: az adatvonalat és az órajelet továbbító vonalat. A kommunikációban szereplő **felek** szerint megkülönböztetünk adót és vevőt, attól függően, hogy melyik fél vezérli a kommunikációt, azaz ki az aki a szinkronitást meghatározza (Master-Slave kapcsolat). Ez a kapcsolat nem szükségszerűen 1:1 számosságú, de egy kapcsolatban csak egy Master van akár kötetlen számú Slave-vel. A perifériák állapotát figyelhetjük úgy, hogy

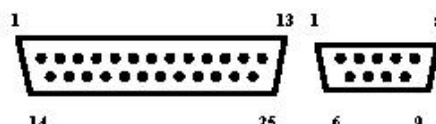
- a futó programhurokban folyamatosan figyeljük a státuszbit(ek) állapotváltását (**polling**), vagy
- a státuszbit(ek) állapotváltása **megszakítást** okoz.

Ezen bitek nem önállóan, hanem a PIC-eknél 8 bites regiszterekben szervezve jelennek meg. A bit szabványos elnevezését és a bitet tartalmazó regiszter nevét, felosztását az egyes adatlapok tartalmazzák. Az elnevezéseknek szabványosaknak kell lenniük, hiszen a C fordító csak ebben az esetben képes értelmezni a forráskódot.

5.2. Univerzális aszinkron adóvevő

A PC-k egyik legsokrétűbben felhasználható része a soros port, az Universal Asynchronous Receiver/Transmitter, vagy röviden **USART**. Ez az eszköz a soros és a párhuzamos interfészek között végez fordítást. Az USART-ot gyakran használják együtt más szabványokkal is (pl. RS-232). A port segítségével kétirányú kommunikáció valósítható meg a számítógép és különböző más eszközök (esetünkben a nyomtatott áramkörtől) között. A fentebb említett RS-232 szabvány szigorú szabályokat határoz meg a hibátlan adatátvitel célja érdekében. Ezek a soros portok 9 valamint 25 tűs SUB-D csatlakozóra vezetnek ki a különböző jeleket.

A csatlakozók szerkezetét az 5.1. ábra mutatja:



Név	Láb (25)	Láb (9)	Írány	Jelentés (a számítógép részére)
TxD	2	3	kimenet	Transmit Data (adatátvivő)
RxD	3	2	bemenet	Receive Data (adatifogadó)
RTS	4	7	kimenet	Request To Send (küldési kérelem)
CTS	5	8	bemenet	Clear To Send (kész a küldésre)
DTR	20	4	kimenet	Data Terminal Ready (adatterminál kész (jelen))
DSR	6	6	bemenet	Data Set Ready (adategység kész (jelen))
RI	22	9	bemenet	Ring Indicator (kicsengés jel?)
DCD	8	1	bemenet	Data Carrier Detect (adatvív? jelen)
GND	7	5	-	Singal Ground (jelföld)
-	1	-	-	Protective Ground (védőföld)

5.1. ábra. UART csatlakozók és láb kiosztásuk²

A kommunikáció folyamán mind a 8 vonalnak szerepe van (5.1. ábra). Az adatkommunikáció kezdetén minden "jelvezeték" alacsony szinten van. A kommunikáció kétféleképpen indulhat: vagy a PC kezdeményez vagy az eszköz. A PC az adási szándékát a DTR vonal alacsony szintre emelésével jelzi, amire az eszköz a DSR jelszint megemelésével válaszol. Abban az esetben, amikor a másik eszköz, amivel kommunikálni szeretne a PC, nem elérhető vagy nem üzemkész, a DSR jelszint alacsony állapotban marad. Ha a kommunikáció elindul, akkor a

²http://www.machinetoolhelp.com/_Images/rs232-DB25-DB9.jpg

két eszköz között (PC és a panel) az RTS és CTS vonalakkal szabályozhatja az adatáramlást. A CTS szint magasan tartásával az adateszköz, míg az RTS jel felemelésével a számítógép jelzi, hogy kész a másik oldal felől érkező adatok fogadására. A TxD és RxD vezeték feladata a két eszköz közötti tényleges adatátvitel biztosítása. Ha aszinkron adatátvitelről beszélünk, akkor a küldő oldalon sorosra alakított adatbájtot a vevő oldalon párhuzamosra alakítva dolgozzák fel. Az adatátvitel kezdetét egy kezdő úgynevezett START bit indítja, ekkor a jelszint a logikai 1-es értékét veszi fel, majd ezután következik a továbbítani kívánt 6, 7 vagy 8 adatbit amit követhet egy paritásbit (ezt a fogadott adatok ellenőrzésére használják) és végül a kommunikáció végét jelző záró vagy más néven STOP bit. A kommunikáció sebessége 120 és 112000 BAUD között változtatható.

5.3. I2C³

A hétköznapi elektronika, a távközlés és az ipari elektronika gyakran sok hasonlóságot mutat a látszólag független konstrukciókban. Minden rendszer rendelkezik intelligens vezérlővel, általános célú áramkörökkel, és/vagy alkalmazás orientált áramkörökkel. Az Inter IC technológia a Philips találmánya, jelentősége abban rejlik, hogy alacsony sebességű perifériákat csatlakoztathatunk alaplapokra, beágyazott rendszerekbe, vagy akár mobiltelefonokba. A buszt Inter IC-nek, vagy csak röviden I2C-busznak nevezik. Minden I2C-busszal kompatibilis eszköz tartalmaz egy on-chip interfészt, ami az I2C buszon lehetővé teszi a többi eszközzel való közvetlen kommunikációt. E tervezési koncepció számos illesztési problémát old meg a digitális vezérlésű áramkörök tervezésekor.

Az I2C-busz néhány jellemzője:

- egy soros adatvonal és egy soros órajel szükséges a működéséhez
- a csatlakoztatott eszközök programból címezhetők egy egyedi cím, és a köztük fennálló master-slave kapcsolat segítségével
- az adatvesztés elhárítására ütközésetektálás
- soros, 8bit-es, kétirányú adatforgalom, melynek sebessége normál üzemmódban 100kbit/s, gyors üzemmódban 400kbit/s
- a chipbe integrált szűrő az adatvonalon lévő zavarokat kiszűri, megőrizve ezzel az adatintegritást
- a buszra csatlakoztatható IC-k számát csak a busz kapacitása korlátozza, ami maximum 400pF lehet

Fizikai protokoll

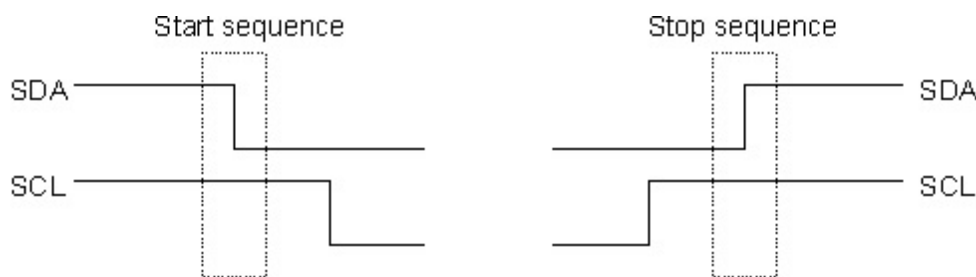
Az I2C busz támogat NMOS, CMOS és bipoláris IC gyártási technológiákat. Két vezeték a soros adat (SDA), és a soros órajel (SCL) továbbítja az információkat a buszra kötött eszközök között. Minden eszközt egy egyedi cím által ismernek fel, - vajon egy mikrokontroller, LCD driver, memória, vagy billentyűzet interfész-e - és működhet akár küldő, akár fogadóként is az

³A fejezet forrása [6]

eszköz funkciótól függően.

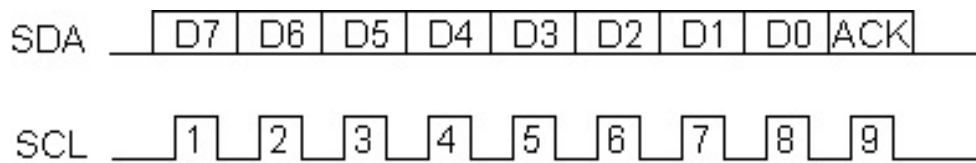
Amikor az eszközök adatátvitelt valósítanak meg, tekinthetjük őket MASTER-nek és SLAVE-nek. A MASTER az az eszköz, amely megkezdi az adatátvitelt a buszon, és generálja az órajeleket. Bármely ezalatt megcímezett eszközt SLAVE-nek tekintünk.

Ha a MASTER kommunikálni szeretne egy SLAVE-vel, egy **START** jelet küld az I2C buszon (5.2. ábra). A START jel az egyike a **két speciális jelzésnek** amit I2C buszra definiáltak. A másik speciális jel a **STOP**. Ha az adattovábbítás megkezdődik, az SDA-nak muszáj stabilnak maradnia, és nem változtathatja állapotát amíg az SCL magasan van. A START és STOP jelzések az adattovábbítás határolói.



5.2. ábra. I2C adat és órajelvonalak⁴

Az 5.3. ábra szemlélteti, hogy az adatok 8 bites sorozatokban kerülnek továbbküldésre. Az SCL vonal ezután magas majd alacsony szinteken váltakozik. Miután 8 bit átvitele megtörtént, az adatot fogadó eszköz visszaküld egy nyugta bitet, így 9 SCL órajel szükséges 8 bit adat átviteléhez. A fogadó eszköz egy alacsony ACK (nyugta) bit visszaküldésével jelzi az adat sikeres fogadását, és hogy készen áll a következő adatbajt fogadására. Ha magas ACK bitet küld vissza, azt jelzi, hogy nem tud több adatot fogadni és a master-nek le kell állítani a küldést egy STOP jelzéssel.



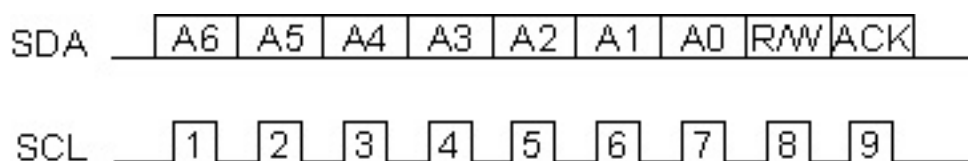
5.3. ábra. Adatbitek, nyugta, órajel⁵

⁴http://www.robot-electronics.co.uk/htm/using_the_i2c_bus.htm

⁵http://www.robot-electronics.co.uk/htm/using_the_i2c_bus.htm

Címzés

Minden **I2C cím**ről elmondható, hogy 7 vagy 10 bitesek. Láthatjuk, hogy mennyire hatékony ez a címzési rendszer, hiszen a 7 bit lehetőséget ad 128db eszköz egy buszra történő csatlakoztatására. Mivel a dolgozathoz felhasznált hőmérőt 7 biten címezzük, a továbbiakban erről a módszerről esik szó az alfejezetben. Az eszköz (TMP101) címzéséhez bár 7 bitre van szükségünk, 8 bitet küldünk (5.4. ábra), hiszen jeleznünk kell a SLAVE-nek, hogy olvasni vagy írni szeretnénk. Ha a plusz bit 0 a master írja a SLAVE-t, ha 1 a MASTER olvas a SLAVE-ből.



5.4. ábra. Címbytek, R/W bit, nyugta bit⁶

Az általam felhasznált TMP101 hőmérő esetén 5.5. ábra:

```

unsigned char WriteRegister(char oAddress, char oData) //Regiszter iras, mibe mi
{
    // regiszter      reg maszk
    char    i;

    IdleI2C();      // varas
    StartI2C();
    IdleI2C();      // varas
    WriteI2C(DEVICE_ADDR & 0xFE); //Irasra LSB=0-> ossze ES-eljuk
}
    
```

5.5. ábra. Eszköz címe és írása

A kódrészlet a 5.5. ábrán világosan elénk tárja, hogy az eszköz címe DEVICE_ADDR (a programban korábban definiált értéke 0x90) és mivel írásra címezzük, olyan maszkot választunk (0xFE), aminek az utolsó bitje 0.

Olvasás a szolgából

Mielőtt a SLAVE-ből **olvasni** szeretnénk előtte meg kell címeznünk az aktuális belső regisztert (írás). Egy START jelzés indítása az I2C eszköz címével, R/W bit alacsony állásával, és a belső regiszter számával ez meg is valósul. Egy következő START jelzéssel, újabb címzéssel,

⁶http://www.robot-electronics.co.uk/htm/using_the_i2c_bus.htm

és az R/W bit magas állásával olvashatjuk az adatokat, amit STOP jelzéssel állíthatunk meg (5.6. ábra).

```
// a konfig vege, innentől az olvasas
//FRAME3

RestartI2C();
IdleI2C();
WriteI2C(DEVICE_ADDR|0x01); // olvasásra LSB=1-> VAGY-oljuk, így biztos c
IdleI2C();
if (SSPCON2bits.ACKSTAT) {StopI2C();return FAIL;}
IdleI2C();
```

5.6. ábra. Olvasás a SLAVE-ből

5.4. Ethernet⁷

Mint ahogyan a 2.2. alfejezetben is megemlítettem, az egyes intézmények összekötése, hálózatba szervezése nélkülözhetetlen a gyors információcsere, az információ megosztása szempontjából. Nagyon sok szabvány és megoldás született a nagyvárosi hálózatok (MAN) vagy csak a helyi, kis méretű hálózatok pl.: LAN (Local Area Network) kialakítására.

A legnépszerűbb LAN az Ethernet. Az Ethernet története 1970-es évek elején kezdődött Hawaii szigeteken, ahol Norman Abramson kutató és a Hawaii Egyetemen dolgozó kollégái a távoli szigeteken dolgozó felhasználókat próbálták Honoluluban található fő számítógéphez kapcsolni. Az egyik megoldást a Csendes-óceán fenekére fektetett kábelek jelentették volna, azonban a kis hatósugarú rádiózást választották. Minden felhasználói terminált felszereltek egy kis rádióval, amely egy felfelé és egy lefelé irányú frekvencián üzemelt. Ha a felhasználó csatlakoztatni akarta a számítógépét, csak elküldött egy csomagot a felfelé irányuló csatornán, és ha senki más nem forgalmazott abban a pillanatban, akkor a csomag valószínűleg célbaért, és nyugtát küldtek a lefelé irányuló csatornán. Ütközés nem következhetett be, hiszen a lefelé irányuló csatornán egyetlen küldő volt (a központi számítógép). Ez az ALOHANET névre hallgató rendszer alacsony forgalmi terhelés mellett igen jól működött, de ha a felfelé irányuló forgalom megnőtt, bedugult. Ha versenyhelyzet alakult ki a felfelé irányuló csatornáért, és a terminál észrevette, hogy nem jött válasz, akkor újrapróbálkozott.

A Xerox PARC központjában a probléma továbbra is az volt, hogy a gépek el voltak szigetelve egymástól. Metcalfe és David Boggs 1976-ban megtervezték, és meg is valósították az első helyi hálózatot, amit **Ethernetnek** neveztek el.

Az átviteli közeg ebben a technológiában a koaxiális kábel volt, amely akár 2,5km hosszú is lehetett, a kábelre csatlakoztatott adóvevőkkel akár 256 gépet is lehetett csatlakoztatni a rendszerhez. Az ALOHANET-tel szemben egy nagy előnnyel bírt: a számítógépek adás előtt behallgattak a csatornába, hogy megbizonyosodjanak arról, éppen senki más nem ad. Ha egy számítógép foglaltak találta a csatornát, megvárta a folyamatban lévő adás végét. Ez a vizsgálat megelőzte, hogy az új adás keveredjen egy már folyamatban lévővel, nagyobb hatékonyságot biztosítva. Kérdés továbbá az, hogy mi történik, ha két vagy több számítógép várja az éppen folyamatban lévő adás végét, és utána mind egyszerre akar adni. A megoldás az, hogy a számítógépek hallgatják a csatornát adás közben is, és ha interferenciát fedeznek fel, akkor

⁷A fejezet forrása [7] és <http://www.ihd.hu/general/ethernet.html>

zavarni kezdik a csatornát, hogy figyelmeztessék a küldőket. Ezután véletlen hosszú ideig várnak, és csak utána próbálkoznak újra. Ha további ütközések történnek, akkor a véletlen időt mindegyik ütközés után megduplázzák, szétszórva ezzel a versenyhelyzetben lévő adásokat, így adva esélyt valamelyiknek az adásra.

Ethernet-kábelezés

Négyféle kábelezés terjedt el:

- A vastag Ethernet kábelezés volt az első, **10Base5**. A csatlakozás ún. vámpír csatlakozókon keresztül lehetséges, amelyekben egy apró tüskét nyomnak óvatosan a koaxiális kábel középső vezetékébe.
- A második kábelezési típus a **10Base2**, más néven a vékony Ethernet volt, amely sokkal könnyebben hajlítható ellentétben elődjével.
- A kábelezés hibáinak bemérésével kapcsolatos problémák olyan kábelezési elrendezés kialakításához vezettek, amelyben minden állomástól egy kábel vezet egy központi elosztóhoz (hub), ahol a kábelek között közvetlen csatlakozást hoznak létre. Az ilyen hálózatokhoz sodrott érpáros telefonvezetékeket használnak, mivel az irodaházak többsége ilyenekkel van felszerelve, és többnyire rengeteg szabad érpár áll még rendelkezésre. Ezt a sémát **10Base-T**-nek nevezik. Az adó-vevőbe kerül beszerelésre az az elektronika, ami a vivőjel, illetve az ütközések érzékelését hivatott elvégezni. Ütközés detektálása esetén az adó-vevők egy speciális jelet kezdenek el adni a kábelen, hogy az összes többi adó-vevő biztosan értesüljön az ütközés bekövetkeztéről.
- Az Ethernet negyedik kábelezési lehetősége a **10Base-F**, mely fényvezető szálakat használ. Ez a megoldás meglehetősen drága a csatlakozók és a lezáró elemek magas ára miatt, viszont kiváló a zajtűrése, és általában ezt a módszert használják az épületek, vagy az egymástól nagy távolságokban lévő elosztók összekötésére.

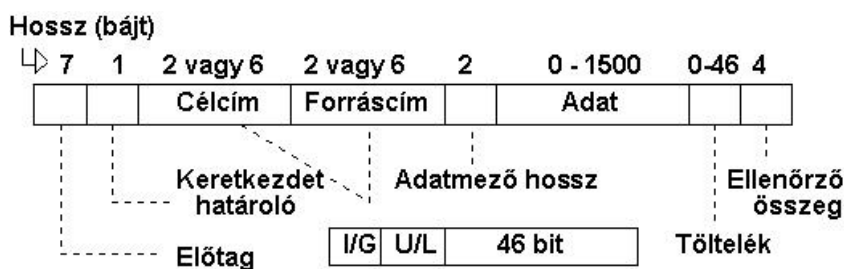
Az Ethernet MAC-protokollja

Az OSI réteg adatkapcsolati rétegének szintjén keretekről beszélünk. A fizikai réteg felől érkező bitfolyam csoportosul keretekké, majd a hálózati rétegben dekapzuláció révén juthat el

egyik csomóponttól a másikra.

A keret formátuma (5.7. ábra):

Hét bájtos *Előtaggal* kezdődik, ami lehetővé teszi, hogy a vevő az adóhoz igazítsa az óráját. A feleknek a keret hátralevő részében szinkronban kell maradniuk, amit úgy érnek el, hogy a Manchester kódolás segítségével nyomon követik a bithatárokat. A következő mező (nyolcadik bájt) a *Keret Kezdet Határoló*, majd ezután a hat-hat bájtos *Cél- és Forráscímek* mezői, amit két bájt *Hossz* mező követ. Az *Adatmező* 0-1500 bájt hosszú lehet. Ha a keret adatrésze 46 bájtnál rövidebb, akkor a *Kitöltés* (következő) mezőt kell használni a keret minimális méretének eléréséhez (64 bájt). Az utolsó Ethernetes keretmező az *Ellenőrző összeg*, ami tulajdonképpen az adatok 32 bites kódja.



5.7. ábra. Az Ethernet keret formátuma⁸

Gyors Ethernet

A 10Mb/s kezdetben óriási sebességnek számított, azonban az igények folyamatos növekedésével nagyobb sebességre volt szükség, "az adat mindig kitölti az átvitelhez rendelkezésre álló sávszélességet"⁹.

Ha több sávszélességre volt szükség, sokszor számos 10 Mb/s-os LAN-t kötöttek össze jelismétlők, hidak, routerek és átjárók sorozatával. Az IEEE 1992-ben indítványozta, hogy a 802.3 bizottság készítsen szabványt egy gyorsabb LAN-ra. Az új, 802.3u szabvány a meglévő 803.3 kiegészítése lett, és **gyors Ethernet** néven hivatkozunk rá. Megtartottak minden régi keretformátumot, interfészt és eljárási szabályt, de a bitidőt lecsökkentették 100ns-ről 10ns-ra. A

⁸<http://szabilinux.hu/konya/konyv/7fejezet/Image261.gif>

⁹Andrew S. Tannenbaum - Számítógép Hálózatok. Páner Könyvkiadó Kft, Budapest, 2004

10BaseT kábelezés előnyei annyira meggyőzőek voltak, hogy az új szabványt teljes egészében erre alapozták. További kérdéseket kellett megválaszolni ezután, például, hogy milyen kábeltípusokat támogassanak. Szóba jöhetett a 3-as kategóriájú sodrott érpár, 5-ös kategóriájú sodrott érpár, vagy a fényvezető szál. Végeredményben mindhárom kábeltípus támogatása mellett döntöttek, de a 3-as kategóriájú megoldást fel kellett javítani úgy, hogy elérje a szükséges szállítási kapacitást.

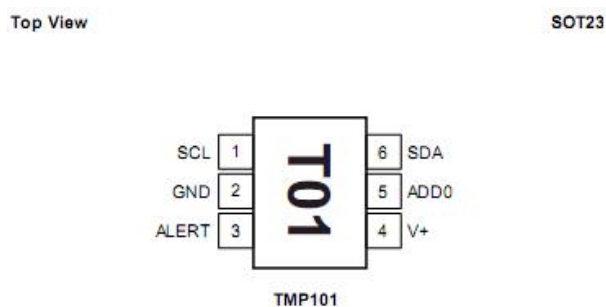
Az Ethernet tovább fejlődött és még most is fejlődik. Azóta már 100 Mb/s-os, 1000 Mb/s-os és még nagyobb sebességű hálózatok is létrejönnek. Az Ethernet már három évtizede létezik, és úgy tűnik sokáig fenn is marad, hiszen nem akadt komoly vetélytársa. Egyszerű, rugalmas, megbízható, olcsó és jól működik együtt a mára már egyeduralkodóvá vált TCP/IP-vel.

6. fejezet

Felhasznált IC-k¹

6.1. TMP101

Az időjárési tényezők egyike a **hőmérséklet**, ennek mérésére a TMP101 nevű digitális hőmérő szenzort alkalmaztam, felülnézetből a 6.1. ábrán látható.



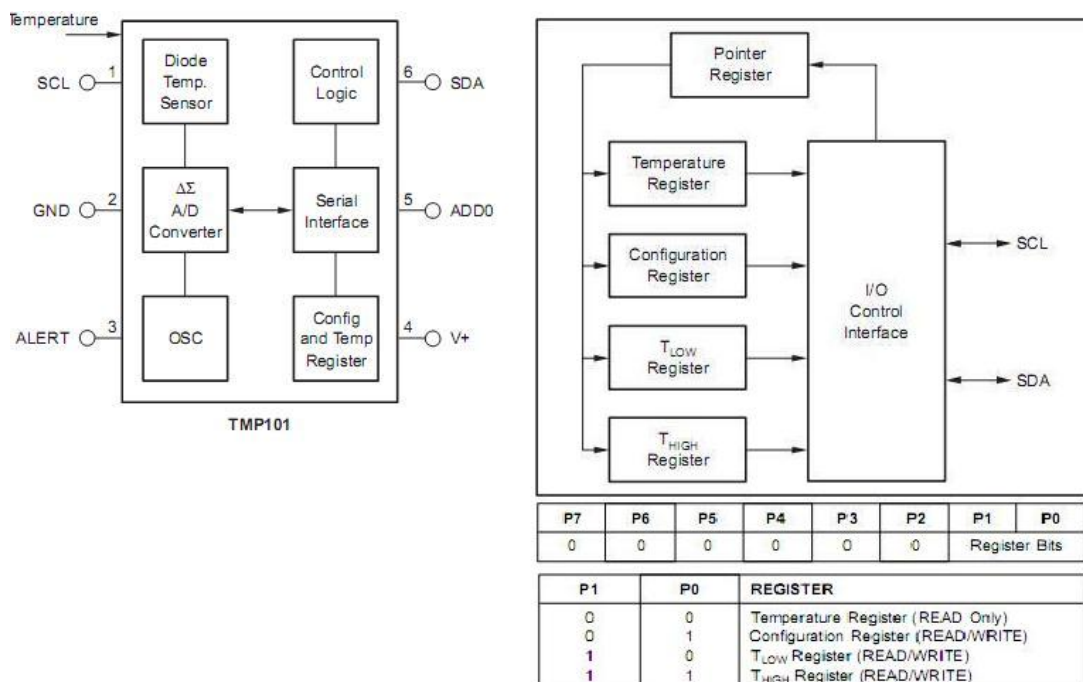
6.1. ábra. TMP101 felülnézet [10]

A hőmérő a kommunikációt SMBuson és I2C-n is lehetővé teszi, üzemeltetéséhez nincs szükség külső eszközre, csupán az SCL, SDA és ALERT vonalakon elhelyezett felhúzó ellenállásokra.

Mint a 6.2. ábra is mutatja, a szenzor áll egy a hőmérséklet mérésére szolgáló diódából, A/D átalakítóból, oscillátorból, logikai vezérlőből, soros interfészből, valamint konfigurációs és hőmérséklet regiszterekből.

Mint ahogyan az 5.3. alfejezetben tárgyaltam, a Master - Slave kapcsolat megköveteli a

¹A fejezet forrása [10] és [11]



6.2. ábra. TMP101 felépítése, regiszterei és regiszterek címei [10]

felek egymással való kommunikációját. Ez a kommunikáció lehet a Master részéről történő *írás* vagy a Slave-ből való *olvasás*. Mindig a Master kezdeményezi a kommunikációt, így a mi esetünkben a mikrokontroller feladata az eszköz, az egyes regiszterek címzése, adatok olvasása a regiszterekből.

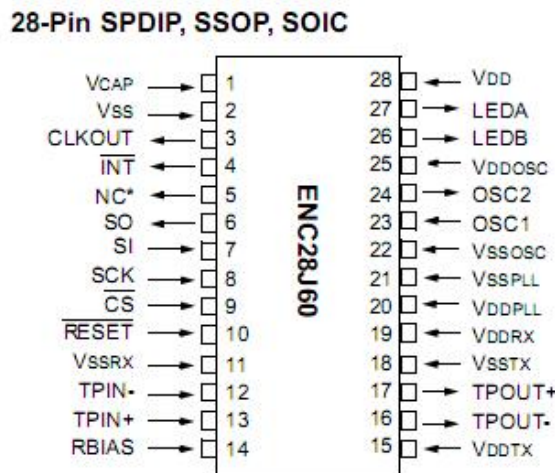
Az egyes regisztereket és felhasználási céljukat (olvasás/írás) a **Pointer Register** (6.2. ábra) írásával és a megfelelő maszk (LSB: 1 olvasás, 0 írás) alkalmazásával választhatjuk ki.

A **Temperature Register** egy 12 bites csak olvasható regiszter. A felhasználó a **Configuration Register**-ben tudja beállítani, hogy 9, 10, 11 vagy 12 bites felbontást szeretne. A Configuration Register egy 8 bites olvasható/írható regiszter, arra használjuk hogy a hőmérséklet szenzor működési módjait tároljuk benne. A *Shutdown Mode* beállításával (SD bit=1) lehetőségünk van minden áramköri kapcsolást leállítani, aminek köszönhetően energiát spórolunk. A *Thermostat Mode* bit segítségével állíthatjuk be, hogy az eszköz *Comparator* vagy *Interrupt* módban üzemeljen-e. A *Polarity* bittel a felhasználó beállíthatja az ALERT lábát, így POL=0 esetén a láb alacsonyan, míg POL=1 esetén az ALERT láb magas állapotban lesz aktív, és az ALERT láb állapota invertálva lesz. A *Converter Resolution Bitek* a belső Analog-to-Digital konverter beállítására szolgálnak. A magasabb hatékonyság vagy a gyorsabb konverziós idő között választhatunk ennek köszönhetően.

Lehetőségünk van még az eszköz olyan finombeállításaira, mint pl.: alvó állapotból egy mérés erejéig való felébredésre stb, de ez a feladat szempontjából nem volt létfontosságú, így csak a legfontosabbak kerültek tárgyalásra.

6.2. ENC28J60

Ezen IC egy egyedülálló Ethernet vezérlő SPI interfésszel, integrált MAC és 10Base-T fizikai réteggel, mely támogat Full Duplex és Half Duplex módokat, programozható automatikus újraküldést ütközés esetén és CRC generálást.



6.3. ábra. ENC28J60 lábkiosztása [11]

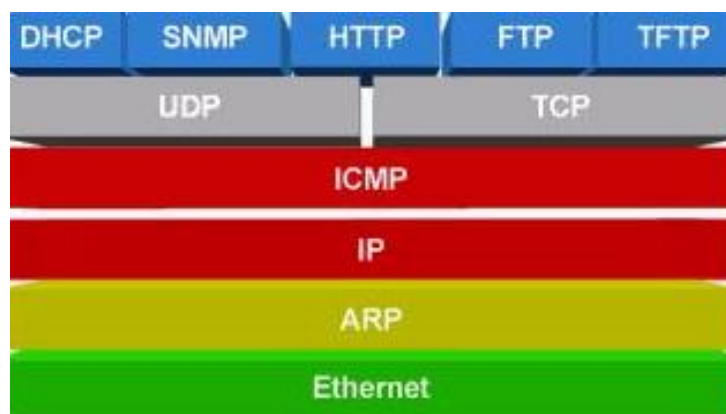
A kommunikáció a kontrollerrel egy megszakítás lábán és SPI-n keresztül történik akár 20 Mhz-es órajelfrekvencián.

A vezérlő hét fő funkcionális egységből áll:

- A mikrovezérlő és ENC28J60 közti kommunikációs csatornaként szolgáló SPI interfész
- Control regiszterek, amik az ENC28J60 vezérlésére és monitorozására szolgálnak
- Egy kettős portú RAM buffer a fogadott és átvitt adatcsomagokhoz
- Irányító a RAM bufferhez való hozzáféréshez
- BUS interfész, ami értelmezi az SPI interfészen fogadott adatokat és parancsokat
- Medium Access Control modult, ami implementálja az IEEE 802.3 MAC logikáját
- Physical Layer modult, ami kódolja és dekódolja az analóg adatokat

Az eszköz tartalmaz továbbá más egységeket mint, órajelgenerátor, on-chip feszültségszabályozó, szint fordítót hogy biztosítsa az 5V toleranciát és rendszer kontroll logikát.

A programozásban segítségemre a Microchip által kiadott **TCP/IP Stack for PIC18, PIC24, dsPIC, and PIC32 MCUs** szoftver [14] volt. Ez nem más, mint a beágyazott hálózati alkalmazásokhoz alapot nyújtó eszköz, mely kezeli a legtöbb elvárt kölcsönhatást a fizikai hálózati port és az alkalmazás között. Modulokat tartalmaz a gyakran használt alkalmazási rétegekhez, beleértve a weboldalak kiszolgálásához szükséges HTTP, e-mail küldéshez SMTP, Serial-to-Ethernet stb. Továbbá tartalmaz implementációkat a TCP és UDP transzport rétegekhez, valamint más támogató modulokat mint IP, ICMP, DHCP, ARP és DNS.



6.4. ábra. TCP/IP Stack rétegei [14]

A Configuration Wizard lépései során kiválaszthatunk applikációs, biztonsági protokollokat, beállíthatjuk eszközünk hálózati nevét, IP címét, MAC Address-ét, és egyéb paramétereit.

A létrejött általános célú program a Microchip által kiadott Demo Kit-ekre van optimalizálva, az általunk használt panel természetesen eltér az előre definiált konfigurációktól, így sajátot kellett létrehozni. Ez azt jelenti, hogy a rengeteg programösszetevő megértése után szükséges volt új hardverprofil létrehozása. A programösszetevők kifejtése nagyon hosszadalmas lenne, így csupán a **hardverprofil** emelném ki.

Nagyon sok előre konfigurált lapkával működik együtt a program, azonban a mi esetünkben ki kellett bővíteni ahhoz, hogy az aktuális konfigurációhoz is illeszkedjen. A mikrokontroller digitális portjainak adatrányának beállítása, portok kapcsolódása, ENC28J60 ethernet illesztő lábainak beállítása került megírásra (6.5. ábra).

A programrészek módosítása miatt (nem használt funkciók eltávolítása) az egymásra épülő

Mikrokontrolleren alapuló időjárás előrejelző hálózat

programrészek együttműködésének helyreállítása, szolgáltatások (ICMP,DHCP_CLIENT) engedélyezése is beállításra kerültek.

```

#define oledD_C_TRIS                (TRISGbits.TRISG4)

#elif defined(__18F46J50)

#define LED0_TRIS                   (TRISDbits.TRISD0)
#define LED0_IO                     (LATDbits.LATD0)
#define LED1_TRIS                   (TRISDbits.TRISD1)
#define LED1_IO                     (LATDbits.LATD1)
#define LED2_TRIS                   (TRISDbits.TRISD2)
#define LED2_IO                     (LATDbits.LATD2)
#define LED3_TRIS                   (TRISDbits.TRISD3)
#define LED3_IO                     (LATDbits.LATD3)
#define LED4_TRIS                   (TRISDbits.TRISD4)
#define LED4_IO                     (LATDbits.LATD4)
#define LED5_TRIS                   (TRISDbits.TRISD5)
#define LED5_IO                     (LATDbits.LATD5)
#define LED6_TRIS                   (TRISDbits.TRISD6)
#define LED6_IO                     (LATDbits.LATD6)
#define LED7_TRIS                   (TRISDbits.TRISD7)
#define LED7_IO                     (LATDbits.LATD7)
#define LED_GET()                  (LATD)
#define LED_PUT(a)                 (LATD = (a))

#define BUTTON0_TRIS                (TRISAbits.TRISA5)
#define BUTTON0_IO                  (PORTAbits.RA5)
#define BUTTON1_TRIS                (TRISEbits.TRISE0)
#define BUTTON1_IO                  (PORTEbits.RE0)
#define BUTTON2_TRIS                (PRODL) // No Button2 on this board
#define BUTTON2_IO                  (lu)
#define BUTTON3_TRIS                (PRODL) // No Button3 on this board
#define BUTTON3_IO                  (lu)

#define oledD_C_TRIS                (TRISGbits.TRISG4)

```

6.5. ábra. Hardverprofil létrehozása

7. fejezet

Eredmények

Két nyomtatott áramkörrel dolgoztunk, az egyik a Microchip PICKIT3 nevű Demo Kit-je, a másik pedig Daróczi Balázs által tervezett és kivitelezett nyomtatott áramkörtől.

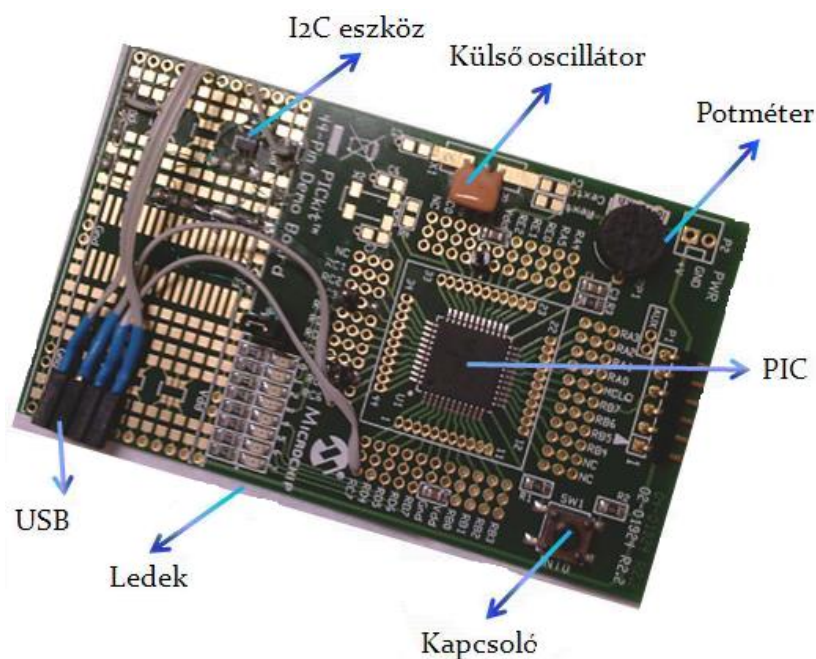
Az alap Demo Kit lehetőséget adott a PIC18-as családjának megismerésére 18F45K20 mikokontrolleren keresztül, az első programok is ezen controllerre készültek, de mivel nem tartalmazott sem szenzort, sem csatlakozót amire a feladat kivitelezéséhez szükség lett volna, betekintést nyertünk az elektronikai tervezés világába is. A lapkára forrasztottunk egy 10MHz-s órajelgenerátort, TMP101-es hőmérséklet szenzort, ellenállásokat, valamint csatlakozót az USART-on való kommunikációhoz(7.1. ábra).

USART kapcsolat kialakítása a 7.2. ábrán látható lapkával lehetséges, számítógéphez Mini USB porton csatlakozik. A számítógépre szükséges USB - Serial ¹ portot telepítenünk, ahhoz hogy a rendszer felismerje eszközünket.

Ezekkel a módosításokkal lehetőségünk nyílt a hőmérővel mért adatok átvitelére számítógépre későbbi felhasználás céljából. Az adatokat egyszerűen egy terminálablakban meg tudjuk jeleníteni, ahogy a kép is mutatja. A terminal kezdetben nem jelenítette meg a szenzor által mért adatokat, ha azokat nem egész értéként tároltuk a mikrokontroller memóriájában, holott az MPLAB Integrated Development Environment ² fejlesztői környezet szimulátorában ez nem jelentett problémát. Megfigyeléseink szerint az egész típusú értékek átvitele számítógépre sikeres volt, de lebegőpontos értékek átvitele sikertelen (7.3. ábra). A megoldást a lebegőpontos típusú értékek szétválasztása jelentette oly módon, hogy két külön változóban került tárolásra az

¹www.ftdichip.com

²http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en019469&part=SW007002



7.1. ábra. Módosított PICKIT3 Demo board



7.2. ábra. USART modul

érték egész része és tört része, majd ezeket egyszerűen "."-tal elválasztva megkapjuk az eredeti értéket 7.4. ábra.

A logikai analizátor képernyőjén (7.5. ábra) jól láthatók az 5.3. alfejezetben ismertetett SDA és SCL vonalak, és ezek viszonya. A felső vonal jelzi az órajelek periódusait, míg az alsó vonal az adatfolyamot. Az analizátor segítségével láthattuk valós időben mindazt, amit a forráskódban kódok formájában írtunk le, elősegítve a hibák felfedezését.

Az új panelre (7.6. ábra, 7.7. ábra) olyan IC-k kerültek, mint:

- SHT15 hőmérséklet és pára szenzor
- MPH6115A nyomás szenzor
- TCS3404 fény szenzor

³Ultrascope for DS1000D Series: <http://www.rigolnpi.com/ds1000d.html?skipIntro=true>


```
        //Delay10KTCYx(2*ReadSW()+6);
        //putsUSART("Hello");
    }
    d1=T; //d1 és d2 int típusúak, T float típusú
    d2=(T-d1)*1000;
    sprintf(pszText, "Homerseklet: %d.%0.3d \r\n", d1, d2);

//d1=T;
//f2=T-d1;
//d2=trunc(f2*10000);
//f3=f2*10000-d2;
//d3=trunc(f3*10000);

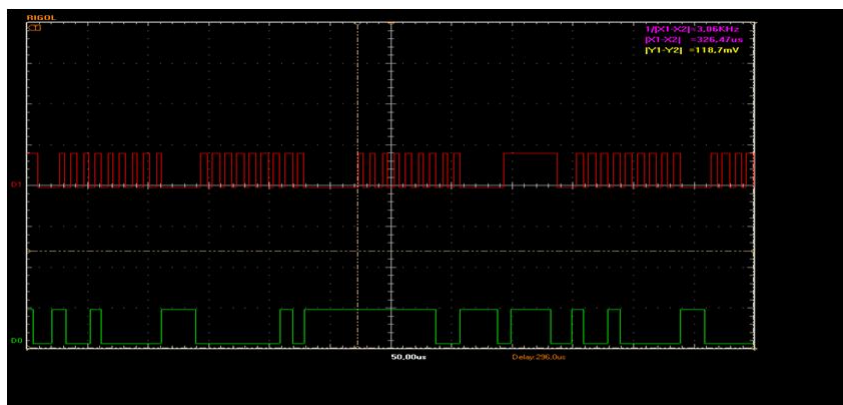
//sprintf (pszText, "Homerseklet: %d.%04d%04d\n", d1, d2, d3);

//    sprintf(pszText, "Homerseklet: %g ", T);
    putsUSART(pszText);
}
}
```

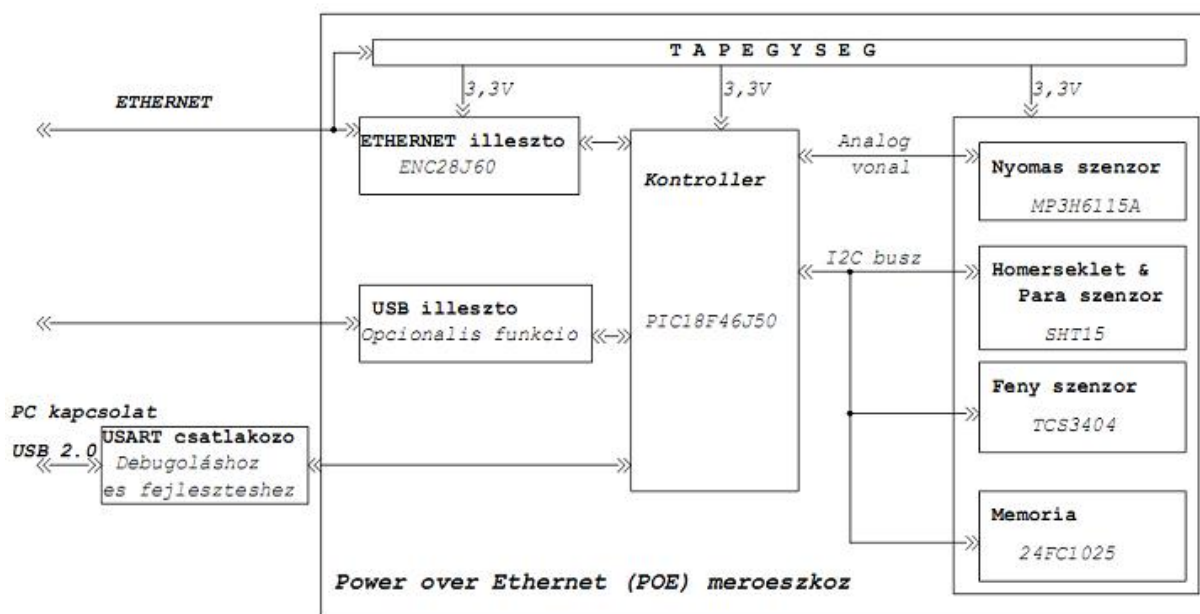
7.4. ábra. Egész-lebegőpontos probléma megoldása

hoztam létre a HardwareProfil.c állományban, azonban a többi komponensben is szükséges definiálni az új beállításokat. A fordító hibaüzeneteinek orvoslása után a program lefordult, de a fizikai réteg feletti rétegek elégtelen működése miatt a hálózati kommunikáció nem valósult meg. A számítógép és a mérőállomás között a fizikai link kiépült, a hálózati kártya LED-je világít az eszköz csatlakoztatásakor, de a Fizikai Réteg feletti rétegek felé a kerettovábbítás egyelőre sikertelen volt.

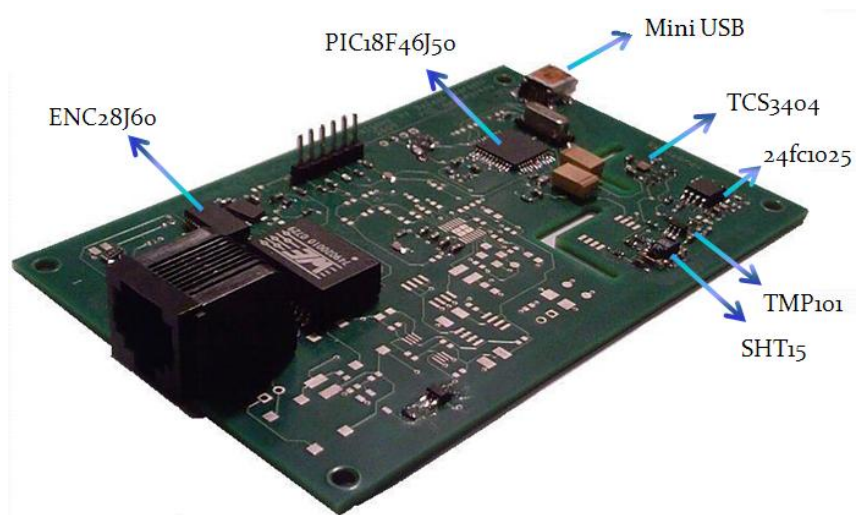
Végeredményben egy olyan általános eszközt sikerült létrehozni, amely a rajta lévő szenzorok segítségével képes lehet az időjárási paraméterek mérésére, a mért adatokat pedig továbbküldheti számítógép felé Ethernet vagy USART kommunikáció révén.



7.5. ábra. Logikai analizátor kimenet³



7.6. ábra. PIC18F46J50 mikrokontrollerrel vezérelt áramköri lap blokk diagramja⁴



7.7. ábra. PIC18F46J50 mikrokontrollerrel vezérelt áramköri lap

8. fejezet

Összefoglalás

Szakedolgozatunkban egy igen hétköznapi tünő, azonban annál sokkal jelentősebb tudománnyal, a meteorológiai észlelések, előrejelzések egy új módjával foglalkoztunk, ahol a részeredmények és a cél eléréséhez meg kellett ismerkednünk több igen sokrétű tudományággal ahhoz, hogy végül ezek összekapcsolódhassanak.

A "légtüneménytan" története nem csupán érdekesség, hanem szerves része a mindennapi életünknek, fejlődése napjainkban is tart. Társadalmunk jelzője "információs társadalom", mely legnagyobb értéknek az információt tekinti. A számítástechnika fejlődése hatással volt a meteorológiára is, pontosabb mérésekre, távolabbi előrejelzésekre, gyorsabb információcserére nyílt lehetőség.

Az időjárás tanulmányozása során betekintést nyertem az időjárás ismeretek fogalomvilágába, megismertem az egyes légmozgások kapcsolatát a domborzati tényezőkkel és ennek következményeit, illetve az erre támaszkodó matematikai egyenleteket.

Ahhoz, hogy ezen ismeretek alkalmazása lehetővé váljon, az egyes időjárás paraméterek mérésére volt szükség. A technikai háttérül szolgáló mikrokontrollerek, mérő szenzorok, kommunikációs szabványok megértése, alkalmazásának technikája jelentette a legnagyobb kihívást.

A rengeteg paraméter mérése közül a hőmérséklet mérése valósult meg, de technikai szinten nem sokban tér el az egyes szenzorok kezelése, így nem a többi mérő szenzor programozása élvezte a második szakaszban a legnagyobb prioritást, hanem az Interneten való kommunikáció megvalósítása. Ehhez rendelkezésemre állt a beágyazott hálózati alkalmazásokhoz alapot nyújtó eszköz, így kiindulási alapként fel tudtam használni az általa kínált általános program-

rendszert, és némi módosítás után a Fizikai kapcsolat kiépült a hálózati kártya és a mérőállomás között.

A programok írása a MPLAB Integrated Development Environment fejlesztői környezetben történt¹.

A szakdolgozat LaTeX-ben² készült, melynek elsajátításához az Irodalomjegyzék 9. bejegyzésben szereplő forrás adott segítséget.

A szakdolgozatban elért eredmények jó alapot adnak a későbbi fejlesztéshez, a készítése közben megszerzett új ismeretek pedig szakmai előrehaladásomban segítettek.

¹Használati útmutatója az Irodalomjegyzék 8. bejegyzésében található

²<http://www.texniccenter.org/>

9. fejezet

Köszönetnyilvánítás

Köszönettel tartozom Dr. Tóth Lászlónak, Nagy Gábornak és a National Instruments Hungary-nak a dolgozatom elkészítéséhez nyújtott segítségükért, útmutató tanácsaikért, ötleteikért és az egyetemi évek alatt eltöltött órákért. Szeretném kiemelni Pásztor Gábort, hiszen ezen szakdolgozat nem csak önálló, hanem csapatmunkában elért eredményeket is tartalmaz, amelyeket a későbbiekben folytatni szeretnénk.

Végül, de nem utolsó sorban köszönetet mondok Daróczi Balázsnak az általa nyomtatott áramkörért és Lehotai Balázsnak az Ethernet vezérlő programozása során adott segítségéért.

Irodalomjegyzék

- [1] **Lakotár Katalin - Meteorológiai előrejelzés** <http://ttmk.nyme.hu/fldi/Documents/LakotárKatalin/Éghajlattan/Meteorologiaielőrejelzés.pdf> (Letöltve: 2010. április 1.)
- [2] **Dr. Péczely György - Éghajlattan. Nemzeti Tankönyvkiadó, Budapest, 2002**
- [3] **Kónya László-Kopják József - PIC Mikrovezérlők alkalmazástechnikája, harmadik bővített kiadás. ChipCAD Elektronikai Disztribúció Kft. Budapest,2003**
- [4] **Microchip - PIC18F46J50 Family Data Sheet. Microchip Technology Inc., 2009**
- [5] **Kónya László-Kopják József - PIC Mikrovezérlők alkalmazástechnikája, második bővített, átdolgozott kiadás. ChipCAD Elektronikai Disztribúció Kft. Budapest,2003**
- [6] **Philips Semiconductors - Az I2C-busz használata (Specifikációkkal), 1995** http://www.muszeroldal.hu/measurenotes/i2c_hu.pdf (Letöltve: 2009. október 02)
- [7] **AndrewS.Tannenbaum - Számítógép Hálózatok. Pánem Könyvkiadó Kft, Budapest, 2004**
- [8] **Microchip - MPLAB IDE User's Guide with MPLAB Editor and MPLAB SIM Simulator, 2009** http://ww1.microchip.com/downloads/en/DeviceDoc/MPLAB_User_Guide_51519c.pdf (Letöltve: 2009. szeptember 10)
- [9] <http://komal.elte.hu/forum/forum.cgi?a=tk>
- [10] **TMP100 TMP101 - Digital Temperature Sensor with I2C Interface, Microchip Technology Inc.,2002** <http://www.datasheetcatalog.org/datasheet/texasinstruments/tmp101.pdf> (Letöltve: 2009. szeptember 10)

- [11] **Microchip - ENC28J60 Data Sheet Stand-Alone Ethernet Controller with SPI Interface, Microchip Technology Inc, 2008** <http://ww1.microchip.com/downloads/en/DeviceDoc/39662c.pdf> (Letöltve: 2010. február 15)
- [12] **Makra László - A léghőmozgásjelenségei, a Poisson-egyenlet** <http://www.sci.u-szeged.hu/eghajlattan/pdf/4-5%20A%20legkor%20mozgasjelensegei.pdf> (Letöltve: 2010. április 03)
- [13] **Dr. Lakotár Katalin - Függőleges Mozgások a léghőmozgásban** <http://ttmk.nyme.hu/fldi/Documents/LakotárKatalin/Léghőmozgás/Függőlegesmozgásokaléghőmozgásban.pdf> (Letöltve: 2010. április 03)
- [14] **Nilesh Rajbharti - The Microchip TCP/IP Stack, Microchip Technology Inc.** (Letöltve: 2010. február 18)

10. fejezet

Mellékletek

Mellékletként az elkészült programokat CD-n csatolom tömörített állományban.