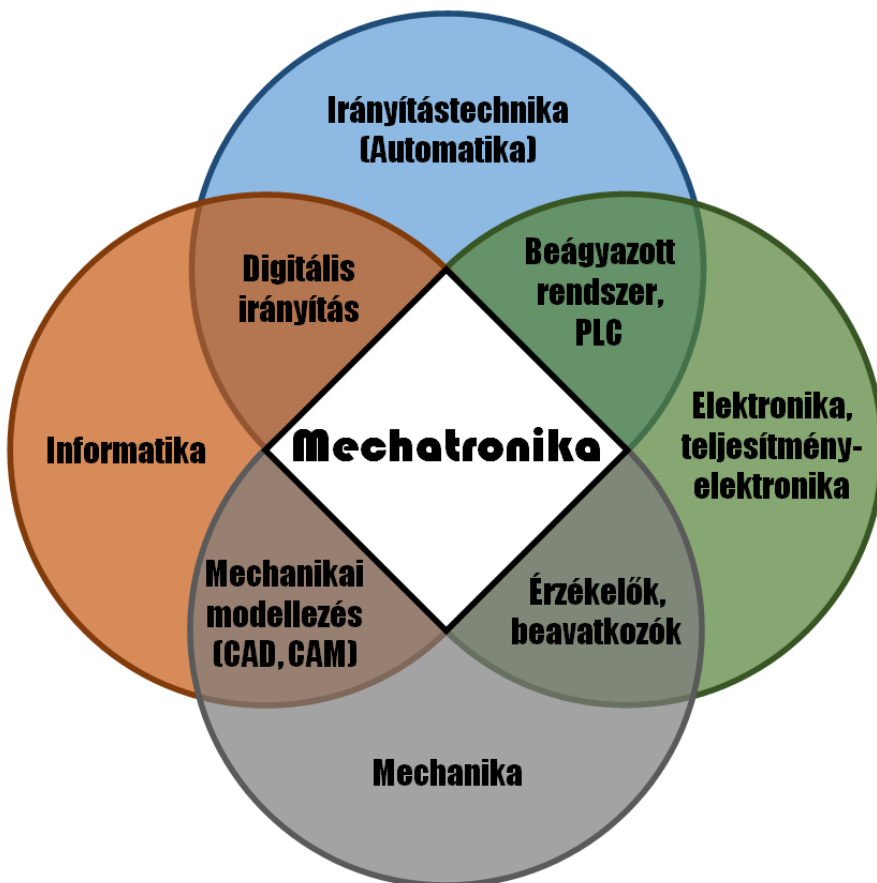


Kővári Attila – Katona József

Mechatronika alapjai

***Mechatronika a gyakorlatban
Arduino és Raspberry Pi 2 alapokon***



Mechatronika alapjai

*Mechatronika a gyakorlatban
Arduino és Raspberry Pi 2 alapokon*

@Kővári Attila – Katona József

Lektorálta: Dr. Odry Péter

Publio Kiadó
Felelős kiadó: Publio Kiadó Kft.
2015
Minden jog fenntartva!
További impresszum adatok...

Előszó

A digitális technika, a számítástechnika összességében az információs technológia fejlődése az összetett irányítóberendezések, az automatizált folyamatok vezérlése és szabályozása terén is új lehetőségeket teremtett meg. Manapság már az ipari gyártófolyamatok és általánosságban az egyes berendezések automatizálása elképzelhetetlen digitális irányítóberendezések alkalmazása nélkül. A piaci versenyhelyzet által diktált költséghatékony működés, magas minőségi követelmények a folyamatok optimalizálását követelik meg. Ennek megvalósításához egyre komplexebb irányítási algoritmusok, termelésirányítás és optimalizálás, folyamatfelügyelet, diagnosztika alkalmazása mára már elengedhetetlen. A gyártó, megmunkáló, mozgató, energiatermelő mechanikai mozgást végző gépek automatizálása terén az igazi fejlődést a digitális számítógépek, a villamos mérés-, vezérlés- és szabályozástechnika, villamos gépek és más villamos működtetésű energia átalakítók alkalmazása hozta meg. A **mechatronika (mechanika és elektronika)** szó is erre utal, vagyis egy multidiszciplináris területet jelöl, mely elsősorban a gépész, villamos és automatikai területhez kapcsolódik, azok alkalmazása a gépészeti berendezések automatizálására, akár intelligenssé tételére. Manapság mechatronikán ennél még többet értünk, mivel a gépek, berendezések egyre komplexebbé válnak, ezért mára a mechatronika már az előbb felsorolt szakterületeken kívül az információfeldolgozás, alkalmazott informatika, méréstechnika, villamos energetika, híradástechnika, távközlés szakterületek együttes alkalmazását jelenti. A mechatronika nem egy tantárgy, tudomány vagy technológia, hanem inkább egy filozófia, szemléletmód mely az előzőkben meghatározott szakterületek egymást segítő integrációjaként definiálható. Mechatronikai megoldások esetén az egyes szakemberek nem kizárólag a saját szakterületükhöz kapcsolódó problémák, hanem egy több szakterületet integráló csapat tagjaként, egy közös cél optimális megoldásán kell dolgozzanak. Ezért egy mechatronikai mérnöknek azonosulnia kell a komplex rendszerben történő gondolkodás filozófiájával, és mindennel, amely e filozófia mögött van. Nem egyes részegységek, hanem a teljes rendszer optimális megoldására kell törekedjen ésszerű kompromisszumok által.

E könyv a mechatronika alapjait gyakorlati példákon keresztül megismertető sorozat első kötete. A könyvsorozat elsősorban a mechatronika villamos és informatikai ismereteit felölelő, információ feldolgozó, érzékelő és beavatkozó egységeivel foglalkozik. A könyv gyakorlati feladatok felhasználásával segítséget kíván nyújtani a szakemberek számára a mechatronikai rendszerek alapjainak megismerésében. Azok számára ajánlott, akik érdeklődnek e terület iránt, és akik nem érik be az elméleti alapok megismerésével, hanem szeretnének a mechatronika területén gyakorlati ismereteket is szerezni, de nincs lehetőségük azt egy komplex mechatronikai laborban elsajátítani. A kötetekben található

példák olyan feladatokat, mechatronikai rendszereket ölelnek fel, amelyek olcsón, bárki számára beszerezhető eszközök felhasználásával otthoni körülmények között is összeállíthatók, tesztelhetők. A sorozat a mechatronikai rendszerek tárgyalása során előzetes ismeretekre támaszkodik elsősorban a középiskolai fizika elektromosság, mágnesesség, testek mozgása, dinamikája, merev testek statikája területén.

Bízunk benne, hogy azok, akik végigolvassák a könyvet, és összeállítják, tesztelik a gyakorlati példákat, azok alkalmazás szintjén megismerik a mechatronikai rendszerekben rejlő lehetőségeket, elsajátítják a mechatronikai rendszerek tervezésének, kialakításának alapjait, és a tárgyalta ismeretek és gyakorlatias példák pedig jó alapot adnak az egyes mechatronikai feladatok megoldásához.

Úgy gondoljuk, hogy ez a könyv nagymértékben hozzájárulhat ahhoz, hogy különböző, a témában kevésbé jártas szakemberek vagy akár középiskolai diákok is gyakorlatorientált példák segítségével megismerhessék a mechatronika alapjait, ezzel alapszintű gyakorlati tudásra is szert tehessenek.

Jelen kötet a mechatronika alapjait mutatja be, valamint a következő kötetekben a gyakorlati feladatok megvalósítása során felhasznált információ feldolgozó egységeket az Arduino fejlesztőlapot és Raspberry Pi 2 miniszámítógépet.

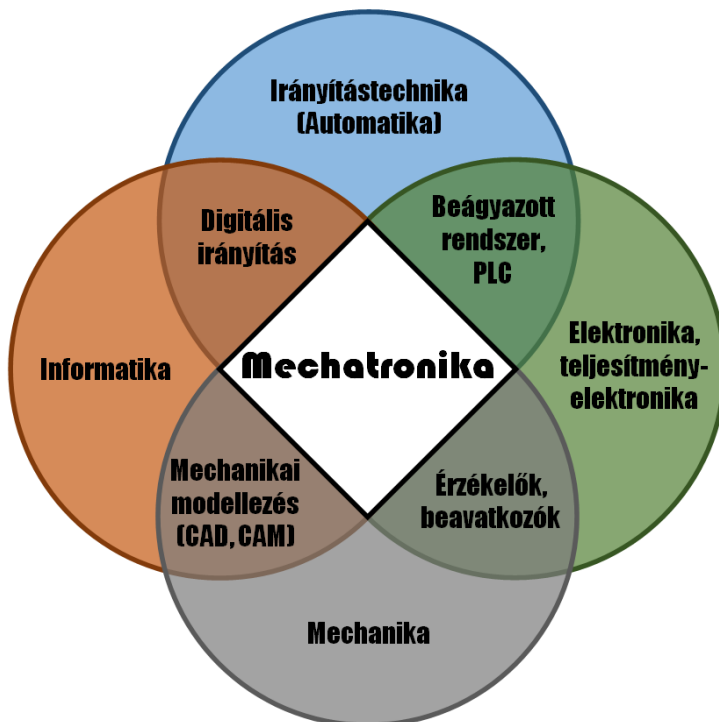
Köszönettel tartozunk a kézirat lektorának, Dr. Odry Péter szakmai tanácsaiért és módszertani javaslataiért.

A Szerzők

Mechatronika

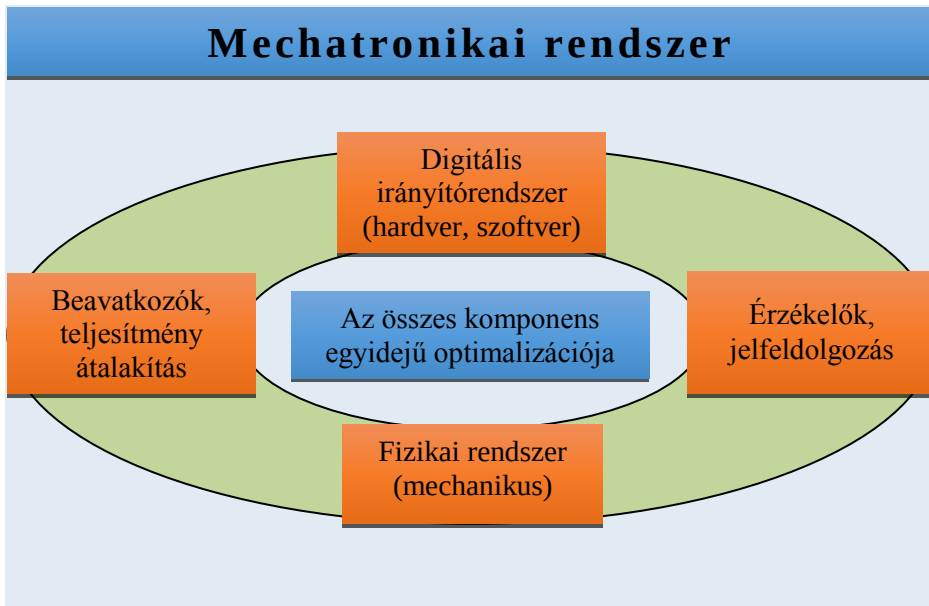
A mechatronika szót a Yaskawa Electric Corporation fiatal mérnöke Tetsuro Mori alkotta meg 1969-ben. A cég gépipari berendezésekhez gyártott elektronikus irányítóberendezéseket. A mechatronika kialakulása a mikrovezérlők, majd a digitális számítógépek megjelenésével az 1970-es, 1980-as évekre tehető, de létrejött egy folyamatos fejlődés eredménye, mely arra irányul, hogy egyre intelligensebb, olcsóbb berendezéseket hozzunk létre. Ennek elérése érdekében egyértelművé vált, hogy a gépészeti berendezések intelligensé tétele az elektronika, digitális technika és a számítástechnika segítségével valósítható meg, ezért azt nem lehet elhatároltan külön kezelni.

A mechatronika egy definíciója: a gépészet, az elektrotechnika/elektronika és az informatika *egymást segítő* integrációja termelőrendszerek és termékek előállítására, tervezésére és működtetésére. A definícióban fontos jelentőséget kap az egymást segítő kifejezés, mely az egyes szakterületek ismeretanyagának együttes alkalmazásával, összekapcsolásával a termék előállítás, tervezés és működtetés folyamatát optimalizálja, mellyel a terméket versenyképesebbé teszi. A mechatronika területe tehát az 1. ábra szerinti szakterületek integrációját jelenti.



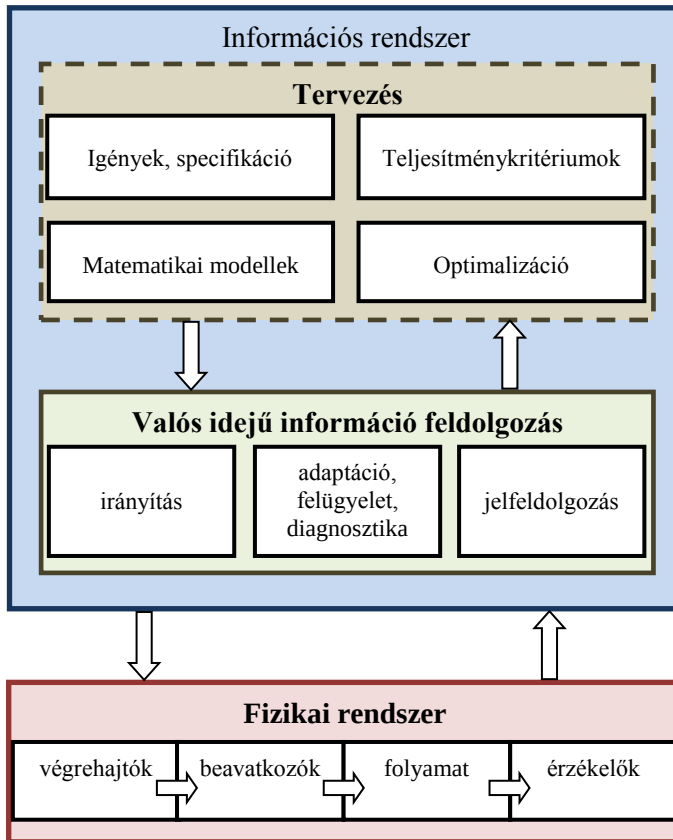
1. ábra: *Mechatronika mint több szakterület integrációja*

A mechatronika nem egy tudomány vagy technológia, hanem sokkal inkább egy komplex szemléletmód, filozófia a gépészeti berendezések számítógépes automatizálására. A mechatronikai rendszer egybefoglalóan tárgyalja a fizikai rendszerek, az érzékelők, mérőátalakítók, végrehajtók, beavatkozók és az elektronikai, valamint a legtöbb esetben beágyazott digitális irányítórendszer tervezésének, kialakításának, optimalizálásának kérdéseit.



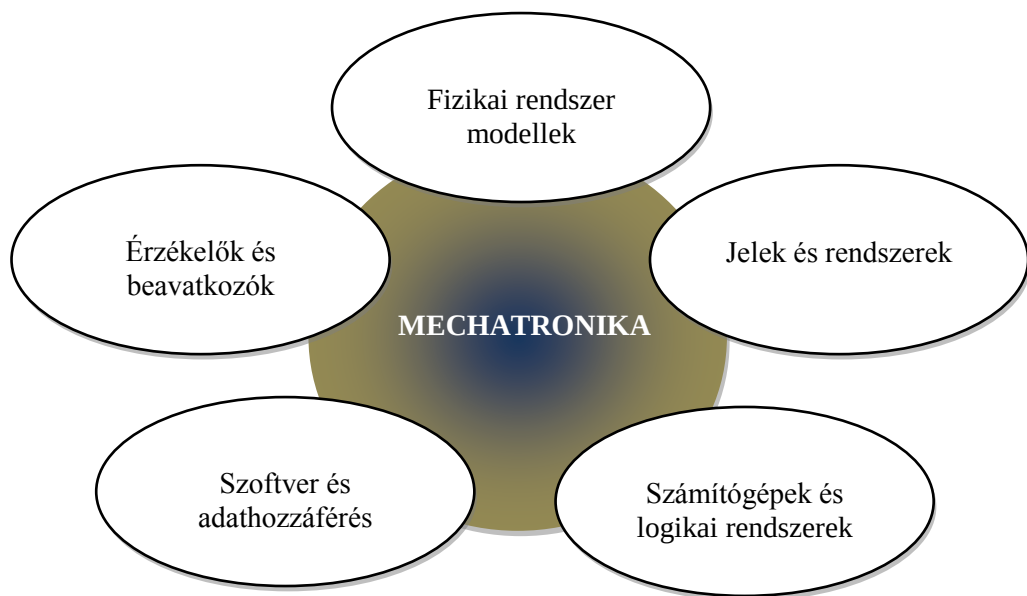
2. ábra: *Mechatronikai rendszerek*

A mechatronikai rendszerek tervezése során a teljes rendszerre kiterjedő optimális megoldásra kell törekedni, meg kell határozni, hogy mely feladatokat érdemes mechanikus rendszer segítségével, melyeket elektronikai berendezésekkel, digitális irányítóegységgel, vagy akár több irányítási szintet felölelő folyamatirányító rendszer segítségével megoldani az egész rendszer optimális kialakítását figyelembe véve. Így nem csak egyes hardver komponensek, hanem egy teljes információ feldolgozó rendszer integrálását jelenti.



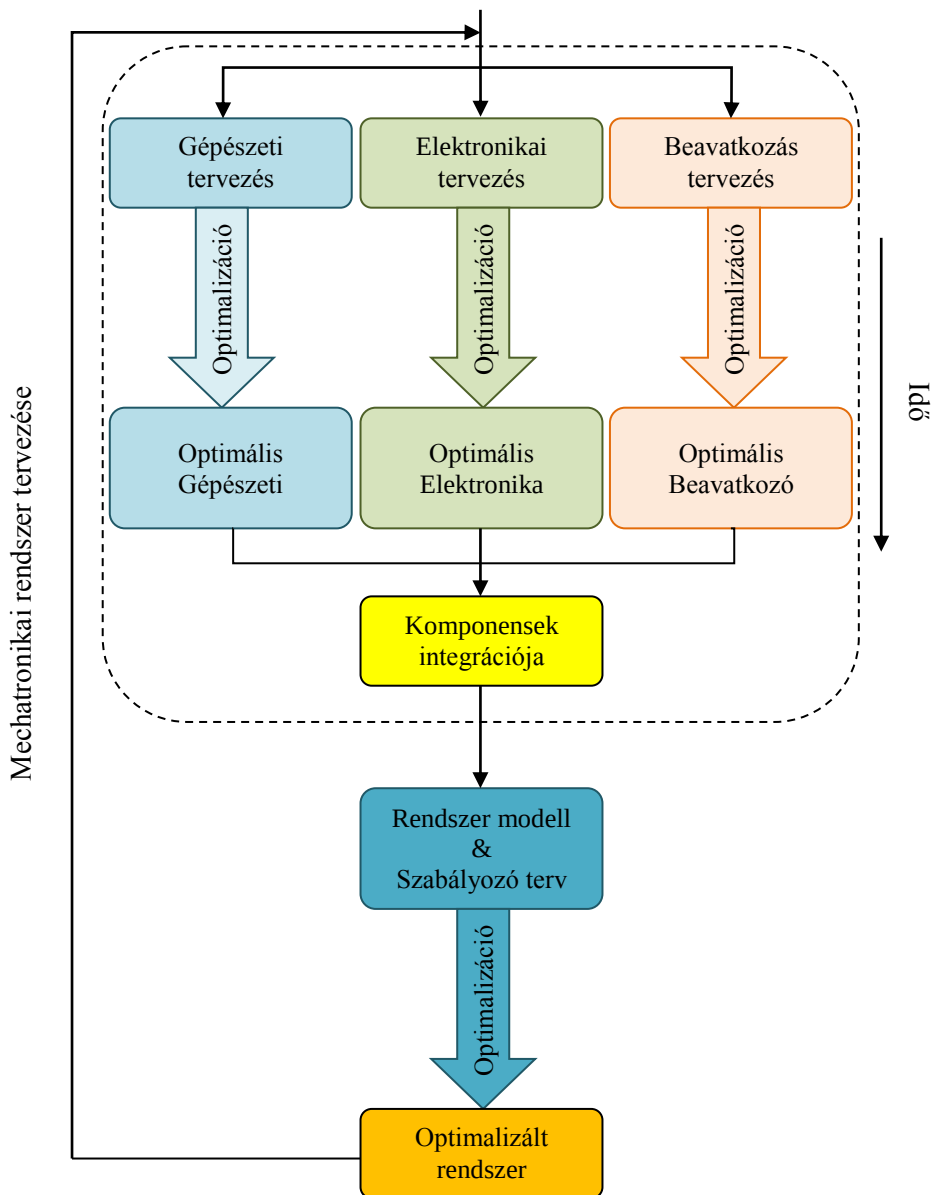
3. ábra: Főbb komponensek és információ feldolgozás integrációja

A tervezés első fázisában tehát a főbb rendszerkomponensek megadása, azok működésének leírása, domináns paramétereinek, közelítő modelljének, valamint az egyes rendszerkomponense közötti interfészek, kommunikációs felületeinek meghatározása történik. A teljes rendszer optimális kialakítására, működésére felállított komplex rendszermodell egyes komponenseinek tervezése, kiválasztása, kialakítása már az előírások szerinti funkciók és interfészek megvalósítását tárgyalja. A mechatronikai rendszerek legfontosabb építőköveit az alábbi ábra mutatja.



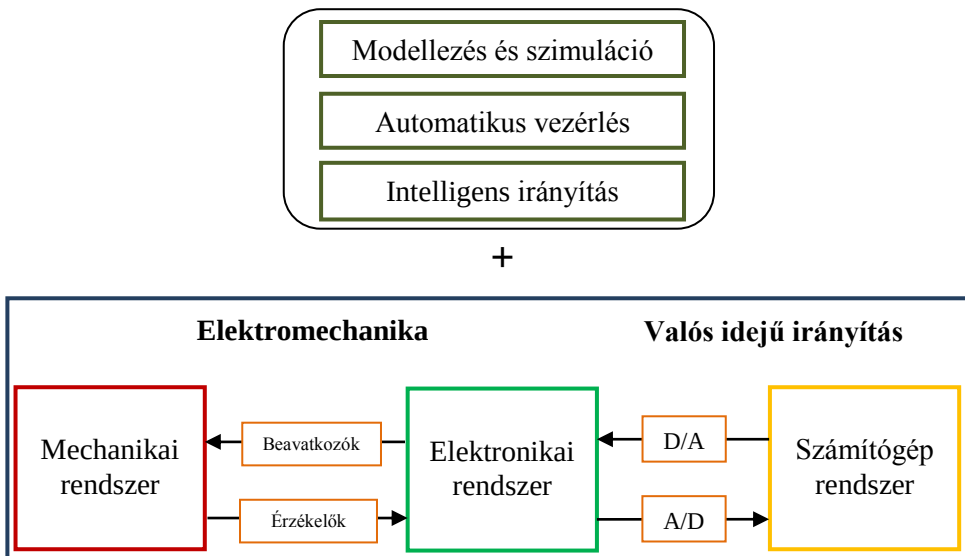
4. ábra: *Mechatronikai rendszerek a legfontosabb építőkövei*

A teljes mechatronikai rendszer optimalizációja azonban nem csak a tervezés első fázisában történik meg, hanem a tervezés teljes folyamatát végigkíséri. Az egyes meghatározó rendszerkomponensek bizonyos részeinek kidolgozása alapján a rendszermodell folyamatosan pontosítható, mely a többi részkomponens tervezésénél figyelembe vehető. Lényegében a teljes rendszer tervezése az egyes rendszerkomponensek egyidejű párhuzamos optimalizációja mentén valósul meg.



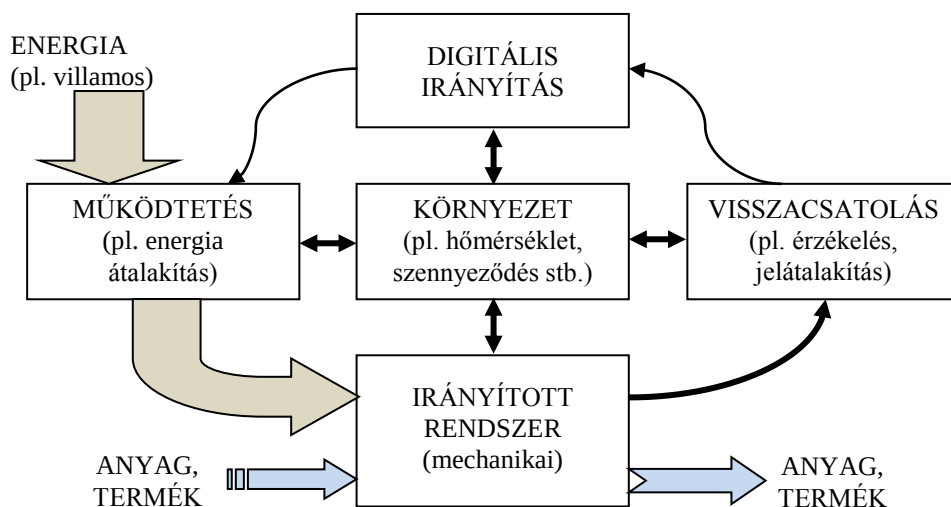
5. ábra: *Mechatronikai rendszerek tervezésének folyamata*

A mechatronikai rendszerekben a gépeszeti egységek elsősorban a mozgás létrehozásában, míg a villamos egységek az automatizálás, vagy akár annak még magasabb szintje az intelligens irányítást valósítja meg. A mechatronikai fő rendszerkomponensnek kapcsolatát mutatja a következő ábra.

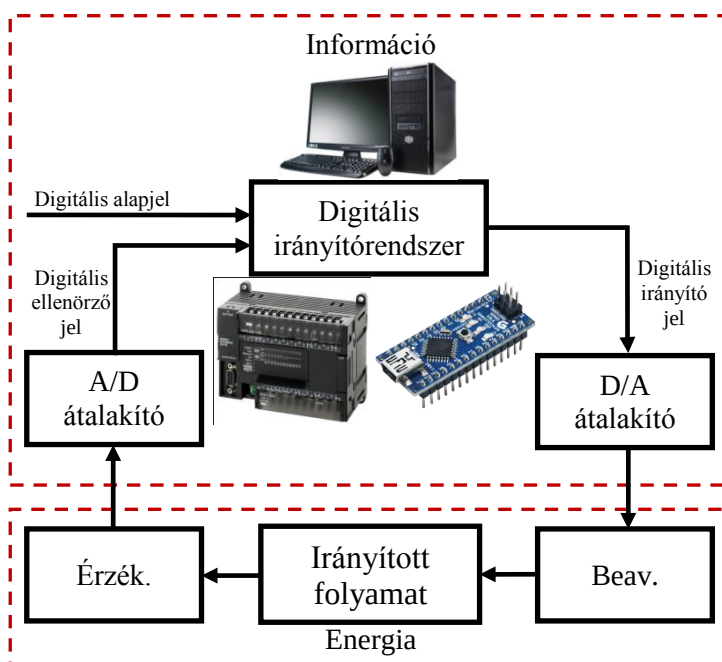


6. ábra: *Mechatronikai rendszerkomponensek*

Ennek megfelelően a mechatronikai rendszerben az irányított berendezés és annak modellje a gépészeti egységek és a velük kapcsolatban álló a makro (esetleg mikro) világ együttesét tartalmazza, mely a teljesítmény termék előállítási folyamatában történő felhasználását valósítja meg. Így a beavatkozók (és esetleg a végrehajtók), a tervezés folyamatában, sok esetben a gépészeti egységek részét képezik, míg az elektronikai egységek, melyek sok esetben digitális irányítórendszerek az információ feldolgozásáért felelősek. A makro világ, a gépészeti és teljesítmény átalakítást végző egységek analóg jelei (sebesség, gyorsulás, erő, forgatónyomaték, nyomás, áramlás stb.) által hordozott információk és a digitális irányítórendszer digitális jelekkel végzett információfeldolgozási eljárásai közötti interfészt, információátadást analóg-digitális, és digitális-analóg átalakítók segítségével lehet megvalósítani.



7. ábra: Digitális irányítóberendezés és az érzékelőket, beavatkozókat és a folyamatot tartalmazó analóg működésű egységek kapcsolata



8. ábra: Termék előállítás nagyteljesítményű irányított és a kisteljesítményű irányítóberendezésének kapcsolata

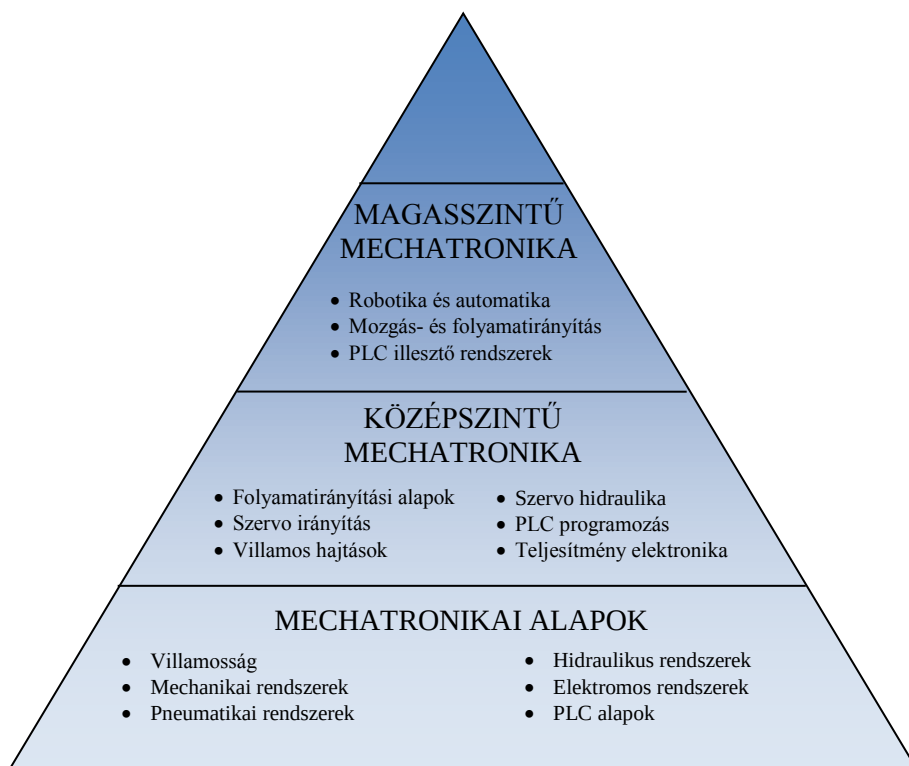
Az előzőekben tárgyalt működés alapján megfogalmazható a mechatronika egy másik definíciója is, mely szerint a mechatronika a gépészet, az elektrotechnika és az informatika tudományterületeire épülő interdiszciplináris terület. A definíció kiegészül még a rendszerek általános definíciójának – a jelet fogad, feldolgoz, előállít – mechatronikára vonatkozó definíciójával, melyben egy tipikus mechatronikai rendszer jeleket fogad, dolgoz fel, majd a jeleket erőkké és mozgásokká alakítva bocsátja ki. A mechatronikai irányítási feladat az általános irányítástechnika alapkérdésével foglalkozik, vagyis milyen módon kell a mechanikai rendszerbe beavatkozni, annak bemenetét módosítani ahhoz, hogy a kimenet, mely elsősorban erő, mozgás előre meghatározott, előírásoknak megfelelő legyen. Az irányítóberendezés megtervezéséhez, kialakításához ismerni kell az általános rendszertechnikai megközelítéseket, leírási módokat, mely arra keresi a választ, hogy egy rendszer bemeneti jelének ismerete esetén hogyan lehet meghatározni annak kimeneti jelét.

A komplex gépészeti berendezések általánosságban az irányítóberendezéseknél sokkal költségesebbek. Ezért a mechatronikai rendszerek tervezése során az egyszerűbb felépítésű gépészeti egységek és a komplexebb, a fizikai rendszer hiányosságait is kompenzáló irányítóberendezések alkalmazására törekednek. De ez nem jelentheti a gépészeti egységek gondatlan megtervezését, mert egy rosszul megtervezett rendszer megfelelő irányítástechnikai tulajdonságait általában nem lehet egy intelligens irányítóberendezéssel sem kompenzálni.

Összességében egy mechatronikai rendszer tervezése és kialakítása során az egyes szakterületek szakembereinek egymást segítő módon kell az optimális, költséghatékony, megbízható megoldás kialakításában együttműködnie. Ehhez elengedhetetlen az egyes szakemberek más szakterületeket érintő kérdéseinek, problémafelvetéseinek, megoldási módszereinek részbeni ismerete, hogy a közösen kialakított mechatronikai rendszer minden szempontot figyelembe vevő optimalizálására törekedjenek.

Felismerve a mechatronikai terület előzőekben tárgyalt előnyeit, a mechatronikai képzések az elmúlt időszakban nemzetközi szinten is nagymértékben növekedtek. Természetesen az ipar és technológia szereplői, mint például a repülőgépipar, autóipar, szerszámgépipar, általánosságban a gyártóipar, egészségügy területén a mechatronikai mérnökök mind egyre keresettebbek és mára már szerves részét képezik a mérnöki gárdának.

A mechatronikai rendszerek, és az azokkal kapcsolatos ismeretek, szaktudás elsősorban három fő szintre bontható: mechatronikai alapok, közép és haladó, magas mechatronikai szint. Az alapszint elsősorban az egyes rendszerkomponensek elméleti alapjainak ismeretét tartalmazza, a középszint, az irányítástechnikai, irányítóberendezés programozási, szervó technika, villamos hajtások témaköreit öleli fel. A haladó szint már az előzőek alkalmazása komplex mechatronikai rendszerek, mint például robotikai, mozgásszabályozó, folyamatszabályozó rendszerek, tervezésében, kialakításában.



9. ábra: *Mechatronikai rendszerek és ismeretek szintjei*

Mechatronika kezdetei, oktatása nemzetközi szinten

A következőkben a mechatronikai filozófia elterjedését megalapozó kezdeti nemzetközileg is jelentős lépések kerülnek röviden bemutatásra. Mára már elmondható, hogy szinte minden országban jelentős területet képvisel a mechatronikai szemléletmód oktatása és alkalmazása az ipari fejlesztések terén.

Az első kifejezetten mechatronika területével kapcsolatosan szervezett konferenciát 1989-ben Japánban, Tokió városában rendezte meg a Japán Gépészmérnöki Egyesület (Japanese Society of Mechanical Engineers - JSME). Akkoriban az egyetemek csak néhány a mechatronika területét érintő speciális tárgyat oktattak, sokkal kevesebbet, mint azt az ipar megkívánta volna a fényképezőgépek, videó felvevők, másológépek, autók, robotok gyártása területén.

Európában, néhány fejlett országban, Japánhoz hasonlóan a magas szintű technológiát képviselő ipar egyre nyitottabbá vált mechatronikával kapcsolatos szakterületek integrációjának megvalósítására a termékfejlesztésben.

A Svájci Műszaki Főiskola (Eidgenössische Technische Hochschule Zürich – ETH) már 1984-ben megalapított egy mechatronikai területtel foglalkozó kutatócsoportot. A Mechatronikai Csoport (ETH Mechatronics Group) a gépészmérnöki és villamosmérnöki tanszékek oktatóiból áll.

Németországban, szintén 1989-ben a Német Mérnökök Egyesülete (Verein Deutscher Ingenieure - VDI) szervezett egy mechatronika területét érintő konferenciát Mozgásirányítás - Mechatronika a Gép- és Járműiparban ("Kontrollierte Bewegungen – Mechatronik im Maschinen- und Fahrzeugbau") címmel. Ebben az időszakban már Németországban is számos egyetemi kutatási projekt indult a területen.

Angliában 1990 szeptemberében került megrendezésre a Mechatronika – Intelligens Gépek Tervezése című nemzetközi konferencia, melyet a nemzetközi szinten több mint százezer tagot számláló, Londoni székhelyű Gépészmérnökök Intézete (Institution of Mechanical Engineers – IMechE) szervezett. A mechatronika területének intenzív támogatója a Tudományos és Műszaki Kutatási Tanács (Science and Engineering Research Council – SERC) is.

Hollandiában 1992-ben alapították meg a Mechatronika Platformot az ipar és az egyetemek képviselőivel, hogy terjesszék a mechatronikai szemléletmódot mind a kutatás mind pedig az oktatás területén.

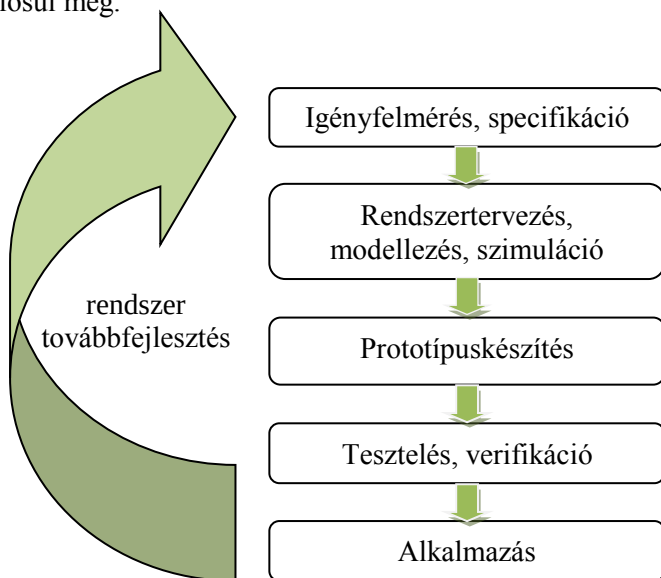
Ausztriában a Linzi Egyetem emelhető ki a mechatronika területének oktatása tekintetében.

Mechatronika oktatása Magyarországon

Magyarországon a mechatronikai mérnöki alapképzési szak a régi rendszerben (a kétciklusú rendszer előtti) nem létezett. A mechatronikai alapképzések 2005-től indultak, a mesterképzéseket pedig 2007-től. A mechatronikai képzést a Budapesti Műszaki és Gazdaságtudományi Egyetem Gépészmérnöki Karán a 2007-ben a Mechatronika, Optika és Műszertechnika Tanszékből és a Gépészeti Informatikai Tanszékből megalakult Mechatronika, Optika és Gépészeti Informatika Tanszék gondozza. A korábbi Budapesti Műszaki Főiskolán, 2009-től Óbudai Egyetemen szintén 2007-ben alapították meg a Mechatronikai és Autótechnikai Intézetet, mely a mechatronika képzést gondozza. A Pannon Egyetemen szintén 2005-től indítanak mechatronikai mérnök képzést, melyet a Mérnöki Kar Fizika és Mechatronika Intézetéhez tartozik. A Miskolci Egyetem Gépészmérnöki és Informatikai Kara a Robert Bosch Mechatronika Tanszéket 2004-ben alapította meg, melynek egyik legfontosabb feladata a mechatronikai mérnökképzésben való részvétel. Alapképzési szakon 2007-ben kezdődött meg a képzés, mesterszak 2011-től indul. Ezeken felül még a Debreceni Egyetem, Nyugat-magyarországi Egyetem, Széchenyi István Egyetem, Szent István Egyetem, Szegedi Tudományegyetem és az Edutus Főiskola is folytat mechatronikai mérnök képzést.

Mechatronikai rendszerek fejlesztése és alkalmazása

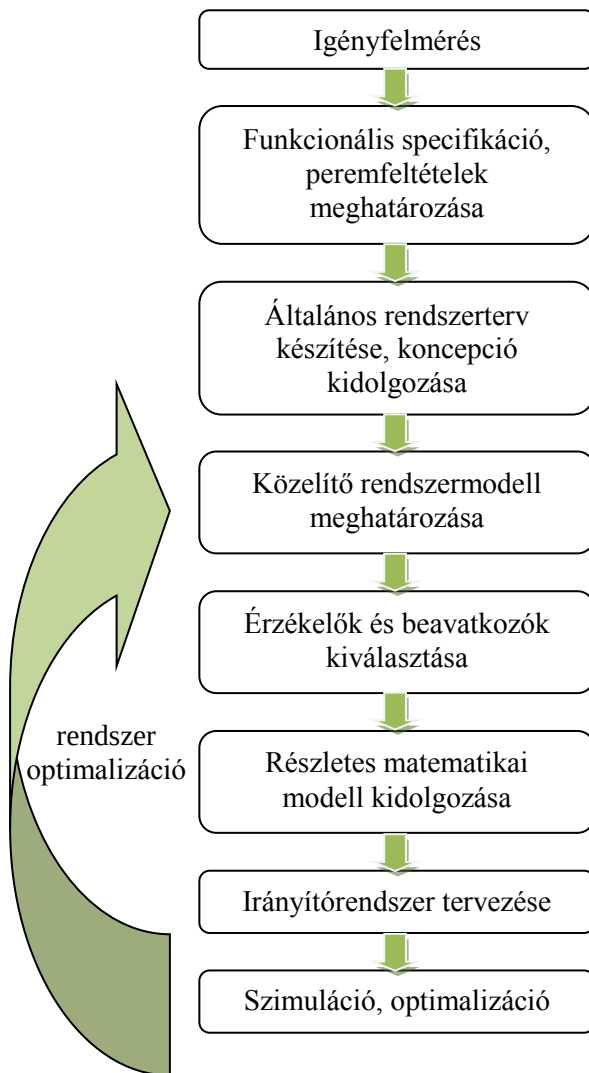
A mechatronikai rendszerek fejlesztésének folyamata az alábbi ábrán látható, mely az igényfelméréssel, specifikációval kezdődik és a rendszertervezés alapja. A rendszerterv modellezés és szimuláció segítségével kerül optimalizálásra és az így kialakított terv szerint készül el a prototípus, mely azonnal éles vagy sok esetben szimulált környezetben tesztelhető és az esetleges módosítások, finomhangolások után a végleges rendszer elkészíthető. A rendszer alkalmazása, életciklusa során számos, a tesztelések alatt nem vizsgált és javításra szoruló tulajdonság derülhet ki, melyek megoldása a rendszer továbbfejlesztése során valósul meg.



10. ábra: *Mechatronikai rendszerek fejlesztésének folyamata*

A rendszertervezés a modellezés és szimuláció segítségével történő elkészítése és optimalizálása az általános rendszerterv, rendszer kialakítására vonatkozó koncepció kialakításával kezdődik. Ez alapján lehet meghatározni a komplett rendszer egy közelítő matematikai modelljét, mely segítséget nyújt az alkalmazandó érzékelők és beavatkozók kiválasztásába. A rendszer analóg, nagyteljesítményű részére részletes matematikai modell dolgozható ki, mely az irányítóberendezés tervezésének alapját adja. Szimuláció segítségével vizsgálható a rendszer viselkedése és az eredmények alapján a nagyteljesítményű és az irányítórendszer iterációs lépésekben optimalizálható. Természetesen több fő komponensből álló rendszer esetén az egyes fő

komponensek fejlesztése párhuzamosan történik, majd a komplett rendszer optimalizálása ezután valósul meg.



11. ábra: Rendszertervezés folyamata

Az iteratív optimalizáció után elkészíthető az irányítórendszer prototípusa. A prototípust sok esetben igen költséges lenne valós körülmények között tesztelni, és bizonyos tesztesetek előállítása igen komplikált és veszélyes is lehet, ezért a prototípus első tesztjeit úgynevezett HIL szimuláció (Hardware-in-the-loop simulation) segítségével végzik el. A HIL teszt során a folyamatot egy számítógép helyettesíti, melyen valós idejű alkalmazás a folyamat, az érzékelők

és a beavatkozók matematikai modellje segítségével emulálja az érzékelők és beavatkozók villamos jeleit, mely az elektromos interfész az irányítórendszer és az irányított folyamat között. A HIL tesztek során kapott eredmények felhasználásával a prototípus tovább optimalizálható és az adott feltételek mellett legoptimálisabb megoldás elérése után a végleges rendszer megvalósítható.

Magában a rendszertervezés folyamatában mind a mechanikai és irányító rendszer modellezése, szimulációja terén számos tervező szoftver alkalmazható (például Solidworks, LabVIEW, MATLAB).

Mechatronikai rendszerek alkalmazásai

Egy mechatronikai rendszer általában az alábbi alap hardver komponenseket tartalmazza:

- mechanikai rendszer, a közvetlenül munkát végző egységgel;
- működtető rendszer a teljesítmény átalakító és meghajtó egységekkel;
- számítógépes vezérlőegység.

Egy mechatronikai rendszerre jó a példa a merevlemez és annak elektromechanikus fejmozgató egysége, mely gyors működésűnek, precíz beállításúnak, és megbízhatónak kell lennie. A mechatronika lényegében a mechanikai, elektromos és digitális információ feldolgozó részt egy funkcionális modulba integrálja.

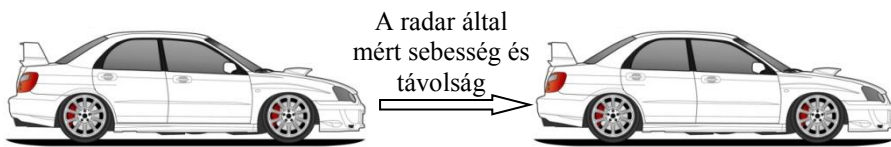
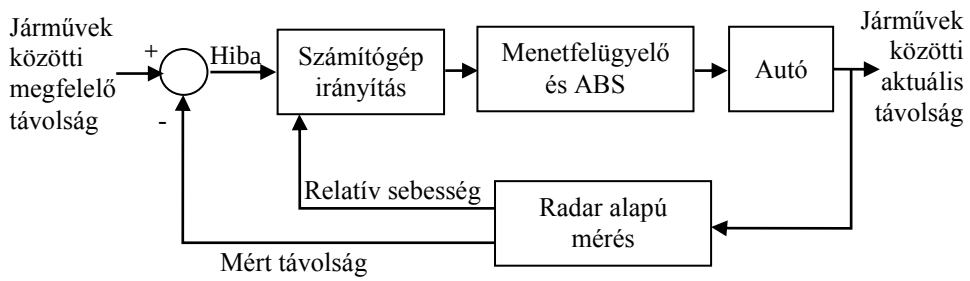
A mechatronikai rendszerben a digitális irányítóegység a mozgás önműködő, automatikus irányításán felül az alábbi funkciókat is biztosítja:

- összehangolt működés a többi külső kapcsolódó folyamattal;
- kezelővel való ember-gép kapcsolat (HMI), kézi üzemmód;
- információs adatkapcsolatok biztosítása, adatok fogadása továbbítása.

A mechatronikai szemléletmód szerint kialakított rendszerek az alábbi főbb előnyöket nyújtják:

- alacsony ár;
- összetett, precíz mozgások végrehajtása;
- egyszerűbb felépítésű mechanikai mozgó egységek;
- magas integráció, egységes, szabványos interfészek;
- nagy megbízhatóság és zajvédelem.

Mechatronikai rendszerre egy másik jó példa a gépkocsi távolságtartó-követő rendszere, mely a gépkocsi mozgását az előtte haladó jármű alapján határozza meg. A rendszer blokkvázlata a következő ábrán látható.



12. ábra: Gépkocsi távolságtartó-követő rendszer

Arduino fejlesztőplatform mint beágyazott rendszer

Az alábbiakban az Arduino fejlesztőplatform kerül bemutatásra, mely a gyakorlati példákban elsősorban az irányítórendszer feladatát ellátó beágyazott rendszer, programozható vezérlő funkciót valósítja meg. Az Arduino egy olyan mikrovezérlő alapú, nyílt forrású elektronikai fejlesztőplatform, melyet elsősorban hobbielektronikai célokra szántak. Az Arduino projekt weboldalán a www.arduino.cc –n igen sok információ megtalálható. A mikrovezérlők programozása korábban viszonylag igen sok speciális ismeretet kívánt, de az Arduino fejlesztői a tervezésekor a minél egyszerűbb, és ez által felhasználók számára könnyebben elsajátítható rendszer kialakítási és programozási platform létrehozására helyezték a hangsúlyt. Mindemellett a platform nyílt forrású, amely lehetővé tette, hogy az Arduino alapú fejlesztőeszközök minél olcsóbbak lehessenek, melynek köszönhetően a platform nagymértékben elterjedté vált. Az Arduino platformú fejlesztőrendszerek segítségével alapszintű programozói és elektronikai ismeretek birtokában is készíthetünk különböző egyszerű vezérlést vagy összetettebb szabályozást, esetleg más kommunikációs vagy egyszerűbb adatfeldolgozást igénylő projekteket bárki számára elérhető, olcsó hardver modulok segítségével.

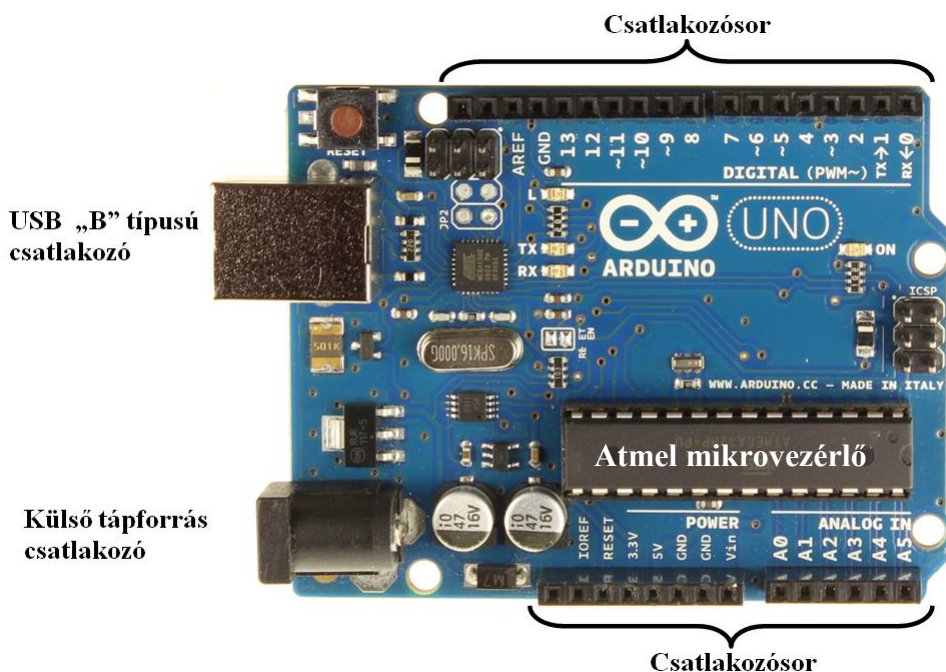
Az Arduino nem egyetlen eszköz, hanem egy eszközcsalád számos kiegészítővel. Az Arduino fejlesztőplatform egy integrált fejlesztői környezetet (IDE – Integrated Development Environment) és fejlesztő valamint ehhez illeszthető kiegészítő elektronikai modulokat tartalmaz. A fejlesztőmodulok elsősorban az Atmel cég által kifejlesztett, módosított Harvard-architektúrájú RISC típusú AVR mikrovezérlő család köré épülő áramköri lapokból állnak, de már elérhető ARM Cortex-M3 CPU-ra épülő változat is. Az AVR volt az első mikrovezérlő család, amely a programtárolásra integrált Flash (nem felejtő típusú adattároló technológia, mely elektronikusan törölhető és újraprogramozható) memóriát tartalmazott, így a szintén integrált adatmemória (regiszterek, RAM) miatt nincs szükség külső memória használatára. Az integrált memória leegyszerűsíti, olcsóbbá teszi ezen mikrovezérlőn alapuló eszközök gyártását. Egyszerű felépítés és alacsony áruk mellett nagy előnyük az ATMEL picoPower technológiájának köszönhető alacsony fogyasztás, valamint a sokféle fejlesztői környezettel való kompatibilitás. A fejlesztőrendszer további előnye, hogy nagyon sok Arduino alapú projekt hardver és szoftver leírása érhető el az interneten, megkönnyítve a fejlesztéshez szükséges ismeretek elsajátítását, valamint ötleteket adva más projektek megvalósításához.

Az Arduino platform nyílt forrású, ami azt jelenti, hogy az elektronikai hardver egységek, a fejlesztő- valamint a kiegészítő modulok kapcsolási és nyomtatott áramköri tervrajzai bárki számára ingyenesen hozzáférhetőek, így akár bárki ezeket felhasználhatja saját projektjének megvalósítására. Természetesen a referencia fejlesztő és kiegészítő modulok, valamint ezek más gyártó általi utánpótlásai, úgynevezett klónjai kereskedelemben beszerezhetőek. A

fejlesztőmodulok csatlakozósorokra kivezetett jelek segítségével illeszthetők egyéb berendezésekhez, valamint további kiegészítő modulokhoz.

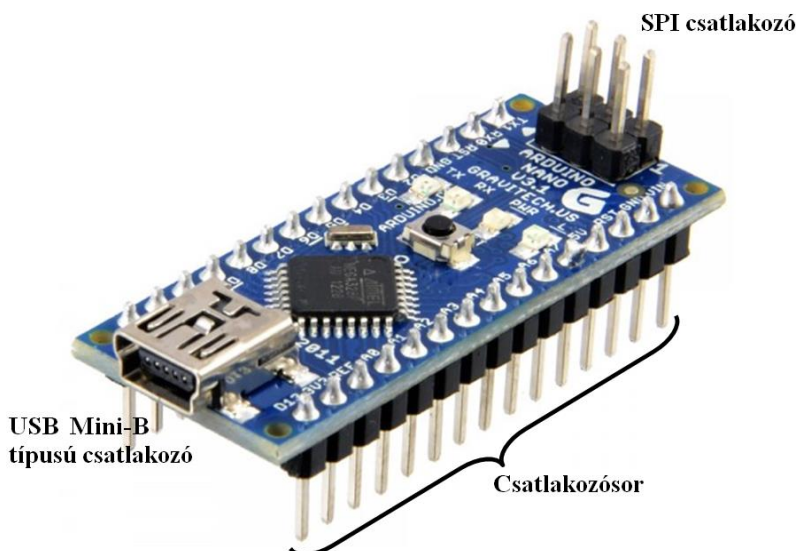
Az integrált fejlesztői környezetet szintén ingyenesen hozzáférhető, nyílt forráskódú, és több operációs rendszeren is biztosítja az alkalmazásfejlesztés lehetőségét. A fejlesztőmodul USB-n keresztül csatlakoztatható a fejlesztőkörnyezetet futtató számítógéphez, a lefordított kód ezen keresztül tölthető le a mikrokontrollerre. Az Arduino platform egyik nagy előnye, hogy az Arduino fejlesztőmodulok gyártása során a mikrovezérlőre feltöltenek egy rendszerbetöltő, úgynevezett bootloader alkalmazást, amely lehetővé teszi, hogy külső speciális és általában költséges programozó nélkül, egyszerűen UART (univerzális soros) interfészen keresztül is fel lehessen tölteni a mikrovezérlőre a megírt programot. Az USB-n történő programozás egy USB-UART illesztő segítségével valósítható meg. Néhány Arduino fejlesztőmodul esetén, mint például az UNO vagy a Mega 2560, egy második ATmega16U2/8U2 típusú mikrovezérlő valósítja meg az átalakítást, a Nano esetén pedig FTDI FT232RL vezérlőegység. A fejlesztőmodul USB csatlakozón vagy külső tápforrásból kaphatja a működéshez szükséges tápfeszültséget.

Az alábbi ábra mutatja az egyik legelterjedtebb Arduino fejlesztőmodult az Arduino UNO R3-at, melynek bal oldalán található az USB és tápforrás csatlakozása, valamint a felső és alsó részen láthatóak az egyes kivezetett jelek csatlakozósorai.



13. ábra: *Arduino UNO R3 fejlesztőmodul*

A 2. ábrán az UNO-val hasonló funkcionalitású, de sokkal kisebb méretű, Arduino Nano fejlesztőlap V3.1 látható.



14. ábra: *Arduino Nano V3.1 fejlesztőmodul (SPI: Soros Periféria Interfész)*

A hivatalos Arduino platformból több modell is elérhető, melyek közös jellemzője, hogy Atmel mikrovezérlőket tartalmaznak, továbbá az integrált fejlesztői környezet egységes programozási felületet biztosít. Az összes hivatalos modell és azok kapcsolási valamint tervrajzai, továbbá mintaprogramok forráskódjai is elérhetőek az Arduino projekt weboldalán. Az 1-4. táblázat összefoglalja néhány Arduino fejlesztőmodul legfontosabb jellemzőit.

1. Táblázat Néhány Arduino fejlesztőmodul főbb jellemzője – processzor, memória, tápfeszültség

Fejlesztőmodul	Processzor	SRAM	FLASH	EEPROM	Órajel	VCC	Külső tápfesz
Arduino DUE	AT91 SAM3X8E	96 kB	512 kB	-	84 MHz	3,3 V	6-16 V
Arduino UNO R3	ATmega328	2 kB	32 kB	1 kB	16 MHz	5 V	6-20 V
Arduino Nano V3.1	ATmega328	2 kB	32 kB	1 kB	16 MHz	5 V	6-20 V
Arduino Pro Mini	ATmega168	1 kB	16 kB	0,5 kB	8 MHz	3,3/5 V	5-12 V
Arduino Ethernet	ATmega328	2 kB	32 kB	1 kB	16 MHz	5 V	6-20 V
Arduino Leonardo	ATmega32u4	32 kB	2,5 kB	1 kB	16 MHz	5 V	6-20 V
Arduino Micro	ATmega32u4	32 kB	2,5 kB	1 kB	16 MHz	5 V	6-20 V
Arduino Mega 2560	ATmega2560	8 kB	256 kB	4 kB	16 MHz	5 V	6-20 V
Arduino Mega ADK	ATmega2560	8 kB	256 kB	4 kB	16 MHz	5 V	6-20 V

2. Táblázat Néhány Arduino fejlesztőmodul főbb jellemzője – interfészek

Fejlesztőmodul	USB-Soros átalakító	USB hoszt	I ² C	SPI	ICSP	Ethernet	CAN	JTAG
Arduino DUE	Integrált	Integrált	2	1	1	-	1	1
Arduino UNO R3	ATmega16U2	-	1	1	1	-	-	-
Arduino Nano V3.1	FTDI FT232RL	-	1	1	1	-	-	-
Arduino Pro Mini	-	-	1	1	1	-	-	-
Arduino Ethernet	-	-	1	1	1	W5100	-	-
Arduino Leonardo	Integrált	-	1	1	1	-	-	-
Arduino Micro	Integrált	-	1	1	1	-	-	-
Arduino Mega 2560	ATmega16U2	-	1	1	1	-	-	-
Arduino Mega ADK	ATmega16U2	MAX3421E	1	1	1	-	-	-

3. Táblázat Néhány Arduino fejlesztőmodul főbb jellemzője – ki- és bemenetek

Fejlesztőmodul	DIO*	PWM*	ADC*	DAC	UART
Arduino DUE	54	12	12 (12 bits)	2 (12 bits)	4
Arduino UNO R3	14	6	6 (10 bits)	-	1
Arduino Nano V3.1	14	6	8 (10 bits)	-	1
Arduino Pro Mini	14	6	6 (10 bits)	-	1
Arduino Ethernet	14	4	6 (10 bits)	-	1
Arduino Leonardo	20	7	12 (10 bits)	-	1
Arduino Micro	20	7	12 (10 bits)	-	1
Arduino Mega 2560	54	15	16 (10 bits)	-	4
Arduino Mega ADK	54	15	16 (10 bits)	-	4

Az egyes fejlesztőmodulok esetén adott kivezetések DIO/PWM/ADC funkciók megvalósítása szerint konfigurálhatóak, ezekben az esetekben átfedések vannak.

4. Táblázat 4. táblázat Néhány Arduino fejlesztőmodul főbb jellemzője – ár

Fejlesztőmodul	Közelítő ár*	Klón ár [5]	Megjegyzés
Arduino DUE	15-25 eFt	10 eFt	Első ARM alapú, 2 db DAC, 32 bites. UNO-val nem kompatibilis, 3,3 V feszültség!
Arduino UNO R3	7-15 eFt	3 eFt	Legtöbb kiegészítő modullal kompatibilis.
Arduino Nano V3.1	5-17 eFt	2-3 eFt	Kis méretű UNO, nem UNO kompatibilis kivez.
Arduino Pro Mini	3-9 eFt	1,5 eFt	Kis méret, soros porton, USB-UART átalakítóval programozható, nem UNO kompatibilis kivez.
Arduino Ethernet	9-20 eFt		USB-UART átalakítóval programozható.
Arduino Leonardo	6-12 eFt	3 eFt	UNO-val nem kompatibilis, natív USB.
Arduino Micro	6-11 eFt	5 eFt	Kis méretű Leonardo, natív USB.
Arduino Mega 2560	13-15 eFt	5 eFt	Jó kompatibilitás az UNO-val, de SPI és I ² C interfész máshol helyezkedik el, mint az UNO esetén.
Arduino Mega ADK	22-28 eFt	12 eFt	Android fejlesztőkörnyezettel kompatibilis.

A fenti táblázatokban használt rövidítések:

- SRAM: statikus memória, a tápfeszültség megszűnéséig tárolja az adatokat.
- FLASH: Törölhető (blokkonként) és újraírható, gyors, nem felejtő memória.
- EEPROM: Törölhető (bájtonként) és újraírható, lassú, nem felejtő memória.
- VCC: processzor tápfeszültség.
- USB-Soros átalakító: A PC-ről történő programozáshoz szükséges USB-Soros vonali átalakító megvalósítása.
- USB hoszt: USB hoszt funkció megvalósítása (amihez USB kliensek csatlakozhatnak).
- I²C: Elsősorban mikrovezérlőknél, beágyazott rendszereknél alkalmazott soros vonali számítógépes busz és protokoll perifériák csatlakoztatásához.
- SPI: Soros periféria interfész, elsősorban beágyazott rendszereknél alkalmazott kommunikáció érzékelők, SD kártyaolvasók stb. számára.
- ICSP: Integrált soros programozó interfész.
- Ethernet: Helyi számítógépes hálózatokhoz használt adatátviteli mód.
- CAN: Elsősorban mikrovezérlőknél, beágyazott rendszereknél alkalmazott üzenet alapú adatbusz és hálózati protokoll (autóipar).
- JTAG: Integrált áramkörök tesztelésére használt csatlakozó.
- DIO: Digitális ki- és bemenet.
- PWM: Impulzusszélesség modulált kimenet.
- ADC: Analóg bemenet, amely jelet analóg-digitális átalakító konvertál át - a mikroprocesszor által feldolgozható – digitális értékké.
- DAC: Analóg kimenet, amelyet – a mikroprocesszor által előállított – digitális értékből digitális-analóg átalakító konvertál át analóg jellé.
- UART: Párhuzamos és soros vonali adatok közti átalakítást végzi, az adattovábbításhoz soros vonali kapcsolatot használ.

Az Arduino fejlesztőmodulok jellemzőit összefoglaló táblázatokból látszik, hogy általános, nem túl bonyolult projektek esetén, a platformon történő fejlesztés elsajátításához szinte bármelyik modul megfelelő. A legtöbb kiegészítő modullal az UNO kompatibilis, így elsősorban ezt a modult célszerű

megismerni, valamint ennek kisebb Nano vagy Pro Mini változatát, valamint ezen modulok klón változatai igen olcsón beszerezhetőek.

Arduino fejlesztőmodulok

A következőkben az Arduino fejlesztőmodulok felépítése, főbb funkciói kerülnek bemutatásra, valamint a legelterjedtebb, olcsó Arduino UNO, valamint annak kisméretű változata, az UNO modullal azonos paraméterű Arduino Nano kerül ismertetésre. Megemlítendő, hogy az Arduino Pro Mini fejlesztőmodul klón változata ATmega328 mikroprocesszorral is beszerezhető, ezért, mint legolcsóbb fejlesztőeszköz ez is bemutatásra kerül. Az Arduino Pro Mini ellentétben az UNO vagy Nano változatokkal, nem tartalmaz USB-UART átalakító egységet, ezért csak külső átalakítóval programozható.

Az Arduino fejlesztőmodulok által az egyes projektek fejlesztéséhez, teszteléséhez biztosított főbb funkciókat megvalósító egységek az alábbiak:

- Atmel mikroprocesszor: vezérlő, szabályozó funkciók, I/O és kommunikáció kezelése.
- Kivezetések: I/O és kommunikáció számára szükséges csatlakozások biztosítása.
- Tesztelést biztosító, segítő egységek: hibakeresés funkciókra használható, kommunikáció, táp meglétének jelzésére LED-es fény visszajelzés, Reset gomb az eszköz újraindítására, ICSP interfész a programozáshoz.
- Tápegység: működéshez szükséges feszültség előállítása.

A legtöbb Arduino platform fejlesztőmodulja az Atmel AVR ATmega 8 bites mikrokontrollerre épül, az Arduino UNO például az ATmega328 mikrokontrollert tartalmazza. Az Arduino DUE kivételnek számít, mivel az ARM architektúrára, Cortex-M3 mikrokontrollerre épülő fejlesztőmodul. A mikrokontrollerbe integrált Flash memória lehetővé teszi, hogy a lefordított kód a mikrokontrollerbe letöltve a tápfeszültség megszűnése után is megmaradjon. A mikrokontroller a tárolt programot futtatja és hozzáférést biztosít az integrált vagy külső perifériákhoz. A mikrokontroller fontosabb funkcióit megvalósító kivezetései az egyes fejlesztőmodulokon csatlakozósorokra vannak kivezelve lehetővé téve a külső egységek és más modulok csatlakoztatását. Néhány fejlesztőmodul esetén a 13. számú kivezetésre egy tesztelésre szánt LED-es fény visszajelző van bekötve, mely segítségével a mikrokontrolleren futó algoritmus visszajelzést adhat a működéséről.

Az egyes fejlesztőmodulok tápfeszültség ellátása többféle módon is biztosítható. Az USB-n keresztül programozható modulok az USB csatlakozó által biztosított 5V feszültségről üzemeltethetőek, de lehetőség van külső tápforrás csatlakoztatására is, így a felprogramozott fejlesztőmodul a számítógéptől független tápforrásról is üzemeltethető. Az Arduino Ethernet

esetén - az adott fejlesztőmodultól függően - lehetőség van PoE (Ethernet vezetéken keresztül történő tápellátás) tápellátás biztosítására is. A fejlesztőmodul tápegység része automatikusan választja ki az alkalmazott tápforrást. A fejlesztőmodulok tápegység része általában 5 V és 3,3 V stabil tápfeszültséget előállító egységet is tartalmaz.

A fejlesztőmodulok általános célú digitális ki- és bemenetei, valamint analóg ki- és bemenetei közül több – az alkalmazástól függő konfiguráció szerint – rendelhetőek az egyes valós kivezetésekhez. Ez azt jelenti, hogy az egyes fejlesztések esetén különösen figyelni kell arra, hogy a konkrét kivezetés a programban milyen funkció szerint van konfigurálva. A fejlesztőmodulok többségénél a digitális ki- és bemenetek 5 V jelszintűek, az analóg bemenet 0...5 V. Az általános célú ki- és bemeneti csatlakozópontok a digitális ki- és bemeneti funkciókon kívül megvalósíthatnak különböző kommunikációs interfészek, PWM kimenet, külső megszakítás kivezetéseket is.

Atmel mikrovezérlők

Az egyes Arduino fejlesztőmodulokban található mikrovezérlők főbb paramétereit az 2-4. táblázatok foglalják össze. A legerterjedtebb fejlesztőmodulban, az Arduino UNO-ban található ATmega328, a több I/O-val rendelkező Mega 2560 modulban lévő ATmega2560, az újabb Leonardo típusban alkalmazott ATmega32u4, és az első ARM alapú DUE-ban található AT91SAM3X8E mikrokontrollerek egyéb főbb tulajdonságai, paramétereit az 5. táblázatban láthatóak. Látható, hogy az Arduinoban használt mikrovezérlők tartalmaznak időzítőt, Watchdogot (programfutási hiba detektálására), energiatakarékos alvó üzemmódot, UART interfészt, megszakításhívásra használható analóg komparátort (mely az alvó üzemmódból is képes felébreszteni az eszközt), időzítési funkciókra használható valósidejű számlálót is. Az USART port az UART soros és párhuzamos átalakítást végző port kiterjesztése, lehetővé téve, hogy az UART interfésznél alkalmazott aszinkron bitfolyamtól eltérően, szinkron módú adattovábbítást is támogatja.

5. Táblázat Atmel mikrovezérlők

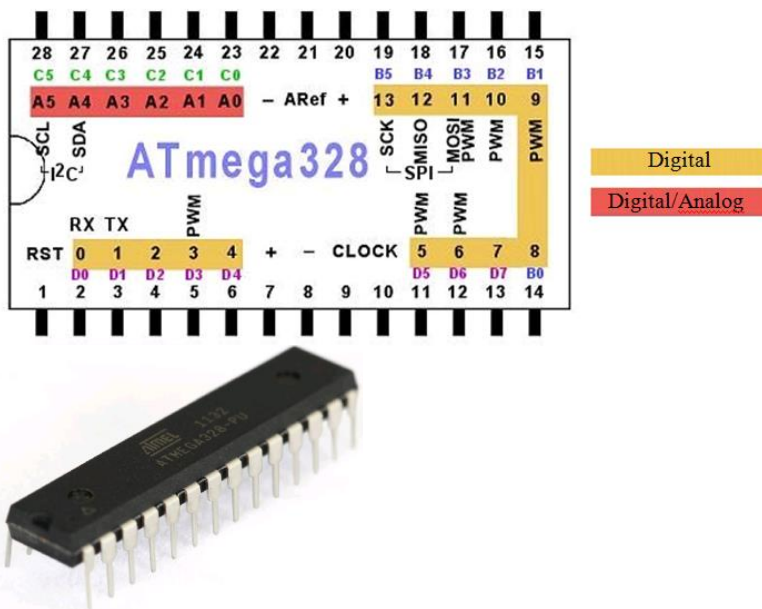
Funkció	ATmega328	ATmega2560	ATmega32u4	AT91SAM3X8E
Regiszterek (db)	32	32	32	25 (core)
Időzítő (db/bit)	2/8, 1/16	2/8, 4/16	1/8, 2/16, 1/10 (PLL)	9/32, 8/16
Watchdog	1	1	1	1
Analóg komparátor	1	1	1	-
USART	1	4	1	4
Watchdog	Igen	Igen	Igen	Igen
Alvó mód	Igen	Igen	Igen	Igen
Valósidejű számláló	Igen	Igen	Nem	Igen

Az Arduino legelterjedtebb fejlesztőmoduljaiban az ATmega328 mikrovezérlő található. Az Atmel a mikrovezérlőt 4 féle tokozásban gyártja, ahogy az a 3. ábrán látható. A 4 féle tokozás kivezetéseinek száma is eltérő: PDIP és QFN/MLF esetén 28, TQFP és QFN/MLF tokozás esetén 32. A 32 kivezetéssel rendelkező változatok nem 6, hanem 8 csatornás analóg-digitál átalakítót tartalmaznak, mint például az Arduino Nano esetén.

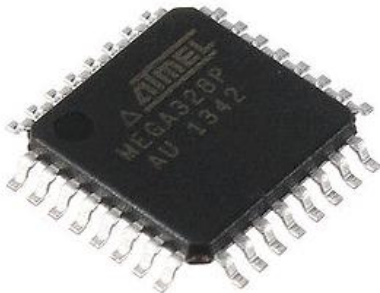
A kivezetések közül VCC a tápfeszültség, GND a föld (nulla), RESET (negált), AVcc analóg-digitál átalakító tápfeszültség, AREF az átalakító referencia feszültség kivezetéseken kívül 3 további portot B,C,D tartalmaz:

- A Port B 6 bites, kétirányú ki- és bemenet, az SPI interfész B3-B5 kivezetéseken érhető el.
- A Port C 6 bites, kétirányú ki- és bemenet, de a bemenetek e port esetén lehetnek analóg bemenetek is. Az I2C interfész a C4-C5 kivezetéseken érhető el.
- A Port D egy 8 bites, kétirányú ki- és bemenet, az UART interfész a D0-D1 kivezetéseken érhető el.

A Portok egyes ki- és bemenetei belső felhúzó (tápfeszültség és kivezetés közé kötött) ellenállásokat is tartalmaznak, valamint az adott kivezetések további eltérő funkciókkal is rendelkeznek. Az egyes kivezetések, portok jelölése PDIP és TQFP tokozás esetén az alábbi ábrákon láthatóak.



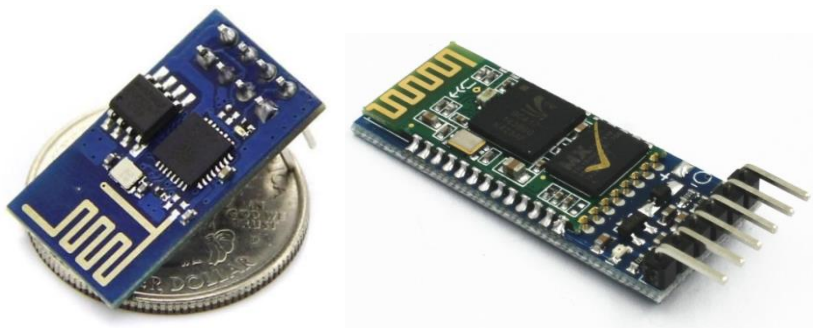
15. ábra: 4. ábra ATmega328 kivezetéseinek jelölése PDIP tokozás esetén



16. ábra: *ATmega328* kivezetéseinek jelölése TQFP tokozás esetén

Kiegészítő modulok

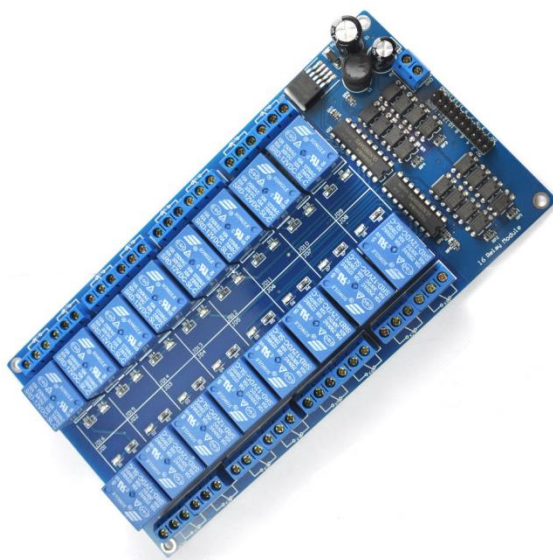
A kiegészítő modulok elsősorban a fejlesztő modul funkcióinak kiegészítésére szolgálnak. Ilyen kiegészítő funkciók például a kommunikációt biztosító modulok, mint az Ethernet, WiFi, Bluetooth vagy GSM kapcsolat, az egyes beavatkozókat működtető modulok, mint a relé vagy a motorvezérlő modul, vagy éppen az egyes érzékelők illesztését biztosító modulok. További fontos funkciókat megvalósító kiegészítő modulok az információk tárolását, az adatgyűjtést biztosító SD foglalatot tartalmazó modulok vagy éppen az információk kijelzését megvalósító LCD modulok is. Néhány kiegészítő modul látható a következő ábrán.



17. ábra: ESP8266 soros WiFi, és HC-05 soros Bluetooth modul



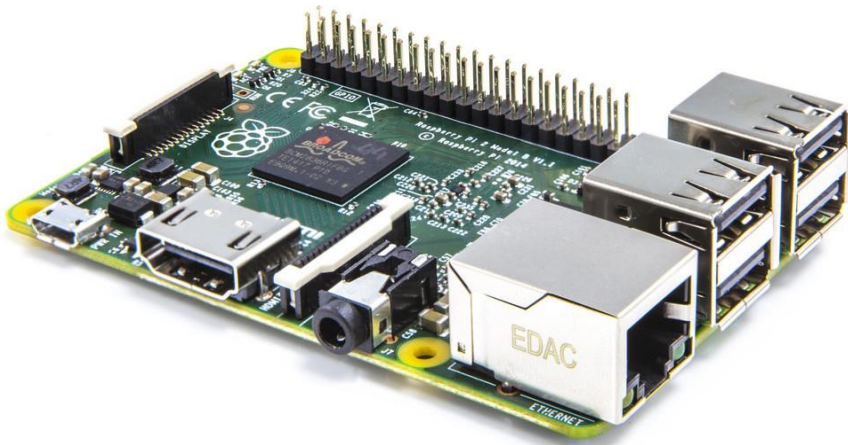
18. ábra: Nokia 5110 LCD modul



19. ábra: 16 csatornás relé modul

Raspberry Pi miniszámítógép mint folyamatirányító, folyamatfelügyeleti rendszer, HMI vagy kommunikációs interfész

A folyamatirányítási, folyamatfelügyeleti funkciók, távvezérlés, egyes szerver funkciók (weboldal, adatbázis stb.) továbbá az ember-gép kapcsolat (HMI: human-machine interface) megvalósítására a gyakorlati példák kidolgozása szempontjából legalkalmasabb egy olcsó mini számítógép. A Raspberry Pi egy kicsi, nagyjából bankkártya méretű, olcsó Broadcom gyártmányú ARM architektúrájú miniszámítógép, melyhez többféle Linux disztribúció is letölthető és microSD kártyáról futtatható. A Raspberry Pi egyetlen lapra integrált (SoC – System on a Chip) BCM2835 egymagos ARMv6 architektúrájú 700 MHz órajelű, míg a Raspberry Pi 2 négymagos BCM2836 ARMv7 architektúrájú 900 MHz órajelű processzorra épül. A Raspberry Pi A változat 256 MB, a B és B+ változat 512 MB, míg a Raspberry Pi 2 1 GB RAM-ot tartalmaz. A továbbiakban a Raspberry Pi 2 kerül részletesebben bemutatásra, a Raspberry Pi weboldalán <http://www.raspberrypi.org> a korábbi változatok specifikációja és sok más információ is megtalálható.



20. ábra: *Raspberry Pi 2 – oldalnézet*



21. ábra: Raspberry Pi 2 - felülnézet

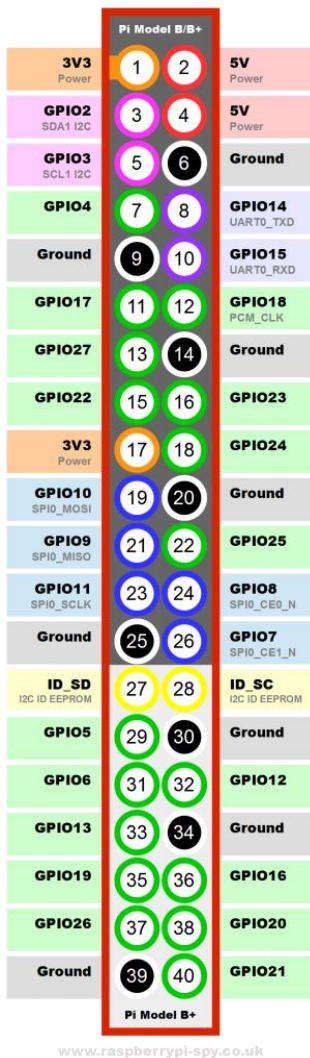
A Raspberry Pi 2 főbb paramétereit:

- BCM2836 ARMv7 architektúrájú 900 MHz órajelű négymagos processzor
- VideoCore IV IGP grafikus kétmagos processzor
- 1 GB RAM
- tápellátás microUSB-n keresztül (max 2A)
- 1db 10/100 Mbit/s Ethernet port
- 4db USB port
- 1db normál HDMI port
- 1db DSI kijelző port érintőképernyő számára (lap szélén a processzortól balra)
- 1db CSI kamera port (HDMI és audio kimenet között)
- 1db 4 pólusú sztereo audio és kompozit video kimenet
- 1db 40 kivezetéses általános ki- bemeneti port (GPIO)
- microSD foglalat az operációs rendszer betöltése és az adatok tárolása számára

A Raspberry Pi-hez ajánlott operációs rendszer a Raspbian, ami a Debian Linux kifejezetten RaspberryPi-re optimalizált változata, de elérhető rá a Fedora optimalizált változata is a Pidora. Az ARMv7 utasításarchitektúrának köszönhetően képes az ARM GNU/Linux disztribúciók és a Microsoft Windows 10 futtatására is, mely a Windows Developer Program for IoT program keretén belül ingyen elérhető lesz.

A Raspberry Pi 2 40 kivezetéses általános ki- bemeneti portot is tartalmaz az alacsonyabb szintű kommunikáció, funkciók számára. A kivezetések 3,3 V logikai szintűek nem 5 V, mint az Arduino esetében. A kivezetések funkciói az alábbiak:

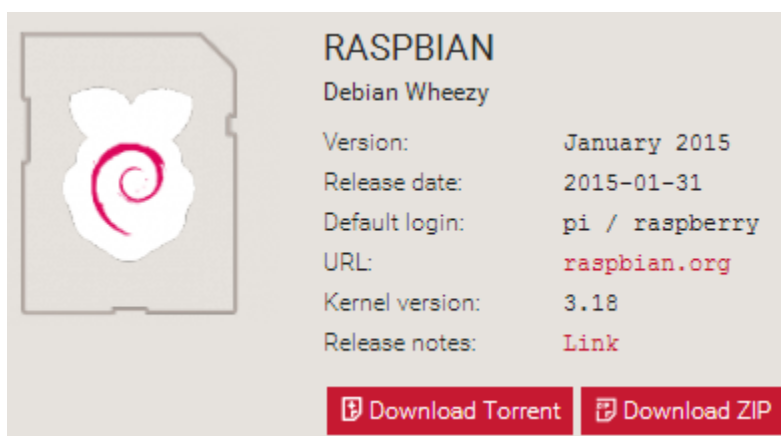
- 5 V és 3,3 V tápfeszültség (max 50mA)
- Ground föld, a nulla feszültségszint
- GPIO2-GPIO27 3,3 V jelszintű digitális be- vagy kimenet (egy max 16mA, de összesen max 50 mA, célszerűen max 0,5 mA kimenetenként)
- kommunikációs protokollok: I²C, SPI, UART



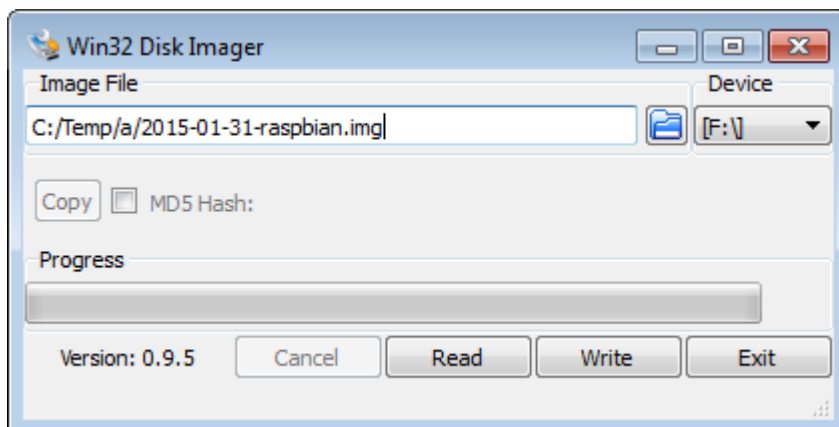
22. ábra: Raspberry Pi 2 GPIO csatlakozójának kiosztása

Raspbian telepítése

A Raspberry Pi 2-re többféle operációs rendszer is telepíthető. Ezek főként Linux disztribúciók, de a jövőben a Windows 10 egy változata is futtatható lesz. A következőkben a Debian disztribúció alapú Raspbian telepítése kerül bemutatásra. Az operációs rendszer képfájlja az alábbi linkről letölthető: <http://www.raspberrypi.org/downloads/>. Az alábbi kép szerint „Download ZIP”-re kattintva a képfájl zip csomagolt fájlként tölthető le. Kicsomagolás után a kapott img kiterjesztésű fájl Windows operációs rendszeren a <http://win32diskimager.sourceforge.net/> oldalról letölthető Win32 Disk Imager programmal legalább egy 8GB kapacitású microSD kártyára írható, az img fájl megadásával, a cél kiválasztásával, majd a „Write” gombra kattintva.



23. ábra: Raspbian Debian alapú operációs rendszer képfájljának letöltése



24. ábra: Win32 Disk Imager program

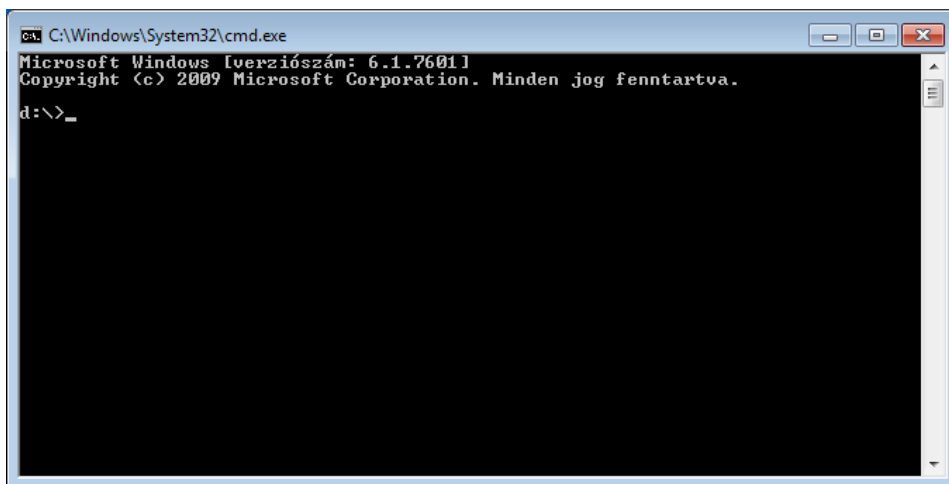
A microSD kártyát ezután be kell helyezni a Raspberry Pi 2-be, HDMI monitor, billentyűzet és egér csatlakoztatása után a microUSB csatlakozón keresztül tápfeszültséget kell adni a mini számítógépnek és a Debian alapú operációs rendszer betöltődik. A grafikus felület nem indul el, az a startx paranccsal indítható. A „sudo raspi-config” konfigurációs menüben lehet megadni, hogy a grafikus felület automatikusan betöltődjön a „3 Enable Boot to Desktop” menüpontot választva. Amennyiben nem szeretnénk vagy nincs lehetőségünk monitor, billentyűzet és egér csatlakoztatására, akkor SSH parancssori vagy grafikus távoli asztali elérés is beállítható.

SSH kapcsolat létrehozása putty alkalmazás segítségével

Ha a Raspberry Pi 2 mini számítógépen az operációs rendszer betöltődött és a számítógép az Ethernet hálózathoz csatlakoztatva van, akkor az IP cím alapján úgynevezett SSH kapcsolaton csatlakozni lehet hozzá. SSH kapcsolat Windows operációs rendszer alatt legegyszerűbben a putty.exe program segítségével, mely az alábbi linkről tölthető le:

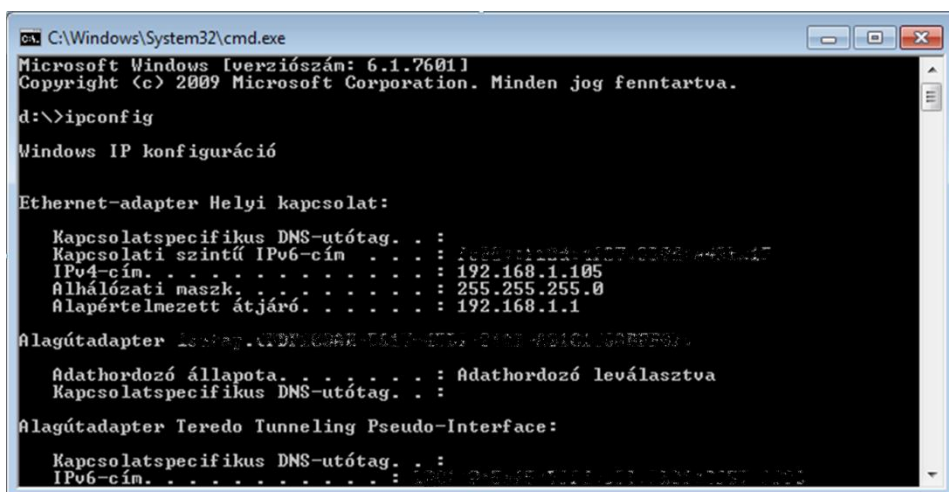
<http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>.

A kapcsolat létrehozásához szükség van a mini számítógép IP címére. Ha monitor és billentyűzet csatlakoztatva van hozzá, akkor az IP cím az ifconfig paranccsal kérdezhető le, az „eth0” hálózati csatoló „inet addr:” címke után található. Amennyiben nincs csatlakoztatva hozzá monitor és billentyűzet, akkor a Raspberry Pi 2 IP címe otthoni hálózati eszközön (router) a csatlakoztatott eszközöknél nézhető meg, vagy – általános beállítású otthoni hálózat esetén – hálózati ismeretek nélkül a legegyszerűbben próbálgatással határozható meg. A Raspberry Pi 2 ugyanannak a hálózati eszköznek (router) az egyik portjához legyen csatlakoztatva, ahova az SSH kapcsolat létrehozására használt kliens PC is csatlakozik, mert ebben az esetben a Raspberry Pi 2 IP címe a PC IP címéhez hasonló lesz. A kliens PC IP címének lekérdezését Windows operációs rendszeren legegyszerűbben az ipconfig parancs segítségével valósíthatjuk meg, ehhez egy parancssori ablakot kell nyitni, mely a menüből elérhető vagy a cmd paranccsal indítható.



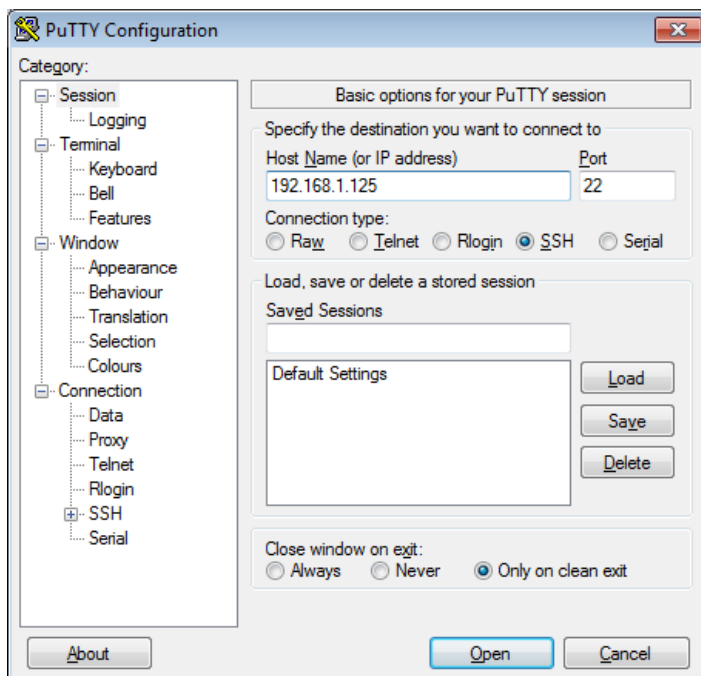
25. ábra: *ipconfig* futtatása parancssorban

A parancs lefuttatása után az „Ethernet-adapter Helyi kapcsolat” pont alatt, az „IPv4-cím” sorban látható a PC IP címe. Általános otthoni háózzati beállítások esetén az IP cím feltehetőleg 192.168.0.XXX vagy 192.168.1.XXX lesz.



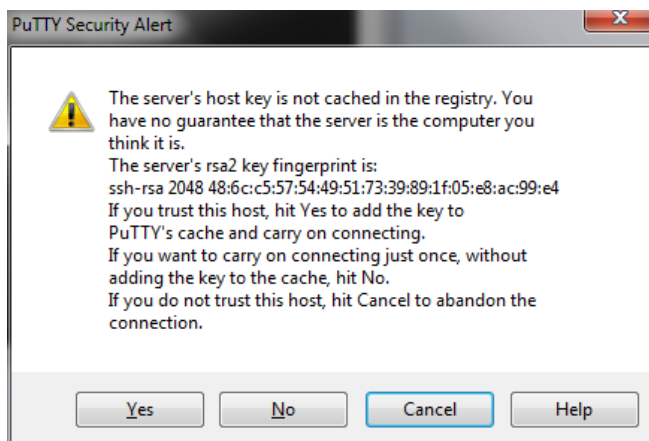
26. ábra: *ipconfig* segítségével a kliens számítógép IP címének meghatározása

A Raspberry Pi 2 IP címe feltehetően ezen IP cím közelében lesz, a putty programban az XXX érték 2-254 értékek közötti változtatásával az „Open” gombra kattintva lehet próbálni az SSH kapcsolaton keresztül történő belépést az alábbi ábrának megfelelően.



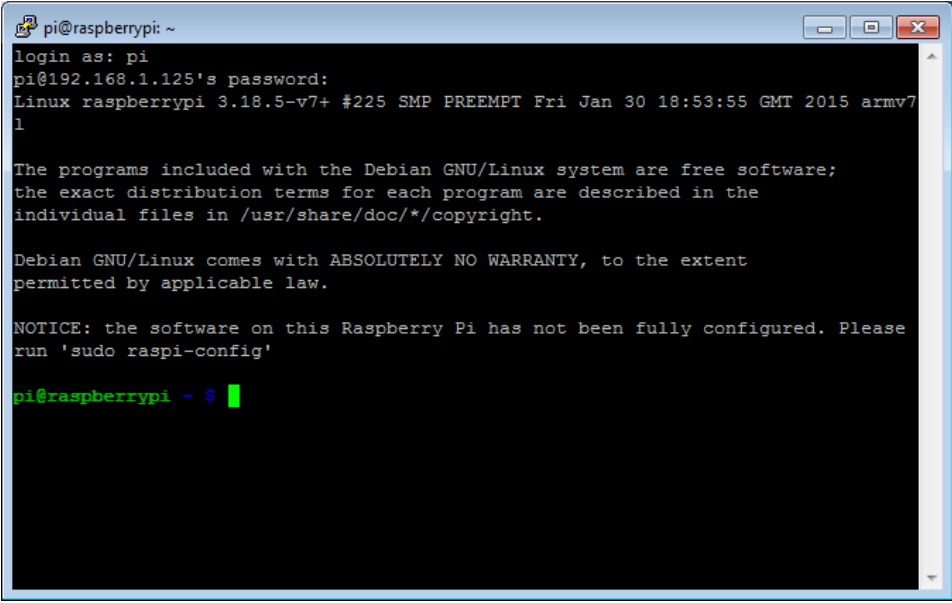
27. ábra: *putty programmal SSH kapcsolat létrehozása a 192.168.1.125 IP címen*

Amennyiben az IP cím helyes, a kapcsolat létrehozása sikerül, úgy az alábbi üzenet jelenik meg, mely az rsa2 kapcsolat titkosítási kulcs elfogadására és elmentésére utal. „Yes”-re kattintva elfogadásra kerül a kulcs.



28. ábra: *rsa2 kulcs elfogadása*

Belépéshez felhasználónévként (login as:) „pi”-t, jelszóként (password:) „raspberry”-t kell megadni és az alábbi parancssor fogad.

A screenshot of a terminal window titled 'pi@raspberrypi: ~'. The terminal shows the login process for the 'pi' user. The prompt is 'login as: pi', followed by 'pi@192.168.1.125's password:'. Below this, the system version is displayed: 'Linux raspberrypi 3.18.5-v7+ #225 SMP PREEMPT Fri Jan 30 18:53:55 GMT 2015 armv7l'. A message follows: 'The programs included with the Debian GNU/Linux system are free software; the exact distribution terms for each program are described in the individual files in /usr/share/doc/*/copyright.' Another message states: 'Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.' A notice then says: 'NOTICE: the software on this Raspberry Pi has not been fully configured. Please run 'sudo raspi-config''. The prompt returns to 'pi@raspberrypi ~ \$' with a green cursor.

29. ábra: Raspberry Pi 2-re történő sikeres bejelentkezés

Parancssorból távoli asztali kapcsolat is beállítható, az alábbiak szerint xrdp vagy VNC segítségével.

Távoli asztali kapcsolat létrehozása xrdp segítségével

Windows operációs rendszereken a „Távoli asztali kapcsolat” alkalmazás az RDP (Távoli Asztali Protokoll – Remote Desktop Protocol) protokollt használja egy másik PC grafikus felülethez hálózaton keresztül történő kapcsolódásához. Ehhez azon a számítógépen, amihez távoli asztallal szeretnénk kapcsolódni egy szerverszolgáltatásra van szükség, mely a kapcsolatokat kezeli. Linux operációs rendszeren ezt valósítja meg az xrdp, mely a „sudo apt-get install xrdp” paranccsal telepíthető. A telepítés végeztével a „Starting Remote Desktop Protocol server : xrdp sesman” sor elején az „[ok]” jelzi, a szerver szolgáltatás sikeres indítását.

```
pi@raspberrypi: ~
login as: pi
pi@192.168.1.125's password:
Linux raspberrypi 3.18.5-v7+ #225 SMP PREEMPT Fri Jan 30 18:53:55 GMT 2015 armv7
l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

NOTICE: the software on this Raspberry Pi has not been fully configured. Please
run 'sudo raspi-config'

pi@raspberrypi ~ $ sudo apt-get install xrdp
```

```
pi@raspberrypi: ~
Unpacking xfonts-base (from .../xfonts-base_1%3a1.0.3_all.deb) ...
Selecting previously unselected package xrdp.
Unpacking xrdp (from .../xrdp_0.5.0-2_armhf.deb) ...
Processing triggers for man-db ...
Processing triggers for fontconfig ...
Setting up tightvncserver (1.3.9-6.4) ...
update-alternatives: using /usr/bin/tightvncserver to provide /usr/bin/vncserver
(vncserver) in auto mode
update-alternatives: using /usr/bin/Xtightvnc to provide /usr/bin/Xvnc (Xvnc) in
auto mode
update-alternatives: using /usr/bin/tightvncpasswd to provide /usr/bin/vncpasswd
(vncpasswd) in auto mode
Setting up xfonts-base (1:1.0.3) ...
Setting up xrdp (0.5.0-2) ...
[....] Generating xrdp RSA keys.....
Generating 512 bit rsa key...

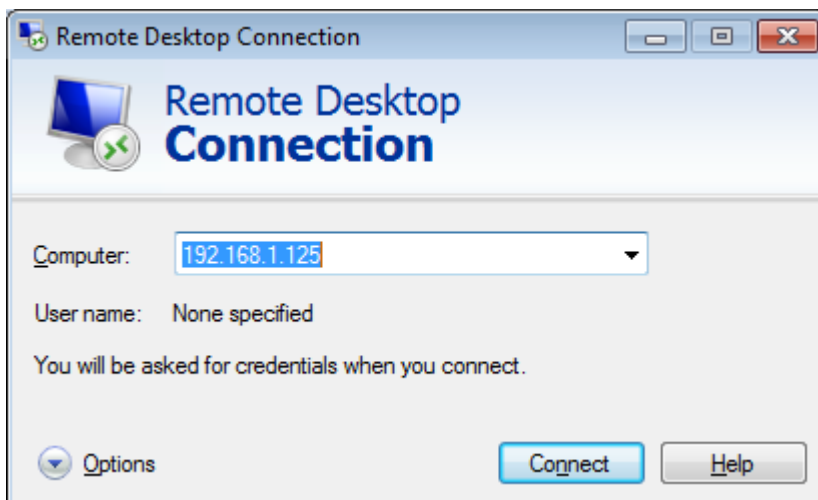
ssl_gen_key_xrdp1 ok

saving to /etc/xrdp/rsakeys.ini

done (done).
[ ok ] Starting Remote Desktop Protocol server : xrdp sesman.
pi@raspberrypi ~ $
```

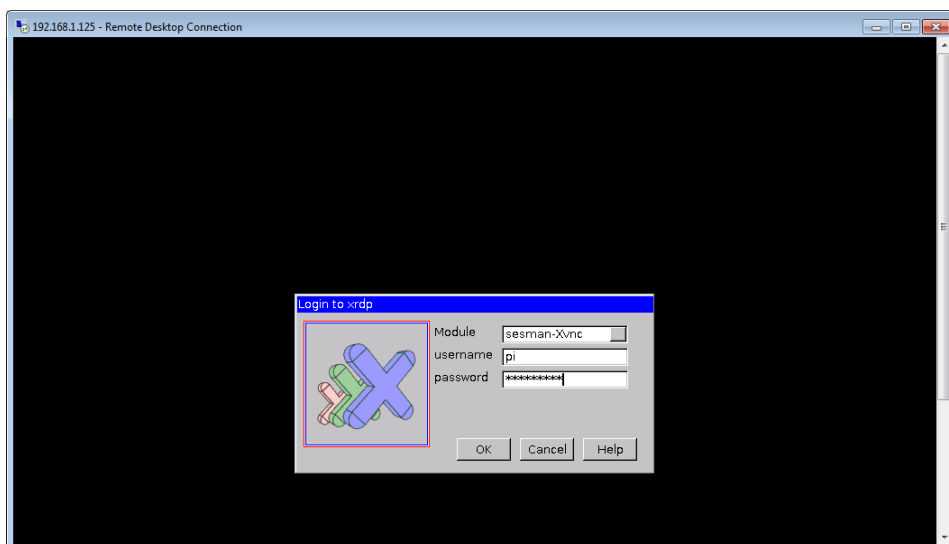
30. ábra: *xrdp telepítése*

Sikeres telepítés után a Raspberry Pi 2 IP címének ismeretében a távoli asztali kapcsolat Windows számítógépről a „Távoli asztali kapcsolat” alkalmazás segítségével hozható létre, mely a Start menüből is elérhető.



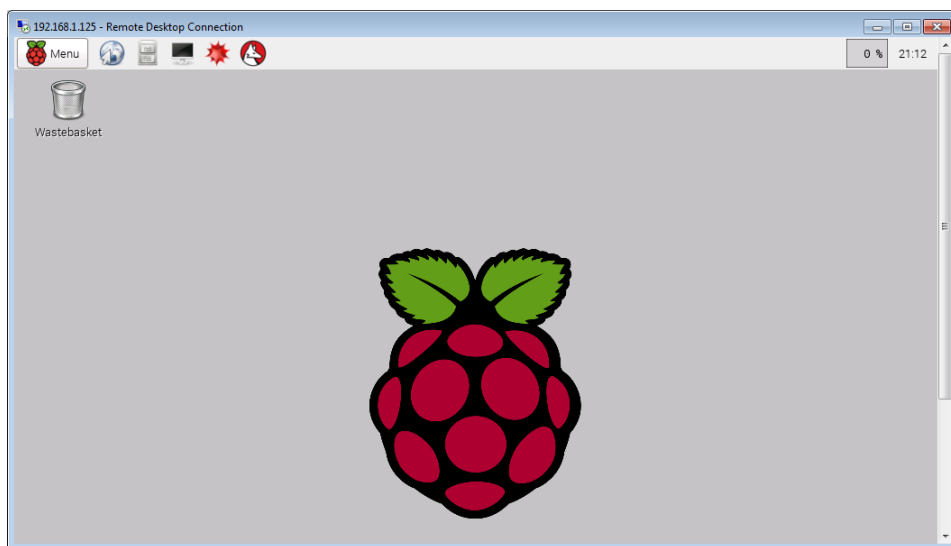
31. ábra: Távoli asztali kapcsolat létrehozása

Amennyiben a kapcsolat létrejön, úgy a bejelentkezési képernyő fogad az alábbi ábra szerint. Itt meg kell adni felhasználónévként (username) „pi”-t, jelszóként (password) „raspberry”-t figyelve arra, hogy a billentyűzetkiosztás angol, így az „y” karakternél a „z”-t kell leütni.



32. ábra: xrdp bejelentkező képernyő

Sikeres bejelentkezés után a feltelepített Raspbian operációs rendszer asztala látható.



33. ábra: Raspbian rendszer asztalának elérése RDP Távoli asztali kapcsolaton

Távoli asztali kapcsolat létrehozása VNC segítségével

A VNC távoli asztali kapcsolat szerver oldali konfigurálásához és a gép elindulásakor történő automatikus indításához több konfigurációs beállítás szükséges. Ennek egyszerűsített lépéssorozata az alábbi parancsok segítségével valósítható meg, egy szkript segítségével (részletes leírás itt található: <http://www.penguintutor.com/linux/tightvnc>):

1. `wget http://www.penguintutor.com/otherfiles/tightvncserver-init.txt`
2. `sudo mv tightvncserver-init.txt /etc/init.d/tightvncserver`
3. `sudo chown root:root /etc/init.d/tightvncserver`
4. `sudo chmod 755 /etc/init.d/tightvncserver`
5. `sudo update-rc.d tightvncserver defaults`
6. `sudo /etc/init.d/tightvncserver start`

```
pi@raspberrypi: ~
login as: pi
pi@192.168.1.125's password:
Linux raspberrypi 3.18.5-v7+ #225 SMP PREEMPT Fri Jan 30 18:53:55 GMT 2015 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Feb 10 22:00:48 2015 from 192.168.1.122
pi@raspberrypi ~ $ wget http://www.penguintutor.com/otherfiles/tightvncserver-init.txt
```

```
pi@raspberrypi: ~
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Feb 10 22:00:48 2015 from 192.168.1.122
pi@raspberrypi ~ $ wget http://www.penguintutor.com/otherfiles/tightvncserver-init.txt
--2015-02-10 22:15:44-- http://www.penguintutor.com/otherfiles/tightvncserver-init.txt
Resolving www.penguintutor.com (www.penguintutor.com)... 213.171.218.150
Connecting to www.penguintutor.com (www.penguintutor.com)|213.171.218.150|:80...
connected.
HTTP request sent, awaiting response... 200 OK
Length: 769 [text/plain]
Saving to: `tightvncserver-init.txt'

100%[=====>] 769      --.-K/s   in 0s

2015-02-10 22:15:44 (12.8 MB/s) - `tightvncserver-init.txt' saved [769/769]

pi@raspberrypi ~ $ sudo mv tightvncserver-init.txt /etc/init.d/tightvncserver
pi@raspberrypi ~ $
```

```
pi@raspberrypi: ~
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Feb 10 22:00:48 2015 from 192.168.1.122
pi@raspberrypi ~ $ wget http://www.penguintutor.com/otherfiles/tightvncserver-init.txt
--2015-02-10 22:15:44-- http://www.penguintutor.com/otherfiles/tightvncserver-init.txt
Resolving www.penguintutor.com (www.penguintutor.com)... 213.171.218.150
Connecting to www.penguintutor.com (www.penguintutor.com)|213.171.218.150|:80...
connected.
HTTP request sent, awaiting response... 200 OK
Length: 769 [text/plain]
Saving to: `tightvncserver-init.txt'

100%[=====>] 769          --.-K/s   in 0s

2015-02-10 22:15:44 (12.8 MB/s) - `tightvncserver-init.txt' saved [769/769]

pi@raspberrypi ~ $ sudo mv tightvncserver-init.txt /etc/init.d/tightvncserver
pi@raspberrypi ~ $ sudo chown root:root /etc/init.d/tightvncserver
pi@raspberrypi ~ $
```

```
pi@raspberrypi: ~
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Feb 10 22:00:48 2015 from 192.168.1.122
pi@raspberrypi ~ $ wget http://www.penguintutor.com/otherfiles/tightvncserver-init.txt
--2015-02-10 22:15:44-- http://www.penguintutor.com/otherfiles/tightvncserver-init.txt
Resolving www.penguintutor.com (www.penguintutor.com)... 213.171.218.150
Connecting to www.penguintutor.com (www.penguintutor.com)|213.171.218.150|:80...
connected.
HTTP request sent, awaiting response... 200 OK
Length: 769 [text/plain]
Saving to: `tightvncserver-init.txt'

100%[=====>] 769          --.-K/s   in 0s

2015-02-10 22:15:44 (12.8 MB/s) - `tightvncserver-init.txt' saved [769/769]

pi@raspberrypi ~ $ sudo mv tightvncserver-init.txt /etc/init.d/tightvncserver
pi@raspberrypi ~ $ sudo chown root:root /etc/init.d/tightvncserver
pi@raspberrypi ~ $ sudo chmod 755 /etc/init.d/tightvncserver
pi@raspberrypi ~ $
```

```
pi@raspberrypi: ~
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Feb 10 22:00:48 2015 from 192.168.1.122
pi@raspberrypi ~ $ wget http://www.penguintutor.com/otherfiles/tightvncserver-init.txt
--2015-02-10 22:15:44-- http://www.penguintutor.com/otherfiles/tightvncserver-init.txt
Resolving www.penguintutor.com (www.penguintutor.com)... 213.171.218.150
Connecting to www.penguintutor.com (www.penguintutor.com)|213.171.218.150|:80...
connected.
HTTP request sent, awaiting response... 200 OK
Length: 769 [text/plain]
Saving to: `tightvncserver-init.txt'

100%[=====>] 769          --.-K/s   in 0s

2015-02-10 22:15:44 (12.8 MB/s) - `tightvncserver-init.txt' saved [769/769]

pi@raspberrypi ~ $ sudo mv tightvncserver-init.txt /etc/init.d/tightvncserver
pi@raspberrypi ~ $ sudo chown root:root /etc/init.d/tightvncserver
pi@raspberrypi ~ $ sudo chmod 755 /etc/init.d/tightvncserver
pi@raspberrypi ~ $ sudo update-rc.d tightvncserver defaults
update-rc.d: using dependency based boot sequencing
pi@raspberrypi ~ $
```

```
pi@raspberrypi ~
2015-02-18 21:11:06 (19.2 MB/s) - `tightvncserver-init.txt' saved [769/769]

pi@raspberrypi ~ $ sudo mv tightvncserver-init.txt /etc/init.d/tightvncserver
pi@raspberrypi ~ $ sudo chown root:root /etc/init.d/tightvncserver
pi@raspberrypi ~ $ sudo chmod 755 /etc/init.d/tightvncserver
pi@raspberrypi ~ $ sudo update-rc.d tightvncserver defaults
update-rc.d: using dependency based boot sequencing
pi@raspberrypi ~ $ sudo /etc/init.d/tightvncserver start

You will require a password to access your desktops.

Password:
Warning: password truncated to the length of 8.
Verify:
Would you like to enter a view-only password (y/n)? n
xauth: file /home/pi/.Xauthority does not exist

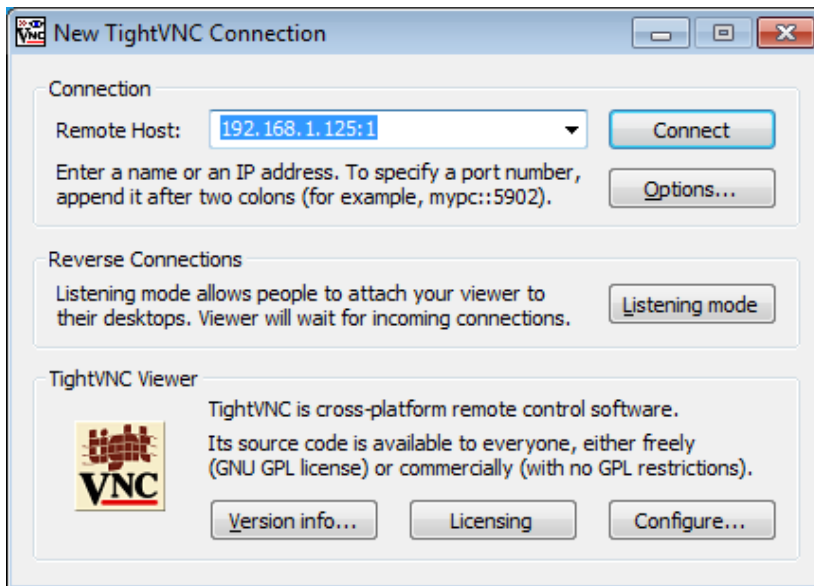
New 'X' desktop is raspberrypi:1

Creating default startup script /home/pi/.vnc/xstartup
Starting applications specified in /home/pi/.vnc/xstartup
Log file is /home/pi/.vnc/raspberrypi:1.log

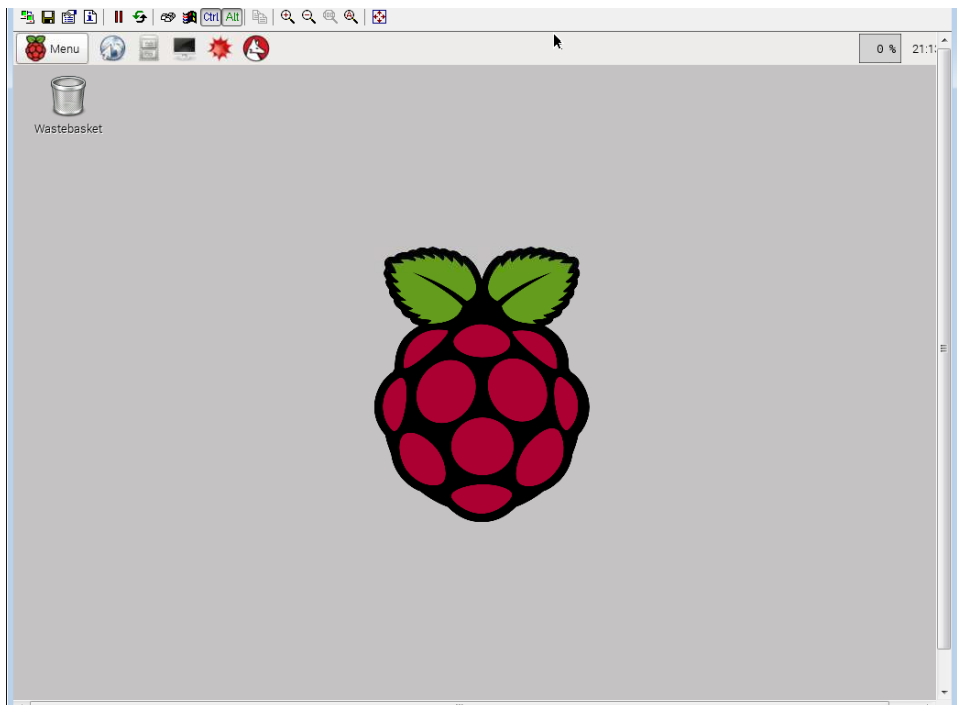
Starting TightVNC server for pi
```

34. ábra: VNC távoli kapcsolat konfigurálásának lépései

A telepítés során meg kell adni egy jelszót (password), amelyet a bejelentkezéshez kell a csatlakozáskor megadni. A <http://www.tightvnc.com/> oldalról letölthető és Windows operációs rendszerre telepíthető kliensalkalmazás a „TightVNC Viewer”, melyen a Raspberry Pi 2 IP címét beállítva a kapcsolat létrehozható a „Connect” gombra kattintva majd a jelszót megadva.



35. ábra: Távoli asztali kapcsolat létrehozása TightVNC Viewer alkalmazással



36. ábra: Raspbian rendszer asztalának elérése TightVNC Viewer segítségével

Az így előkészített rendszer már alkalmas arra, hogy további fejlesztés alapját adja, akár ember-gép kapcsolatot grafikus felület létrehozása, vagy akár szerver szolgáltatások, például weboldal, adatbázis elérése céljából. Ezek az eszközök a következő kötetekben ismertetésre kerülő gyakorlati példákban az információ feldolgozó egységek szerepét fogják betölteni.

Felhasznált irodalom

- Horváth Péter: *A mechatronika alapjai*. Széchenyi István Egyetem, Győr, 2006.
- Forgó Zoltán: *Bevezetés a mechatronikába*, Erdélyi Múzeum-Egyesület, Kolozsvár, 2009
- Arduino projekt weboldala: www.arduino.cc (2015.02.01)
- Raspberry Pi projekt weboldala: www.raspberrypi.org (2015.02.01)
- TightVNC telepítési leírás: <http://www.penguintutor.com/linux/tightvnc> (2015.02.01)