

Saját fejlesztésű vezérlőrendszer Open-Source alapokon

Self-developed controller system – based on Open-Source applications

T.I. ERDEI¹, ZS. MOLNÁR², G. HUSI³

¹Debreceni Egyetem Villamosmérnöki és Mechatronikai Tanszék, timoteierdei@gmail.com

²Debreceni Egyetem Villamosmérnöki és Mechatronikai Tanszék, zsoltt.molnar94@gmail.com

³Debreceni Egyetem Villamosmérnöki és Mechatronikai Tanszék, husigeza@eng.unideb.hu

Absztrakt: A Debreceni Egyetem Épületmechanikai kutatóközpontjában, a központban a legújabb épületautomatizálási eszközök kaptak helyett az oktatás színvonalának növelése érdekében. Az általunk kifejlesztett rendszer egy egyedülálló beágyazott rendszer, amely újrakonfigurálható és szabályozható az adott feladatok ellátása végett. A vezérlő rendszer bizonyítottan alkalmas lehet ipari gyártósori rendszerekbe való integrálásra.

Kulcsszavak: beágyazott rendszer; ipari gyártósor; vezérlő rendszerek; Raspberry Pi; Linux, Timotei-Robotics

Abstract: In the University of Debrecen Building Mechatronics Research Centre faced by the application of building automation technology, are tackled by cutting edge research performed at the Research Centre. The embedded device system, the focus of the ongoing project at the Research Centre, is programmable, reconfigurable, and adjustable. The design of an efficient embedded system must meet regulatory requirements, for optimizing building energy efficiency. The system provides solutions for a range of industrial applications, and peaks the interest of building services providers (e.g. SMEs).

Keywords: embedded system; production line; controller systems; Raspberry Pi; Linux, Timotei-Robotics

Bevezető

Napjainkban egyre nagyobb igény mutatkozik a megújuló energiaforrások használatára. A jövőben kulcsfontosságú szerepet fog játszani az energia felhasználás optimalizációja. A Debreceni Egyetem Villamosmérnöki és Mechatronikai szakán, az épületmechanikások ebben játszanak fontos szerepet. Kiemelten fontos, hogy az adott épület szabályzásához a legköltséghatékonyabb rendszereket valósítsuk meg, optimalizált programok megírása mellett.

Az épületmechanikai kutatóközpont az évek során több nemzetközi kapcsolatot alakított ki más országok egyetemével, példának okáért a Japán Tokyo-i Chou Egyetemmel. A nemzetközi kapcsolatok

révén, sok kutatásba sikerült bekapcsolódni és ezzel egyidejűleg egy egységes tudás bázist létrehozni, amely a későbbi hallgatók vagy PhD-sok munkáját könnyítheti meg [1].

A mechatronikának egyik meghatározója a robotika, amelyben rengeteg előrelépés történt az elmúlt években. A Villamosmérnöki és Mechatronikai Tanszék bevonásával, 3D nyomtatók kerültek megépítésre, amelyekkel lehetőségünk nyílt robotok vázszerkezeteinek vagy más alkatrészeinek a gyártására. Az oktatásban rendkívül fontos szerepet tölt be a 3D nyomtató jelenléte, hiszen a CAD programokban előzetesen megtervezett 3D modellek nyomtatásra kerülhetnek. Továbbá a Mestersége Intelligencia „AI” kutatásában is részt vesz, amiben a ROS (Robot Operation System) használjuk fel olyan algoritmusok írására, amik képesek komplex utasításokat-feladatokat végrehajtani.

A soron következő cikkben az alábbi taglalásban kerülnek tárgyalásra a vezérlő rendszerünk programozása és működtetése: 1. Tudományos Linux disztribúció. 2. TiMo programozható beágyazott rendszer bemutatása. 3. A vezérlő programozásának leírása. 4. Otthoni és Ipari automatizálás. 5. Összegzés. 6. Köszönetnyilvánítás és 7. Hivatkozások.

1. Tudományos Linux Disztribúció

A Debreceni Egyetem Villamosmérnöki és Mechatronikai Tanszéke, a fejlesztéseinkhez biztosította az infrastruktúrát. Az egyik legjelentősebb projectünk a Smart Home témakörében volt, amiben egy, az egyetem tulajdonában lévő légcserélő hűszivattyút vezéreltünk a National Instruments által gyártott sbRio 9606 FPGA eszközzel [3]. Az FPGA beintegrálásával az Intelligens Épületünkbe lehetőségünk nyílt a fogyasztói szokások mérésére, ennek megfelelően a vezérlés újraprogramozására, mindezt valós időben [4].

A tervezés legelső fázisa egy saját Linux disztribúció megalkotása volt Debian-Ubuntu alapon. A Linux egy Open-Source és UNIX szerű professzionális rendszer egyéni felhasználók és szuper számítógépek részére [5].

A mi Linux disztribúciós rendszerünk leegyszerűsíti a robot programozási feladatokat flexibilitása révén. A legfontosabb aspektusa a disztribúciónak, hogy megtalálhatóak rajta olyan előre telepített alkalmazások, mint a ROS Os, Arduino Integrated Development Environment (IDE), KiCAD, Blender. Továbbá fontos a rendszer változásainak pontos monitorozása, mivel mind a robotvezérléseknek, mind pedig a szimulációknak eltérőek a lefutási sebességei és a felhasznált erőforrás igényei. A rendszernek köszönhetően képesek vagyunk valós időben mérni és menteni az adott számítógép processzor használatának szintjét, RAM és swap memória értékét, hőmérséklet változását a CPU és HDD -nak és a WiFi-ethernet forgalmat, hálózaton történő robotvezérlések alkalmával. Ezen mérési adatok birtokában, pontos képet kapunk arról, hogy az egyes szimulációk vagy komplexebb vezérlések, milyen erőforrás igényűek és segítséget nyújt abban, hogy milyen irányba fejlesszük tovább a Linux alapú disztribúciónkat.



1. ábra: Debian-Ubuntu alapú Linux disztribúciós rendszer
(Forrás: tervezők által létrehozott tartalom)

A Timotei-Robotics Pre-Realise Linux disztribúciós rendszer követelménye:

- 700 MHz x32 or x64 CPU
- 512 MB RAM
- 8 GB HDD/SSD szabad hely
- Minimum felbontás: 1024x786

A teszt során a relatíve erős netbook-ot használtunk, aminek egy Intel DualCore N2840 CPU van beépítve, 2.16GHz órajellel és 1MB L2 cache memóriával. Valamint 4GB DDR3 RAM-al van ellátva, aminek frekvenciája 1600 MHz. A megjelenítésért egy Intel HD integrált VGA felel és háttértárolónak egy 128GB SSD kapott helyett [6].

2. TiMo Programozható Beágyazott Rendszer

Minden olyan rendszert, ami specifikáltan egy feladat ellátására alkalmas beágyazott rendszernek nevezzük. Gyakran kerülnek alkalmazásra mivel üzembiztos megoldást jelentenek. Beágyazott rendszerekkel találkozunk hibrid járművekben, forgalom irányításnál és az ipar által használt vezérlőknél. A modern beágyazott rendszerek olcsó mikrokontrollereken alapulnak, éppúgy ahogy a mi vezérlő panelünk is. Általánosan elmondható, hogy a beágyazott rendszerek, nem rendelkeznek újraprogramozható képeséggel, emiatt felhasználhatóságuk limitált [7].

Mi egy olyan beágyazott rendszert akartunk létrehozni, ami rendelkezik az újrakonfigurálás képességével, így az alkalmazhatósági körét jelentősen kibővíthetjük. Az újraprogramozhatóság előnye komoly gazdasági és termelési javulást von maga után, hiszen elég mindössze egy darab panel egy adott feladatra, nincs szükség tovább panelek beszerzésére, amik többletköltséget eredményeznek. Ez

egy költséghatékony megoldásnak bizonyulhat az otthoni és bizonyos ipari automatizálási folyamatokban.

A projektben egy Raspberry Pi B+ modell került felhasználásra, mivel közismerten olcsó és kisméretű, emellett automatizálási és oktatási célokra használják. A B+ Raspberry Pi modell, rendelkezik egy Broadcom BCM2835 ARM11 700MHz CPU -val, 512 SDRAM, microSD kártya socket-el, 5V micro USB, CSI camera konnekttorral, HDMI port-al és quad USB portokkal [8].

A rendszer specifikációi alkalmasak Linux alapú disztribúciók futtatására, úgymint Raspbian, Ubuntu Mate/Snappy Core vagy Pidora [9].

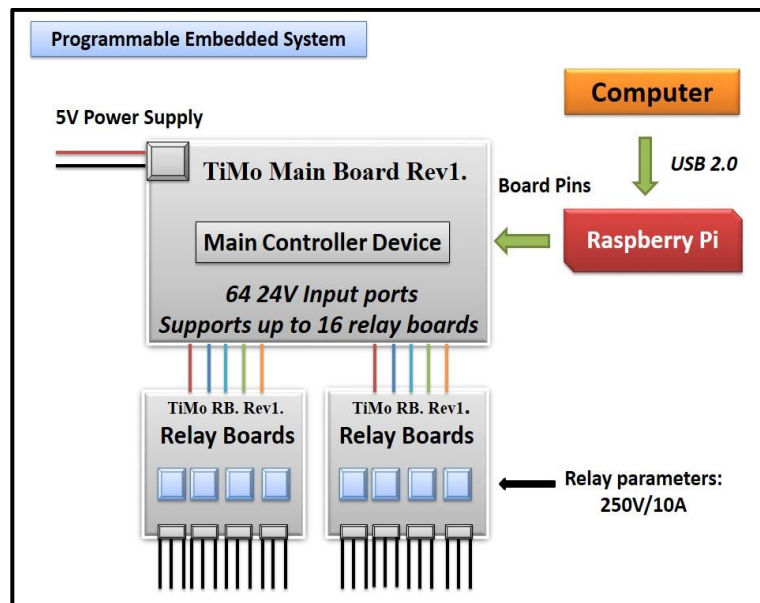
A rendszertelepítés és konfigurálás relatíve komplikált folyamat. Először a Timotei-Robotics Linux rendszer kerül telepítésre a notebook-ra. Továbbá a Raspberry Pi B+ -ra kell feltelepíteni a Raspbian disztribúciót a frissített package listával. Ez fogja létrehozni a kommunikációs hidat, a Raspbian és Timotei-Robotics Linux rendszerek között.

Amikor programozzuk a vezérlő panelt, akkor a programot a notebook-on írjuk, szabvány PLC létra diagramban, amit a megírás után generálunk egy programkódot, ami Raspberry Pi B+ -hoz kerül kiküldésre.

A program tervezés létra logikában és a következő fejezetben kerül taglalásra.

A vezérlő rendszerünk négy darab részre bontható:

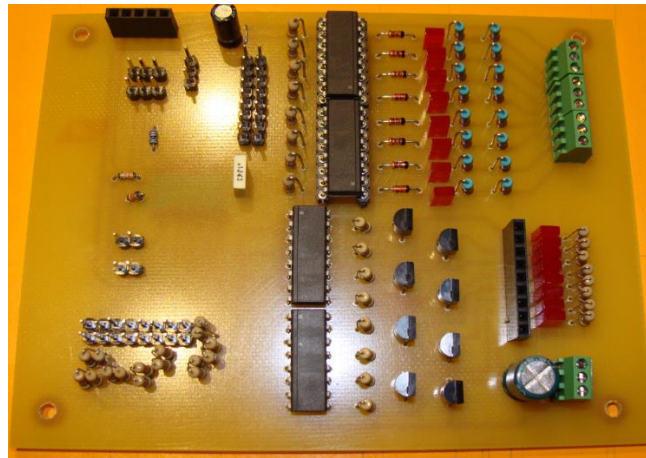
- Számítógép vagy Notebook
- Raspberry Pi eszköz
- TiMo main board
- Relé panelok



2. ábra: Programozható Beágyazott Rendszer - blokk diagram
(Forrás: tervezők által létrehozott tartalom)

A programozható beágyazott rendszernek a koncepciójának értelmében, a különböző programok implementációja és bementek kiértékelése, valós időben történik a Raspberry Pi -on. Az eszköz használata ilye téren azért lehetséges, mivel magas órajelű ARM architektúrára épülő CPU -val rendelkezik. Továbbá az eszköz folyamatos kapcsolatot létesít a notebook-al, amin a programok megírása történik az adott feladat elvégzése érdekében létra diagramban. Az épület mechatronikai kutatóközpont hozzáférést biztosított a tesztek ideje alatt a nagysebességű belső hálózathoz, annak érdekében, hogy szélessávú internetet tudjuk biztosítani az eszközök között [10].

A TiMo főpanel áramköre galvanikusan le van választva és a kommunikációt végző IC (Integrated Circuit) egy speciális soros-párhuzamos konverter. Az Integrált áramkör típusa egy PCA9555D [11], ami 16 bit-es párhuzamos input/output-okkal (GPIO) és 16 I/O -val rendelkezik. A TiMo főpanel végzi az I²C kommunikációt, vagyis párhuzamos input/output-oká alakítja át.



*3. ábra: TiMo főpanel – nyomtatott áramkör
(Forrás: tervezők által létrehozott tartalom)*

Az I/O port konverter IC I2C kommunikációs buszon keresztül kapcsolódik a Raspberry Pi portjaira. Itt elérhető a rendszer által biztosított 3,3V-os feszültség, továbbá a Raspberry Pi processzora is ezt a feszültségszintet használja az I2C kommunikáció megvalósításához. Ezen okokból kifolyólag kézenfekvő és egyben biztonsági szempontokat tekintve optimális megoldásnak bizonyult, ha az IC tápellátását a Raspberry Pi által biztosított 3,3V-ról kerül megvalósításra. [12].

A konverter IC teljes mértékben galvanikusan le van választva optocsatolók által az I/O portokról, ezért nem károsodhat még túlfeszültség hatására sem az IC, ezáltal a Raspberry Pi sem. Továbbá mivel az IC tápja a Raspberry Pi processzorának tápjával megegyezik, ezért a feszültségcsúcsok és a zavarok megjelenésének mértéke is a lehető legkisebbre van csökkentve. A galvanikus leválasztást az ACPL847 optocsatolók végzik. A kiválasztásuk során azért erre a típusra esett a választás, mert az IC egy tokban tartalmaz 4 darab optocsatolót. Így a 8 kimenet és 8 bemenet leválasztásához 4 darab IC-re volt szükség.

A kimeneti optocsatolók LED részének a meghajtását a konverter IC közvetlenül végzi.

Az optocsatoló LED-jével záró irányba beépítésre került egy védődióda csatornánként, annak érdekében, hogy fordított polaritás esetén a teljesítmény a diódán és az ellenállásokon disszipálódjon el és ne az optocsatoló LED részét tegye tönkre. A kimeneti optocsatolók esetén a leválasztott oldalon tranzistorokat hajt meg. Ezeken keresztül kisebb teljesítményű, 5V feszültséget igénylő eszközök vezérelhetők. Az 5V-os feszültséget a Raspberry Pi-től eltérő tápegység biztosítja, így zárlat esetén a vezérlő egység nem károsodik. Az elkészített vezérlő esetén a meghajtott eszközök 5V-al vezérelhető relék, amelyekkel akár hálózati feszültség kapcsolása is lehetséges.

A reléket tartalmazó panelek a könnyű szerelhetőség és modularitás miatt két darab, egyenként 4 relét tartalmazó panelra kerül kialakításra. A relék paramétereit tekintve 250V, 10A kapcsolására alkalmasak váltakozó áram esetén. A reléknek alaphelyzetben nyitott és zárt érintkezői is vannak, így alkalmazási helye függvényében különbözőféle képen is beköthetőek.

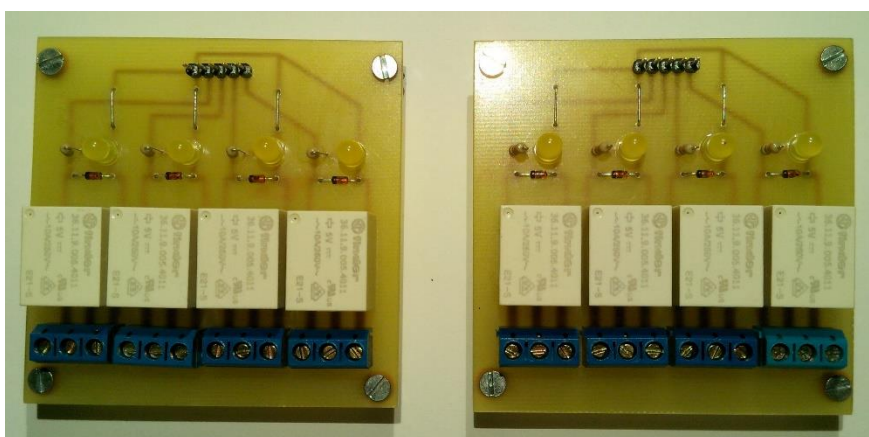
Továbbá feszültségcsúcs és interferencia sem okoz problémát a működésben, mivel a konverter IC és a Raspberry Pi B+ azonos tápegységet igényel.

3. A Vezérlő Programozása

Ahogy korábban említésre került a vezérlő programozható létra logika szerint is. A PLC (Logikai Vezérlők) az iparban gyakran használt eszközök, amik kiforrott szabvány létra diagramban programoznak [13].

Általánosan a HMI (Human Machine Interface) felprogramozása is létra logika szerint történik. Az egyik korábbi projektben a légszűrő hűtőszivattyú vezérlését valósítottuk meg [14].

A Linux rendszereken rengeteg PLC program tervező érhető el, esetünkben ClassicLadder került feltelepítésre a notebook-ra [15]. A telepítési eljárást követően, létrehoztuk a kapcsolatot a notebook és a Raspberry Pi B+ között a softwaren belül. Innentől pedig bármilyen szabványos PLC létra logika szerint írt program feltölthető a Raspberry Pi B+, ami I2C buszon keresztül csatlakozik a TiMo főpanelra. Raspberry Pi jelet küld ki a TiMo főpanelnak, ami továbbítja a jelet a relé panelek felé.



4. ábra: Relé panelek - nyomtatott áramkör
(Forrás: tervezők által létrehozott tartalom)

Ezzel a kidolgozott metodikával képesek vagyunk, újrakonfigurálni a komplex rendszereket, saját Linux disztribúción és szabványos létra logika szerint.

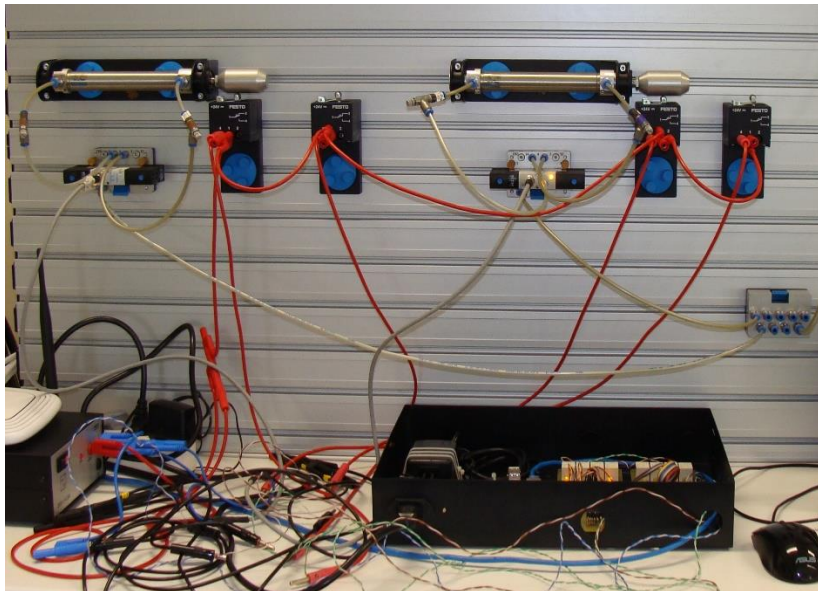
4. Otthoni és Ipari Automatizálás

A Debreceni Egyetem, mint épület mechatronikás mérnökök számos automatizálási feladatot végeztünk el. Az elkészített vezérlők alkalmas mágneszárak, világítás vagy riasztó rendszerek vezérlésére. A Smart Home témakörben egy hőszivattyút vezéreltünk kezdetben az Ni sbRio 9606 FPGA-val [16], később Schneider PLC -vel oldottuk meg a vezérlést. Jelenleg a teljes program átültetése zajlik a mi általunk tervezett TiMo vezérlő panelra. Az épületmechatronikai kutatóközpontban rengeteg ipari eszköz található, amik közül mi a FESTO pneumatikus elemeit használtuk fel, egy kapcsolás megépítésére, amit a PLC létra diagramban programoztunk fel hálózaton keresztül. A kapcsolásban két kettősműködésű munkahengert (DAC), négy végállás kapcsolót és további két darab 5/2 bistabil szolenoid szelepet használtunk fel.

Alaphelyzetben a két munkahenger alapállapotban van, a két alaphelyzetet jelző kapcsoló nyomva van

A pneumatikus kapcsolás működési folyamata:

- Az első munkahenger végállásig kimegy (végálláskapcsoló megnyomódik)
- A második munkahenger végállásig kimegy (végálláskapcsoló megnyomódik)
- Az első munkahenger alaphelyzetbe visszamegy (alaphelyzetet jelző kapcsoló megnyomódik)
- A második munkahenger alaphelyzetbe visszamegy (alaphelyzetet jelző kapcsoló megnyomódik)
- Az egész indul előről – állandósult állapot



5. ábra: TiMo vezérlő által működtetett pneumatikus kapcsolás
(Forrás: tervezők által létrehozott tartalom)

A program a pneumatikus kapcsolás részére Linux rendszeren került megtervezésre és feltöltésre Raspberry Pi B+, ami a jelet továbbítva a TiMo főpanel részére a relé panelokon keresztül képes kapcsolni az FESTO eszközöket a megadott időn belül. Az első teszt során 6 órán keresztül ment a vezérlő pontosan és üzembiztosan. A rendszer előnye hogy hálózatról is könnyedén újraprogramozható létra diagramban, így akár távolról is új vezérlést lehet adni az adott rendszernek vagy gyártósornak.

5. Összegzés

A vezérlési metodika, ami elkészítésre került, újraprogramozható, saját Linux rendszert futtat és képes komplex vezérlések végrehajtására. Továbbá szabvány PLC létra logikában programozható, ami nem teszi szükségessé más programozás nyelvek megtanulását és távolról felprogramozható-felügyelhető. Ezenfelül képes FESTO ipari eszközök vezérlésre és moduláris felépítése következtében könnyen bővíthető és olcsó a TiMo vezérlő rendszer elkészítése. Alternatív megoldást jelenthet a kis- és középvállalkozásoknak, illetve az ipar specifikus területein is integrálható. Elsősorban jelenleg Hibrid 3D nyomtatók vezérlését látják el, a Timothy-Robotics-nál [17].

6. Köszönetnyilvánítás

A fejlesztési és a kutatási források rendelkezésünkre való bocsátásáért, ezúton is szeretnénk köszönetet mondani, a Timothy-Robotics-nak! A tesztelésért szükséges engedélyek megadásáért pedig Dr. Tóth János Tanár Úrnak!

7. Hivatkozások

- [1] <http://eem.eng.unideb.hu/index.php/department/laboratories> - 2015.04.14. 5:21
- [2] G. Husi, P. T. Szemes, N. Tóth, É. Dudás, Establishment of New Research Infrastructure at Department of Electrical Engineering and Mechatronics, Faculty of Engineering, University of Debrecen - Annals of the Oradea University 05/2013; XII (XXII)(1):111-116.
- [3] G. Husi, P. T. Szemes, E. Dávid, T. I. Erdei, Building Mechatronics Research Centre as energy aware Intelligent Space - Industrial Electronics Society, IECON 2013 - 39th Annual Conference of the IEEE; 01/2013
- [4] G. Husi, P. T. Szemes, E. Dávid, T. I. Erdei, G. Petó, Reconfigurable Simulation and Research Toolset for Building Mechatronics. Proceedings of CERiS'13 - Workshop on Cognitive and Eto-Robotics in iSpace. Budapest, 2013. július. ISBN 978- 963-313-086-5
- [5] L. Torvalds, Linux: a Portable Operating System - Helsinki January 31, 1997, Master of Science Thesis, University of Helsinki, Department of Computer Science
- [6] <http://www.hpmarket.cz/productOpt.asp?konfId=K3W99EA> 2015.05.18. 8:02

- [7] S. J. Sharma, A. C. Balharpure, A. S. Pande, S. U. Dubey, G. K. Singh, V. M. Ghodki, S. Rajagopalan, Design of Embedded Sing-around System for Ultrasonic Velocity Measurements in Liquids - Journal of Embedded Systems. 2014, 2(1), 15-17, doi: 10.12691/jes-2-1-3
- [8] E. Upton, Raspberry Pi - IEEE Computer Society 31.10.2013, 10.1109/MC.2013.349
- [9] A. Horvath, The software package dependency networks of some Linux distributions - Conference: Nonlinear Science and Complexity (NSC), 2012 IEEE 4th International Conference on, DOI: 10.1109/NSC.2012.6304711
- [10] G Husi, C Szász, H. Hashimoto, M. Niitsuma, Building Mechatronic Simulation System - Annals of The Oradea University Fascicle of Management and Technological Engineering ISSUE #1, MAY 2014, <http://www.imtuoradea.ro/auo.fmte/> - 2015.05.15 8:20
- [11] http://www.nxp.com/products/interface_and_connectivity/i2c/i2c_general_purpose_i_o/PCA9555D.html - 2015.08.15 12:41
- [12] E. Upton, G. Halfacree, Raspberry Pi User Guide - <http://www.cs.unca.edu/~bruce/Fall14/360/RPiUsersGuide.pdf> 2015.05.18 13:22
- [13] J. R. Hackworth, F. D. Hackworth, Jr., Programmable Logic Controllers: Programming Methods and Applications - <http://electrical-engineering-portal.com/download-center/books-and-guides/automation-control/plcs-programming-apps> - 2015.05.02 1:42
- [14] G. Husi, H. Hashimoto, Cs. Szász, Application of Reconfigurable Hardware Technology in The Development and Implementation of Building Automation Systems - Environmental Engineering and Management Journal, November 2014, Vol.13, No. 11, 2899-2905
- [15] M. Le Douarain, ClassicLadder project - <https://sites.google.com/site/classicladder/> - 2015.06.09 5:12
- [16] G. Husi, Cs Szász, Building Automation Technology in Electrical Engineering and Mechatronics Department in Debrecen - Ulusal Makína Teorisi Sempozyumu, 2013
- [17] <https://www.facebook.com/Timotei-Robotics-468710276611086/> - 2015.04.0414:18