Szegedi Tudományegyetem

Informatikai Tanszékcsoport

**Ipari folyamat szimulációja és irányítása programozható logikai vezérlővel**

Diplomamunka

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Készítette: |  | Témavezető: |  |
|  | Miklós Árpád |  | Dr. Kincses Zoltán |  |
|  | mérnök informatikus szakos hallgató |  | egyetemi adjunktus |  |

Szeged

2016

Ipari folyamat szimulációja és irányítása programozható logikai vezérlővel.



**Diplomamunka mérnök-informatikus MSc szakos hallgató számára**

**Témavezető: Dr. Kincses Zoltán**

**Témakör: ipari informatika, rendszer szimuláció, irányítás**

**Műszaki Informatika Tanszék**



**A feladat leírása, a munka célja**

A hallgató feladata egy ipari folyamat emulációjának és irányításának elkészítése. A munka célja, egyrészt az Országos Ajtonyi István Irányítástechnikai Programozó Versenyen szereplő technológia emulációjának elkészítése egy HIL szimulátor eszköz segítségével. A munka másik célja az így elkészített rendszer irányításának megvalósítása egy programozható logikai vezérlővel. Ez elkészült munka később jól alkalmazható a későbbi PLC versenyekre történő felkészítésben

**A munkavégzés fontosabb lépései:**

* A HIL szimulátor eszköz és a hozzá tartozó szoftverek megismerése, ismertetése
* Az emulálni kívánt technológia megismerése
* A programozható logikai vezérlő és a hozzá tartozó szoftver megismerése, ismertetése
* A HIL emuláció elkészítése
* Az elkészült emuláció irányításának megvalósítása programozható logikai vezérlővel
* Hibakezelés
* Tesztelés
* A dolgozat megírása

**A fejlesztéshez rendelkezésre álló erőforrások:**

* OMRON CJ2M PLC, I/O egységek és a programozásához szükséges szoftver
* Lucas Nülle I/O interfész PRO/TRAIN-hez
* Lucas Nülle BORIS szoftver csomag
* Lucas Nülle PRO/TRAIN



**A jelentkezés feltételei:**

* Érdeklődés a PLC alapú vezérlések és vizualizációjuk iránt
* Angol nyelvtudás

# Tartalmi összefoglaló

* **A téma megnevezése:**

Ipari folyamat emulálása Hardware-In-the-Loop (HIL) eszközzel és ennek irányítása programozható logikai vezérlővel.

* **A megadott feladat megfogalmazása:**

Emulálnom és irányítanom kell egy ipari folyamatot, amelyik egy korábbi Országos Ajtonyi István Irányítástechnikai Programozó Versenyen szerepelt. Az irányításhoz Programozható Logikai Vezérlőt (PLC) kell alkalmaznom, az emulációhoz pedig a WinFACT szoftvercsomagot. Az alap feladatkiíráson túl plusz feladatként meg kell terveznem egy olcsóbb alternatívát a Lucas Nülle I/O interfész helyettesítésére.

* **A megoldási mód:**

A feladatkiírásban szereplő szoftvercsomagok és a hozzájuk tartozó technológiák megismerése és ismertetése. A kiválasztott ipari folyamat emulációjának és a hozzá tartozó irányításnak a megtervezése és megvalósítása, majd ezek tesztelése. Az I/O interfész és a hozzá tartozó szoftver közötti kommunikáció feltárása, tanulmányozása és az eredmények alapján egy saját eszköz megtervezése és tesztelése.

* **Alkalmazott eszközök, módszerek:**
* Alkalmazott eszközök: Lucas Nülle I/O interfész, Omron CJ2M-CPU32 PLC, CX-Programmer 9.1, mikroC PRO for PIC 6.6.2 és Microsoft Visual Studio 15 fejlesztőkörnyezetek, WinFACT 7 BORIS tervezőszoftver, CadSoft EAGLE 7.5.0 NYÁK-tervező szoftver, MikroElektronika EasyPIC v7 fejlesztőlap.
* Alkalmazott módszerek: kommunikáció, fejlesztőkörnyezet és fejlesztőeszközök tanulmányozása, fejlesztés, tesztelés, áramkör-szimuláció, mérés és kiértékelés.
* **Elért eredmények:**

A WinFACT 7 szoftvercsomag BORIS (Block Oriented Simulation System) szoftverét kiismertem és dokumentáltam. Megterveztem és teszteltem a kiválasztott ipari folyamatot – karosszéria gyártósor –, illetve elkészítettem a hozzá tartozó irányítást. Megvizsgáltam az I/O interfész és a BORIS közötti kommunikációt, az I/O interfész felnyitása nélkül. Az eredmények alapján megterveztem egy alternatív áramkört, amelyik képes az említett eszközt helyettesíteni. A terveket szimulációval és egy fejlesztőlap segítségével teszteltem, majd az elkészíttetett eszközzel sikeresen működtettem az emulációt.

* **Kulcsszavak:**

ipari informatika, rendszer szimuláció, irányítás, HIL, PLC, technológia-visszafejtés

# Tartalomjegyzék

[Tartalmi összefoglaló 4](#_Toc466835180)

[Tartalomjegyzék 5](#_Toc466835181)

[Bevezetés 6](#_Toc466835182)

[1. A HIL szimulációhoz rendelkezése álló eszközök 8](#_Toc466835183)

[1.1. Az ipari folyamat emulációját lehetővé tevő programok 8](#_Toc466835184)

[1.2. Az emulált technológia és az irányítást végző eszköz közötti kapcsolat 8](#_Toc466835185)

[1.3. Az irányítást megvalósító programozható logikai vezérlő 8](#_Toc466835186)

[2. Az emuláció bemutatására kiválasztott ipari folyamat 9](#_Toc466835187)

[3. A BORIS használatának gyakorlati bemutatása 10](#_Toc466835188)

[3.1. A példaprogram ismertetése 10](#_Toc466835189)

[3.2. A példaprogram megvalósításának bemutatása 10](#_Toc466835190)

[3.3. A példaprogramhoz tartozó vizualizáció megvalósításának bemutatása 10](#_Toc466835191)

# Bevezetés

Az iparban használt rendszerek üzembe helyezése rendkívül magas költségekkel jár. A költségek egyik forrása a ráfordított idő, amit az ilyen rendszerek tervezésének és fejlesztésének körülményessége okoz. Sok esetben az automatizálást végző beágyazott rendszerek fejlesztése az irányítani kívánt rendszerre kapcsolva történik, ami a körülményektől függően akár azt is megkövetelheti, hogy az ilyen munkálatok a beüzemelés helyszínén történjenek. Az ilyen nehézségek csökkentése érdekében ma már HIL (Hardware-In-the-Loop) szimulátorok is alkalmazhatók, amik nem csak az irányítani kívánt rendszerek helyettesítését képesek ellátni a fejlesztési munkálatok alatt, de a napjainkban egyre kiemeltebb szempontot képviselő minőség biztosításában is jelentős szerepet kaphatnak. A végtelenítve futó automata-teszt rendszerek információt adnak a rendszer megbízhatóságáról, amik a HIL szimulátorok alkalmazásával költséghatékonyan megvalósíthatók. Mindezek mellett a HIL szimulátorok jól alkalmazhatók az oktatásban és a különböző képzési programokban is.

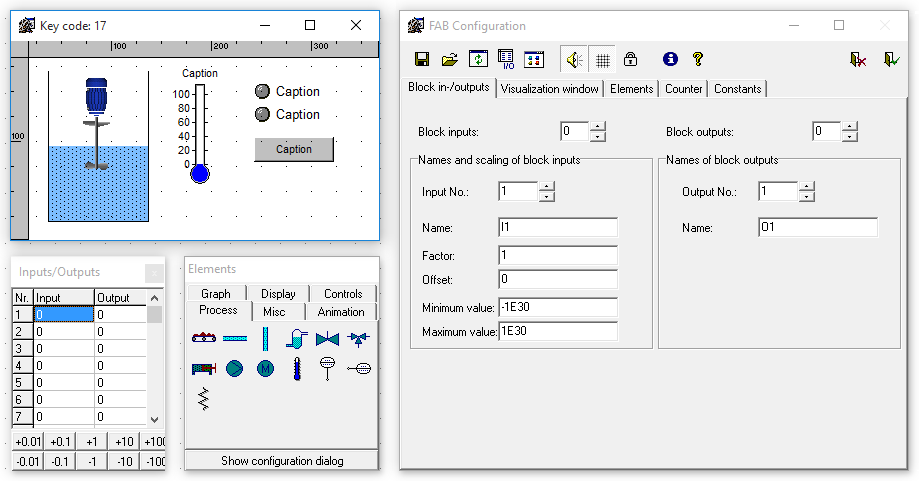
A diplomamunkám elsődleges célja egy létező HIL szimulátor tanulmányozása, a használatának a bemutatása egy konkrét példán keresztül és egy lehetséges helyettesítő eszköz megtervezése. A bemutatásra szánt példa egy karosszéria gyártósor lesz, aminek az irányítását az ipari automatizálásban elterjedt módon, egy programozható logikai vezérlővel fogom megvalósítani. A helyettesítő eszköz megtervezéséhez a helyettesíteni kívánt I/O interfészt annak megbontása nélkül fogom tanulmányozni, ellenkező esetben ugyanis az I/O interfész elvesztené a garanciális javítás lehetőségét. A másik célom a helyettesítő eszköz megvalósítása és összehasonlítása a helyettesíteni kívánt I/O interfésszel alkalmazhatósági és ráfordítási szempontokból, illetve a bemutatásra szánt példa futtatásával mindkét eszközön.

Az 1. fejezetben azokat az eszközöket mutatom be, melyeket a Műszaki Informatika Tanszék biztosított a HIL szimuláció megvalósítására. Ezeknek az eszközöknek a tanulmányozását Görbedi Ákos kollégámmal együtt végeztük. A 2. fejezetben ismertetem az emulálni kívánt ipari folyamatot, a karosszéria gyártósort, majd a 3. fejezetben a vizualizációra fókuszálva egy gyakorlati példán keresztül mutatom be a BORIS használatát. A 2. fejezetben ismertetett technológia és a hozzá tartozó irányítás megvalósítását a 4. fejezetben foglalom össze. Az 5. fejezetben a BORIS projektek hordozhatóságának problémájáról és az általam megvalósított megoldásról írok. Ezeket követően a 6. fejezetben rátérek az alkalmazott I/O interfész és az általam tervezett helyettesítő eszköz műszaki jellemzőire, majd a 7. fejezetben összegzem a diplomamunka készítése során szerzett tapasztalatokat és az elért eredményeket.

# A HIL szimulációhoz rendelkezése álló eszközök

## Az ipari folyamat emulációját lehetővé tevő programok

FAB…



1.. ábra: Flexible Animation Builder

## Az emulált technológia és az irányítást végző eszköz közötti kapcsolat

## Az irányítást megvalósító programozható logikai vezérlő

# Az emuláció bemutatására kiválasztott ipari folyamat

# A BORIS használatának gyakorlati bemutatása

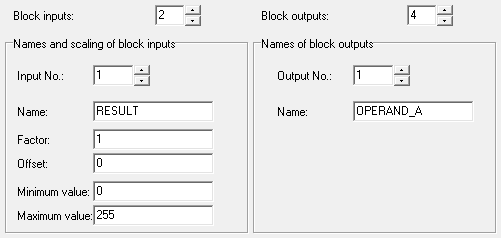
## A példaprogram ismertetése

## A példaprogram megvalósításának bemutatása

## A példaprogramhoz tartozó vizualizáció megvalósításának bemutatása

A 3.2. fejezet jól példázza, hogy a BORIS-ban készült szimulációk nem feltétlenül igényelnek vizualizációt, az eszköztára számos olyan blokkot biztosít, amikkel a felhasználó kölcsönhatásba kerülhet a futó szimulációkkal. Ennek ellenére a vizualizációk sokkal áttekinthetőbb interfészeket biztosíthatnak a szimulációk számára, illetve lehetőségeket adnak a rendszerek könnyebb megértését biztosító animációk elkészítésére is.

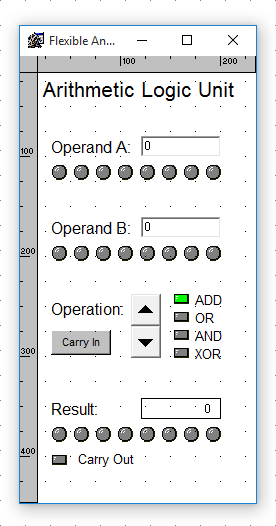
Egy új vizualizáció létrehozásához a blokkdiagramon a *Blocks 🡪 User-DLL-Block 🡪 Flexible Animation Builder* menüponton keresztül elérhető blokkot kell elhelyezni. Ez a blokk az eszköztár *User* fülén is megtalálható ugyanilyen néven. A vizualizáció szerkesztéshez elő kell hívni a FAB szerkesztőfelületét (1.1. ábra), ami az imént említett blokk szerkesztőablakából a *Dialog…* gombra kattintva érhető el. Itt legjobb első lépésként beállítani a be- és kimeneteket, ami a *FAB Configuration* ablak *Block in-/outputs* fülén végezhető el. A példában a vizualizáció legyen egyben beviteli és megjelenítő felület is. Ehhez kettő bemenetre és négy kimenetre lesz szükség (3.1. ábra).



3.. ábra: a vizualizáció be- és kimeneteinek beállító felülete

Az első bemenet fogadja majd az ALU eredményét, a másik bemenet pedig az átvitel jelzőbitjét. A négy kimenet közül az első kettő szolgáltatja az operandusokat, a harmadik a művelet sorszámát, a negyedik pedig a kezdeti átvitelt jelző bitet.

A munkafelületre új elemeket az *Elements* ablakban megjelenő eszköztárról lehet behelyezni. Amennyiben ez az ablak nem jelenne meg, a munkafelületen előhívható helyi menüből előtérbe hozható az *Elementfenster* menüpontra kattintva. Az operandusok beviteléhez két beviteli mezőre lesz szükség, melyek a *Controls* fülön *Edit field* néven találhatók meg. Emellett a logikai műveletek jobb szemléltetése érdekében az operandusok és az eredmény binárisan is meg lesznek jelenítve, amikhez LED elemekre is szükség lesz (3.2. ábra). A LED-ek a *Display* fülön *LED* néven találhatók meg.



3.. ábra: az ALU példaprogram vizualizációja