

Párhuzamos adatgyűjtés és feldolgozás Raspberry Pi klaszterekkel

Parallel data acquisition and processing with Paspberry Pi clusters

A. AGONÁS¹, P. T. SZEMES²

²Debreceni Egyetem, Műszaki Kar, Villamosmérnöki és Mechatronika Tanszék, 4028 Debrecen, Óttemető utca 2-4
szemespeter@eng.unideb.hu

Absztrakt. A kutatásaink során egy Raspberry Pi számítógépekből álló számítógép hálózatot hoztunk létre, amely összehangolt működéssel épületmechatronikai kutatási feladat megoldásához használtunk. Az eredmények alapján megállapítható, hogy a Raspberry Pi számítógépekből álló klaszter segítségével hatékonyan feldolgozható egy épületen belül keletkező mérési adatok.

Abstract. Computer network was built from Raspberry Pis, what can solve research task in the filed of building mechatronics. According to the results, it can state the cluster built from Raspberry Pis, can process indoor sensor data effectively.

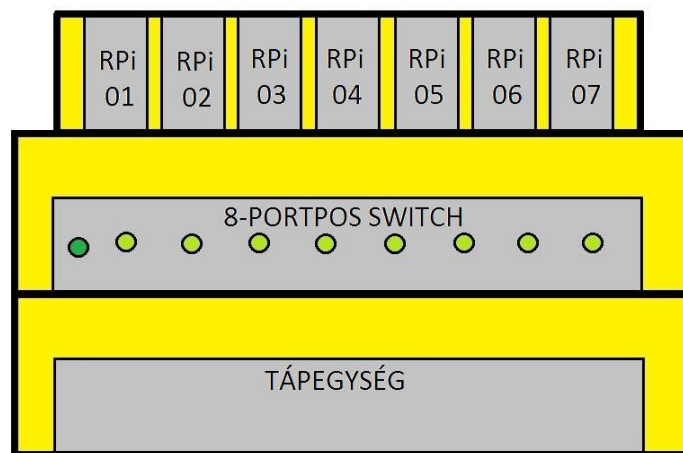
Bevezetés



1. ábra: „UCMote” Vezeték nélküli szenzorhálózat egy mérő csomópontja.

Jelen cikk témája a következő: a Debreceni Egyetem Műszaki Karának Villamosmérnöki és Mechatronikai Tanszékén a Dr. Szemes Péter Tamás tanár úr vezetésével épített Raspberry Pi klaszterek felhasználása az épületmechatronikában és információ tárolásában, illetve feldolgozásában. A szakirodalomban sokféle Raspberry Pi klaszter építési technológia ismeretes, de

alkalmazástechnikai leírás kevesebb. Jelen cikk egy alkalmazási példát mutat be; különböző, beltéri és kültéri szenzorok által mért jelek adatbázisba történő tárolását és rendezését, illetve, az adatok grafikus vizualizációját.



2. ábra: Egy Raspberry Pi klaszter fizikai felépítése

1. Mechatronikai háttér bemutatása

A Raspberry Pi (RPi) egy áramkörü lapra integrált mini-számítógép [1], mely különböző Linux-disztribúciókkal működtethető. A kártya rendelkezik SD kártya hely foglalattal, Ethernet csatolóval. A klaszter, amely több számítógép rendezett és koordinált együttműködését jelenti [4], Ethernet csatlakozás segítségével, TCP/IP protokollon keresztül kapcsolódik a klaszter többi tagjához. A RPi-k különlegessége a szabadon programozható GPIO portok, amelyek lehetővé teszik a különböző szenzorok közvetlen fogadására is. Ezek a lábak tartják a kapcsolatot az eszköz és a külvilág között, betölthetnek bemeneti és kimeneti funkciókat is a felhasználástól függően.

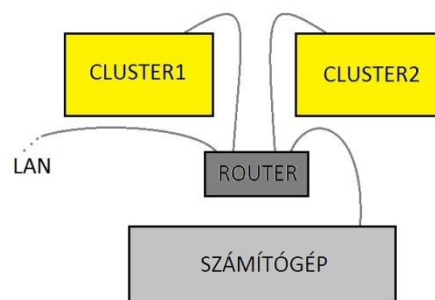
A Raspberry Pi jó ár-érték arányának köszönhetően megfelelő eszköz épületmechatronikai kutatás támogatásához. Viszonylag olcsó, elegendő tudással rendelkezik, keveset fogyaszt, így lehetővé teszi az épületmechatronikai rendszerek kiépítési és üzemeltetési költségeinek csökkentését úgy, hogy közben a teljesítmény ne romoljon. A fejlesztett számítógép klaszter fizikai felépítése a 2. ábrán látható; alsó részen tápegység, a középső részen egy Layer 2 switch, valamint a felső részen a miniszámítógépek helyezkednek el.

A feldolgozandó adatokat ADXL345 gyorsulásmérő [3] illetve UCMote mini szenzorokból (1. ábra) lettek gyűjtve. A gyorsulásmérő közvetlenül a Raspberry Pi I2C buszára lett kötve, amely az eszköz gyorsulását méri a derékszögű Descartes koordinátatengelyek szerint. Az UCMote mini egy vezeték nélküli érzékelő hálózat, mely képes gyorsulást, légnyomást, fényerősséget és hőmérsékletet mérni. Ez a szenzor Ethernet átjárón keresztül csatlakozik közvetlenül a RPi-hez, így kiterjedt épületek mérésére is alkalmazható.

2. Mérés és adatgyűjtés

2.1. A rendszer felépítése

A Raspberry Pi klaszterek egyenként 7 darab RPi-t tartalmaznak. Az eszközök Ethernet kábel segítségével egy switchen keresztül csatlakoznak egymáshoz és az internethez, illetve kommunikálhatnak ilyen módon más számítógéppel, vagy más Internetre kötött, eszközzel is. A RPi-k kapcsolata kétféle lehet. Működhetnek egymástól függetlenül párhuzamosan, illetve egymástól függően, ún. „master-slave”, azaz mester-szolga módban is. A mester küldhet vagy fogadhat információt a szolgától, és mindig ő mondja meg, mi lesz az adott feladat. Egy adott mester-szolga felállásban mindig csak egy mester van, szolgál azonban lehet több is. A klasztereket általában közös tápfeszültséggel látják el. Az általam használtak Lego elemekből álló tokba lettek beépítve.



3. ábra: Számítógép klaszterek összeköttetése

A klaszterek routeren (útválasztón) keresztül összeköthetők egymással és a számítógéppel, így képesek az egymás közötti információcserére és feladatmegosztásra is. Ezt használtam ki a folyamatok felgyorsítása szempontjából. Amennyiben az egyes eszközöknek kevesebb feladatot kell ellátniuk, sokkal nagyobb sebességgel dolgozhatnak, így növelhető az összteljesítményük is.

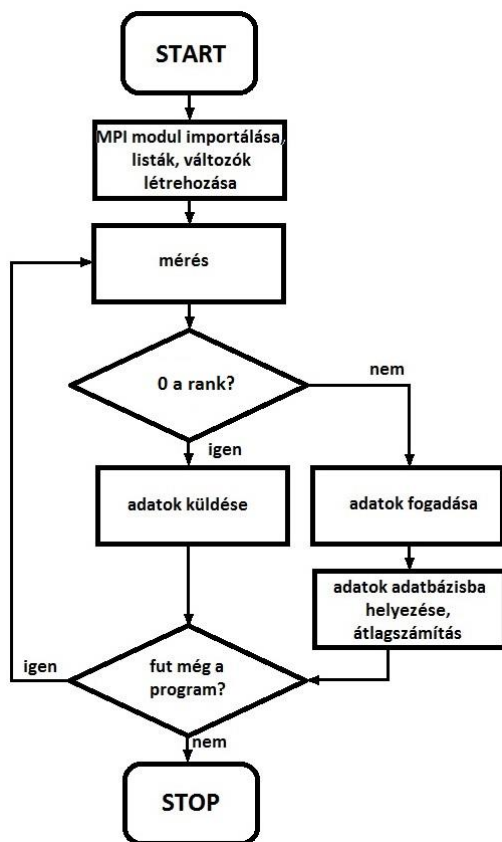
2.2. Az adatgyűjtés menete

A szenzorokból külön specifikált programok segítségével olvashatunk be adatokat. Ezt egy adott Pi-n végezzük, mely tovább küldi ezeket az adatokat a klaszter egy másik elemére feldolgozás céljából. Az eszközök egymással történő adatcseréje az ún. MPI (Message Passing Interface) segítségével valósítható meg.

Az MPI egy általánosított üzenetküldő rendszer, amely arra lett tervezve, hogy széles körben alkalmazható legyen párhuzamosan működő számítógépekre. A kifejezetten Pythonra kifejlesztett MPI csomagot mpi4py-nak hívják. Ez támogatja a pontról pontra történő (küldés és fogadás) és a kollektívizáló (információ szétsugárzás, szétoosztás, begyűjtés) folyamatokat is.

A szerzők által fejlesztett adatbeolvasó és -küldő program a mért értékeket az MPI által az eszközöknek adott rang (prioritás) alapján osztja szét, illetve fogadja, feldolgozza és adatbázisban

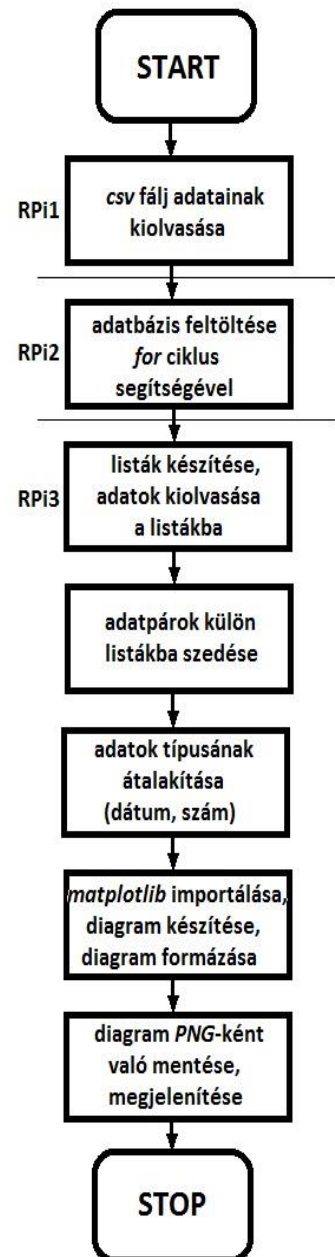
helyezi el. A 0 rangú elemen történik a mérés és a küldés, a 0-tól eltérő elemeken a fogadás és a feldolgozás.



4. ábra: Az MPI modul működése



5. ábra: MySQL adatbázis létrehozása



6. ábra: UCMote adatainak feldolgozása

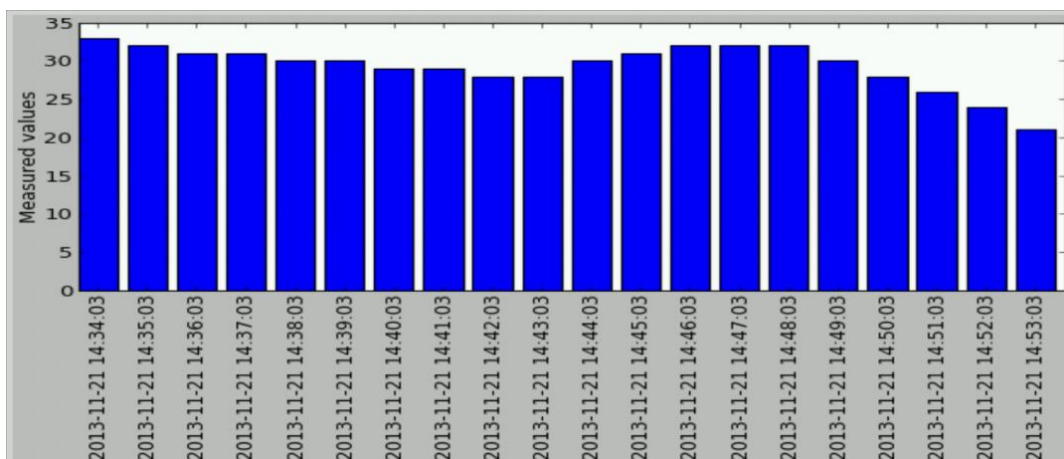
3. Rendszerezés és feldolgozás

A begyűjtött adatokat MySQL [6,7] adatbázisban lett elhelyezve, amely az egyik számító csomópontra lett telepítve. A MySQL az egyik legelterjedtebb többfelhasználós, többszálú, SQL-alapú relációs adatbázis-kezelő szerver. A MySQL-kiszolgáló egy vagy több adatbázist tartalmaz, melyekben táblák,

ezekben belül pedig oszlopok találhatók. Minden oszlopnak külön neve van, mely utal az oszlop elemeire. A tábla minden sorának és oszlopának a keresztmetszetében van egy, a táblameghatározásnak megfelelő érték.

A fogadó Raspberry Pi egy előre létrehozott adatbázis meghatározott oszlopaiban helyezi el a kapott adatokat, melyek újabb program segítségével olvashatók ki belőle. A gyorsulásmérő szenzor által mért értékek átlagolással feldolgozásra kerültek, azaz egy perces időközönként átlagot számoltam a három koordinátengelyen ért értékből. Természetesen az átlagszámítás ideje tetszés szerint növelhető illetve csökkenthető.

Az UCMote mini esetében a szenzor egy .csv fájlba menti a mért értékeket. Ezek előzetesen elkészített mérések voltak, én csupán a felhasználásukat valósítottam meg. Az első számú Pi kiolvassa az adatokat, majd továbbküldi a második számú Pi-nek, mely feltölti egy MySQL adatbázis táblájának "mért érték" és "mérés ideje" oszlopát.



7. ábra: A matplotlib által készített diagram a 2013-ban előre lement adatok felhasználásával

Az adatok grafikus megjelenítését a Python programozási nyelvhez készült matplotlib könyvtárban található modul segítségével végeztem el. Az adatbázisból történő kiolvasást egy harmadik Pi végzi. Erre az eszközre megírt program listákat készít, melyekbe beleteszi az oszlopokban található információkat, majd a megfelelő formátumba alakítja őket. Mivel a diagram elkészítéséhez az ordináta tengelyen lévő értékeknek számoknak, az abszcissa tengelyen lévőknek pedig dátumnak kell lenni, ez az átalakítás elengedhetetlen. A matplotlib alkalmazásával különböző típusú diagramok készíthetők. Az egyes értékeket jelölhetik különálló vagy összekötött pontok, oszlopok, stb. Ezek színezhethetők, formázhatók, változtatható a méretük, megadható a diagram és a tengelyek neve is. A program elkészült diagramot PNG képformátumban menti, így a későbbiekben egyszerű grafikus módon megjeleníthetővé válnak a mérési eredmények.

Összegzés

Kutatásunk során megvizsgáltuk a Raspberry Pi B típusú változatát, a belőlük felépített klaszterek működését és felhasználási lehetőségeit. Ennek kivitelezéséhez Python [2,8],[5] programozási nyelvet alkalmaztuk, mely segítségével programokat fejlesztettünk mért adatok fogadására, feldolgozására,

tárolására és megjelenítésére. A fejlesztést a RPi csomópontokon végeztük. Megállapítottuk, hogy az eszközök ilyen módon történő felhasználása rendkívül hasznos és kifizetődő az épületmechatronikai alkalmazásban. Véleményünk szerint a jövőben a Raspberry Pi mini-számítógépekből és vezeték nélküli szenzorhálózatból (UCMote) jól használható lesz különféle okos-épületek (okos-házak, okos-gyárak) jellemző paramétereinek feltárására, információszolgáltatásra a beavatkozók számára és statisztikák felépítésére. Mivel számtalan különböző fajtájú szenzor csatlakoztatható hozzá, melyek működésére, érzékenységének beállítására más-más programok írhatók, széles körben válik alkalmazhatóvá.

Hivatkozások

- [1] Raspberry Pi https://hu.wikipedia.org/wiki/Raspberry_Pi
- [2] Python-programozás 1: Alapfogalmak <http://blog.molnardenes.hu/python-programozas-1-alapfogalmak/>
- [3] ADXL345 Data sheet <http://www.analog.com/media/en/technical-documentation/data-sheets/ADXL345.pdf>
- [4] SSH Using Windows <https://www.raspberrypi.org/documentation/remote-access/ssh/windows.md>
- [5] Quick Guide To nano Text Editor On The Raspberry Pi <http://www.raspberrypi-spy.co.uk/2013/11/quick-guide-to-nano-text-editor-on-the-raspberry-pi/>
- [6] MySQL <https://hu.wikipedia.org/wiki/MySQL>
- [7] Zak Greant, Chris Newman: MySQL zsebkönyv, Budapest, 2007
- [8] Brad Dayley: Python zsebkönyv, Budapest, 2008