

Interdiszciplináris műszaki gyakorlatok az informatikatanár szakon

Makan Gergely¹, Antal Dóra², Mingesz Róbert³, Gingl Zoltán⁴, Kopasz Katalin⁵,
Mellár János⁶, Vadai Gergely⁷

{¹makan,³mingesz,⁴gingl,⁶mellar,⁷vadaig}@inf.u-szeged.hu,
²antal74dora@gmail.com

Szegedi Tudományegyetem, Műszaki Informatika Tanszék

⁴kopaszka@titan.physx.u-szeged.hu

Szegedi Tudományegyetem, Optikai és Kvantumelektronikai Tanszék

SZTE Gyakorló Gimnázium és Általános Iskola

Absztrakt. A technikai fejlődésnek köszönhetően az eszközök egyre nagyobb része elektronikus, működésükért szoftverek felelősek. Mivel az eszközök a külvilág jeleit érzékelik, és ezek feldolgozásával végeznek beavatkozást, az informatika egyre inkább interdiszciplináris jellegűvé válik. A hétköznapi eszközök technológiája, az autópár, az Ipar 4.0, az IoT – Internet of Things, dolgok internete – az informatika mellett épít a műszaki, elektronikai, fizikai, biológiai és más területekre is. Ennek megfelelően fontos, hogy a képzésekben is megjelenjen ez az interdiszciplináris jelleg, a hallgatók alapvető ismeretekhez juthassanak kapcsolódó műszaki megoldásokban, aminek később, a tanári pályájuk során is hasznát vehetik. Fontos kiemelni, hogy bár az eszközök fejlődése továbbra is igen gyors, a működési elvek nem változnak, így az utóbbiak oktatása egyre fontosabb, szemben egy-egy adott fejlesztőkörnyezet, szoftver vagy hardver használatának tanításával. A fenti elvek követésével különböző területekhez tartozó gyakorlatokat dolgoztunk ki az informatikatanár-képzésünk támogatására. A laboratóriumi gyakorlatok során létrehozzák az áramköri kapcsolásokat, megírják a szoftvereket és gyakorolják a beágyazott szoftverfejlesztés alapvető módszereit is. Egy hétköznapi, egy iparhoz kötődő és egy orvosi méréshez és jelfeldolgozáshoz tartozó példán keresztül mutatjuk be, hogyan értik meg és tanulják meg a hallgatók a legfontosabb elveket. Az élményszerű tanulás jobban motiválja a hallgatókat, segíti a rendszerintű megértést, a kreativitás fejlesztését. Tapasztalataink alapján a hallgatók magabiztossága is erősödik a modern műszaki rendszerek használatakor, amit különösen fontosnak tartunk.

Kulcsszavak: műszaki informatika, Arduino, interdiszciplináris oktatás

1. Bevezetés

Az oktatás egyik legfontosabb kérdése az, hogy hogyan képes alkalmazkodni a gyors technikai fejlődéshez. Ehhez tartozik a mai, modern eszközeink használata és ismerete mellett az is, hogy a tanulók jelentősen megváltozott körülmények közt élnek, más módon jutnak ismeretekhez, így a hozzáállásuk is más. Komoly gondot okoz, hogy az iskolai közeg egyre idegenebb a tanulók számára, nem értik, mi szükség van számos tananyagrészt megtanulására. Ez a közgondolkodásban is egyre elterjedtebb szemlélet: az iskola olyat oktasson, amire a gyakorlatban szükség van. Szakmai körökben gyakori a vita, hogy szükséges-e ma is univerzálisabb érvényű ismereteket tanítani ahelyett, hogy közvetlenül a mai eszközök alkalmazását tanítanánk. Szükséges-e fejben számolni, a Pitagorasz-tételt ismerni, ráadásul bizonyítani, szükséges-e külön tárgyként fizikát, biológiát és más természettudományos tárgyat oktatni.

Az informatika területén különösen jól látszik a probléma: nem tudjuk megmondani, hogy pár év múlva milyen eszközök, programozási környezetek jelennek meg, mivel fognak dolgozni a végzetek egy informatikai cégnél. Így tehát a legjobbnak az látszik, ha olyan ismeretekhez jutnak a tanulók,

melyek alapján könnyen lesznek képesek alkalmazkodni a változó környezethez. Fontos, hogy megtanuljanak logikusan gondolkodni, hogy értsék a legfontosabb működési elveket, hiszen ezek változnak a legkevésbé. Mivel egyre több területen jelennek meg az informatikai megoldások (oktatás, ipar, gyógyászat, kommunikáció, navigáció, szórakoztatás, stb.), igen hasznos egy adott szintű interdiszciplináris tudás, nyitottság is.

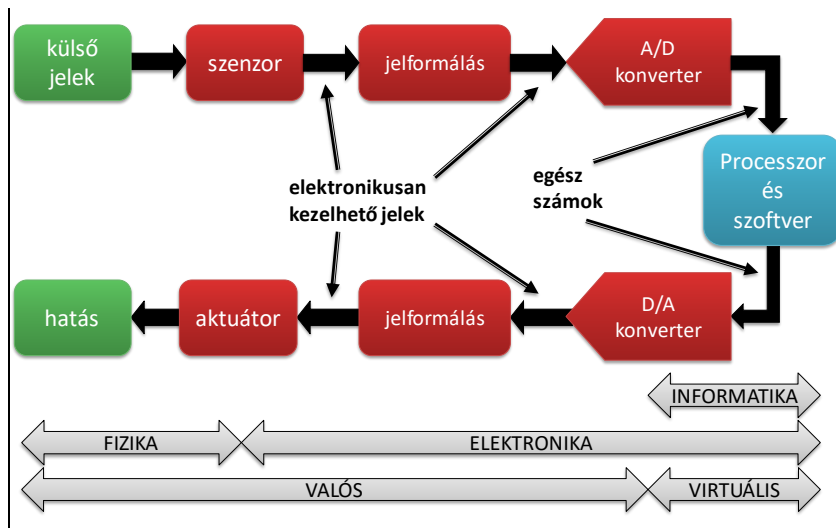
Sokan úgy látják, hogy reménytelen a modern eszközök működési elveinek megértése, egyre inkább "fekete dobozokként" használjuk eszközeinket: nem látunk bele a működés részleteibe. Ezek az eszközök sok esetben a legújabb eredményeket alkalmazzák, működésükben igen sok szakterülethez kötődnek. Ugyanakkor, ha a megoldások részletei helyett megpróbáljuk a háttérben meghúzódó elveket megtalálni, akkor megfelelő általánosításhoz juthatunk, ami sokak számára érthetővé válik, az eszközök működése sokszor a hétköznapi életben is észlelhető elveken alapul. Jól segíti ezt az is, hogy az oktatás számára is széles körben elérhetővé váltak univerzális építőelemek, melyekkel egészen komoly eszközöket lehet akár játékosan készíteni [1, 2, 3, 4].

Fontos tehát, hogy a tanárszakos hallgatók képzésük során igen alapos tudást szerezzenek és kialakuljon a megfelelő szemléletük. A tanár magas szintű és széleskörű szakmai képzettsége nélkülözhetetlen a mai oktatásban, amit nem pótolnak, csak hatékonyabbá tesznek az oktatási módszerek és megoldások.

Az oktatás fejlesztésével foglalkozó kutatócsoportunk [5] a fentieknek megfelelően alakította ki tevékenységét. Az informatikatanárok képzésében kurzusfejlesztéssel, oktatási anyagok, eszközök és módszerek kidolgozásával veszünk részt [6]. Súlyt helyezünk a gyakorlatias képzésre, a megfelelő elméleti tudás és általános elvek ismeretének átadására, melyek laboratóriumi gyakorlatokon mélyíthetők el.

2. Műszaki módszerek és eszközök az informatikatanár-képzésben

A mai eszközök túlnyomó többsége már elektronikus, processzort tartalmaz, szoftverek vezérlik a működését. Hamarosan lényegében az összes körülöttünk levő eszköz ilyen lesz, a mérettől és az alkalmazási területtől függetlenül. A felépítés és működési elv ugyanakkor viszonylag egyszerű és egységes, ahogy az 1. ábra is mutatja [részletesebb leírás: 6]. A külvilág jelei átalakulnak olyan elektronikával kezelhető jelekké, melyeket számokká konvertálhatunk és így processzorokkal kezelhetünk. A feldolgozás során információt nyerhetünk, ami felhasználható akár beavatkozásra is. Az ilyen felépítésű eszközöket beágyazott rendszereknek is nevezzük, a fő működtető egység, azaz a beépített processzor jelenléte miatt [7].



1. ábra: A modern elektronikus eszközök felépítésének blokkvázlata.

Több olyan oktatási eszköz kapható, melyeknél jól látható a fentebb részletezett felépítés, a tanárok és diákok maguk rakhatják össze az egyes univerzális építőelemeket, megtervezhetik a vezérlő algoritmust és elkészíthetik az ezt megvalósító programot.

Fontos kiemelni, hogy az eszközöktől (beágyazott rendszerektől) nagyfokú megbízhatóságot várunk el. Előzések, egy légszák kinyitások, betegőrző monitor működése közben nem keletkezhet szoftver- vagy hardverhiba, de egy mobilalkalmazástól is elvárjuk, hogy ne navigáljon szembe egyirányú utcában, megfelelő számlára utaljon, biztonságosan kezelje adatainkat. Sarkos példaként szokták említeni, hogy egy egész szám túlsordulása okozta egy rakéta katasztrófáját és dollár százmilliók füstté válását [8].

Már tanulókban fontos tehát az egyfajta igényes, műszaki szemlélet, hozzáállás kialakítása, amit a felsőoktatási képzési és kimeneti követelmények közvetlenül meg is jelölnek. Tanulságos megnézni, hogy milyen előírások érvényesek az informatikatanárok esetén. Ebből pár dolgot emelünk itt ki, melyek közvetlenebbül kötődnek a fentebb említettekhez [9]:

- „Ösztönzi a tanulók önálló véleményalkotását, törekszik a kritikus gondolkodásmód kialakítására, különös tekintettel az informatikai alkalmazás veszélyeinek figyelemfelhívására.”
- „Rendelkezik azokkal az ismeretekkel, amelyek lehetővé teszik, hogy szaktárgyának új eredményeit megismerhesse, értelmezhesse. Ismeri a szaktárgy alapvető kutatási módszertanát.”
- „Képes - elsősorban a természettudományokkal és a matematikával - a különböző szakterületek tudás- és ismeretanyaga közötti összefüggések felismerésére, integrációjára.”
- „Képes a szaktárgyában elsajátított elméleti ismeretek gyakorlati alkalmazására, ennek közvetítésére a tanuló felé.”
- „Tisztában van azzal, hogy a szaktárgyában közvetített tudás, kialakított kompetenciák más műveltségterületen is hatnak.”
- „Szakszerűen tudja használni az iskola informatikaoktatási eszközeit, bevonni oktatómunkájába az informatikai eszközöket, távoktatási anyagokat. Alkalmas informatikai tananyagfejlesztésre, más szakos tananyagfejlesztés informatikai megvalósításának támogatására.”
- „Együttműködik a szaktárgyával rokon tárgyak tanáraival. Képes arra, hogy a rokon tárgyakban is megjelenő, egymásra épülő ismeretanyagok ütemezését egyeztesse.”

- „Elkötelezett az igényes tanári munkára, a folyamatos önművelésre.”

A szakmai tudás előírása igen bőséges és szerepel benne közvetlenül műszaki terület, robotika is.

2.1. Népszerű műszaki informatikai oktatási eszközök

Egyre jobb, nagyobb tudású és egyre olcsóbb eszközök állnak rendelkezésre, melyekkel jól segíthető egy mai eszköz felépítésének, működési elveinek oktatása játékos, élvezetes, gyakorlatias módon. Ilyenek a Lego robotok [1], az egylapos számítógépek is, a micro:bit [2], az Arduino [3] és a Raspberry Pi [4]. Ezekhez sokféle szenzor köthető, számos különféle jelet lehet velük mérni, melyek adataihoz közvetlenül hozzáférhetnek a tanulók, műveleteket végezhetnek rajtuk, különféle jeleket, hatásokat állíthatnak elő. Ennek megfelelően tehát számottevő átláthatóságok biztosítanak, a szenzorok fontossága és szerepe kiderül anélkül is, hogy az alkalmazott elektronikai kapcsolásokat ismerni kellene. Természetesen sokféle szintje van az ismereteknek, a tanároknak alaposabb műveltségre van szükségük, nem jöhetnek zavarba, ha például egy diák megkérdezi, hogyan képes egy robot érzékelni a közelében levő tárgyakat.

„Az informatika műszaki alkalmazásai” tantárgy oktatása során az alapvető elméleti és gyakorlati ismereteket egy félévben 15 illetve 30 órában tanítjuk informatikatanár-szakos hallgatóknak [10]. A képzés a robotika, a Raspberry Pi mellett jelentős mértékben épít a világszerte is rendkívül népszerű Arduino platformra.

2.2. Arduino alkalmazások – előnyök és hátrányok

Az Arduino egy egylapos számítógépre és programozási környezetre épülő platform. Rendkívül átlátható, gyakorlatilag egy mikrovezérlő (processzort és perifériákat tartalmazó csip) kivezetései vannak kényelmesebben használható csatlakozókra kötve. Sokféle kiegészítő áramkör, szenzor kapható hozzá, így a diákok közvetlenebbül találkozhatnak elektronikával, kapcsolásokkal. Ezek mellett a programozást is nagyon jó arányérzékkel oldották meg a fejlesztők. Az egyik leggyakrabban alkalmazott programozási nyelv a C++, az Arduino használatához azonban a legtöbb esetben elegendő a szinte minden nyelvben előforduló alapok elsajátítása. A környezet egyszerűsége törekszik, kevés, de jól használható függvényt, építőelemet biztosít a hardver kezeléséhez és a programok felépítése is jól megfelel az operációs rendszer nélkül működő mikrovezérlők általános alkalmazási szokásainak (a main helyett setup és loop függvények). A mikrovezérlő egy fogyasztási, ipari, gyógyászati és más területeken is használt professzionális műszaki komponens, alapvető szerepű a beágyazott rendszerekben, ahol kiemelten fontos a megbízhatóság, a kapcsolódó elvek, szabványok követése is. Természetesen az oktatás során nem lehet ezek maradéktalan követését elvárni, de el kell kerülni a helytelen berögződéseket, a nem kellően tudatos, szakszerűtlen használatot.

Az interneten szinte mindenféle feladathoz található megoldás, letölthető forráskóddal, az összeállítás részletező receptekkel, melyeket szakemberek mellett diákok, hobbisták adnak meg. Sok jó ötlettel lehet találkozni, de az előnyök mellett sajnos komoly hátrányok is mutatkoznak. Egyrészt ez arra ösztönözheti a tanulókat, hogy saját munka helyett inkább reprodukáljanak, másoljanak, sokszor nem is értik, miért kéri őket a tanárok olyan feladatok megoldására, amelyeket könnyen meg lehet találni. Másrészt a nem elég alapos tudás miatt rengeteg látszólag jól működő, műszaki szempontból mégis hibás megoldás terjed. Az egyetlen probléma esetén is rendkívül sok látalattól ráadásul igen nehéz lehet megtalálni a megfelelőt még képzetesebb tanárok számára is. Sajnos még az Arduino hivatalos oldalán, szakkönyvekben is találhatók helytelen iránymutatások, melyek később nehezen alakítható rossz beidegződéseket okozhatnak.

Fontos feladatunknak tartjuk ezért a működés és elvek minél jobb megértését, a szakmailag korrekt alkalmazások segítségét, az önállóságra és gondosságra nevelést. Nem célunk receptszerű megoldások megadása, az esettanulmányokat példának szánjuk, melyekben a megfelelő módszerek alkalmazását és a helyes hozzáállást szemléltetjük [6]. A laboratóriumi gyakorlatokon élményszerű tanulást jelenthetnek a beágyazott eszköz és szoftverfejlesztési elemek, segíthetik a magabiztosság kialakulását

az egyes problémákkal való szembesülés és azok megoldása útján. A képzési és kimeneti követelményekkel is összhangban fejlődhet az igényes gondolkodásmód és hozzáállás is, aminek főbb elemei közé tartozik a feladat gondos mérlegelése, megfelelő megoldás megadása, az ismert elvek és módszerek helyes, szakszerű alkalmazása, a megoldás elkészítése, tesztelése, javítása is.

3. Arduino gyakorlatok

A laboratóriumi gyakorlatok során többféle területet érintő feladatokat oldanak meg az informatikatanár hallgatók. Ennek megfelelően többféle szennorral, megoldással, szakterülettel és szemlélettel is találkozhatnak. A munka során tűz- és munkavédelmi ismereteket szereznek, a feladataik elvégzéséről jegyzőkönyvet készítenek, gyakorolják a közös munkát is. Az elkészült programokat és jegyzőkönyveket az oktató értékeli. Az előadás és gyakorlati órák 1:2 aránya jól szolgálja a gyakorlatorientáltságot.

A hallgatók szabadon választhatják meg a megoldást, használhatják ehhez az interneten elérhető segédanyagokat is. Olyan feladatokat kapnak, melyekhez szükség van a személyes problémamegoldó képességeikre is. Néhány feladatra kezdetben egyszerűbb, elterjedtebb megoldást adhatnak, majd ezek után kerülhet sor a szakmailag igényesebb kidolgozásra, melyhez az oktató útmutatást és magyarázatot adhat.

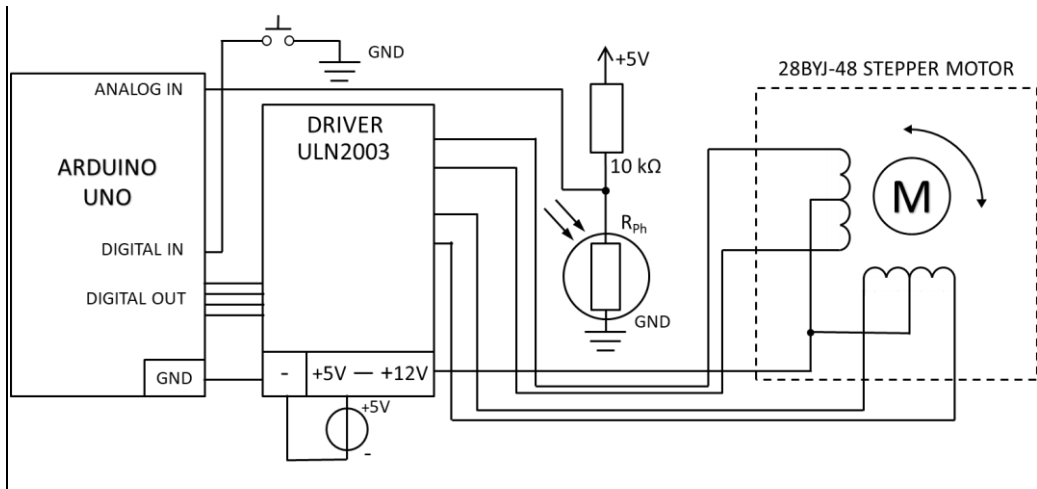
Fontos kiemelni azt is, hogy tapasztalatokat szereznek arról, hogy kell egy laboratóriumi/műszaki környezetben dolgozni, milyen főbb szabályokat, előírásokat kell betartani a megbízhatóság és minőség érdekében, függetlenül akár attól is, hogy ezek céljai számukra nem feltétlen egyértelműek.

A következő példák azt is szemléltetik, hogy a megoldásokhoz alkalmazni kell bizonyos matematikai, fizikai és akár más szakterületi ismereteket is, melyek szükségességére máshogy nem feltétlen gondoltak volna a hallgatók.

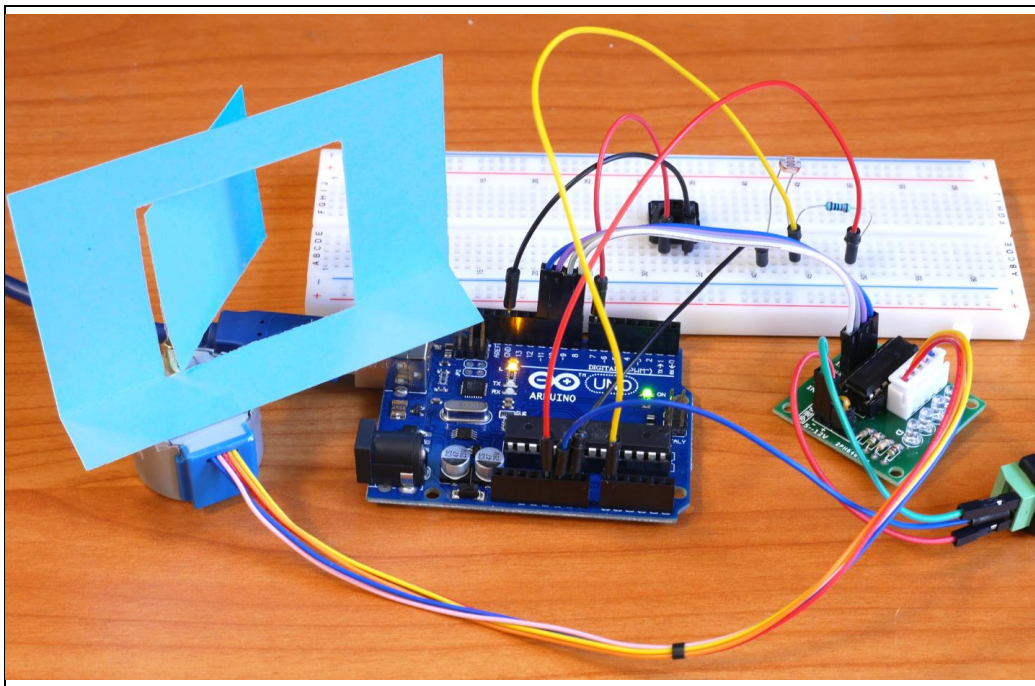
3.1. Kapuvezérlő rendszer készítése

Igen hasznosak az olyan gyakorlatok, melyek egy hétköznapi eszköz működését modellezik. Ez közvetlenül mutatja a gyakorlati hasznosságot és a mögöttes elveket is. Ilyen feladat egy garázkapu-mozgató berendezés makettjének összeállítása és a működtető szoftver megírása.

A munka során egy léptetőmotor mozgatását kell megoldani, amihez a megfelelő algoritmust ki kell dolgozni az elméleti alapismeretek alapján. A motor forgása egy kapu mozgatásának felel meg, a nyitás és zárás 90 fokos elfordulásokat jelent. Jó példa az elvi elvonatkoztatási lehetőségekre is a rendszer, ugyanis a valóságban nem léptetőmotort alkalmaznak ilyen célokra, ugyanakkor ez csak egy részegység közvetlen kezelését jelenti, a főbb elveket nem érinti. A feladat része az is, hogy egy optikai szenzor segítségével vészleállítás történjen, azaz ha a kapu útjába akadály kerül, akkor a mozgás álljon le. Erre a célra egy fotoellenállást kapnak a hallgatók, amit az Arduino áramkörhöz kell illeszteniük és az érzékeléshez szükséges kódrészletet is meg kell írniuk. A kapunyitás egy digitális bementre kötött nyomógomb segítségével indítható.



2. ábra: A kapuvezérlő rendszer felépítése. Az Arduino a meghajtó áramkörrel vezérli a léptetőmotort. A kapu nyitása gombnyomásra indul, valamint a mozgás szünetel, amíg a fényérzékelő akadályt észlel.



3. ábra: A képen az összeállított kapuvezérlő rendszer látható. A kapu egyszerű makettje kartonlapból van kialakítva.

A feladat változatos lehetőségeket ad az egyes részesegységek működtetésének megoldására, számos elv alkalmazására. A motor vezérlésében használható fél vagy egész lépés, szabályozhatóvá tehető a lépési sebesség, detektálhatók a végállások. Fotoellenállás helyett használható fototranzisztor, látható helyett infravörös fény. A nyitás vezérelhető távirányítóval is, a működési logika is sokféleképp variálható. Ki

lehet egészíteni a feladatot a motorok sebességének változtatásával a végpontok körül vagy egy további LED-el, ami villog, ha mozgásban van a kapu. A hallgatók az adott számú léptetést tipikusan for ciklussal oldják meg, mely helyett használható maga a loop függvény is megfelelő feltételvizsgálattal. A lépések időzítését meg lehet oldani precízebben is a timer periféria használatával. Ezek nagy szabadságot adnak a nehézségi szint megválasztásában, a különböző tempóban haladó hallgatók megfelelő terhelésében is.

Tapasztalataink alapján a feladat megoldása során a hallgatóknak leginkább a következők jelentettek nehézséget: a léptetőmotor áttételének megértése és a mozgató szüneteltetése arra az időre, amíg nincs elindítva a nyitás, és amikor akadály került az érzékelő elé. A léptetőmotor működésében a mechanikai felépítés és mozgatósi elv hathatott újdonságként. A mikrovezérlő programozása operációs rendszer nélküli környezetben történik, ami szintén szokatlan a hallgatók számára. A léptetőmotor működésének megértésében több előzetes segédanyag, szimuláció lehet segítség [11, 12], míg a mozgató megállításának ésszerű megoldásához a mikrovezérlőprogramozásban alkalmazott végtelen ciklus (az Arduino környezetben a loop függvény) szükségességének alaposabb elmagyarázása illetve több példa segíthet.

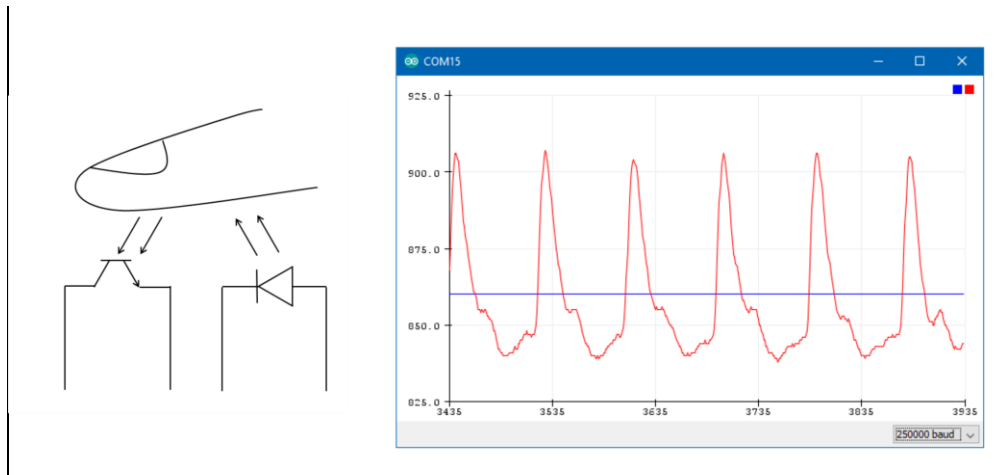
3.2 Fotopletizmográf alapú szívritmusmérés

Ma már okostelefonok és okosórák is képesek szívritmusmérésre, melynek egyik egyszerű alapelve az úgynevezett fotopletizmográfia. Egy izgalmas, élményszerű gyakorlat építhető fel ennek megvalósítására.

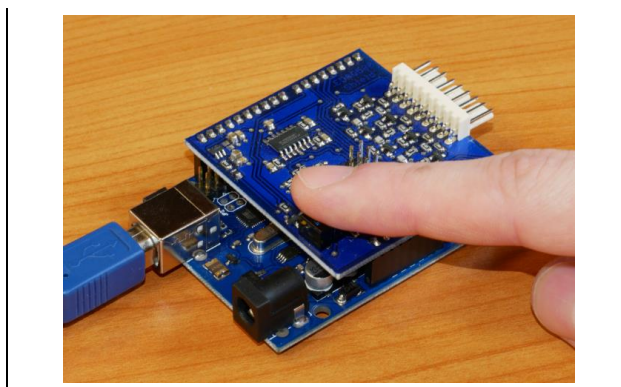
A mérési elv egyszerű: az ujjba infravörös fényt bocsátunk, majd mérjük a visszavert vagy átjutó fényt, aminek erőssége függ az ujj pillanatnyi vértelítettségétől [13]. Ennek mérésével lényegében a vérnyomással arányos időfüggő jelet kaphatunk, amit felhasználhatunk a szív működés megfigyelésére is. A jel változó komponense igen kicsi, amely egy sokkal nagyobb átlagértékhez adódik [14]. A hallgatók megismerik, hogyan lehet a változókomponenst megfelelő tartományba erősíteni, mely így közvetlenül az Arduino áramkör analóg bemenetére köthető és digitalizálható. A megoldási elv igen általános, sok más (pl. mikrofon) jel kezelésében is elterjedten alkalmazzák.

A hallgatók a feladat során az általunk fejlesztett EDAQ530-as szenzorinterfész [15] Arduino alapú megvalósítását használják. Az EDAQino nevű Arduino „shielden” [6, 16, 17] többek között található egy infra fényű fotodetektor a pletizmográfiahoz szükséges előerősítéssel és szűréssel. Az áramkör fotopletizmográf részét akár próbapanelen össze is lehet állítani, aminek során fontos áramkörépítési alapismeretekre tehetnek szert a hallgatók. Ezt követően mintavételezéses mérést kell végezni, ami igen egyszerűen is megoldható, de gondosabb megoldások esetén számos megfontolást igényel. Az Arduino integrált környezet valós idejű grafikus megjelenítést biztosít az úgynevezett „serial plotter” segítségével, így a hallgatóknak hamar lehet komoly sikerélményük a saját szív működésük grafikus szemléltetésével.

Valós idejű jelfeldolgozással lehetséges egy LED-et a szív működésnek megfelelő ritmusban kapcsolni, a megfelelő detektáló algoritmus megvalósításával az egyes szívdobbanások közti időtartamot meg lehet mérni és meg lehet jeleníteni. A feladat jelentősen bővíthető az orvosi gyakorlatban is használatos szívritmus-variabilitás (heart rate variability, HRV) mutatók kiszámításával. Ez tulajdonképpen az egymást követő szívütések között eltelt idő ingadozásának statisztikai analízisét jelenti. Ezek közé tartozik a szívütések közti idő (RR) átlagának (Mean RR), szórásának (SD RR), valamint a pNN50 nevű indikátornak a meghatározása, ami százalékosan azt adja meg, hogy az előző RR értéktől hány szívütés tér el 50 ms-nál jobban [18, 19]. Továbbá az RR értékekből hisztogram is számolható.



4. ábra: Az ábra bal oldalán a fotopletizmográf működési elve látható, a jobb oldalán pedig a fototranzisztor által mért jel a „serial plotter”-ben megjelenítve.



5. ábra: A pulzus mérésére az általunk fejlesztett EDAQuino nevű Arduino shieldet használják a hallgatók.

A tapasztalataink alapján a feladatnál a szintmérés-detektálás algoritmusának megtalálása okozott kisebbfajta nehézséget a hallgatóknak, amit indokol, hogy digitális jelfeldolgozással még nem találkozottak. Ez rávilágít arra, hogy milyen fajta jelfeldolgozási alapokat érdemes a tananyag részévé tenni. Felmerül annak a lehetősége is, hogy a programozási alapkurzusokon olyan jellegű gyakorló feladatokat kapjanak a hallgatók, melyek hasonló algoritmusokkal oldhatók meg.

3.3 Hőmérséklet szabályzás

Az egyik legegyszerűbb és legegyszerűbb szabályozási elv az úgynevezett on-off szabályzás, mely számos hétköznapi eszköz működésében is kulcsszerepű. A szabályozások általános módszere szerint mérni kell a kívánt értéktől való eltérést, és ennek megfelelően növelő vagy csökkentő hatást kell gyakorolni. Az on-off szabályzás esetén a hatás nagysága előre adott, nem függ a kívánt értéktől való eltérés nagyságától.

A szabályzás megvalósításához egy teljesítményellenállást használnak a hallgatók, melyre feszültséget kapcsolva fűteni lehet. A fűtőteljesítmény a következőképpen függ a fűtőfeszültségtől:

$$P = \frac{U^2}{R}, \quad (1)$$

ahol U a fűtőfeszültség, R a fűtőellenállás értéke.

Ez jelentős áramot igényelhet, amit az Arduino digitális kimeneteivel vezérelt meghajtó áramkörrel vagy egyszerűen az egyik digitális kimenetére megfelelően kötött tranzisztorral biztosíthatunk. A hűtést egy kisméretű ventilátorral lehet megoldani. Az ellenállás aktuális hőmérsékletét a hallgatók egy termisztorral [20] (hőmérsékletfüggő ellenállással) mérik meg. A hőmérsékletet a termisztor képlete segítségével számíthatjuk ki:

$$T = \frac{1}{\frac{1}{T_{25}} + \frac{1}{B} \cdot \ln\left(\frac{R}{R_{25}}\right)}, \quad (2)$$

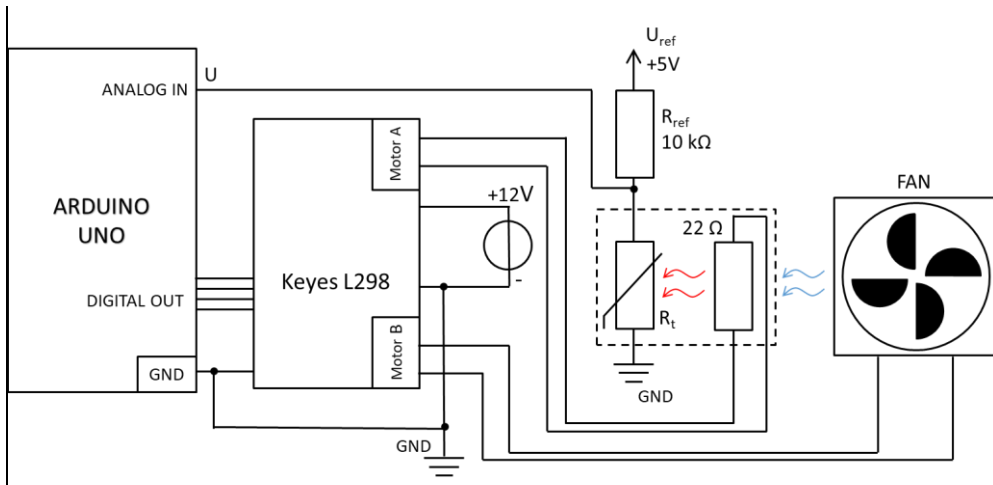
ahol T a hőmérséklet kelvinben, T_{25} a szobahőmérséklet szintén Kelvinben ($25\text{ °C} = 298,15\text{ K}$), a B az adott termisztorra jellemző konstans, R_{25} pedig a termisztor ellenállása 25 °C -on. Mivel az analóg-digitál konverterrel csak feszültséget lehet mérni, ezért szükség van egy olyan kapcsolásra, ami az ellenállásmérést feszültségmérésre vezet vissza. Ezt legegyszerűbben egy feszültségosztóval tehetjük meg. A feszültségosztó bemenete az A/D konverter referenciafeszültsége (U_{ref}), a kimeneti feszültséget (U) mérve kifejezhetjük a termisztor ellenállását (R_t):

$$R_t = R_{ref} \frac{U}{U_{ref} - U} \quad (3)$$

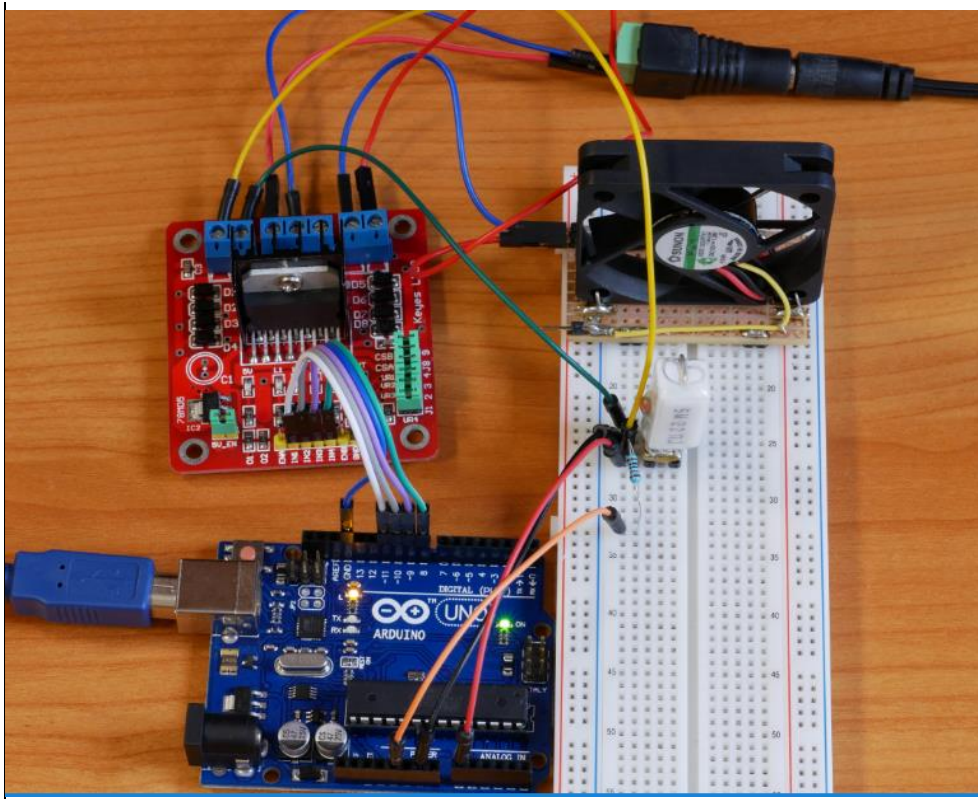
A feladat egy olyan program megírása, ami az ellenállást egy adott hőmérsékletre fűti fel, majd ezen az értéken tartja. A ki- és bekapcsolási küszöbszinteket (hiszterézist) be kell állítani, ezek hatását a hőmérséklet valós idejű grafikus megjelenítésével (serial plotter) lehet megfigyelhetővé tenni.

Pusztán a program módosításával pulzusszélesség-modulációs (PWM) módban, azaz változtatható erősséggel is vezérelhető a fűtés és hűtés, így általános elvet láthatnak a hallgatók arra, hogy kétállapotú jelekkel hogyan lehet sokféle szintet létrehozni.

A kísérlet könnyen átvihető más területekre is, alkalmazható például egy kondenzátor feszültségének adott szinten tartására, melyet egy ellenálláson keresztül lehet tölteni és kisütni. Ehhez elegendő az Arduino kimeneti árama is, így az ellenálláson és kondenzátoron kívül nincs szüksége semmilyen más alkatrészre. A pillanatnyi feszültség A/D konverterrel mérhető.

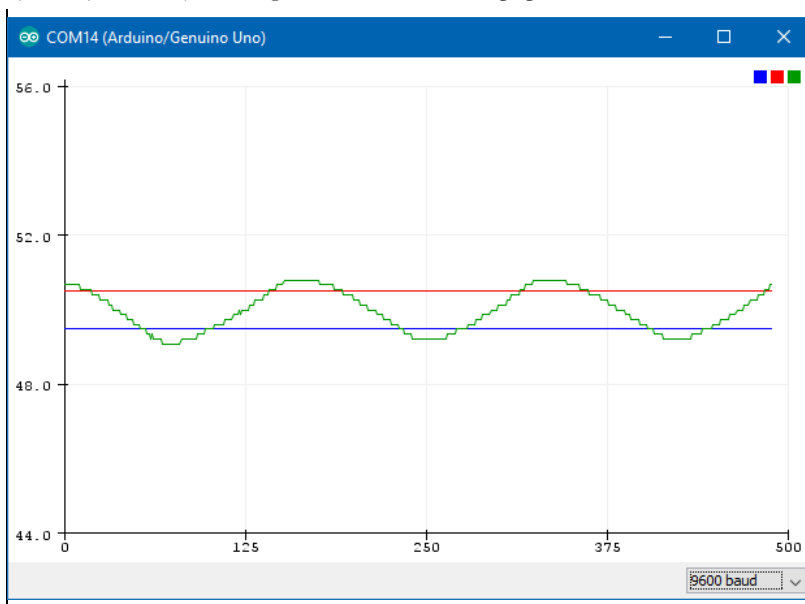


7. ábra: A hőmérséklet szabályzási feladat diagrammja. Az Arduino egy meghajtó áramkör segítségével tudja ki-, bekapcsolni a fűtést és a hűtést is.



8. ábra: A képen a próbapanelen összeállított hőmérséklet szabályzás látható. Az Arduino csak a meghajtó áramkör segítségével tud elegendő áramot kiadni a fűtés és a hűtés számára.

Ennél a feladatnál azt tapasztaltuk, hogy a hiszterézis (a két küszöbszint) megvalósítása okozott nehézséget, illetve nehezen volt érthető számukra, hogy a hőmérséklet miért nem marad a küszöbszinteken belül (9. ábra). Ennek jobb megértését akár hétköznapi példák említésével is lehet segíteni.



9. ábra: A szabályzott hőmérséklet időfüggése. A fűtés be- és kikapcsolási küszöbszintjeit (49,5 °C és 50,5 °C) kék és piros vonalak jelzik.

A hőmérséklet képletének kódolásánál gyakran futottak bele a hallgatók egész osztási problémákba. Nem mindig vették figyelembe, hogy C és C++ nyelvekben két egész szám osztásakor az eredmény is egész és ez igen nagy hibát okozhat, könnyen lehet az osztás eredménye nulla is. A mikrovezérlő-programozás során gyakran előforduló hibákra, a fontosabb programozási elvek ismertetésére a bevezető órákon érdemes kitérni.

Mivel a hallgatóink az egyetemi tanulmányaik során most először találkoztak az elektronika alkalmazásával, ezért az egyszerű kapcsolások összeállítását és a próbapanel használatát is meg kellett ismerniük.

4. Összefoglalás

Az informatikatanárok képzésének fontos része az egyre elterjedtebb modern, processzorokat tartalmazó, szoftvereket futtató eszközök működési elveinek alapvető megértése. Az informatikai, műszaki megoldások egyre nagyobb szerephez jutnak más területeken is, így az interdiszciplináris jelleg is erősödik. Az eszközök, szoftverek igen gyors fejlődése miatt alapvető az univerzálisabb elvek megismerése, alkotó alkalmazása, melyet az elméleti oktatás mellett laboratóriumi gyakorlatokon lehet a leghatékonyabban elmélyíteni. A hallgatók az élményszerűbb gyakorlatias tanulás révén magabiztosabbá, igényesebbé válhatnak, számos szakterület módszereit ismerhetik meg, jelentősen fejlődik a problémamegoldó képességük is.

Bemutattunk három, Arduino platformmal elvégezhető laboratóriumi gyakorlatot, melyet a tanárszakos képzésünkben használtunk. A tapasztalatok alapján a hallgatók sikeresen el tudják végezni a feladatokat és élvezetesnek találják a munkát. A feladatok igen sokrétűen bővíthetők, alkalmazhatók középiskolai és egyetemi környezetben, különböző nehézségi szinteken. Az Arduino áramkörök és

kiegészítőik olcsón hozzáférhetőek, így tantermek kis ráfordítással felszerelhetőek, akár minden diák számára juthat külön készlet. A diákokok otthoni használatra is beszerezhetik az eszközöket, ami a gyakorlati lehetőség mellett segíti saját ötleteik megvalósítását is.

Köszönetnyilvánítás

A tanulmány elkészítését a Magyar Tudományos Akadémia Tantárgypedagógiai Kutatási Programja támogatta.

Irodalom

1. *Lego robotok információs oldalai*
<https://www.lego.com/en-us/mindstorms>, (utoljára megtekintve: 2018.10.24.)
2. *Micro:bit információs oldalak*
<https://microbit.org/hu/> (utoljára megtekintve: 2018.10.24.)
3. *Arduino információs oldalak*
<https://www.arduino.cc/>, (utoljára megtekintve: 2018.10.24.)
4. *Raspberry Pi információs oldalak*
<https://www.raspberrypi.org/>, (utoljára megtekintve: 2018.10.24.)
5. Az MTA-SZTE Műszaki Informatika Szakmódszertani Kutatócsoport honlapja
<http://www.inf.u-szeged.hu/miszak/> (utoljára megtekintve: 2018. 10. 24.)
6. Gingl Zoltán, Kopasz Katalin, Makan Gergely, Mingesz Róbert, Mellár János, Szépe Tamás, Vadai Gergely, *Műszaki informatikai megoldások a modern középiskolai oktatásban*, INFODIDACT 2017 konferencia, Zamárdi, 2017. november 23-25. (2017)
7. *Beágyazott rendszer*
https://en.wikipedia.org/wiki/Embedded_system
8. *Rakétakatasztrófa*
<https://hownot2code.com/2016/09/02/a-space-error-370-million-for-an-integer-overflow/>
9. *KKK Informatika műveltségi terület*
<https://net.jogtar.hu/jogszabaly?docid=a1300008.emm>
10. *Az informatika műszaki alkalmazásai” tantárgy oktatása*
<http://www.noise.physx.u-szeged.hu/Education/IMA/>
11. *Mechatronika jegyzet*
https://www.tankonyvtar.hu/hu/tartalom/tamop412A/2011_0104_SZTE-2_Mechatronika/adatok.html, (utoljára megtekintve: 2018.10.28.)
12. *Léptetőmotor*
https://en.wikipedia.org/wiki/Stepper_motor, (utoljára megtekintve: 2018.10.28.)
13. Nagy, Tamás, and Zoltán Gingl., *Low-cost photoplethysmograph solutions using the Raspberry Pi*, Computational Intelligence and Informatics, CINTI 2013, IEEE 14th International Symposium on. IEEE Budapest, Hungary (2013), pp. 163-167.
14. Zoltán Gingl, *A photoplethysmograph experiment for microcontroller labs*, INTERNATIONAL JOURNAL OF ELECTRICAL ENGINEERING EDUCATION 49:(1) pp. 42-60. (2012)
15. Katalin Kopasz, Péter Makra and Zoltán Gingl, *Edaq530: a transparent, open-end and open-source measurement solution in natural science education*, Eur. J. Phys., (2011), **32** 491
16. Zoltán Gingl, János Mellár, Tamás Szépe, Gergely Makan, Róbert Mingesz, Gergely Vadai, Katalin Kopasz, *Universal Arduino-based experimenting system to support teaching of natural sciences* pp. 1-2. Spanyolország Extended abstract, GIREP-MPTL 2018 - Research and Innovation in Physics education: two sides of the same coin. 9th-13th July 2018, Donostia-San Sebastian, Spain (2018)
17. *EDAQuino*
<http://www.inf.u-szeged.hu/miszak/projektjeink/edaquino>, (utoljára megtekintve: 2018.10.28.)

18. *McKinley P S, Shapiro P A, Bagiella E, Myers M M, Meersman R E D, Grant I and Sloan R P 2003 Deriving heart period variability from blood pressure waveforms Journal of Applied Physiology 95 1431–8*
19. *Acharya U R, Joseph K P, Kannathal N, Lim C M and Suri J S 2006 Heart rate variability: a review Med Bio Eng Comput 44 1031–51*
20. *Termisztor*
<https://en.wikipedia.org/wiki/Thermistor>, (utoljára megtekintve: 2018.10.28.)