DFIR-PLC INTERFÉSZ

műszaki leírás

**Tartalomjegyzék**

[1. A DFIR rendszer funkciói 3](#_Toc413322607)

[2. A DFIR rendszer felépítése 4](#_Toc413322608)

[2.1 A DFIR rendszer felosztása üzemeltetők szerint 6](#_Toc413322609)

[2.2 A DFIR rendszer működése 7](#_Toc413322610)

[3. VATECH rendszer 8](#_Toc413322611)

[3.1 A PLC feladata 8](#_Toc413322612)

[3.2 A PLC Hardver ismertetése 8](#_Toc413322613)

[3.3 A PLC kommunikáció beállításai 8](#_Toc413322614)

[3.3.1 A PLC Ethernet kártya beállítása 8](#_Toc413322615)

[3.3.2 Az Ethernet hálózat topológiája 9](#_Toc413322616)

[3.3.3 Az Ethernet hálózat beállítása 9](#_Toc413322617)

[3.4 Szúrásterv kezelés 10](#_Toc413322618)

[3.4.1 A PLC-be leküldött szúrásterv adatok 10](#_Toc413322619)

[3.4.2 A PLC által küldött válasz a DFIR-nek 11](#_Toc413322620)

[3.4.3 PLC Szúrásterv-kezelés folyamatábrája 12](#_Toc413322621)

[3.5 Hengerlési adatok 13](#_Toc413322622)

[3.6 Lecsévélőről levett tekercsadatok 13](#_Toc413322623)

[3.7 Felcsévélőről levett tekercsadatok 14](#_Toc413322624)

[3.8 Tekercslevétel nyugtázó telegram 15](#_Toc413322625)

[3.9 Életjelküldés 15](#_Toc413322626)

[3.10 Mérlegelés 16](#_Toc413322627)

[3.10.1 A mérlegelés kommunikációs protokollja 16](#_Toc413322628)

[3.10.2 DISOMAT\_F\_MEAS 16](#_Toc413322629)

[3.10.3 Soros kommunikációs kábel 17](#_Toc413322630)

[3.11 MOXA ETHERNET – RS232 átalakító 18](#_Toc413322631)

[3.11.1 MOXA átalakító beállítása 20](#_Toc413322632)

[3.11.2 A beállítások: 23](#_Toc413322633)

[3.11.3 A beállítások visszaállítása 23](#_Toc413322634)

[4. DFIR – PLC Interfész 25](#_Toc413322635)

[4.1 Virtuális számítógép 25](#_Toc413322636)

[4.2 MySQL adatbázis 26](#_Toc413322637)

[4.3 A DFIRPLC program 29](#_Toc413322638)

[4.3.1 DFIRPLC Program felépítése 30](#_Toc413322639)

[4.3.2 Szúrásterv kezelése 32](#_Toc413322640)

[4.3.3 Hengerlési adatok 33](#_Toc413322641)

[4.3.4 Tekercslevétel 34](#_Toc413322642)

[4.3.5 A DFIRPLC használata 35](#_Toc413322643)

[4.3.6 PLC Üzenetek panel 36](#_Toc413322644)

[4.3.7 Dresszírozó panel 37](#_Toc413322645)

[4.3.8 – Esemény Napló panel 38](#_Toc413322646)

[4.3.9 Esemény napló fájlok 39](#_Toc413322647)

[5. Mellékletek 40](#_Toc413322648)

[5.1 Disomat\_F\_Meas SCL kódja 40](#_Toc413322649)

[5.2 Moxa Nport beállítása 47](#_Toc413322650)

# A DFIR rendszer funkciói

A DFIR[[1]](#footnote-1) a következő funkciókat látja el:

* SAP[[2]](#footnote-2)–VATECH[[3]](#footnote-3) adatkapcsolata:

A dresszírozásra küldött tekercseket az SAP átadja a DFIR –nek. A kezelő kiválasztja a <http://dfirapp.ms.dunaferr.hu/page.html> „Tekercsléptető” oldalról a leküldhető tekercsek közül a küldeni kívánt tekercseket. A leküldés után a tekercsek megjelennek a VATECH rendszerben a HMI[[4]](#footnote-4)-n a szúrásterv kezelő ablakban. A kihengerelt tekercsek visszaadásra kerülnek az SAP felé.

* Technológia adatok tárolása:

A DFIR tárolja a kihengerelt Tekercsek minősítési, technológiai és egyéb adatokat, valamint a VATECH rendszerből kapott részletes nyúlási, hengerlési erő, síkkifekvés, sebesség, stb. rekordokat, melyek a <http://dfirapp.ms.dunaferr.hu/regf.html> oldalon megtekinthetőek.

* Automatikus súlymérés

A DFIR minden tekercslevételnél lekérdezi a tekercsleszedő kocsiba épített mérleg által mért súlyt, és továbbítja az SAP felé.

* Vonalkód tikett nyomtatás

A tekercs elkészítése után a DFIR rendszer minden tekercshez vonalkód tikettet nyomtat.

# A DFIR rendszer felépítése

A DFIR rendszer sematikus ábrája az 1-es ábrán látható. A rendszer a következő elemekből épül fel:

1. HMI
2. Mérleg
3. Moxa Ethernet - RS232 Átalakító[[5]](#footnote-5)
4. Zebra tikett nyomtató
5. Siemens S7-400 PLC[[6]](#footnote-6)
6. DFIR –PLC Interfész (Virtuális PC)
7. DFIR Szerver (Virtuális PC)
8. SAP
9. Web kliensek

# DFIRrendszerfelépítés.jpeg

. ábra – DFIR rendszer sematikus ábrája

A rendszer elemeinek feladata:

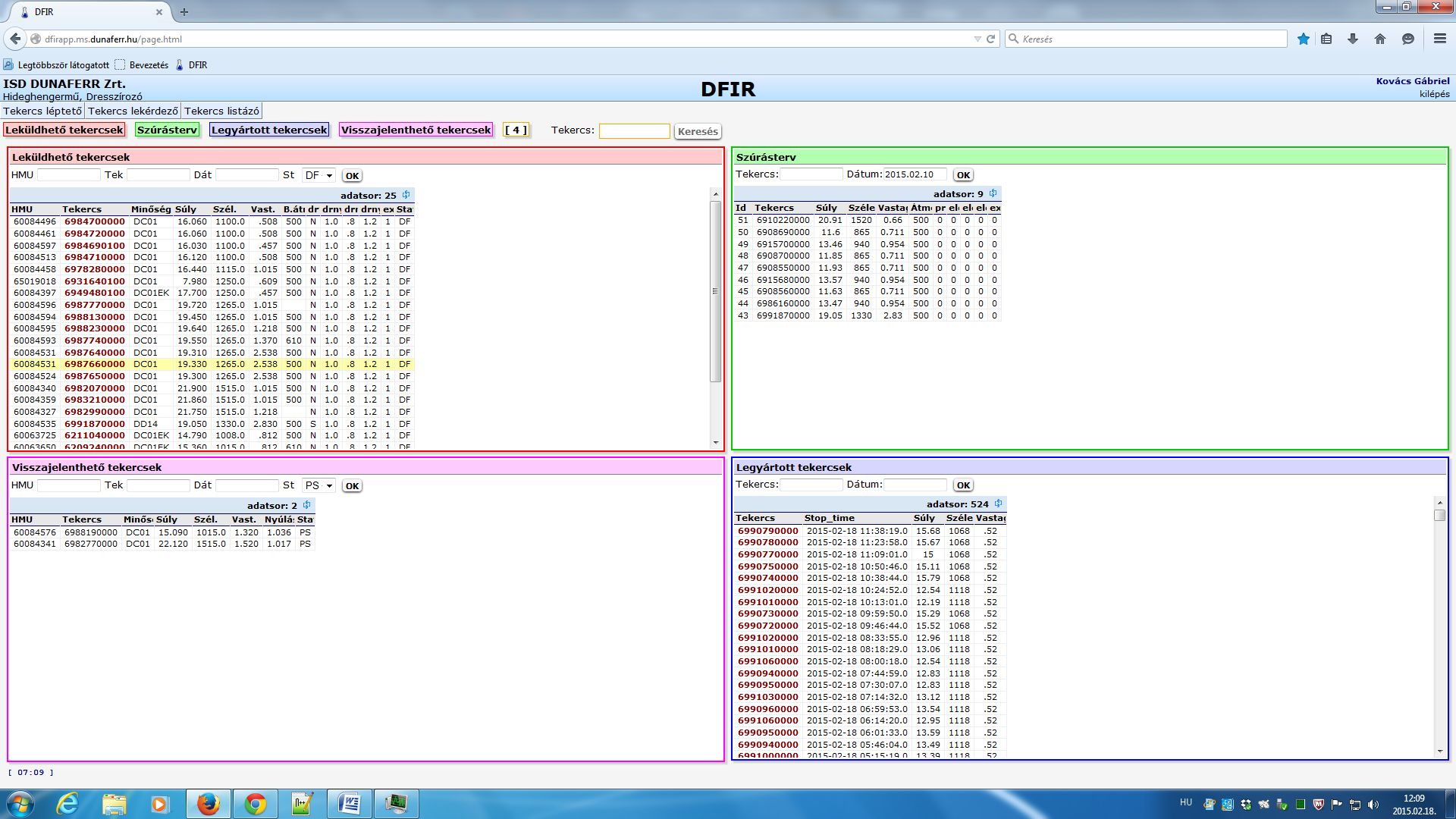
1. HMI: A főpulton elhelyezett számítógépes vezérlő képernyő, amely a technológiai folyamatirányító rendszer része. A HMI-n a szúrásterv képernyő kapja meg a DFIR rendszerből a leadott tekercseket, és a hozzá tartozó adatokat.
2. Mérleg: A felcsévélőn a levett tekercsek súlyát méri. A Disomat mérleg kiértékelő a főpulton az operátor mellett van elhelyezve.
3. Moxa Ethernet – RS232 átalkító: A Disomat mérleg kiértékelője soros porton kommunikál. A PLC-ben Ethernet interfész van. Az átalakítón keresztül történik a kommunikáció a mérleg és a PLC között.
4. Zebra tikett nyomtató: A vonalkód nyomtató a DFIR szerverről nyomtat. A nyomtatást a web kliensen keresztül indíthatjuk el. A nyomtató a Dresszírozói étkezőben található.
5. Siemens S7400 PLC: A PLC a technológiai folyamatirányító rendszer feladatát látja el. A PLC –én keresztül küldjük el a DFIR rendszer felé a DFIR-ben rögzíteni kívánt technológiai adatokat, illetve az anyagkövetéshez tartozó eseményeket. PL.: Tekercslevétel. A DFIR –ből leadott tekercsek adatait a PLC küldi tovább a technológiai képernyőre.
6. DFIR – PLC Interfész (Virtuális PC[[7]](#footnote-7)): Az informatika szerverén fut egy virtuális számítógép. Ez a számítógép virtuális Ethernet hálózaton keresztülkommunikál az AP1-es PLC-vel. A PLC-től érkező hengerlési adatokat összegyűjti, és letárolja a MYSQL adatbázisba. A DFIR szervertől érkező szúrásterv adatokat átküldi a PLC-nek.
7. DFIR Szerver (Virtuális PC): Az informatika szerverén fut egy másik virtuális számítógép is. A DFIR – PLC Interfész adatbázisból átmásolja az adatokat az SAP rendszerbe. Ez a számítógépen keresztül lehet a DFIR rendszert vezérelni bármelyik web kliensről. Ez a pc nyomtatja ki a tiketteket.
8. SAP rendszer: A vállalatrányitási rendszer. A 3-as szintű termelésirányítást valósítja meg.
9. Web kliensek: Azok a számítógépek melyek rendelkeznek modern web böngészővel, és fizikailag össze van kötve a Dunaferres belső hálózattal. A web böngészőn keresztül lehet megnézni a tekercsek hengerlés során regisztrált adatait, illetve a DFIR rendszert vezérelni.

## A DFIR rendszer felosztása üzemeltetők szerint

1. Villamos üzem : A villamos üzem üzemelteti a HMI-t, a Moxa Ethernet-RS232 átalakítót, a Siemens S7400 PLC-t, és a DFIR –PLC Interfészen futó programokat.
2. Informatika: Az informatika üzemelteti a Zebra tikett nyomtatót, a DFIR szervert, a Dunaferr hálózatot, és az SAP rendszert.
3. Mérlegszerelők: A mérlegszerelők üzemeltetik a mérleget.

## A DFIR rendszer működése

A dresszírozón dolgozó operátor az SAP –ből leküldi a dresszírozni kívánt tekercset a DFIR –be. A DFIR rendszert a <http://dfirapp.ms.dunaferr.hu/page.html> weboldalra történt bejelentkezés után lehet vezérelni. A tekercsléptető képernyőn látható a DFIR-be leadott tekercsek helyzete. Az SAP –ból leadott tekercsek megjelennek a leküldhető tekercsek között. A tekercs kiválasztásával a szúrástervet át lehet küldeni a VATECH rendszerbe. Ekkor a tekercsadatok megjelennek a DFIR –ben a szúrásterv képernyőn. Ezzel egy időben megjelenik a tekercs a technológiai HMI –n a szúrásterv kezelő táblázatban. A tekercs dresszírozása közben a PLC 100ms időnként elküldi a hengerlési adatokat a DFIR rendszer felé. Amikor a tekercs elkészült a mérleg megméri a tekercs súlyát, és a sikeres mérlegelés után a PLC elküldi a levett tekercs adatait a DFIR rendszer felé. Ekkor a tekercs a DFIR-ben a Legyártott tekercsek között fog megjelenni, és a Zebranyomtató kinyomtatja a kis tikettet. A legyártott tekercs kijelölésével át lehet küldeni a Visszajelenthető tekercsek közé. Ekkor a felugró ablakban a mérlegelt súlyt tudjuk hitelesíteni. A hitelesített tekercsek átkerülnek a visszajelenthető tekercsek közé, és addig lesznek láthatóak, amíg a tekercseket legyártását nem rögzítik az SAP-ban.



. ábra - Tekercs léptető

# VATECH rendszer

## A PLC feladata

Az Ethernetes kommunikáció AP1 –es PLC –be lett megírva. A PLC a DFIR-PLC interfésszel. és a mérleggel kommunikál. A PLC feladata a kommunikációk vezérlése és az adatok feldolgozása.

## A PLC Hardver ismertetése

A PLC Hardver konfigurációjában a „=10AP1+E37E1.VH1-A09” –es tervjelű CP443-1 kommunikációs kártya lett beüzemelve. Így most két Ethernet vezérlő kártya van a PLC-ben. Az eredeti vezérlő kártya kommunikál a simítói belső hálózattal, illetve a mérleggel. Az új vezérlő kártya kommunikál a vasműs hálózaton keresztül a DFIR – PLC interfésszel. Az új kártyára azért volt szükség, hogy a simítói hálózat továbbra is el legyen szeparálva a Dunaferr hálózattól.

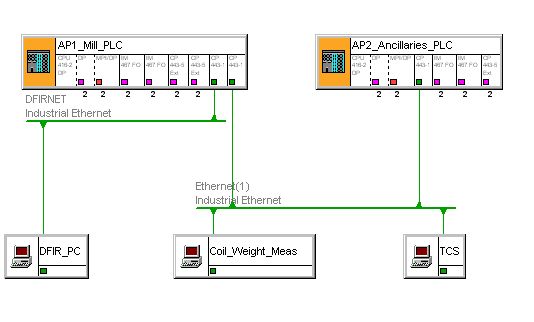
## A PLC kommunikáció beállításai

### A PLC Ethernet kártya beállítása

|  |  |  |
| --- | --- | --- |
|  | Ethernet kártya CP443-1 | Ethernet kártya CP443-1 |
| slot | 11 | 12 |
| Name | CP 443-1 DFIR | CP 443-1 |
| IP address | 192.168.210.11 | 172.24.2.160 |
| Subnet mask | 255.255.255.0 | 255.255.255.192 |
| Gateway | Do not use router | Do not use router |

. táblázat - Ethernet kártya konfigurálása

### Az Ethernet hálózat topológiája



. ábra PLC hálózati topológiája.

### Az Ethernet hálózat beállítása

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Megnevezés | Local ID | Local Address  (PLC IP cím:port) | Partner Address (Partner IP cím:port) | Sender (Küldő) DB | Receiver (Fogadó) DB |
| Tekercslevétel nyugtázó telegram | 0007 A050 | 192.168.210.11:2010 | 192.168.210.10:2010 |  | DB881 |
| Hengerlési adatok | 0008 A050 | 192.168.210.11:2011 | 192.168.210.10:2011 | DB882 |  |
| Felcsévélőről levett tekercsadatok | 0009 A050 | 192.168.210.11:2012 | 192.168.210.10:2012 | DB883 |  |
| Lecsévélőről levett tekercsadatok | 000A A050 | 192.168.210.11:2013 | 192.168.210.10:2013 | DB884 |  |
| Szúrásterv adatok | 000B A050 | 192.168.210.11:2014 | 192.168.210.10:2014 |  | DB885 |
| Szúrásterv nyugtázás | 000C A050 | 192.168.210.11:2015 | 192.168.210.10:2015 | DB886 |  |
| Életjel | 000D A050 | 192.168.210.11:2016 | 192.168.210.10:2016 | DB887 |  |
| Tekercssúly lekérés | 000F A020 | 172.14.1.160:2100 | 172.24.1.138:2100 | DB3001 | DB3001 |

. táblázat - Ethernet hálózat beállítása

## Szúrásterv kezelés

A HMI-n a szúrásterv kezelő képernyőn 50 tekercset lehet tárolni. Az eltárolt tekercsek a HMI szerver belső memóriájába kerülnek. A PLC-ben az első 20 tekercsek azonosítóit lehet megtalálni. Ezért a HMI újraindítás után a tekercsek elvesznek. A HMI a leadott szúrástervet a szúrásterv-kezelő képernyő első szabad helyére teszi be. Ha már nincs szabad hely a HMI egyszerűen eldobja a tekercs adatokat. Ezért nagyon fontos hogy csak akkor küldjünk a HMI –nek új tekercsadatot, ha van szabad hely. Ezt egy másodpercenként futó scripttel ellenőrizzük a HMI-n, és PLC-ből elküldjük a DFIR felé. A DFIR csak akkor küld újabb szúrástervet, ha a PLC ezt engedélyezi.

### A PLC-be leküldött szúrásterv adatok

A PLC-be a szúrásterv adatokat a DB885 –ös adatblokkba tároljuk le. A PLC-be a következő adatok érkeznek meg:

|  |  |  |  |
| --- | --- | --- | --- |
| Address | Name | Type | Comment |
| 0.0 | DwaPssSchTelId | INT |  |
| 2.0 | DwaCoilId | STRING [ 14 ] | DWA beállítási adat: tekercsazonosító |
| 18.0 | DwaSteelGrade | STRING [ 20 ] | DWA beállítási adat: Acélminőség |
| 40.0 | DwaWidth | REAL | DWA beállítási adat: Szélesség [mm] |
| 44.0 | DwaThickness | REAL | DWA beállítási adat: Vastagság [mm] |
| 48.0 | DwaWeight | REAL | DWA beállítási adat: tekercssúly [kg] |
| 52.0 | DwaLength | REAL | DWA beállítási adat: tekercshossz [m] |
| 56.0 | DwaExitCoilsNo | INT | DWA beállítási adat: gyártandó tekercsek száma |
| 58.0 | DwaExitCoil1Length | REAL | DWA beállítási adat:az 1. Kész tekercs hossza [m] |
| 62.0 | DwaExitCoil2Length | REAL | DWA beállítási adat: a 2. Kész tekercs hossza [m] |
| 66.0 | DwaProcessType | INT | DWA beállítási adat: folyamat típusa |
| 68.0 | DwaElongation | REAL | DWA beállítási adat: nyúlás [%] |
| 72.0 | DwaRollForce | REAL | DWA beállítási adat: hengerlési erő [kN] |
| 76.0 | DwaBendingForce | REAL | DWA beállítási adat: hajlító erő [kN] |
| 80.0 | DwaLineSpeed | REAL | DWA beállítási adat: sori sebesség [m/min] |
| 84.0 | DwaBasicSprayAmount | REAL | DWA beállítási adat: alap felszórt mennyiség [l/min] |
| 88.0 | DwaTensionPorEsBr | REAL | DWA beállítási adat: feszítés FCS-Beo S-ggő [N/mm2] |
| 92.0 | DwaTensionEsBrStd | REAL | DWA beállítási adat: FEszítés Beo S ggő-állvány [N/mm2] |
| 96.0 | DwaTensionStdXsBr | REAL | DWA beállítási adat: Feszítés Állv-Kio Sggő [N/mm2] |
| 100.0 | DwaTensionXsBrTr | REAL | DWA beállítási adat: Feszítés KioSggős-FCS [N/mm2] |
| 104.0 | DWACoreDiameter | REAL | DWA beállítási adat: magátmérő (belső menetek) [mm] |
| 108.0 | DwaElongLowLim | REAL | DWA beállítási adat: nyúlás alsó határ |

. táblázat - DB885 Szúrásterv adatok

A „*DwaPssSchTelId”* az üzenet azonosítót tartalmazza. A DFIR számozza a telegramok számát. 0–től 999 -ig. A számozást egyesével növeli, majd 999 után újra az 0-ás jön. Ebből az azonosítóból tudja a PLC, hogy új telegram érkezet. A „*DwaCoilId*” a hidegtekercs számot tartalmazza. A többi mező egyértelmű a táblázatban.

### A PLC által küldött válasz a DFIR-nek

A PLC 100mS időnként elküldi a nyugtázó üzenetet a DFIR-nek . A telegram üzeneteit a DB886 –os adatblokk tárolja:

|  |  |  |  |
| --- | --- | --- | --- |
| Address | Name | Type | Comment |
| 0.0 | PassSchedTelId | INT | Üzenetazonosító |
| 2.0 | PassSchedRecHMI | INT | Pass Schedule Received by HMI (1- ready for next telegram) |

. táblázat - DB886 szúrásterv válasz telegram

A *„PassSchedTelId”* az utoljára fogadott szúrásterv azonosítóját küldi vissza (*DwaPssSchTelId*). A DFIR ebből az üzenetből fogja tudni, hogy az elküldött üzenetet a PLC megkapta.

A *„PassSchedRecHMI”* üzenet jelzi a DFIR-nek, hogy a HMI képes fogadni új üzenetet, vagy nem. (Feltöltöttük az 50 mezőt a HMI-n). Amíg nem rendelkezünk újabb üres hellyel addig blokkolva lesz az új szúrásterv adatok küldése.

### PLC Szúrásterv-kezelés folyamatábrája

Az alábbi ábrán a szúrásterv-kezelés folyamatábrája látható:

4. ábra - szúrásterv kezelés folyamat

START

Érkezett új szúrásterv adat?

nem

igen

nem

„Szúrásterv küldés folyamatban van” jelzőbit beállítása.

igen

Szúrásterv kezelése a PLC-ben.  
Minden PLC ciklusban lefut.

END

Szúrásterv küldés folyamatban van?

Nyugtázó üzenet küldése a DFIR felé.

Szúrásterv küldés engedélyezése a DFIR-nek

nem

igen

+

+

Van még szabad hely a HMI-n a szúrásterv számára?

Szúrásterv küldés tiltása a DFIR-nek

„Szúrásterv küldése a HMI –nek.

A HMI a szúrásterv beillesztése után törli a „szúrásterv folyamatban van” jelzőbitet.

## Hengerlési adatok

A PLC hengerlés közben 100 msec-onként küldi a DFIR felé a hengerlési adatokat. A telegram üzeneteit a DB882 adatblokk tárolja:

|  |  |  |  |
| --- | --- | --- | --- |
| Address | Name | Type | Comment |
| 0.0 | TelegrId | INT | távirat azonosító szám [1..999] |
| 2.0 | CoilId | STRING[14] | Tekercsazonosító (tekercsszám) |
| 18.0 | CoilPartNo | INT | A vágás után képzett tekercsrész-szám [0..999] |
| 20.0 | ProcType | INT | A feldolgozás típusa [0-száraz, 1-nedves] |
| 22.0 | ElongRef | REAL | Nyúlás alapjel (a szúrástervből) |
| 26.0 | ElongAct | REAL | Tényleges nyúlás (a TCS -től) |
| 30.0 | FlatDevZone1 -32 | REAL[32] | Síkfekvés-hiba (a TCS-től) 1-32 zóna |
| 158.0 | RollingSpeed | REAL | Hengerlési sebesség [m/min] (a TCS-től) |
| 162.0 | HGCRollForceAct | REAL | Hengerlési erő összesen CSO + HO [kN] |
| 166.0 | HGCRollForceDiffAct | REAL | Hengerlési erő eltérés HO - CSO [kN] |
| 170.0 | WRBending | REAL | Hengerhajlítás [kN] |
| 174.0 | TensionPOR\_EsBr | REAL | Feszítés a lecsévélő és a bemenő S-ggő között [kN] |
| 178.0 | TensionEsBr\_Ms | REAL | Feszítés a bemenő S-ggő és a hengerállvány között [kN] |
| 182.0 | TensionMs\_XsBr | REAL | Feszítés hengerállvány és a kimenő S-ggő között [kN] |
| 186.0 | TensionXsBr\_TR | REAL | Feszítés kimenő S-ggő és a felcsévélő között [kN] |
| 200.0 | TrRollStripLength | REAL | Kihengerelt hossz a felcsévélőn [m] |

5. táblázat - DB882 hengerlési adatok

Az üzenetekből a DFIR fogja elkészíteni a tekercshez tartozó rekordokat.

## Lecsévélőről levett tekercsadatok

Ha a tekercsfeladó kocsival leveszik a tekercseket, vagy a tekercsmaradékot akkor a tekercsszámot tartalmazó telegramot a PLC elküldi a DFIR-nek. A lecsévélőről levett tekercs adatait a DB884 adatblokk tárolja.

|  |  |  |  |
| --- | --- | --- | --- |
| Address | Name | Type | Comment |
| 0.0 | RemovePORTelId | INT | Anyag levéve az LCS-ről távirat ID |
| 2.0 | CoilId | STRING[14] | Tekercsazonosító (tekercsszám) |
| 18.0 | CoilPartNo | INT | A vágás után képzett tekercsrész-szám [0..999] |

. táblázat - DB884 Lecsévélőről levett tekercsadatok

A „*RemovePORTelId*” 1… 999-ig növekedő azonosító.

## Felcsévélőről levett tekercsadatok

Ha a tekercsleszedő kocsival leszedjük a tekercset, a PLC lekérdezi a mérlegtől a tekercs súlyát. Ha a mérlegelés sikerült, vagy 30s alatt nem sikerült a mérlegelés a tekercsadatokat elküldjük a DFIR felé. A tekercsleszedés folyamatábrája:

5. ábra - Felcsévélői tekercslevétel folyamatábrája

Tekercslevételkor adatok küldése

START

Tekercslevétel van?

nem

igen

nem

**Tekercslevétel telegram küldés** jelzőbit beállítása. DB2000.DBX76.1

igen

END

Mérlegelés sikerült?

**Tekercslevétel telegram küldés** jelzőbit törlése. DB2000.DBX76.1

nem

igen

+

+

**Tekercslevétel telegram-küldés** van?

Mérlegelt súly tárolása a DB883-ban

Tekercsadatok másolása a DB883-ba

+

igen

A mérlegelés több mint 30s ideje kezdődött?

+

nem

+

Levett tekercsadatok küldése a DFIR-nek.

Mérleg súly lekérdezése

+

Mérlegelés sikerült?

nem

Levett tekercsadatok küldése sikerült?

igen

nem

A felcsévélőről levett tekercsadatok a DB 883-ban tároljuk:

|  |  |  |  |
| --- | --- | --- | --- |
| Address | Name | Type | Comment |
| 0.0 | RemoveTRTelId | INT | Anyag levéve az FCS-ről távirat ID |
| 2.0 | CoilId | STRING[14] | Tekercsazonosító (tekercsszám) |
| 18.0 | CoilPartNo | INT | A vágás után képzett tekercsrész-szám [0..999] |
| 20.0 | CoilWidth | INT | Lemez szélesség (mm) |
| 22.0 | CoilMeasWeight | INT | Tekercs mért súly [kg] |
| 24.0 | CoilThickness | INT | Lemez vastagság |
| 26.0 | CoilCalcWeight | INT | Tekercs számított súly [kg] |

7. táblázat - DB883 Felcsévélőről levett tekercsadatok

## Tekercslevétel nyugtázó telegram

A Lecsévélő és a felcsévélő telegram küldésére a DFIR visszaküldi a telegram azonosítót. A PLC ebből fogja tudni, hogy megkapta a DFIR a telegramot. Ha nem érkezik válasz a DFIR-től akkor a PLC újraküldi a telegramot. A nyugtázó telegramot a DB881-es adatblokkban tároljuk.

|  |  |  |  |
| --- | --- | --- | --- |
| Address | Name | Type | Comment |
| 0.0 | RemoveTRTelId | INT | Anyag levéve az FCS-ről távirat ID |
| 2.0 | RemovePORTelId | INT | Anyag levéve az LCS-ről távirat ID |

. táblázat - DB881 lecsévélőről levett tekercsadatok

## Életjelküldés

A DFIR és a PLC adatokat a Dunaferr hálózatán keresztül küldjük. Vannak olyan hálózati eszközök, amik újraépítik a kapcsolat jelet, így kommunikációs hiba esetén a DFIR nem mindig veszi észre, hogy kommunikációs hiba van. Az életjelet a DB887 adatblokkból küldjük.

|  |  |  |  |
| --- | --- | --- | --- |
| Address | Name | Type | Comment |
| 0.0 | LifeSignal | INT | Életjel |
| 2.0 | LifeSignalError | INT | ÉletjelHiba |

. táblázat - DB887 életjelküldés

A LifeSignal 1…9999 között számol. Az értékét eggyel növeljük, ha sikerült elküldeni az üzenetet. 9999 után 1 következik. A LifeSignalError akkor számol, ha nem sikerült elküldeni az üzenetet.

## Mérlegelés

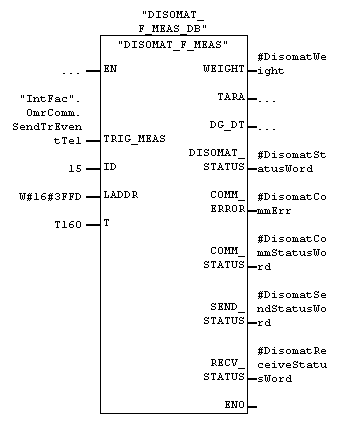
A mérlegelés a Disomat F típusú mérleggel történik. A PLC a Moxa Ethernet - RS232 Átalakítón keresztül kommunikál a mérleggel

### A mérlegelés kommunikációs protokollja

A mérleggel a **SCHENCK Poll Protokollal (DDP 8785)** kommunikálunk. A mérlegnek a „*<STX>00#TG#<ETX><BCC>”* üzenetet küldjük, ha le szeretnénk kérni a mérlegtől a súlyt. Az <STX> az üzenet kezdetét az <ETX> az üzenet végét jelenti. A köztük lévő üzenet az adat. A <BCC> az ellenőrző adat. Az értéke a kezdő <STX> kivételével a csomag minden bájtja XOR-olva. A TG jelenti, hogy a súlyt szeretnénk lekérdezni. Ha a mérleg válaszol, akkor a következő üzenetet kapjuk: „*<STX>00#TG#súly (7 karakter)#tára (7 karakter)#dG/dt (7 karakter)#státusz (2 karakter)#<ETX><BCC>”.* A súly lekérdezés után másodperce van a mérlegnek arra, hogy válaszoljon. Ha ez nem történik meg, akkor újra lekérhetjük a súlyt. Ha a válaszban a státusz 128 akkor a mért adatok hitelesek. Ilyenkor fogja a PLC letárolni az adatokat.

### DISOMAT\_F\_MEAS

A PLC –ben a súly lekérését a *„DISOMAT\_F\_MEAS”* Blokk végzi el.



6. ábra - DISOMAT\_F\_MEAS blokk

*Bemenetek:*

**TRG\_MEAS:** Súly lekérdezése a mérlegtől

**ID:** Partner id (NetPro-ból)

**LADDR:** CP címe

**T:** Az időtúllépéshez használt Timer

*Kimenetek:*

**WEIGHT:** Mért tömeg [kg]

**TARA:** Tára [kg]

**DG\_DT:**dG/dt [kg]

**DISOMAT\_STATUS:** Mérleg állapota **COMM\_ERROR:** Kommunikációs hiba

**COMM\_STATUS**: Kommunikáció állapota

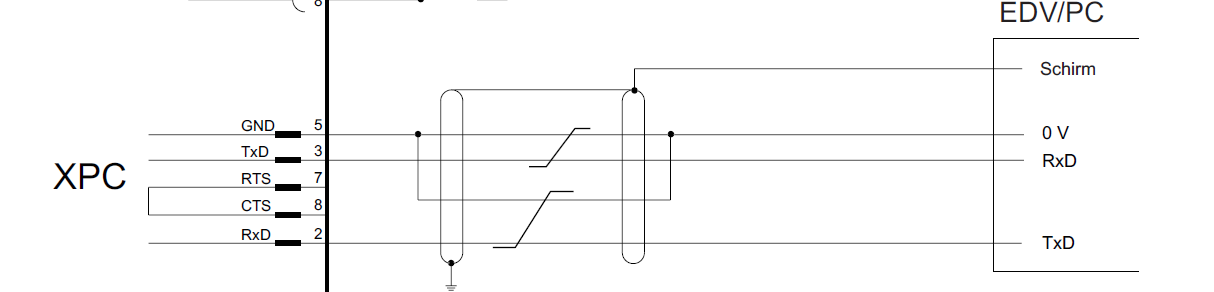
**SEND\_STATUS**: ETH küldés állapota

**RECV\_STATUS:** ETH fogadás állapot

A Disomat\_F\_Meas Blokk működésének a programját az fejezetben lehet megtekinteni.

### Soros kommunikációs kábel

A mérleg és a MOXA közötti kommunikációs kábel bekötését az alábbi ábra mutatja.

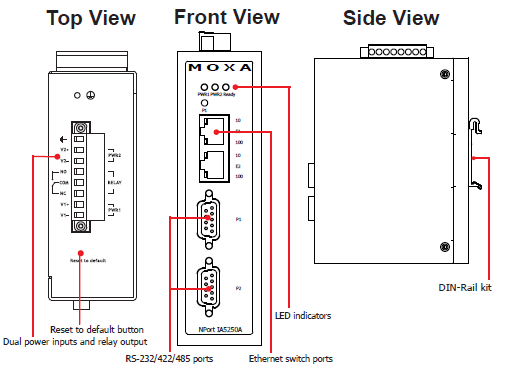


. ábra - Mérleg kommunikációs kábel

A mérlegre a kommunikációs kábelt az XPC pontjára kell bekötni.

## MOXA ETHERNET – RS232 átalakító

A PLC-nek nincs soros portja, ezért közvetlenül nem tud kommunikálni a mérleggel. Megoldásként egy MOXA gyártmányú NPort IA5250I típusú Ethernet –RS232 átalakító lett beüzemelve. Az átalakító a 10AP1.VP4 szekrény (főpult) bal oldalára van felszerelve. A bekötése a „=10AP1.VP4+D43P1.VP4-A1” tervjelen található a kapcsolási rajzban.

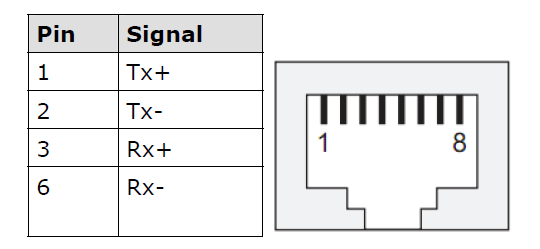


. ábra - NPort IA5250AI

Az NPortIA5250 –es átalakítónak, kettő RS-232/422/485 3 DB9 soros portja van az adatkommunikációhoz. A relés kimeneteket nem használjuk. A készülék tetején található a reset gomb. Ha a reset gombot 5 másodpercen keresztül folyamatosan benyomva tartjuk, a készüléket visszaállítjuk a gyári beállításokra.

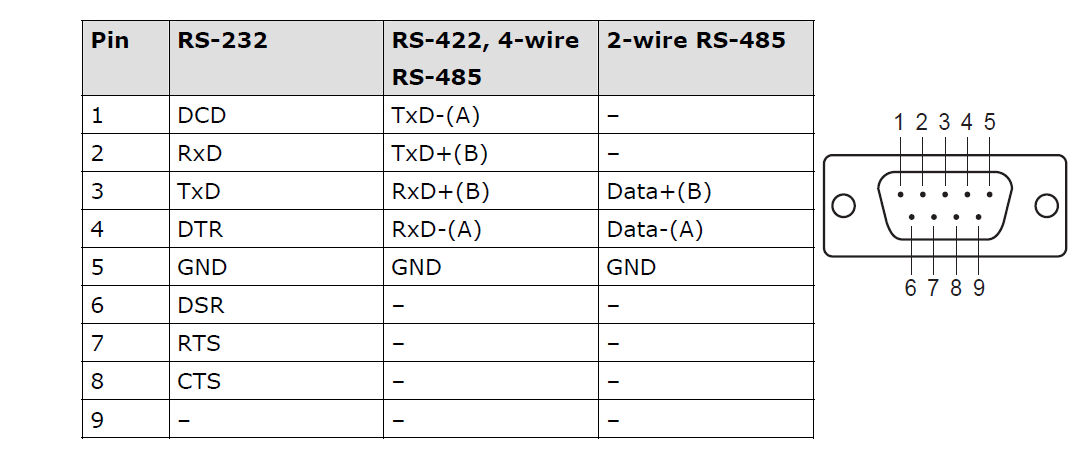
* Portok lábkiosztása:

Ethernet port lábkiosztása:



. ábra - MOXA Ethernet port lábkiosztása

A soros port lábkiosztása:



. ábra - MOXA RS232/422/485 (apa DB9) lábkiosztása

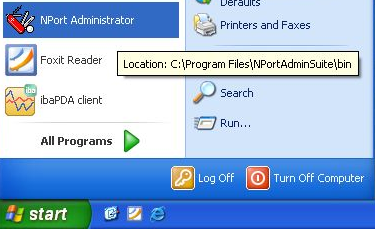
* Az előlapi ledek jelentései:

|  |  |  |
| --- | --- | --- |
| **Név** | **Szín** | **Jelentés** |
| PWR1 PWR2 | piros | A tápfeszültség rendben van a PWR1, PWR2 bemeneten. |
| Ready | Piros | Feszültség rendben van, a készülék boot-ol. |
| piros villogó | IP hibát jelez. A DHCP vagy a BOOTP szerver nem válaszol. |
| zöld | A tápfeszültség rendben van a készülök rendben működik |
| zöld villogó | Az eszköz szerveren az Adminisztrátor aktiválta a "Locate" funkciót |
| nem világit | Tápfeszültség kikapcsolva, vagy tápfeszültség hiba van. |
| E1, E2 | narancs | 10 Mbps Ethernet kapcsolat |
| zöld | 100 MBPS Ethernet kapcsolat |
| nem világit | Ethernet kábel nincs bedugva, vagy meghibásodott. |
| P1,P2 | narancs | Soros port fogadja az adatot |
| zöld | Soros port küldi az adatot |
| nem világit | Nincs adatátvitel a soros porton |

. táblázat - Ledek jelentései

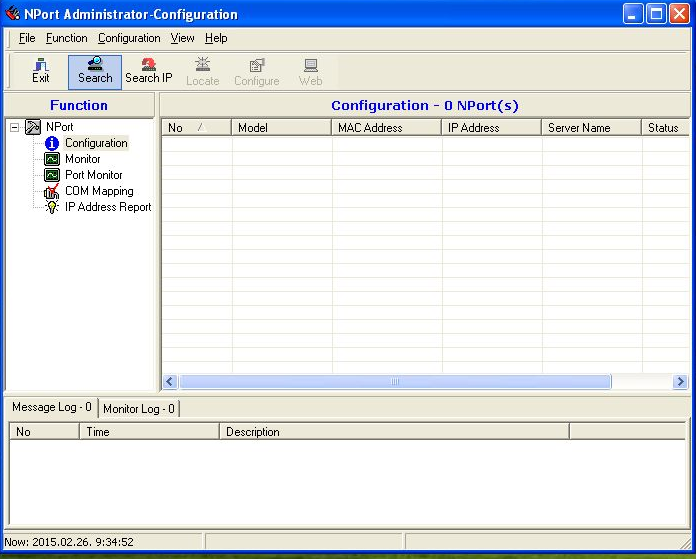
### MOXA átalakító beállítása

A MOXA átalakítót a dresszírozói főpulti PLC programozó számítógépről lehet elérni. A beállítások eléréséhez az „NPort Administrator” programot kell elindítani.



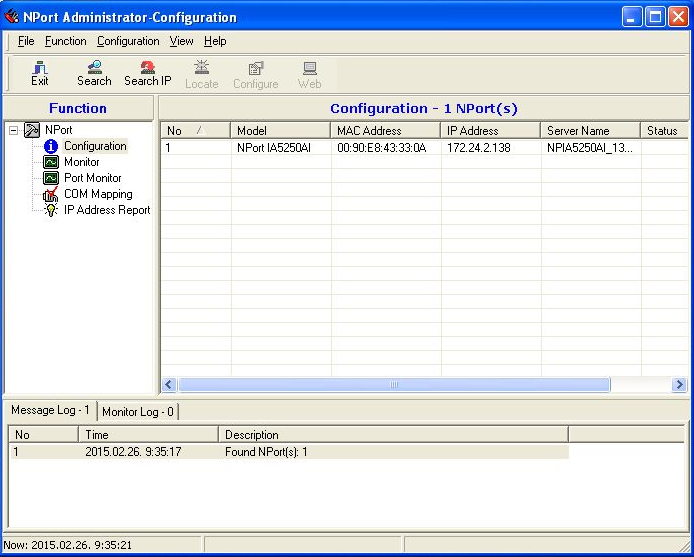
. ábra - program elindítása

A program elindítása után meg kell keresnünk az Ethernet hálózaton az eszközünket. Ehhez kattintsunk a „Search” nyomógombra.



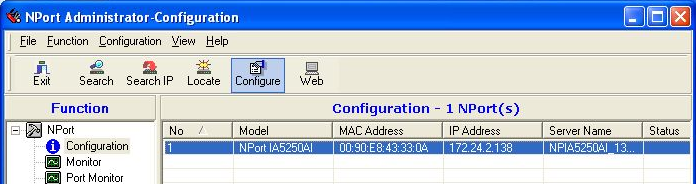
. ábra - eszköz keresése

Ha a keresés sikeres volt, akkor a táblázatban megjelennek a megtalált eszközök:



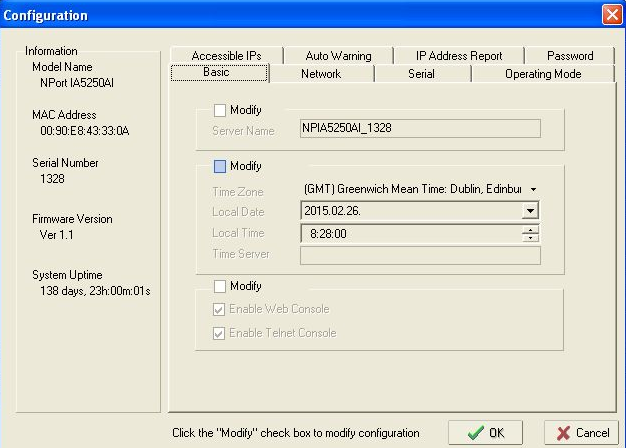
. ábra - a keresés eredménye

Válasszuk ki a 172.24.2.138-as IP címen működő eszközt: és kattintsunk a „Configure” nyomógombra:



. ábra - konfigurálás

A „Basic” fül beállítása:



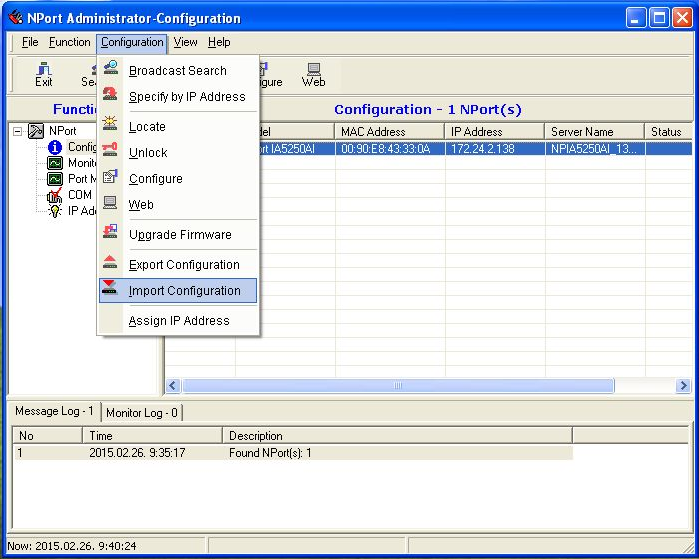
. ábra- Basic beállítás

### A beállítások:

* Basic:
  + Server Name: „NPIA5250AI\_1328” (Tetszőlegesen választható)”
  + Time Zone: „(GMT) Greenwitch Mean Tim: Dublin…”
  + „Enable Web console” kiválasztva
  + „Enable Telnet Console” kiválasztva
* Network:
  + IP Address: „172.24.2.138”
  + Netmask: „255.255.255.192”
  + „Enable SNMP” aktiválva
* Serial:
  + 1. Settings: „9600,N,8,No flowctrl”
  + 2. Settings: 115200,N,8,1,RTS/CTS (Nem használjuk)
* Operating Mode
  + 1. OP Mde: „TCP Server Mode”
  + 2. Disable (Nem használjuk)

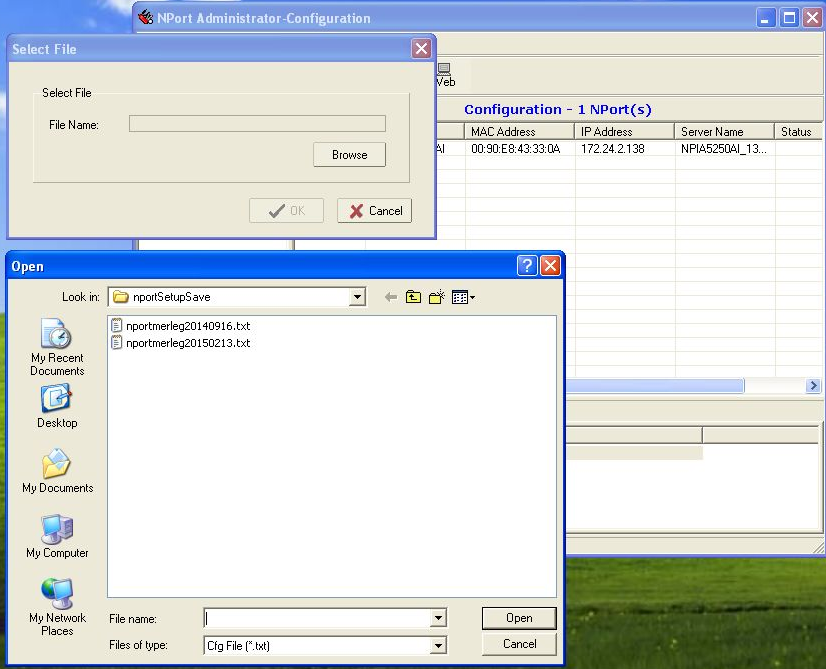
### A beállítások visszaállítása

Ha elvesztek a beállítások, akkor a korábbi mentés alapján vissza lehet állítani. A 5.2 –es „Moxa Nport beállítása” fejezetben tekinthető meg az elmentett beállítás. A beállítás visszaállításához indítsuk el az NPort Administrator programot, majd keressük meg az eszközünket a hálózaton. Válasszuk ki a 172.24.2.138-as IP címen működő eszközt és a menüből válasszuk ki a Configuration / Import Configuration menüpontot.



. ábra - konfiguráció importálása

A felugró „Select File” ablakban kattintsunk a Browse nyomógombra, és válasszuk ki a legutolsó konfigurációs fájlt. Majd kattintsunk az Open gombra, majd az Ok gombra. Az eszköz újraindítás után az új beállításokkal fog működni.



. ábra - konfiguráció importálása 2

# DFIR – PLC Interfész

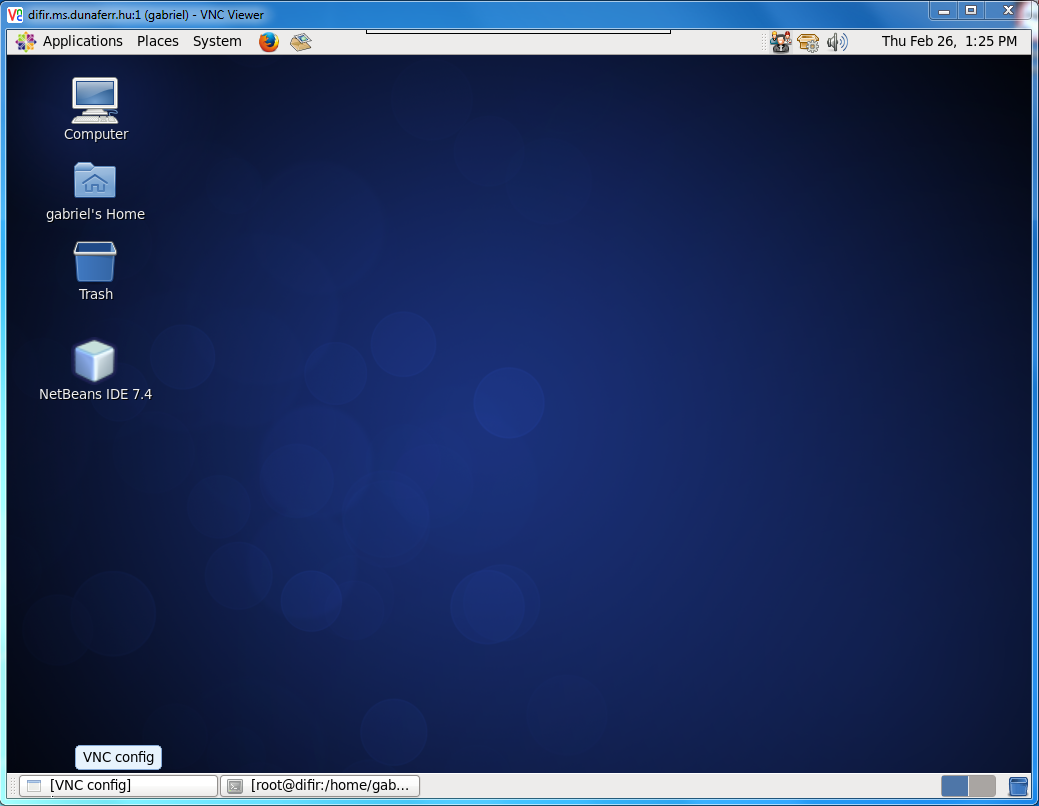
A PLC és DFIR rendszer között nincs közvetlen kapcsolat. A kommunikációt egy DFIRPLC nevű java nyelven írt alkalmazással oldottuk meg. Ez az alkalmazás Etherneten kommunikál a PLC-vel, és MySQL adatbázison keresztül adja át az adatokat a DFIR-nek. UDP üzeneteken keresztül „sor termel, vagy nem termel jelet küld a Dresszírozón elhelyezett operátor számítógépének, és ez alapján tud működni a Centralográf Terminál program[[8]](#footnote-8).

A Program az informatikusoknál valamelyik szerverre telepített virtuális számítógépről fut. ezen a számítógépen van a MYSQL adatbázis is.

## Virtuális számítógép

A számítógépet a Dunaferr belső hálózatáról lehet elérni VNC –n Keresztül. A VNC szerver címe: difir.ms.dunaferr.hu:5901. A belépéshez szükséges jelszó: Qwer1234. A belépés után egy Linuxos bejelentkező képernyő fogad minket. Itt a belépéshez szükséges felhasználónév: gabriel, jelszó: Qwer1234.

Bejelentkezés után egy CENTOS Linux operációs rendszer fogad minket.



. ábra – Virtuális számítógép

Telepített alkalmazások:

* NetBeans IDE 7.4 : Egy ingyenesen használható Integrált fejlesztő környezet. A java program fejlesztése ezzel az eszközzel történt. A DFIRPLC programot futtatás előtt mindig ezzel lett lefordítva.
* Java: A virtuális számítógépen telepített java verziószáma: 1.7.1\_51
* Apacche 2.2.15 : Webszerver program
* PHP 5.3.3: Általános szerveroldali szkriptnyelv dinamikus weblapok készítésére
* MySQL 5.1.73 : Adatbázis
* phpMyAdmin 4.1.9: Webes felületű adatbázis kezelő MySQL-hez.
* Wireshark : Hálózati forgalomfigyelő.

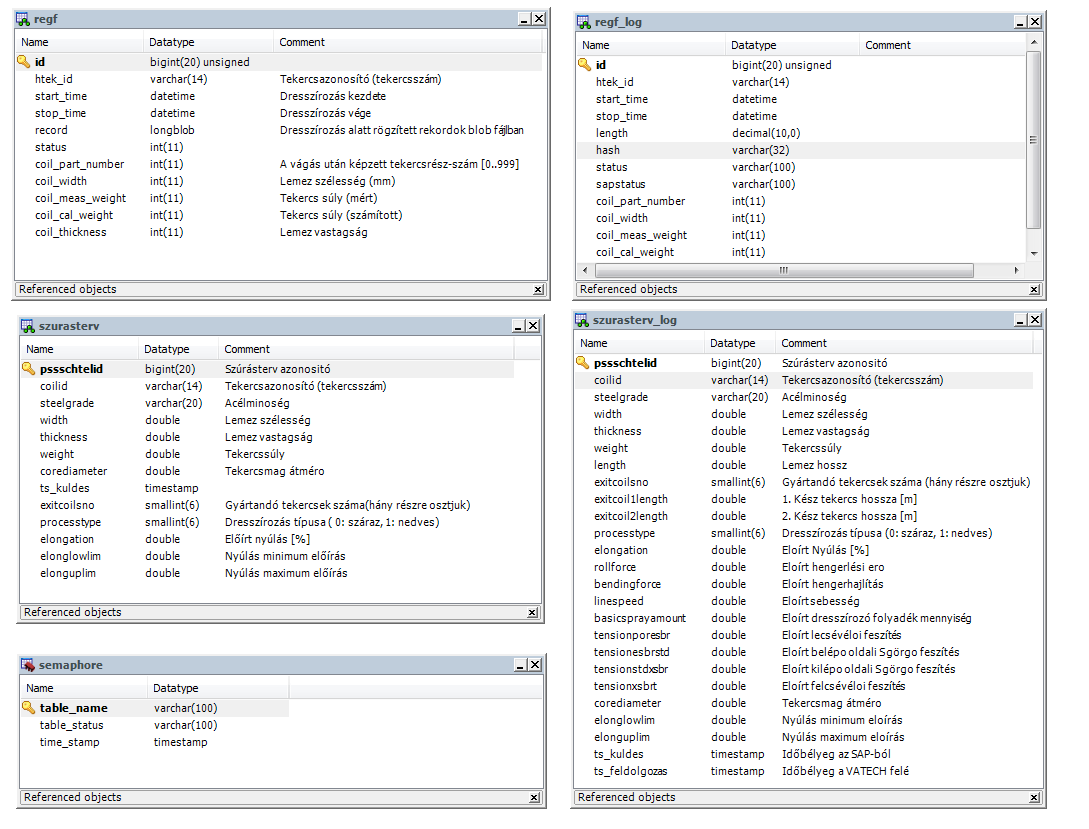
## MySQL adatbázis

MySQL adatbázisba lettek létrehozva a DFIR rendszerhez tartozó interfész táblák. Az adatbázist a difir.ms.dunaferr.hu címen lehet elérni. A felhasználó név gabriel, a jelszó Qwer1234, a port száma 3306. Az adatbázis neve: dfir.

Az adatbázis a következő táblákból áll:

* regf: A hengerlés során regisztrált adatok kerülnek ide letárolásra.
  + id: automatikusan növekedő futósorszám
  + htek\_id: hidegtekercs szám,
  + start\_time: a dresszírozás kezdete:
  + stop\_time: a dresszírozás vége.
  + record: Az összegyűjtött rekordokat tartalmazza BLOB[[9]](#footnote-9) típusban. Az 1 db rekord megegyezik az „” táblázatban látható adatokkal, de a tekercsazonosító nem szerepel benne. A status mező egy, ha rendben van minden adat.
  + coil\_width a PLC által számolt lemez szélesség
  + coil\_meas\_weight a mérleg által mért tekercs súly
  + coil\_cal\_weight a PLC által számított tekercs súly
  + coil\_thickness a lemez vastagsága.
* regf\_log: Az eredeti terv szerint az adatok törlésre kerülnek a regf táblából miután a DFIR rendszer feldolgozta őket. Azért hogy a folyamatokat nyomon lehessen követni, készítünk a bejegyzésekről egy másolatot aregf\_log-ba ami megegyezik regf tábla adataival a record mező nélkül. A length mezőbe a rekord hossza kerül, a hash[[10]](#footnote-10) mezőbe a rekord hash-elt értéke kerül. Illetve van két státusz mező is.
* szurasterv: a DFIR –ből leadott tekercsek szúrásterv adatai találhatók itt. Csak olyan tekercsek szerepelnek a táblában, amelyek még nem lettek elküldve a PLC-nek. Ha minden jól működik, akkor a tábla üres.
  + psssschtelid: automatikusan növekedő futósorszám
  + coilid: hidegtekercs szám
  + steelgrade: acélminőség
  + width: lemez szélesség
  + thickness: lemez vastagság
  + weight: tekercssúly
  + corediameter: Tekercsmag átmérő
  + ts\_küldes: A dátum, amikor le lett adva a szúrásterv az adatbázisba,
  + exitcoilsno: a gyártandó tekercsek száma
  + processtype: a dresszírozás típusa
  + elongation: nyúlás,
  + elonglowlim: nyúlás minimum előírás
  + elonguplim: nyúlás maximum előírás.
* szurasterv\_log: Ha a DFIRPLC program feldolgozta a szúrástervet, akkor leküldi a PLC-be és a leküldött adatok megjelennek a szurasterv\_log táblába. A szurasterv táblából egy az egybe átkerülnek az adatok a szurasterv\_log táblába, és kiegészítjük számított értékekkel: pl.: exitcoil1length. Vannak mezők, amit a PLC feldolgozna, de nem rendelkezünk a szükséges adatokkal, ezért 0 értékkel fogjuk elküldeni. Pl.: rollforce.
* semaphore: A szemafor táblába akkor ír be a DFIRPLC program, ha a regf táblába új mezőt tároltunk le. Innen fogja tudni a DFIR hogy új bejegyzés került a regf táblába.
  + table\_name: „dfir”
  + table\_status: „0”
  + time\_stamp: a bejegyzés ideje.

Az alábbi ábrán a dfir adatbázis táblái láthatóak:



. ábra - dfir adatbázis

## A DFIRPLC program

A DFIRPLC program java programnyelven lett megírva. A program a /home/gabriel/DFIRProject/DFIRPLC könyvtárban található. A program a számítógép elindításával azonnal indul. Ehhez az operációs rendszerben a System/Preferences/Startup Applicatons menüvel indítható programba a DFIRPLC nevű indító lett létre. Ez automatikusan elindítja a /home/gabriel/DFIRProject/DFIRPLC/startDFIRPLC.sh -t. Ez a shell szkript indítja el a DFIRPLC.jar fájlt, és gondoskodik arról, hogy program leállás esetén a program automatikusan újrainduljon.

A program a szúrástervet megkapja adatbázison keresztül a DFIR-től, és továbbküldi a PLC-nek a Dunaferres hálózaton keresztül. A PLC-vel a kommunikáció TCP/IP. A szúrásterv sikeres leküldése után a szurasterv\_log tábla feltöltésével tudatja a DFIR-nek, hogy a PLC megkapta a szúrástervet.

Hengerlés közben a DFIRPLC összegyűjti a hengerlési adatokat. Tekercslevétel esetén az összegyűjtött adatokat, és a tekercslevételkor elküldött tekercsadatokat beírja az SQL adatbázis regf táblájába.

### DFIRPLC Program felépítése

A program a MainApp.java osztállyal indul. Itt történik a változók inicializálása, és innen indítjuk a program taszkok futását. A MinApp.java induláskor a következő változókat inicializálja:

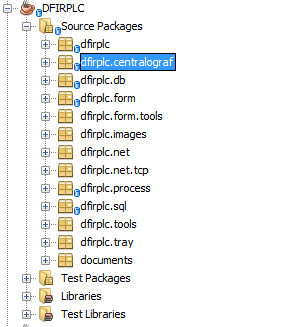
|  |  |  |  |
| --- | --- | --- | --- |
| Név | Típus | Alapértelmezett érték | Leírás |
| BUFFERSIZE | int | 1024 | kommunikációhoz használt Byte buffer mérete |
| IPADDRESS | String | "192.168.210.11" | PLC IP cime |
| FRAMEISENABLED | boolean | true | Frame engedélyezése |
| DRESSPANELISENABLED | boolean | true | DressPanel engedélyezése |
| LOGPANELISENABLED | boolean | true | Loggolás textareában engedélyezése |
| SENDTELEGRAMISENABLE | boolean | true | Telegram küldésének engedélyezése |
| CLASSTOSTRINGENABLE | boolean | false | Az osztályok printelésének engedélyezése |
| MESSAGELENGTHPRINTENABLE | boolean | false | Fogadott üzenetek hosszának kíírásának engedélyezése |
| CENTRALOGGRAFMESSAGEENABLE | boolean | true | Centralográf terminál programnak üzenet küldés engedélyezése |
| CENTRALOGGRAFIPADDRESS | String | "10.1.39.154" | Centrál terminál ip címe |
| LOGMODE | int | 1 | 0:=Minen log új fájlba, 1:= 31 napos naptár szerinti loggolás |
| PASSSCHEDULEENABLE | boolean | true | Szúrásterv leküldés engedélyezése |

. táblázat - Program futását vezérlő jelek

A változók megváltoztatása, csak a program lefordítása, és újraindítása után lép érvénybe.

A program több csomagba lett helyezve, a csomagok jól elkülönítik a programot funkciók szerint, így a program könnyebben áttekinthető. A Programban a következő csomagok találhatók:

* centralograf: A csomagban az állásidő terminál programnak a sor termel vagy nem termel jelet küldő osztályok találhatóak. A CentalografMessage osztály tárolja hogy a berendezés termel vagy áll. Az UDPConnectionServer és az UDP osztályba van megírva az UDP hálózati kapcsolat. A FillDataToBuffer osztály átalakítja a küldendő adatokat úgy hogy el tudjuk küldeni a hálózaton keresztül.
* db: Struct[[11]](#footnote-11) felépítésű osztályok. A PLC adatbázissal megegyezik a szerkezete. DB881 – DB887 –ig. A PLC-ben és a Java nyelvben az elemi típusok nem egyeznek meg. Az int –ből short lesz, a real-ből float, a String a lefoglalt string hossz + 2 byte. Az első byte a String lefoglalt hossza, a második byte a String hossza, majd a String következik.
* form: A megjelenítéshez tartozó form-ok vannak megírva ebben a csomagban.
* images: Az ikon képeket tartalmazza.
* net: A hálózati kommunikáció feldolgozásával kapcsolatos osztályokat tartalmazza.
* process: A program vezérlését végzi. Itt dől el, hogy mit csináljon a program ha üzenet kapott a PLC-től, vagy új bejegyzés került az adatbázisba.
* sql: Az adatbázis kapcsolatok kerültek ide.
* tools: Minden egyéb hasznos konvertáló stb. program.
* tray: A tálca programja került ide.



. ábra - program strukturálása

### Szúrásterv kezelése

A PLC 100 ms-onként küldi a DB886 adatblokkban található üzeneteket. Az üzenetben a legutoljára leadott szúrásterv azonosítója (nyugta), és egy jelzőbit található, ami azt tartalmazza, hogy a HMI képes fogadni az újabb szúrástervet, vagy nem. Ha a szúrásterv kezelés engedélyezve van, akkor a program (ProcessDB886) megnézi, hogy van e leküldendő szúrásterv az adatbázisban, ha van akkorleküldi a PLC-be, majd ellenőrzi, hogy a leküldés sikerült-e, és szükség szerint újraküldi. A sikeresen leadott tekercs adatait kitörli az adatbázis szurasterv táblájából és kibővített információkkal bemásolja a szurasterv\_log táblába. Az alábbi ábrán a szúrásterv küldés folyamat ábrája látható:

START

Szúrásterv kezelés engedélyezve?

nem

igen

Tekercsleküldés sikerült?

igen

SQL: Szúrástervlog tábla frissítése. SQL: Szúrásterv táblából tekercs eltávolítása. „Tekercsküldés volt” jelzőbit törlése

igen

Ez a ciklus akkor fut le, ha a plc-től kapjuk a szúrásterv triggert. (PLC DB886)

END

nem

PLC készen áll az újabb szúrásterv fogadására?

igen

nem

Tekercsleküldés volt?

Szúrásterv újraküldése.

Van leküldendő szúrásterv az adatbázisban?

SQL: Szúrástervlog új szúrásterv bejgyzése. Szúrásterv küldése. Tekercsküldés volt” jelzőbit beállítása

igen

nem

nem

21. ábra - szúrásterv küldés folyamatábrája

### Hengerlési adatok

A PLC hengerlés közben 100 ms-onként küldi a hengerlési adatokat. A program a hengerlési adatokat addig gyűjti, míg a hengerlés közben meg nem változik a tekercs száma, vagy a tekercset le nem vesszük a felcsévélőről. Ha a tekercset levettük a felcsévélőről, akkor a tekercshez tartozó összegyűjtött hengerlési adatokat letároljuk az adatbázisba. A DFIR ebből az adatból fogja elkészíteni a hengerlési diagramokat. Az alábbi ábrán a hengerlési adatokat feldolgozó program (ProcessDB882) folyamat ábráját mutatja:

22. ábra - hengerlési adatok feldolgozása

start\_time beállítva ?

A start\_time beállítása a jelenlegi időre

Tekercsszám ua. és nem volt tekercsleszedés ?

A tekercsleszedés volt a jelzőbitünket nullába állítjuk

Van rekord ?

A „tekercsleszedés volt” jelzőbit nullába állítása. Blob összeállítása. Adatok tárolása SQL adatbázisba.

A start\_time beállítása a jelenlegi időre

START

nem

igen

Tekercsszám nem egyenlő ”” –vel ?

igen

A rekordot hozzá adjuk a többi rekordhoz.

igen

Ez a ciklus akkor fut le, ha új rekord érkezik a hengerlés közben. (PLC DB882) Vagy tekercslevétel volt a felcsévélőn.

END

nem

nem

igen

nem

### Tekercslevétel

A PLC minden tekercslevételnél elküldi a tekercs azonosítót, a súlyt, a vastagságot, és a szélességet. A tekercslevétel üzenetből tudja a program, hogy a tekercs elkészült, és letárolja a hengerlési adatokat. A felcsévélő tekercslevételt feldolgozó program (ProcessDB883) folyamat ábrája:

23. ábra - tekercslevétel folyamatábrája

START

Tekercsleszedés telegram azonosító megváltozott?

A „tekercsleszedés volt” jelzőbit 1 be állítása. Nyugtázó telegram küldése a plc felé. Hengerlési adatokat feldolgozó programrész meghívása.(ProcessDB882)

nem

igen

Ez a ciklus akkor fut le, ha tekercslevétel volt a felcsévélőn. (PLC DB883)

END

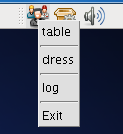
### A DFIRPLC használata

A program a virtuális számítógép elindulása után azonnal elindul. A program ikonja a tálca jobb oldalán látható.



24. ábra - DFIRPLC ikon

Az ikonra jobb egérrel kattintva a következő menü látható:



25. ábra - DFIRPLC menü

A table menüpont a PLC üzenetek nevű panelt nyitja, vagy zárja be. A PLC üzenetek ablakban a PLC-től érkező üzeneteket lehet megnézni, táblázat formájában. A dress menüben az aktuális dresszírozást áttekintő panelt nyitja, vagy zárja be. A log menü, egy egyszerűsített eseménynaplót nyitja meg, vagy zárja be. Az Exit menü bezárja a programot.

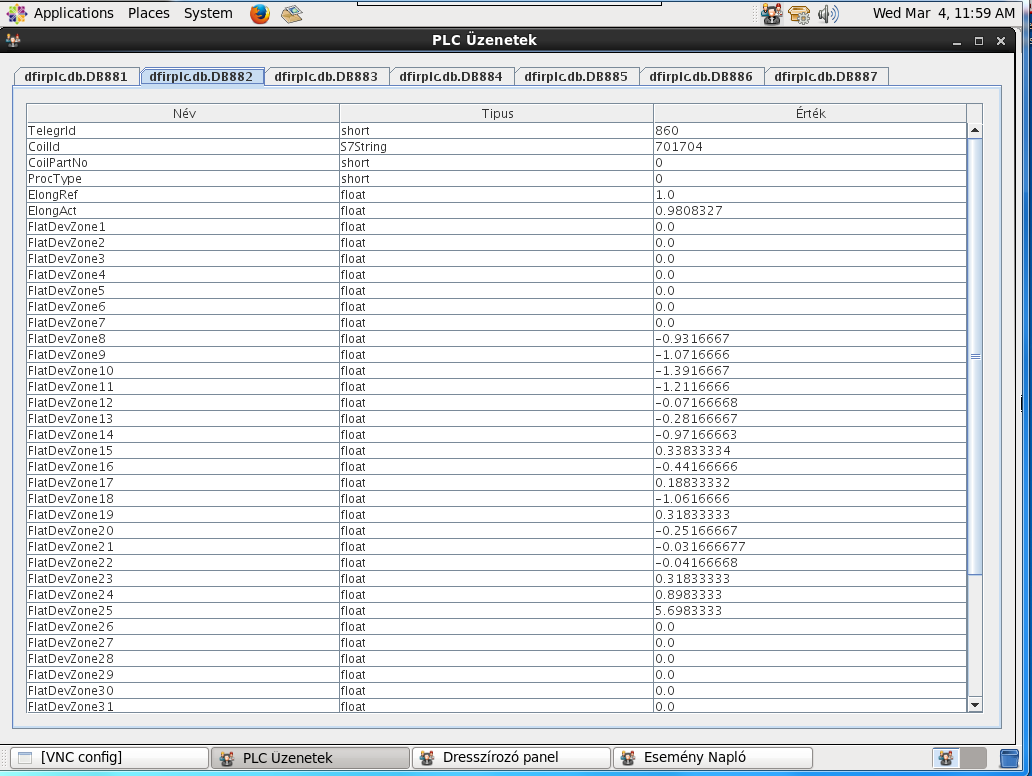
Figyelem! A programot csak akkor zárjuk be, ha a berendezés áll és a program letárolta a levett tekercs adatait. Egyéb esetben a program el fogja veszíteni az utoljára hengerelt tekercs adatait.

Mivel fut a háttérben a startDFIRPLC.sh szkript[[12]](#footnote-12) ezért a program automatikusan újra fog indulni. Ha a programot véglegesen le szeretnénk állítani, akkor először a szkriptet kell leállítani parancssorból. Az ablakok megnyitása vagy zárása attól függ, hogy éppen nyitva vagy zárva van.

Ha az ikonra a bal egérrel kétszer kattintunk, akkor a nyitott paneleket bezárja, és a bezárt paneleket megnyitja. A paneleket a jobb felső sarokban található kereszttel be lehet zárni. A program ettől még tovább fog dolgozni.

### PLC Üzenetek panel

A PLC üzenetek ablakban az utoljára érkezett, vagy küldött üzenet tartalmát lehet megtekinteni. Az ablak diagnosztikai célra szolgál.

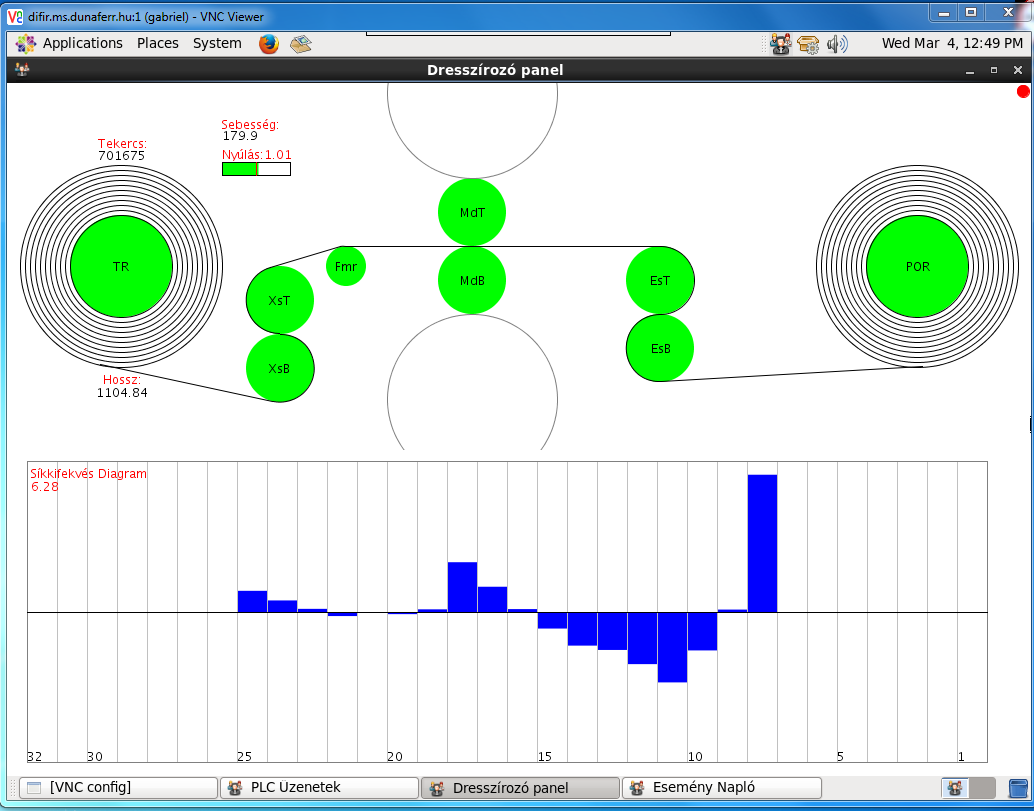


26. ábra - PLC Üzenetek

Az adatbázisok között a megtekinteni kívánt fül rákattintásával lehet váltani. Az adatokat nem lehet módosítani. A fülek elnevezése a <csomagnév>.<alcsomagnév>.osztálynév

### Dresszírozó panel

A dresszírozó panel az aktuális hengerlés ábráját mutatja:

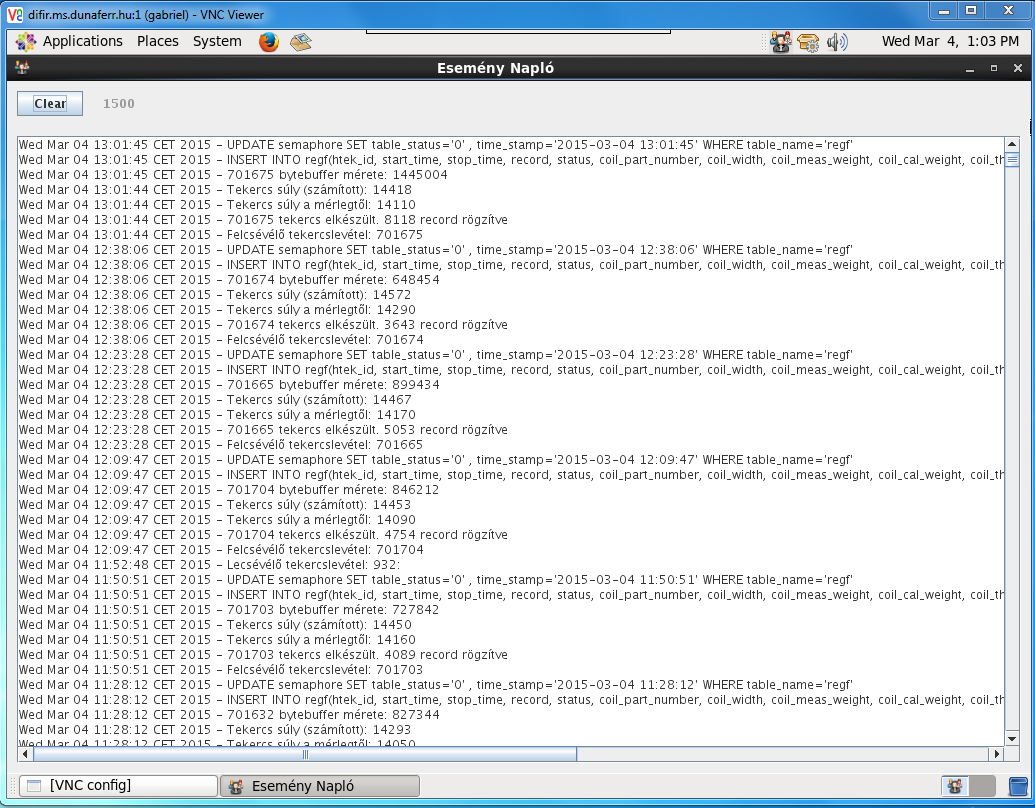


27. ábra - Dresszírozó panel

A dresszírozó panelen a hajtások zöldek, ha a berendezés termel, és kékek, ha a berendezés áll. Mikor a berendezés termel, akkor a jobb felső sarokban egy piros kör villog, ez azt jelzi, hogy a rekordokat rögzíti a program. Ha nincs felvétel, akkor egy fekete négyzet látható. Az ábrán látható a sebesség, a nyúlás, a tekercsszám, és a tekercs hossz. Az ábra alsó felében a síkkifekvés diagram látható.

### – Esemény Napló panel

Az eseménynapló panelre a java program működése közben eredetileg konzolra kiírt üzenetek kerültek átirányításra. A DFIRPLC program működése közben fellépő fontosabb eseményeket lehet beállítani. A legújabb üzenetek kerülnek felülre, és a legrégebbiek alulra. A Clear nyomógomb megnyomására az üzenetek törlődnek a naplóból. Kézzel nem lehet az üzeneteken módosítani.



. ábra - Esemény napló

### Esemény napló fájlok

A program a részletesebb eseményeket log fájlba menti. Ezekből a log fájlokból lehet visszaellenőrizni, hogy a DIFIRPLC program működése közben milyen események, hibák történtek. A program naptári nap szerint naplóz. A log fájlok a /home/gabriel/log könyvtárban vannak. A log fájl neve: debug.log.<hónap napja> . A log fájlok havonta felülírásra kerülnek.

Példa a log fájlra:



. ábra - debug.log

A log fájl első részében az időpont található, majd a következő szögletes zárójellel elválasztott részben a programrész neve, amelyik az eseményt írta, majd kerek záró jelek közé kerül, a „warning” vagy az „error” jelző, ha van. Ezután az esemény leírása következik.

# Mellékletek

## Disomat\_F\_Meas SCL kódja

FUNCTION\_BLOCK DISOMAT\_F\_MEAS

TITLE **=** 'DISOMAT F mért tömeg lekédezés'

// DISOMAT F mért tömeg lekédezés

// ------------------------------

//

// Protokoll: SCHENK DDP8785

//

// COMM\_STATUS értékei:

// 00 - A kommunikáció nincs megkezdve

// 01 - A kommunikáció folyamatban van

// 02 - A kommunikáció sikeresen lezajlott

// 10 - Hiba történt küldés közben (AG\_LSEND blokk hibára futott)

// Az AG\_LSEND blokk által küldött státusz a SEND\_STATUS kimeneten található.

// 20 - Hiba történt fogadás közben (AG\_LRECV blokk hibára futott)

// Az AG\_LRECV blokk által küldött státusz a RECV\_STATUS kimeneten található.

// 21 - Nem érkezett válasz a mérlegtol 5 sec-en belül (idotúllépés)

// 22 - A fogadott adatok hossza nem megfelelo (nem 39 karakter)

// 23 - A fogadott adatcsomag nem értelmezheto

AUTHOR**:** 'FD'

VERSION**:** '1.0'

// Bemeneti változók

**VAR\_INPUT**

TRIG\_MEAS**:** BOOL**;** // Súly lekérdezése a mérlegtol

ID**:** INT**;** // Partner id (NetPro-ból)

LADDR**:** WORD**;** // CP címe

T**:** TIMER**;** // Idotúllépés figyeléshez használt timer

**END\_VAR**

// Kimeneti változók

**VAR\_OUTPUT**

WEIGHT**:** INT**;** // Mért tömeg [kg]

TARA**:** INT**;** // Tára [kg]

DG\_DT**:** INT**;** // dG/dt [kg]

DISOMAT\_STATUS**:** WORD**;** // Mérleg állapota

COMM\_ERROR**:** BOOL**;** // Kommunikációs hiba

COMM\_STATUS**:** WORD **:=** 0**;** // Kommunikáció állapota

SEND\_STATUS**:** WORD**;** // ETH küldés állapot

RECV\_STATUS**:** WORD**;** // ETH fogadás állapot

**END\_VAR**

// Statikus változók

**VAR**

SEND\_ERROR**:** BOOL**;** // ETH küldés hiba

RECV\_ERROR**:** BOOL**;** // ETH fogadás hiba

DISOMAT\_SEND\_DATA**:** **ARRAY[**0**..**8**]** **OF** BYTE**;** // Mérlegnek küldendo adatok

DISOMAT\_RECV\_DATA**:** **ARRAY[**0**..**35**]** **OF** BYTE**;** // Mérlegtol érkezett adatok

COMM\_ENABLE**:** BOOL**;** // Kommunikáció engedély

SEND\_REQ**:** BOOL**;** // Küldés parancs

SEND\_LEN**:** INT**;** // Küldendo adatok hossza

SEND\_DONE**:** BOOL**;** // Küldés kész

RECV\_REQ**:** BOOL**;** // Fogadás parancs

RECV\_NDR**:** BOOL**;** // Fogadás - új adatok érkeztek

RECV\_LEN**:** INT**;** // Érkezett adatok hossza

TEMPINT1**:** INT**;** // Ideiglenes INT változó 1

TEMPINT2**:** INT**;** // Ideiglenes INT változó 2

TEMPBOOL1**:** BOOL**;** // Ideiglenes BOOL változó 1

TEMPCHAR1**:** CHAR**;** // Ideiglenes CHAR változó 1

TEMPWORD1**:** WORD**;** // Ideiglenes WORD változó 1

COMM\_TIMEOUT**:** BOOL**;** // Kommunikáció idotúllépés

TIMERTIME**:** S5TIME**;** // Idotúllépés hátralévo ido

TIMEOUT\_TIME**:** S5TIME**;** // Idotúllépés idotartam

TIMEOUT\_TIMER\_ON**:** BOOL**;** // Idotúllépés figyelo timer engedélyezése

TIMEOUT\_TIMER\_RESET**:** BOOL**;** // Idotúllépés figyelo timer resetelése

WEIGHT\_TMP**:** INT**;** // Ideiglenes tároló a súly ASCII->INT konverzióhoz

TARA\_TMP**:** INT**;** // Ideiglenes tároló a tára ASCII->INT konverzióhoz

DG\_DT\_TMP**:** INT**;** // Ideiglenes tároló a dG/dt ASCII->INT konverzióhoz

STATUS\_TMP**:** WORD**;** // Ideiglenes tároló a státusz ASCII->WORD konverzióhoz

SEQ\_STEP\_NR**:** INT**;** // Folyamat aktuális lépés

END\_VAR

**VAR\_TEMP**

DISOMAT\_RECV\_DATA2**:** **ARRAY[**0**..**35**]** **OF** BYTE**;** // Mérlegtol érkezett adatok

PTR**:** INT**;** //Mérlegtol érkezett adat eltolodás figyeléshez használt mutató

**END\_VAR**

// Kommunkiáció indítása és inicializálás

**IF** **(NOT** COMM\_ENABLE**)** **AND** TRIG\_MEAS **THEN**

COMM\_ENABLE **:=** TRUE**;**

SEQ\_STEP\_NR **:=** 0**;**

COMM\_ERROR **:=** FALSE**;**

COMM\_STATUS **:=** W#16#01**;**

SEND\_ERROR **:=** FALSE**;**

SEND\_STATUS **:=** 0**;**

SEND\_DONE **:=** FALSE**;**

SEND\_REQ **:=** FALSE**;**

RECV\_ERROR **:=** FALSE**;**

RECV\_NDR **:=** FALSE**;**

RECV\_STATUS **:=** 0**;**

RECV\_REQ **:=** FALSE**;**

TIMEOUT\_TIMER\_RESET **:=** TRUE**;**

TIMEOUT\_TIMER\_ON **:=** FALSE**;**

WEIGHT **:=** 0**;**

TARA **:=** 0**;**

DG\_DT **:=** 0**;**

DISOMAT\_STATUS **:=** W#16#0**;**

**END\_IF;**

// Kommunikáció lebonyolítása

**IF** **(**COMM\_ENABLE**)** **THEN**

// Lépésszámnak megfelelo muvelet végrehajtása

**CASE** SEQ\_STEP\_NR **OF**

0 **:** // Adatkérelem (<STX>00#TG#<ETX><BCC>)

DISOMAT\_SEND\_DATA**[**0**]** **:=** B#16#02**;** // <STX>

DISOMAT\_SEND\_DATA**[**1**]** **:=** B#16#30**;** // 0

DISOMAT\_SEND\_DATA**[**2**]** **:=** B#16#30**;** // 0

DISOMAT\_SEND\_DATA**[**3**]** **:=** B#16#23**;** // #

DISOMAT\_SEND\_DATA**[**4**]** **:=** B#16#54**;** // T

DISOMAT\_SEND\_DATA**[**5**]** **:=** B#16#47**;** // G

DISOMAT\_SEND\_DATA**[**6**]** **:=** B#16#23**;** // #

DISOMAT\_SEND\_DATA**[**7**]** **:=** B#16#03**;** // <ETX>

DISOMAT\_SEND\_DATA**[**8**]** **:=** B#16#10**;** // <BCC> (a kezdo STX kivételével a csomag minden bájtja XOR-olva)

SEND\_LEN **:=** 9**;**

SEND\_REQ **:=** TRUE**;**

SEQ\_STEP\_NR **:=** SEQ\_STEP\_NR **+** 1**;**

1 **:** // Küldés sikeres volt-e?

**IF** **(NOT** SEND\_REQ **AND** SEND\_DONE**)** **THEN**

// SEND\_DONE := FALSE;

// SEND\_ERROR := FALSE;

// SEND\_STATUS := 0;

SEQ\_STEP\_NR **:=** SEQ\_STEP\_NR **+** 1**;**

**END\_IF;**

// Küldés hiba figyelése

**IF** SEND\_ERROR **THEN**

COMM\_STATUS **:=** W#16#10**;**

COMM\_ERROR **:=** TRUE**;**

**END\_IF;**

2 **:** // Várakozás adatra a mérlegtol (<STX>00#TG#súly (7 karakter)#tára (7 karakter)#dG/dt (7 karakter)#státusz (2 karakter)#<ETX><BCC>)

TIMEOUT\_TIME **:=** T#5s**;**

TIMEOUT\_TIMER\_ON **:=** TRUE**;**

RECV\_REQ **:=** TRUE**;**

SEQ\_STEP\_NR **:=** SEQ\_STEP\_NR **+** 1**;**

3**:** // Kapott adatok ellenorzése

// Folyamatos lekérdezés, ameddig az ethernet adatfogadó puffer ki nem ürül

// IF (RECV\_LEN > 0) THEN

// RECV\_REQ := TRUE;

// END\_IF;

**IF** **(NOT** RECV\_REQ**)** **AND** **(NOT** RECV\_ERROR**)** **THEN**

// Értelmezheto adat érkezett (a visszaadott cím és parancs ugyanaz)

TEMPBOOL1 **:=** TRUE**;**

TEMPINT1 **:=** 0**;**

**WHILE** **(**TEMPBOOL1 **AND** **(**TEMPINT1 **<** 7**))** **DO**

**IF** **(**DISOMAT\_RECV\_DATA**[**TEMPINT1**]** **<>** DISOMAT\_SEND\_DATA**[**TEMPINT1**])** **THEN**

TEMPBOOL1 **:=** FALSE**;**

END\_IF**;**

TEMPINT1 **:=** TEMPINT1 **+** 1**;**

END\_WHILE**;**

**IF** TEMPBOOL1 **THEN**

RECV\_ERROR **:=** 0**;**

RECV\_STATUS **:=** 0**;**

SEQ\_STEP\_NR **:=** SEQ\_STEP\_NR **+** 1**;**

**ELSE**

// Ha egyéb más érkezett, akkor hiba

COMM\_STATUS **:=** W#16#23**;**

COMM\_ERROR **:=** TRUE**;**

END\_IF**;**

END\_IF**;**

// Idotúllépés figyelése

**IF** COMM\_TIMEOUT **THEN**

COMM\_STATUS **:=** W#16#21**;**

COMM\_ERROR **:=** TRUE**;**

END\_IF**;**

// Fogadás hiba figyelése

**IF** RECV\_ERROR **THEN**

COMM\_STATUS **:=** W#16#20**;**

COMM\_ERROR **:=** TRUE**;**

END\_IF**;**

4 **:** // Adatok kiírása a kimenetre

WEIGHT\_TMP **:=** 0**;**

TARA\_TMP **:=** 0**;**

DG\_DT\_TMP **:=** 0**;**

STATUS\_TMP **:=** W#0**;**

// Az érkezett adatok konvertálása (tizedes vesszo utáni értékek elhagyásával)

**FOR** TEMPINT1 **:=** 0 **TO** 6 BY 1 **DO**

// Tömeg

TEMPINT2 **:=** BYTE\_TO\_INT**(**DISOMAT\_RECV\_DATA**[**TEMPINT1 **+** 7**])** **-** INT#16#30**;**

**IF** **((**TEMPINT2 **>=** 0**)** **AND** **(**TEMPINT2 **<=** 9**))** **THEN**

WEIGHT\_TMP **:=** WEIGHT\_TMP **+** TEMPINT2 **\*** REAL\_TO\_INT**(**10 **\*\*** **(**6 **-** TEMPINT1**));**

**END\_IF;**

// Tára

TEMPINT2 **:=** BYTE\_TO\_INT**(**DISOMAT\_RECV\_DATA**[**TEMPINT1 **+** 15**])** **-** INT#16#30**;**

**IF** **((**TEMPINT2 **>=** 0**)** **AND** **(**TEMPINT2 **<=** 9**))** **THEN**

TARA\_TMP **:=** TARA\_TMP **+** TEMPINT2 **\*** REAL\_TO\_INT**(**10 **\*\*** **(**6 **-** TEMPINT1**));**

**END\_IF;**

// dG/dt

TEMPINT2 **:=** BYTE\_TO\_INT**(**DISOMAT\_RECV\_DATA**[**TEMPINT1 **+** 23**])** **-** INT#16#30**;**

**IF** **((**TEMPINT2 **>=** 0**)** **AND** **(**TEMPINT2 **<=** 9**))** **THEN**

DG\_DT\_TMP **:=** DG\_DT\_TMP **+** TEMPINT2 **\*** REAL\_TO\_INT**(**10 **\*\*** **(**6 **-** TEMPINT1**));**

**END\_IF;**

// Státusz

**IF** **(**TEMPINT1 **<=** 1**)** **THEN**

TEMPINT2 **:=** BYTE\_TO\_INT**(**DISOMAT\_RECV\_DATA**[**TEMPINT1 **+** 31**]);**

// ASCII -> hexa átalakítás (9-nél nagyobb helyiértékek mindig kisbetuvel jönnek)

**IF** **(**TEMPINT2 **>** INT#16#39**)** **THEN**

TEMPINT2 **:=** 10 **+** **(**TEMPINT2 **-** INT#16#61**);**

**ELSE**

TEMPINT2 **:=** TEMPINT2 **-** INT#16#30**;**

**END\_IF;**

**IF** **((**TEMPINT2 **>=** 0**)** **AND** **(**TEMPINT2 **<=** 15**))** **THEN**

STATUS\_TMP **:=** INT\_TO\_WORD**(**WORD\_TO\_INT**(**STATUS\_TMP**)** **+** TEMPINT2 **\*** REAL\_TO\_INT**(**16 **\*\*** **(**1 **-** TEMPINT1**)));**

**END\_IF;**

**END\_IF;**

**END\_FOR;**

WEIGHT **:=** WEIGHT\_TMP**;**

TARA **:=** TARA\_TMP**;**

DG\_DT **:=** DG\_DT\_TMP**;**

DISOMAT\_STATUS **:=** STATUS\_TMP**;**

COMM\_ENABLE **:=** FALSE**;**

END\_CASE**;**

// Idotúllépés figyelés

TIMERTIME **:=** S\_ODT**(**T\_NO **:=** T**,**

S **:=** TIMEOUT\_TIMER\_ON**,**

TV **:=** TIMEOUT\_TIME**,**

R **:=** TIMEOUT\_TIMER\_RESET**,**

BI **:=** TEMPWORD1**,**

Q **:=** COMM\_TIMEOUT**);**

TIMEOUT\_TIMER\_RESET **:=** FALSE**;**

**IF** COMM\_TIMEOUT **THEN**

SEND\_REQ **:=** FALSE**;**

RECV\_REQ **:=** FALSE**;**

**END\_IF;**

// Küldés/fogadás

**IF** **(**SEND\_REQ **AND** **NOT** RECV\_REQ**)** **THEN**

AG\_LSEND**(**ACT **:=** SEND\_REQ**,**

ID **:=** ID**,**

LADDR **:=** LADDR**,**

SEND **:=** DISOMAT\_SEND\_DATA**,**

LEN **:=** SEND\_LEN**,**

DONE **:=** SEND\_DONE**,**

ERROR **:=** SEND\_ERROR**,**

STATUS **:=** SEND\_STATUS

**);**

**IF** **(**SEND\_DONE **OR** SEND\_ERROR**)** **THEN**

SEND\_REQ **:=** FALSE**;**

**END\_IF;**

**END\_IF;**

PTR**:=**0**;**

**IF** **(**RECV\_REQ **AND** **NOT** SEND\_REQ**)** **THEN**

AG\_LRECV**(**ID **:=** ID**,**

LADDR **:=** LADDR**,**

RECV **:=** DISOMAT\_RECV\_DATA**,**

NDR **:=** RECV\_NDR**,**

ERROR **:=** RECV\_ERROR**,**

STATUS **:=** RECV\_STATUS**,**

LEN **:=** RECV\_LEN

**);**

**IF** **(**RECV\_NDR **OR** RECV\_ERROR**)** **THEN**

**FOR** TEMPINT1 **:=** 0 **TO** 35 **DO**

**IF** **(**DISOMAT\_RECV\_DATA**[**TEMPINT1**]=**B#16#02**)** **THEN**

PTR**:=**TEMPINT1**;**

**END\_IF;**

**END\_FOR;**

**FOR** TEMPINT1 **:=** 0 **TO** 35 **DO**

**IF** **(**TEMPINT1**+**PTR**)<**36 **THEN**

DISOMAT\_RECV\_DATA2**[**TEMPINT1**]:=**DISOMAT\_RECV\_DATA**[**TEMPINT1**+**PTR**];**

**ELSE**

DISOMAT\_RECV\_DATA2**[**TEMPINT1**]:=**DISOMAT\_RECV\_DATA**[**TEMPINT1**+**PTR**-**36**];**

END\_IF**;**

END\_FOR**;**

DISOMAT\_RECV\_DATA**:=**DISOMAT\_RECV\_DATA2**;**

RECV\_REQ **:=** FALSE**;**

**END\_IF;**

**END\_IF;**

PTR**:=**PTR**;**

**IF** COMM\_ERROR **THEN**

COMM\_ENABLE **:=** FALSE**;**

**END\_IF;**

**END\_IF;**

**END\_FUNCTION\_BLOCK**

## Moxa Nport beállítása

[NPort Configuration File]

CheckCode=cfgIA

[Basic Information (not changeable)]

APID=0x80005201

HWID=0x52AA

Serial No=1328

MAC Address=00:90:E8:43:33:0A

Firmware version=0x1010000

[Basic Settings]

Server Name=NPIA5250AI\_1328

Time Zone=0

Time Zone index=23

Date\_Year=115

Date\_Month=1

Date\_Day=13

Time\_Hour=9

Time\_Minute=37

Time\_Second=46

Time\_Milliseconds=0

Time Server=

Console Enabled=3

[Network Settings]

IP Address=172.24.2.138

Netmask=255.255.255.192

Gateway=

IP Configuration=0

DNS Server 1=

DNS Server 2=

[SNMP]

Enable SNMP=0

Community=public

Location=

Contact=

Trap=

[Mail]

Mail Server=

Mail Server Login=0

User Name=

Password=

From Address=NPIA5250AI\_1328@NPIA5250AI

To Address1=

To Address2=

To Address3=

To Address4=

[Accessible IP List]

Enabled=0

Rule1=0,,

Rule2=0,,

Rule3=0,,

Rule4=0,,

Rule5=0,,

Rule6=0,,

Rule7=0,,

Rule8=0,,

Rule9=0,,

Rule10=0,,

Rule11=0,,

Rule12=0,,

Rule13=0,,

Rule14=0,,

Rule15=0,,

Rule16=0,,

[Serial]

Port1=9600,3,0,1,0,

Port2=115200,3,1,1,0,

[Operating Mode]

Port1=10

Port2=7

[Operating Mode Option 2]

Port1=16,0x00,0x00,0,0

Port2=16,0x00,0x00,0,0

[Operating Mode Option 1]

Port1=1000,0,1,2100,966,0,0

[Auto Warning]

Mail=0x0

Trap=0x0

Relay=0x0

[Port Auto Warning]

Port1=0x00,0x00,0x00

Port2=0x00,0x00,0x00

[IP Address Report]

Server=

Port=4002

Period=10

[Password]

Password=

LCM & Reset Protect=0

1. Dresszírozói Folyamat Irányító Rendszer. A továbbiakban csak DFIR [↑](#footnote-ref-1)
2. 3-as szintű vállalatrányitási rendszer [↑](#footnote-ref-2)
3. Dresszírozói 1-es szintű folyamatirányító rendszer [↑](#footnote-ref-3)
4. Human Machine Interface (kezelő felület) [↑](#footnote-ref-4)
5. NPort IA5250I [↑](#footnote-ref-5)
6. A dresszírozói AP1-es PLC. A továbbiakban csak PLC [↑](#footnote-ref-6)
7. A virtuális számítógép fizikailag nem létezik: a felépítése csupán egy *szimuláció,* egy olyan számítógépes program, ami egy létező fizikai számítógépet, vagy egy fizikailag nem felépített számítógép működését szimulálja. Ez valójában egy "teljes számítógép egy másik számítógépen belül". [↑](#footnote-ref-7)
8. Centralográf Terminál program: Az eredeti supervisor-ra írt program ezzel a programmal lett kiváltva. A program működését ez a dokumentáció nem tartalmazza. [↑](#footnote-ref-8)
9. BLOB: Karakterek és binárisok (byte-ok) tárolására alkalmas típus. A BLOB típusba 65.535 (2^16 – 1) bájt tárolását teszi lehetővé [↑](#footnote-ref-9)
10. A **hash függvények** (kiejtése: hes, magyarul **hasítófüggvények**) olyan informatikában használt eljárások, amelyekkel bármilyen hosszúságú adatot adott hosszúságra képezhetünk le. Az így kapott véges adat neve *hash/hasító érték*. [↑](#footnote-ref-10)
11. struct: összetett adatstruktúra, az adattípust címkézett **mezők** vagy **tagok** alkotják.  [↑](#footnote-ref-11)
12. **Szkript**: Szkriptnyelven íródott utasítássor, program. [↑](#footnote-ref-12)