



A generikus kétrétegű háló modell PIC bázisú implementációjának lehetőségei

Bánkuti¹ Gy., Bálint² J.

¹Kaposvári Egyetem Matematika és Fizika Tanszék 7400 Kaposvár Guba Sándor utca 40.

²Folyamatinformatika Kutatófejlesztő Betéti Társaság 7478 Bárdudvarnok, Bárd utca 26.

ÖSSZEFOGLALÁS

Munkánkban a Kaposvári Egyetemen működő folyamatinformatikai kutatási iskola által szabadalmaztatott modellezési metodológiájának mikrokontroller alapú hardver megvalósításával foglalkozunk. Mivel a jól bevált szoftver technika adekvát hardver megvalósításához nem áll rendelkezésre megfelelő eszköz, így célunk csak a struktúra létrehozása és a célhardver lehetőségeinek és problémáinak tanulmányozása volt. Bemutatjuk a mikrokontrollerek felhasználásával elkészült struktúrát és ennek programozását - a felprogramozástól, az inicializálástól, teszteléstől, a négyütemű szimuláció végrehajtásáig - az elkészült panelek kapcsolási rajzát, és esetenként nyomtatott áramköri fotóját is.

Kulcsszavak: kétrétegű háló modell, mikrokontroller, párhuzamos hardver implementáció

ABSTRACT

The Possibilities of the Microcontroller based implementation of the Generic Bi-layered Net Model

Gy. Bánkuti¹, J. Bálint²

¹University of Kaposvár, Department of Mathematics and Physics, H-7400. Kaposvár Guba Sándor utca 40.

²Process-Informatics Ltd., H-7478 Bárdudvarnok, Bárd utca 26.

In our paper we deal with the PIC based hardware implementation of the Generic Bi-layered Net Model, patented by the Process-informatics Research Group of the University of Kaposvár. As there is no adequate hardware to implement this well-tried software methodology, our aim was only to construct the structure of a simple small problem, and to examine it. We present the microcontroller based implementation of the structure and its programming, from the initialing, testing, to the four step execution, the circuit schematics and the photos of the integrated circuits as well.

Keywords: Generic Bi-layered Net Model, PIC, parallel hardware implementation

BEVEZETÉS

A kétrétegű hálómodell alapú modellezési metodológia alapjai megtalálhatóak az irodalomban (Czukás és mtsai., 2004; Czukás és mtsai., 2005). Lényege, a fizikai mennyiségek, (vagy jelek) elemi egységeit reprezentáló entitások (passzív elemek) rétegének az ezek elemi megváltozását reprezentáló (aktív elemek) réteggel történő kommunikációja (1. ábra). Mindez az alábbi négy ütemben zajlik:

1. az aktív elemek leolvassák a passzív elemek állapotát
2. aktívak számolnak

3. a passzív elemek leolvassák az aktív elemek állapotát
4. a passzív elemek elszámolják a változásokat

1. ábra

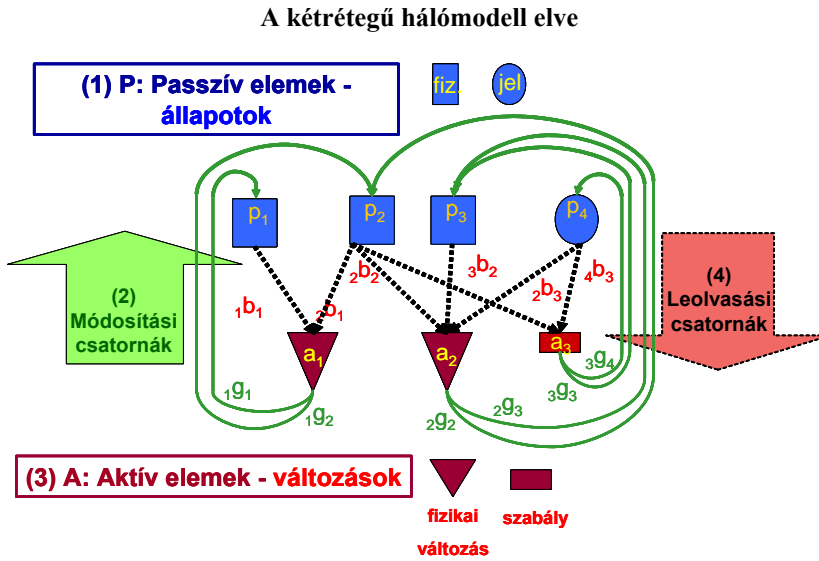


Figure 1: The idea of the Generic Bi-layered Net Model

Passive elements(1), Modifying chanelns(2), Active elements(3), Reading chanelns(4),

2. ábra

Egy példa kétrétegű hálómodell alapú hardver elvi felépítése

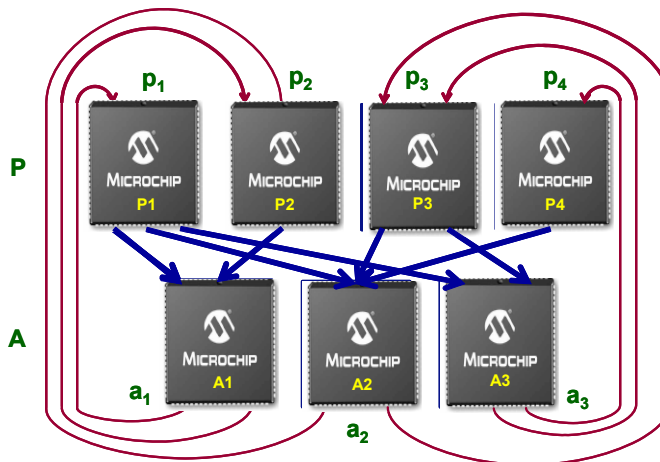


Figure 2.:Example Structure of the Generig Bi-layered Net Hardver

A kétrétegű hálómodell alapú modellezés hardver megvalósítása két lényegileg azonos, - a valós modellekben általában nagyszámú - egyenként kisebb teljesítményű egységek egy sajátos architektúra szerinti strukturált programozásával és ütemezett működtetésével oldható meg (2. ábra). Ennek fizikai hardver megvalósításához jelenleg nem állnak rendelkezésre adekvát eszközök. A jövőbeli fejlődés lehetőségeit figyelembe véve, lényeges azonban, hogy a jelenleg elérhető kevésbé adekvát eszközökkel is tanulmányozzuk a jól bevált szoftver architektúra célirányos hardver leképezésének lehetőségeit és problémáit.

A STRUKTÚRA KIALAKÍTÁSA

Az eszköz kiválasztása

A munka első lépése a létező hardver elemek közül a célnak megfelelő kiválasztása volt. Igaz, hogy a mikrokontrollereket (PIC-eket) vezérlésekhez fejlesztették ki, de a kereskedelemben kaphatóak már olyan típusok, amelyek a számunkra szükséges műveletek (összeadás, szorzás, osztás) elvégzésére alkalmasak, és megfelelő számú be-, és kimenettel rendelkeznek. Választásunk a **PIC18F2550** típusú mikrokontrollerre esett. Melynek paraméterei az alábbiak:

- 20MHz órajel,
- 8 bites architektúra,
- Hardveres 8 bites szorzó,
- Szoftveres osztó,
- Áramköri környezetben programozható (ICSP),
- Futás közbeni hibakeresés (ICD) támogatás,
- USB 2, USART, I2C támogatás,
- 83 elemű assembly utasításkészlet,
- 28 kivezetéses DIL tokozás,
- 21 db kétirányú Input/Output PORT,
- 7 db technikai kivezetés,
- Relatív alacsony ár (<1500 Ft/db), Kereskedelmi forgalomban könnyen beszerezhető

Megjegyezzük, hogy mikrokontrollerek nem Neumann elven működnek. Adat- és a programmemóriájuk fizikailag is külön vannak választva, külön buszokat használnak, és különböző szóhosszúságúak, azaz Harvard architektúra szerint vannak szervezve. (Tanenbaum, 2001). (A klasszikus Neumann-elvű számítógépnél az adatok és az utasítások azonos memóriában helyezkednek el, és azonos buszt használnak.)

Szükségünk volt még a mikrokontrollerek felprogramozását végző eszközre, melyet szintén beszereztünk.

A részegységek felprogramozása

Ennek segítségével a két processzorosztály elemeit a feladatnak megfelelő műveleti kóddal a PC-hez USB-n keresztül csatlakoztatott ICD2 programozó készülékkel áramköri helyükön (ICSP) egyenként felprogramoztuk (3. ábra) a Microchip cég által ingyenesen közzétett MPLAB IDE fejlesztőkörnyezet segítségével.

A passzív és aktív processzor modulokat a PC-hez USB kapcsolattal csatlakoztatott, vezérlő processzor segítségével SN 74154 demultiplexeren keresztül konfiguráltuk. A konfiguráció során megadtuk a passzív és aktív elemek kezdeti értékeit is (4. ábra). A sikeres konfigurálást a processzorosztályok elemei Conf_{OK} porton keresztül nyugtázzák a vezérlő processzor felé.

3. ábra

A beépített mikrokontrollerek ICD2 programozóval a történő felprogramozása

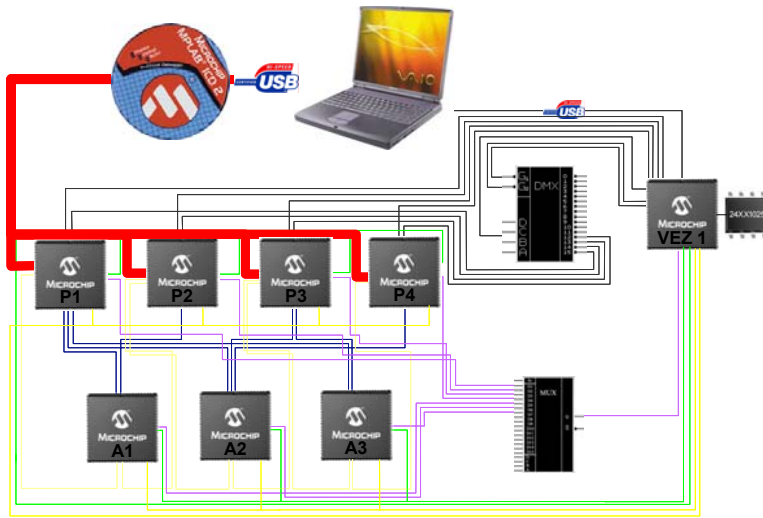


Figure 3.: Programming of the build in microcontrollers by ICD2 programmer

4. ábra

A passzív elemek kezdeti értékeinek megadása:

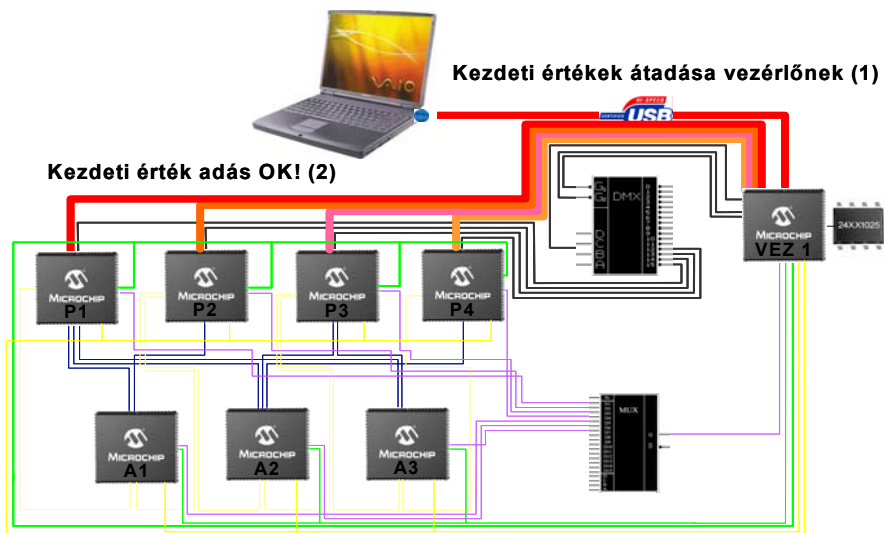


Figure 4: Initialization of the passive elements

Transferring the Initial Values to the controller(1), Initialization is OK! (2)

A VÉGREHAJTÁS LÉPÉSEI

A végre hajtás lépései az 5.ábrán a lépéseknek megfelelően számozott csatornákon látható útvonalakon valósulnak meg:

5. ábra

A végrehajtás lépései (1) – (5)

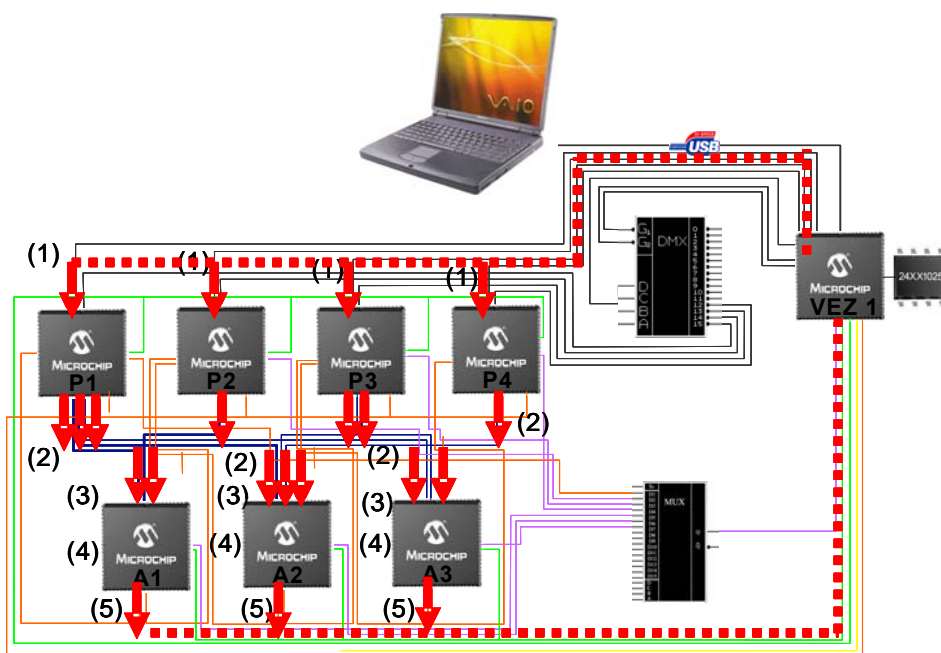


Figure 5: Steps of the Execution

Supervisory Microcontroller sends Activating Message to the passive elements(1), Passive elements export data to the output ports(2), Active elements read the input data(3), Active elements execute embedded programmes(4), Active elements send Ready Message to the Supervisory Microcontroller and exports data to the output ports(5),

(1) Vezérlő 1 – start jel küldés a passzív elemek számára

Az inicializálás befejezése után szükséges, hogy a vezérlő processzor a P_{start} porton start jelet küldjön a passzív elemeknek, hogy helyezték ki output portjaikra a kívánt információkat. (A további lépésekben ez lehetséges, hogy megtakarítható lépés.)

(2) Passzív elemek: adatkihelyezés output csatornáikra

A passzív elemek adatokat helyeznek a kimenetként konfigurált P_L portjaikra (5. ábra).

(3) Aktív elemek adatbeolvasása

Ennek megoldása kétféleképpen történhet. Az aktív elemek a vezérlő processzor utasítására kezdik csak el figyelni a leolvasási csatornájuk közül a struktúra által

meghatározott bemeneteiket (A_L port), vagy belső programjuk szerint, folyamatosan. Most épített kis modellünkben az első túlbiztosító megoldás nem szükséges, de nagyobb modellnél problémát jelenthet ha esetleg az aktív elemek ütemükön kívül is végrehajtják programjukat.

(4) Az aktív típusú processzormodulok programjainak végrehajtása

A vezérlő processzor utasítása, vagy az összes szükséges adat megjelenése után az aktív elemek programjai lefutnak, (5. ábra).

(5) Az aktív típusú processzorok a futás végrehajtás befejeződését jelentik a vezérlő processzor felé, output adatmegjelenítés a kívánt portjain

A programok lefutásának befejeződését az A_{NY} porton keresztül nyugtázzák passzív elemek a vezérlő processzor felé. Majd a struktúra által meghatározott output portokra kihelyezik a kiszámított adatokat (5. ábra).

6. ábra

A végrehajtás lépései (6) – (8)

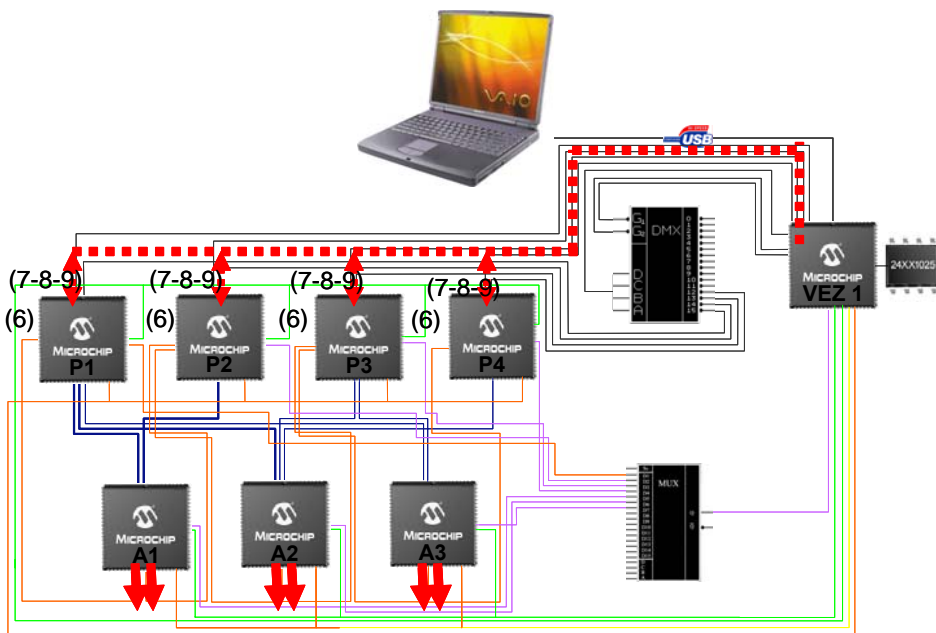


Figure 6: Steps of the Execution:

Passive elements read the input ports(6), Passive elements: run embedded programmes(7) Passive elements: send ready status message to the Supervisory Microcontroller (8), Supervisory Microcontroller send activate Message to the passive elements (9), Passive elements state report to the Supervisory Microcontroller (10)

(6) Passzív elemek: input csatornáik leolvasása

A passzív típusú elemek – az aktívakhoz hasonlóan – kétféleképpen, a vezérlő processzor utasítására vagy programjuk szerint folyamatosan figyelik aktív bemeneteiket (A_L port).

(7) Passzív elemek programjainak futása

A vezérlő processzor utasítása, vagy az összes szükséges adat megjelenése után a passzív elemek programjai lefutnak, (6. ábra).

(8) Passzív elemek: futás befejezés visszajelzése

A művelet befejezését P_{NY} porton keresztül nyugtázzák a vezérlő processzor felé.

(9) Vezérlő: Adatküldési start jel a passzív processzorok számára,

A vezérlő start jelet küld a passzív processzorok számára, hogy állapotjelentésüket jelentésüket kezdjék meg. Ennek szintén nagyobb feladatnál lesz jelentősége, mert a nagyszámú adat multiplexeren keresztül történő megfelelő beérkezése, a PC felé történő továbbküldése rendezett adatbeérkezést igényelhet.

(10) Passzívak: állapot jelentés a vezérlőnek

A passzív processzorok SN 74150 multiplexeren keresztül a vezérlő processzornak továbbítják a módosított értékeket, mely azután az USB csatlakozáson keresztül azt a PC-nek továbbítja. A módosított adatokat a PC-n futó felhasználói program dolgozza fel.

Végezetül megjegyezzük, hogy a programozottan szelektív kommunikáció megvalósítása akkor legegyszerűbb, ha az alkalmazott processzorok outputjainak és inputjainak száma legalább akkora, mint a passzív és aktív elemek száma. Ekkor természetes módon biztosítható, hogy tetszőleges számú elemből álló architektúra teljesen párhuzamosan működhessen, azaz egy ciklusban történjen az összes leolvasás.

Általános célú hardver esetén a $2 \times \left(\frac{n}{2}\right)^2$ képlet alapján számítható ki a szükséges kapcsolatok száma, azonos számú (n db) passzív és aktív elemet feltételezve. Például 10 – 10 passzív és aktív elemet feltételezve, 2000 összeköttetés szükséges az egy ütemben történő leolvasáshoz. A kevesebb beépített kapcsolattal rendelkező, de teljes párhuzamosságot megoldó struktúra megvalósíthatóságának kérdésére ezen munkánk nem keresi a választ.

EGYSZERŰ KÍSÉRLETI PÉLDÁNY KIALAKÍTÁSA, TESZT ÁRAMKÖR

Az ismertetett hardver terv alapján az Altium Ltd. Protel DXP Design Explorer analóg és digitális nyomtatott áramkör és kapcsolási rajz tervező program felhasználásával elkészítettük a PIC alapú hardver kapcsolási rajzát és nyomtatott áramköri tervét, valamint a szoftver alkatrész könyvtárában eredetileg rendelkezésre nem álló PIC18F2550 mikrokontroller elvi és nyomtatott áramköri beültetési rajzát.

Célszerűnek tartottuk beépítés előtt a felhasználandó mikrokontrollerek tesztelését, hogy garantálni lehessen a tervezett hardver implementációban lévő egyes mikrokontroller kapcsolások működésének hibamentességét. A tesztelés lehetséges volt, mivel a gyártó szerinti alapkapcsolás került kivitelezésre. Elkészítettünk egy tesztdiagramot, amelyben egy mikrokontroller esetében a megadott elvi kapcsolást (7. ábra) megvalósítva annak működőképessége kipróbálható volt.

7. ábra

PIC18F4550 típusú Mikrokontroller tesztáramkörének kapcsolási rajza

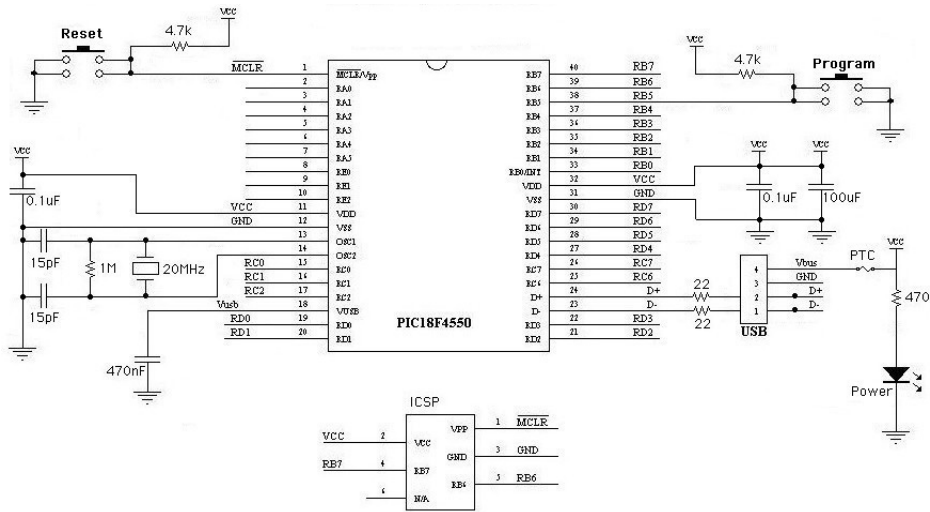


Figure 7: Circuit Schematics of the Test Panel

Ezen tesztáramkörbe egy PIC18F4550 típus került beépítésre (8. ábra), mivel az állt rendelkezésünkre, és ez a típus a több I/O kivezetéstől eltekintve, nem különbözik a választott típusunktól.

8. ábra

PIC18F4550 típusú mikrokontroller tesztáramkör

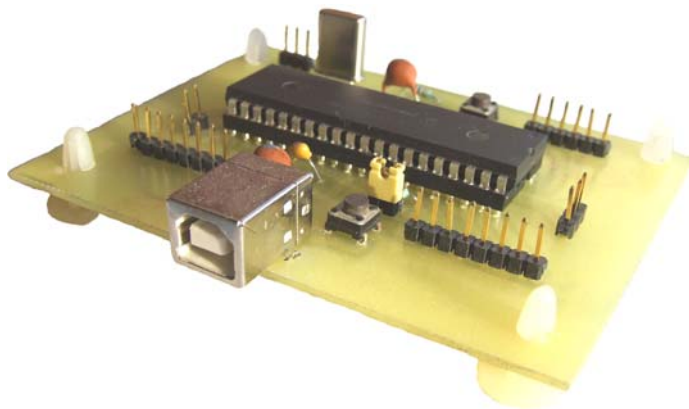


Figure 8: Test Panel for PIC18F4550 Microcontroller

ÖSSZEFOGLALÁS

Elmondhatjuk, hogy munkák eredményeként nemcsak a hardver részletes felépítésének terve, és a nyomtatott áramkörök rajzai készültek el, hanem (pl. a teszt áramkör) fizikailag is megvalósult.

Megállapítható, hogy

- a. Ezen nem adekvát eszközökkel kisméretben létrehozható a kívánt struktúra.
- b. A mikrokontrollerek portjait az adatútnak megfelelően inputként, illetve outputként lehetett programozni, vezérlő processzor start-stop jeleit kezelő portokat mint megszakítás érzékelőket programozva.
- c. A vezérlő utasításának hatására (vagy az egységek belső programjában definiált módon a mikrokontrollerekbe beépített programok futása elindítható.
- d. Az aktív és passzív elemek funkcióit megvalósító választott eszköz (PIC18F2550) képes a struktúra által megadott portra a megfelelő információt, jelet, kihelyezni, és a másik réteg processzorai hasonló módon dolgozva biztosítják az ütemek egymásutániságát.
- e. A visszajelző csatornákon, multiplexeren keresztül a mikrokontrollerek képesek állapotjellemzőik PC-hez történő eljuttatására.

Mindezek alapján megállapítható, hogy az elvárásokat teljesítő kisméretű hardver a választott eszközökkel létrehozható és működtethető. Nem szükséges hozzá új operációs rendszer kifejlesztése, ezen kis méretben nem okoz problémát a jelek megjelenése a nem használt portokon.

Azonban a nagyméretű feladatok megoldására alkalmas hardver megvalósíthatóságának kérdésére munkánk nem ad választ. Az összeköttetések elemszám (felének) négyzetével arányos növekedése ugyanis gyártási és működési nehézségeket okozhat.

Ezen matematikai – geometriai strukturális feladat megoldása további kutatásokat igényel. Megjegyezzük, hogy kutatócsoportunk FPGA alapú munkája sem adott még választ ezen utóbbi kérdésekre.

KÖSZÖNETNYILVÁNÍTÁS

A cikk a GVOP-3.3.1 200404-0091/3.0. projekt támogatásával készült.

IRODALOMJEGYZÉK

- Tanenbaum A.S. (2001). Számítógép architektúrák. Budapest : Panem Kiadó. 31-34.
- Sima D., Terence F., Kacsuk P. (1998.). Korszerű számítógép architektúrák tervezési tér megközelítésben. Budapest : Szak Kiadó. 6-25.
- Bánkuti, Gy., Csukás, B. (2005). Generic Bi-layered Net Model – General Methodology for Process Simulation, In: Daoliang Li and ÍBaoji Wang Eds.: Artificial Intelligence Applications and Innovations II. 2nd IFIP Conference of TC12 WG 12.5, Springer, 691-700.
- Csukás, B., Balogh, S., Bánkuti, Gy. (2005). Generic Bi-layered Net Model – General Software for Simulation of Hybrid Processes, In: Daoliang Li and ÍBaoji Wang Eds.: Artificial Intelligence Applications and Innovations II. 2nd IFIP Conference of TC12 WG 12.5, Springer, 701-710.

- Csukás, B., Bánkuti, Gy. (2004). Generic Bi-layered Net of the Functional Nodes in Process Modelling, IJCNN'2004, International Joint Conference on Neural Networks, 25-29 July, 2004, Budapest. Proceedings T-094. CD IEEE catalog Number: 04CH37541C, ISBN: 0-7803-8360-5.
- Csukás B., Bánkuti Gy., Balogh S.(2004) : Folyamatinformatika I. Elméleti alapok és alkalmazási példák. Egyetemi jegyzet. Kaposvár : Kaposvári Egyetem Matematikai és Informatikai Intézet, 235.

Levelezési cím (*Corresponding author*):

Bánkuti Gyöngyi

Kaposvári Egyetem, Gazdaságtudományi Kar

Matematika és Fizika Tanszék

7401, Kaposvár, Pf. 16.

University of Kaposvár, Faculty of Economic Science

Department of Mathematics and Physics

H-7401, Kaposvár, P.O.Box 16.

Tel.: 36-82-505-950

e-mail: bankuti@matinf.gtk.u-kaposvar.hu