Herczig Ádám

Szakdolgozat

Hőmérő hálózat

IoT felület

kialakítása

PÉCSI TUDOMÁNY EGYETEM

MŰSZAKI ÉS INFORMATIKAI KAR

VILLAMOSMÉRNÖKI SZAK

Pécs, 2017

Nyilatkozat

Alulírott Herczig Ádám diplomázó hallgató, kijelentem hogy a szakdolgozatomat a Pécsi Tudományegyetem Műszaki és Informatikai karán készítettem a villamosmérnöki BSC diploma megszerzése véget.

Kijelentem, hogy a szakdolgozatom érdemi részét egyedül végeztem el. Az érdemi részen kívül csak meghivatkozott forrásokat használtam fel a szakdolgozatomba (szakirodalom, kód). Tudomásul veszem, hogy a szakdolgozatom leírt forráskódot a Pécsi Tudományegyetem, valamint a szakdolgozat témát kiíró konzulens saját céljaira szabadon felhasználhatja.

2017.10.24.

Herczig Ádám

Tartalomjegyzék helye

1. Bevezetés

A mai hétköznapi életnek lassan már szerves része lesz hogy mindenféle eszközt rá fogunk tudni csatolni az internet nagy világára. A legáltalánosabb szenzoroktól kezdve, az autóiparban használt robotokon, aktuátorokon át egészen a hétköznapi használatú eszközökig (Televízió, hűtő, riasztóközpont) és még sorolhatnánk tovább is, szinte mindent rá tudunk csatolni az internetre. Ezáltal minden olyan eszköz mely IP címmel rendelkezik, kapcsolatban lehet más egyéb hálózati eszközökkel, amennyiben ez szükséges. Gondoljunk csak bele, megyünk haza munkából, okostelefonunkkal egy pillanat alatt leellenőrizhetjük a hűtőnk tartalmát vagy hogy a lakásunk különböző pontjain milyen hőmérséklet van. Hideg téli estéken akár, ezáltal előre feltudjuk fűteni számunkra megfelelő hőmérsékletűre a lakásunkat hőmérő szenzorok és egy termosztát vezérlésével, mindezt távolról, telefonunk segítségével mire hazaérnénk. Ezt nevezzük a magyar fordításban „Dolgok Internetének”, szakkifejezéssel IoT-nek azaz Internet of Things-nek. „Gyakorlatilag az IoT a fizikai tárgyak, eszközök, járművek, épületek és egyéb beágyazott elektronikai elemek hálózati kapcsolatát és adatcseréjét jelenti.” Az IoT eszközök nagy részének (szinte az összes beágyazott eszköz) az alapját valamilyen Linux alapú rendszer biztosítja.

Szakdolgozatom témájának pont egy hasonló IoT feladatot választottam, melyet az előbb példának hoztam fel. Miután érdekeltnek tartom magam a Linux rendszerek programozásában, fejlesztésében, feladatként egy olyan hálózat kialakítását kellett megterveznem aminek a vezérlője egy Linux alapú vezérlő mely maximum 99 db hőmérő szenzorral tudja tartani a kapcsolatot. Az eszköz programját C nyelven írtam meg. A vezérlő feladata egy konfigurációs fájl felolvasása, majd a kommunikáció inicializálása. Inicializálás után több szálon futva, bizonyos időközönként hőmérséklet adatokat kell lekérnie, felolvasnia majd a felolvasott értékeket ki is kell értékelnie. Végül a kiértékelt adatokat egy webes felületen meg is kell tudnia jeleníteni a vezérlőnek. A szolga eszközökről beolvasott adatokat FIFO1-ba kellett gyűjtenem majd a feldolgozó szállnak a FIFO-ból kivett értékékkel kell tovább dolgoznia kettő különböző mérési algoritmussal.

Hogy le tudjam fordítani a C nyelven megírt forráskódomat, külön egy Makefile-t kellett létrehoznom. A Makefile segítségével tudtam a Linux számára futtatható programmá fordítani a forrás fájlokat. Az adatok feldolgozására vagy a program futása során előforduló hibák logolására a Linux rendszer beépített syslog-ját használtam. Ezen kívül még egy másik beépített programot használtam fel a logok megfelelő tárolására. Ez a program a logrotate. Ennek a segítségével biztosítom hogy a sok loggolt adat és hiba miatt ne teljen meg az amúgy is kevés tárhellyel rendelkező Linux rendszerem. A fejlesztések során egy git repository-t hoztam létre ennek a projektnek. Ez a repository 2 a [http://github.com](http://github.com/)/Dikeszmen/thesis.git oldalon nyilvánosan elérhető. A lefordított futtatható bináris memóriakezelésére a Valgrind programot használtam.(?) A hardware, melyre a fejlesztés készült egy általam előszeretettel használt Raspberry Pi 2B típusú SoC 3 volt.

2. Hardware