

DFIR-PLC INTERFÉSZ

MŰSZAKI LEÍRÁS

Tartalomjegyzék

1.	A DFIR rendszer funkciói	3
2.	A DFIR rendszer felépítése	4
2.1	A DFIR rendszer felosztása üzemeltetők szerint	6
2.2	A DFIR rendszer működése	7
3.	VATECH rendszer	8
3.1	A PLC feladata	8
3.2	A PLC Hardver ismertetése	8
3.3	A PLC kommunikáció beállításai	8
3.3.1	A PLC Ethernet kártya beállítása	8
3.3.2	Az Ethernet hálózat topológiája	9
3.3.3	Az Ethernet hálózat beállítása	9
3.4	Szűrőterv kezelés	10
3.4.1	A PLC-be leküldött szűrőterv adatok.....	10
3.4.2	A PLC által küldött válasz a DFIR-nek.....	11
3.4.3	PLC Szűrőterv kezelés folyamatábrája	12
3.5	Hengerlési adatok.....	13
3.6	Lecsévélőről levett tekercs adatok	13
3.7	Felcsévélőről levett tekercs adatok	14
3.8	Tekercslevétel nyugtázó telegram.....	15
3.9	Életjelküldés	15
3.10	Mérlegelés.....	16
3.10.1	A mérlegelés kommunikációs protokollja	16
3.10.2	DISOMAT_F_MEAS.....	16
3.10.3	Soros kommunikációs kábel.....	17
3.11	MOXA ETHERNET – RS232 átalakító.....	18
3.11.1	MOXA átalakító beállítása	20
3.11.2	A beállítások:	23
3.11.3	A beállítások visszaállítása.....	23
4.	DFIR – PLC Interfész.....	25
4.1	Virtuális számítógép.....	25
4.2	MySQL adatbázis.....	26
4.3	A DFIRPLC program.....	28
4.3.1	DFIRPLC Program felépítése.....	29
4.3.2	Szűrőterv kezelése.....	30
4.3.3	Hengerlési adatok	32
4.3.4	Tekercslevétel.....	33
4.3.5	A DFIRPLC használata	34
4.3.6	PLC Üzenetek panel	35
4.3.7	Dresszírozó panel	36
4.3.8	– Esemény Napló panel	37
4.3.9	Esemény napló fájlok	38
5.	Mellékletek	39

5.1	Disomat_F_Meas SCL kódja	39
5.2	Moxa Nport beállítása	46

1. A DFIR rendszer funkciói

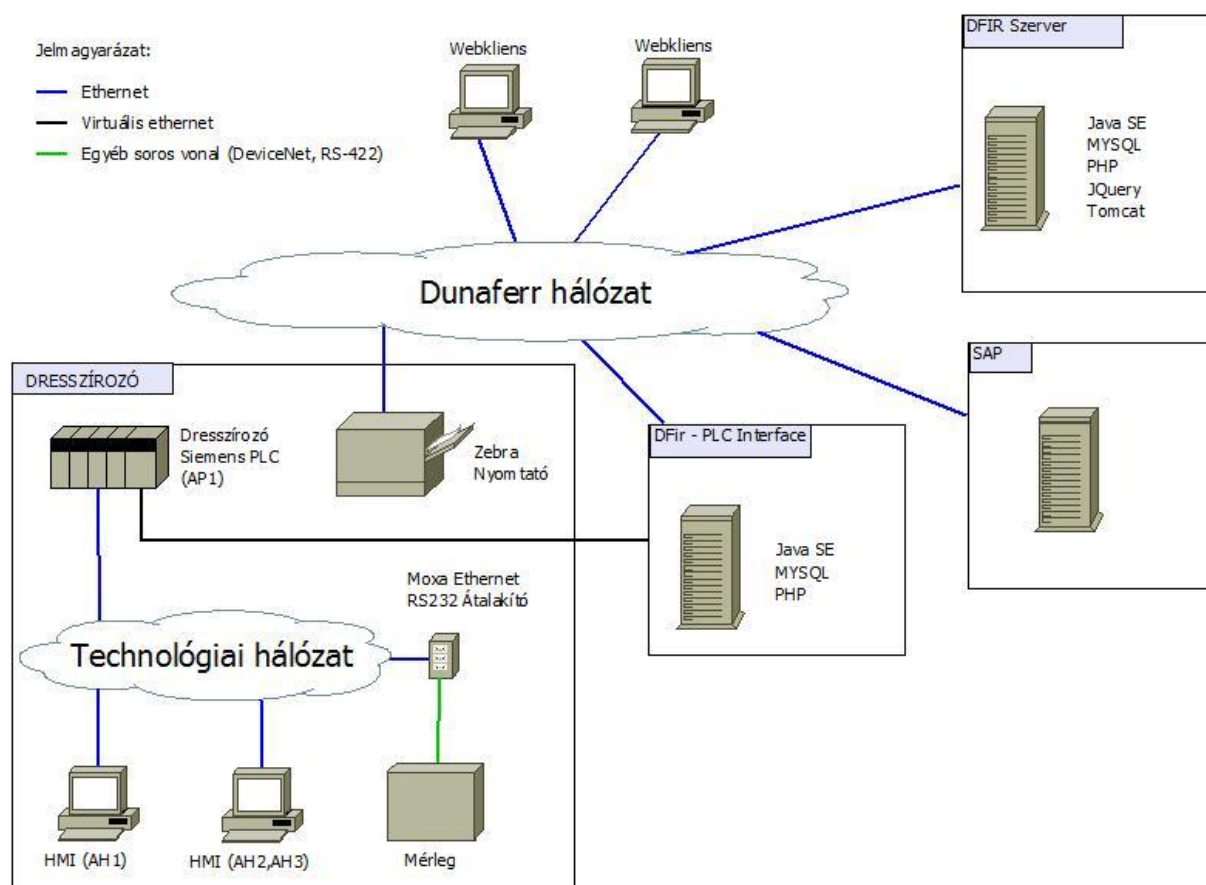
A Dresszírozói Folyamat Irányító Rendszer (DFIR) a következő funkciókat látja el:

- SAP –VATECH adatkapcsolat
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 tekercsek megjelennek a VATECH rendszerben a HMI-n (Human Machine Interface) a szűrésterv kezelő ablakon. 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, síkkifejvés, sebesség, stb. rekordokat, melyek a <http://dfirapp.ms.dunaferr.hu/regf.html> oldalon megtekinthetők.
- 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.

2. A DFIR rendszer felépítése

A DFIR rendszer sematikus ábrája az alábbi ábrán látható. A rendszer a következő elemekből épül fel:

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



1. ábra – DFIR rendszer sematikus ábrája

A rendszer elemeinek feladata:

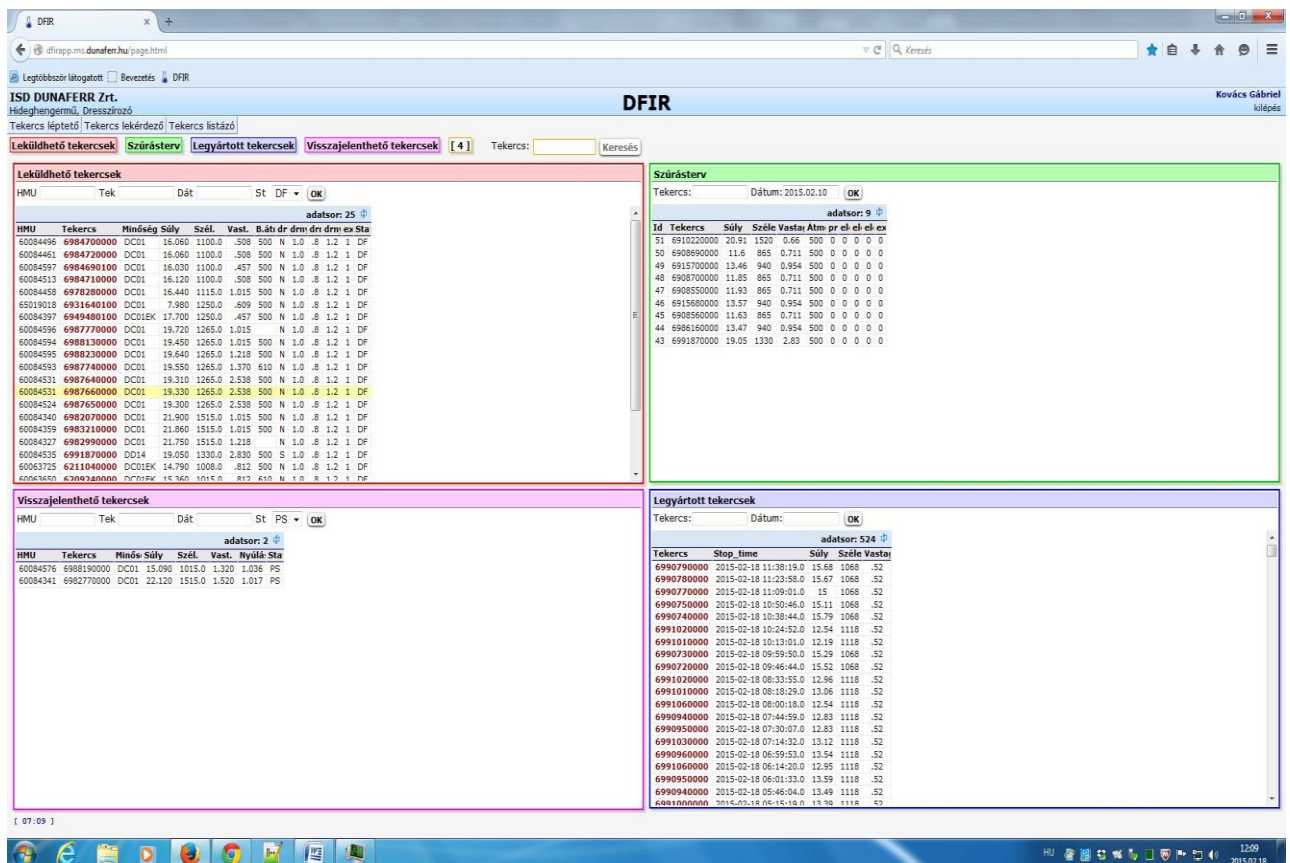
1. HMI: A főpulton elhelyezett számítógépes vezérlő képernyő, amely a technológiai folyamatirányító rendszer része. A HMI-n a szúrásterv képernyő kapja meg a DFIR rendszerből a leadott tekercseket, és a hozzá tartozó adatokat.
2. Mérleg: A felcsévélőn a levett tekercsek súlyát méri. A Disomat mérleg kiértékelő a főpulton az operátor mellett van elhelyezve.
3. Moxa Ethernet – RS232 átalakító: 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): Az informatika szerverén fut egy virtuális számítógép. Ez a számítógép virtuális Ethernet hálózaton keresztül kommunikál az S7400 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 Dunaferre SAP rendszere. A level 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.

2.1 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, a DFIR –PLC Interfészen futó programokat.
2. Informatika: Az informatika üzemelteti a Zebra tikett nyomtatót, a DFIR szervert, a Dunafer hálózatot, és az SAP rendszert.
3. Mérleg szerelők: A mérleg szerelők üzemeltetik a mérleget.

2.2 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> weboldala történt bejelentkezés után lehet vezérelni. A tekercsléptető képernyőn látható a DFIR-be leadott tekercsek helyzete. Az SAP –ből leadott tekercsek megjelennek a „leküldhető tekercsek” között. A tekercs kiválasztásával a szűréstervet át lehet küldeni a VATECH rendszerbe. Ekkor a tekercsadatok megjelennek a DFIR –ben a szűrésterv képernyőn. Ezzel egy időben megjelenik a tekercs a technológiai HMI –n a szűrésterv 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.



The screenshot displays the DFIR web application interface. The top navigation bar includes the company name 'ISD DUNAFERR Zrt.' and the system name 'DFIR'. Below the navigation bar, there are four main panels:

- Leküldhető tekercsek (Sendable coils):** A table listing coils with columns for HMU, Tekercs, Minőség, Súly, Szél, Vast., B.átl, dr, drn, drn, ex, and Sta. The table shows a list of coils with their respective specifications.
- Szűrésterv (Filtering plan):** A table showing the filtering plan for a specific coil. It includes columns for Id, Tekercs, Súly, Szél, Vast., Atm, pr, el, el, ex. The table shows a list of coils with their respective specifications.
- Visszajelenthető tekercsek (Reportable coils):** A table listing coils that have been reported. It includes columns for HMU, Tekercs, Minőség, Súly, Szél, Vast., and Nyúló-Sta. The table shows a list of coils with their respective specifications.
- Legyártott tekercsek (Produced coils):** A table listing coils that have been produced. It includes columns for Tekercs, Stop_time, Súly, Szél, and Vast. The table shows a list of coils with their respective specifications.

2. ábra - Tekercs léptető

3. VATECH rendszer

3.1 A PLC feladata

Az AP1 –es PLC –be lett megírva a DFIR-rel és a mérleggel az Ethernet kapcsolat. A PLC feladata a kommunikációk vezérlése és az adatok feldolgozása.

3.2 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.

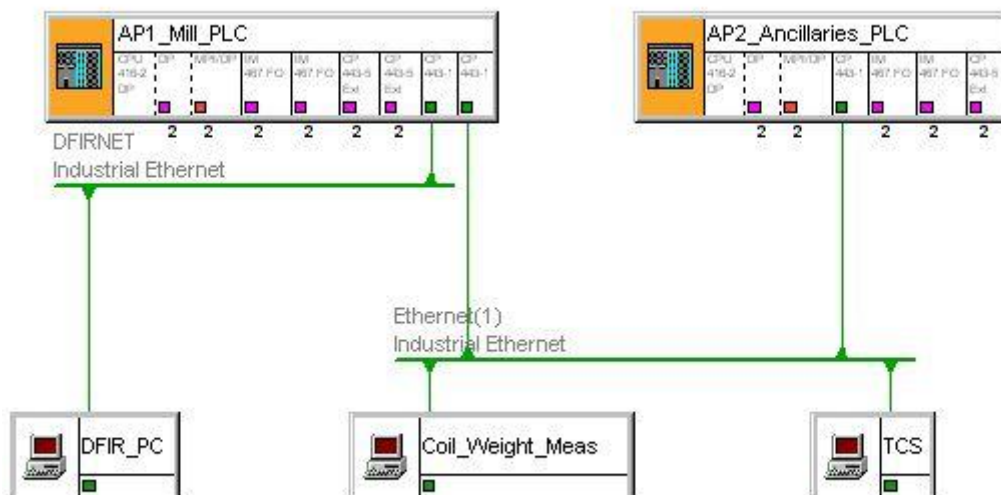
3.3 A PLC kommunikáció beállításai

3.3.1 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

1. táblázat - Ethernet kártya konfigurálása

3.3.2 Az Ethernet hálózat topológiája



3. ábra PLC hálózati topológiája.

3.3.3 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	
Felcsévéőlőről levett tekercs adatok	0009 A050	192.168.210.11:2012	192.168.210.10:2012	DB883	
Lecsévéőlőről levett tekercs adatok	000A A050	192.168.210.11:2013	192.168.210.10:2013	DB884	
Szűrőrásterv adatok	000B A050	192.168.210.11:2014	192.168.210.10:2014		DB885
Szűrőrásterv 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

2. táblázat - Ethernet hálózat beállítása

3.4 Szűrásterv kezelés

A HMI-n a szűrásterv 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 szűrástervet a szűrástervkezelő 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 tekercs adatot, 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 szűrástervet, ha a plc engedélyezi.

3.4.1 A PLC-be leküldött szűrásterv adatok

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

Address	Name	Type	Comment
0.0	DwaPssSchTelld	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

3. táblázat - DB885 Szűrásterv adatok

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

3.4.2 A PLC által küldött válasz a DFIR-nek

A PLC 100mS időnként elküldi a nyugtázó üzenetet a DFIR-nek . A telegram a 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)

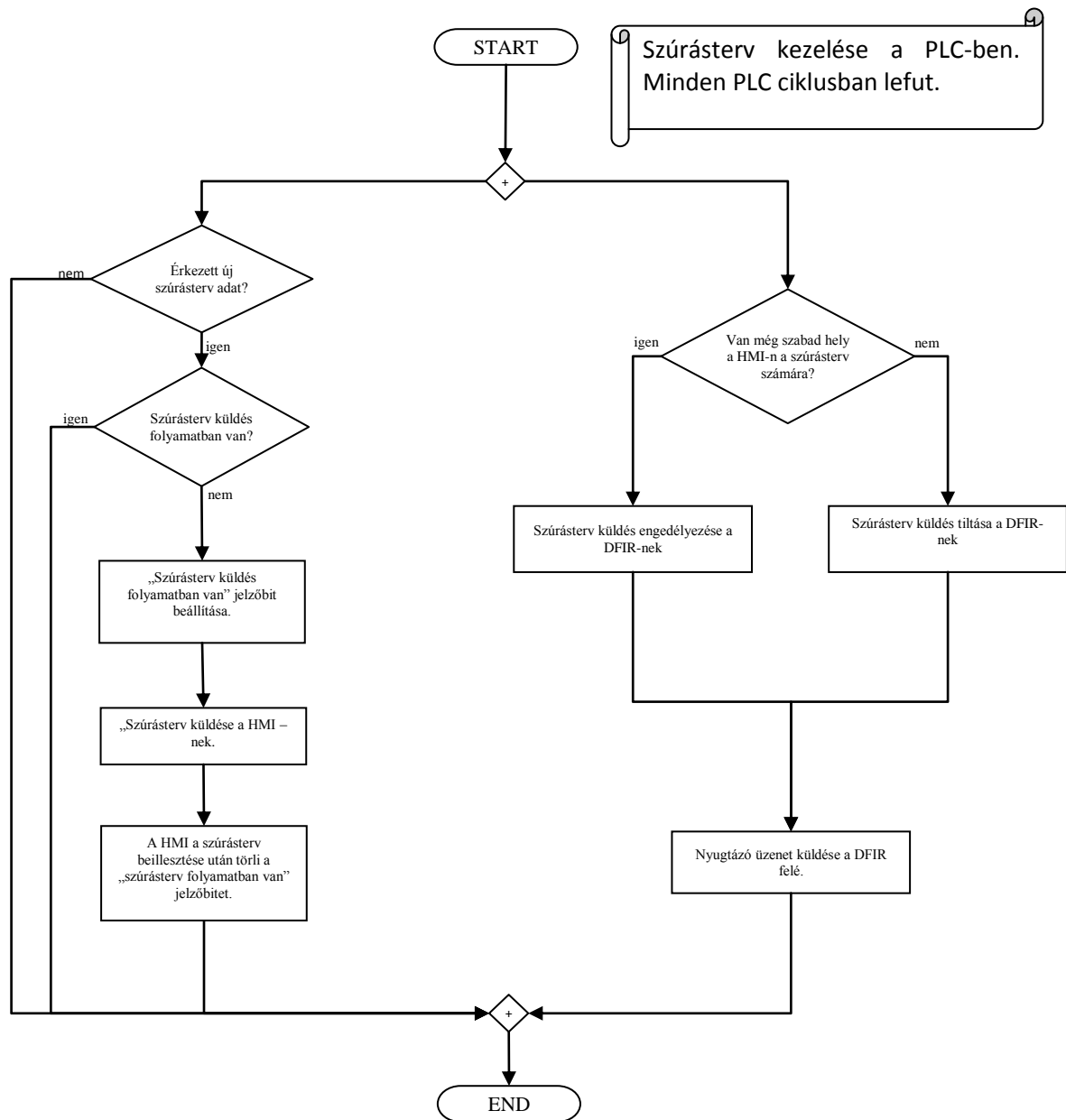
4. táblázat - DB886 szúrásterv válasz telegram

A „*PassSchedTelId*” az utoljára fogadott szúrásterv azonosítóját küldi vissza (*DB885.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.

3.4.3 PLC Szűrősterv kezelés folyamatábrája

Az alábbi ábrán a szűrősterv kezelés folyamatábrája látható:



4. ábra - szűrősterv kezelés folyamat

3.5 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.

3.6 Lecsévélőről levett tekercs adatok

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.

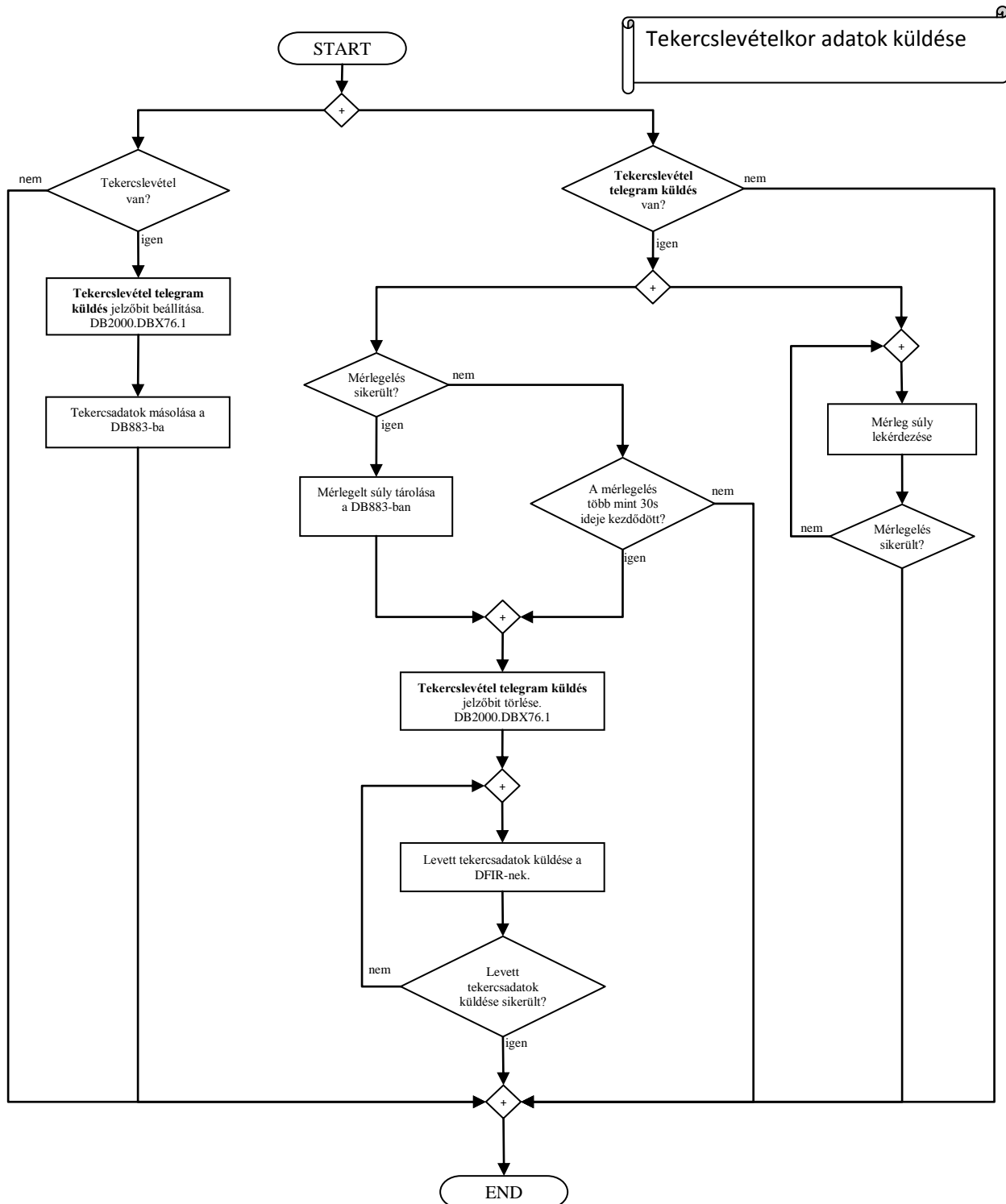
Address	Name	Type	Comment
0.0	RemovePORTelId	INT	Anyag levéve az LCS-ről távirat ID
2.0	CoilId	STRING[14]	Tekercsazonosító (tekercsszám)
18.0	CoilPartNo	INT	A vágás után képzett tekercsrész-szám [0..999]

6. táblázat - DB884 Lecsévélőről levett tekercs adatok

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

3.7 Felcsévéől levett tekercs adatok

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 tekercs adatokat elküldjük a DFIR felé. A tekercsleszedés folyamatábrája:



5. ábra - Felcsévéői tekercslevétel folyamatábrája

A felcsévlőről levett tekercs adatok a DB 883-ban tároljuk:

Address	Name	Type	Comment
0.0	RemoveTRTelld	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 Felcsévlőről levett tekercs adatok

3.8 Tekercslevétel nyugtázó telegram

A Lecsévlő és a felcsévlő 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	RemoveTRTelld	INT	Anyag levéve az FCS-ről távirat ID
2.0	RemovePORTelld	INT	Anyag levéve az LCS-ről távirat ID

8. táblázat - DB881 lecsévlőről levett tekercs adatok

3.9 Életjelküldés

A DFIR és a PLC adatokat a Dunafer hálózaton 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

9. táblázat - DB887 életjelküldés

A „LifeSignal” 1-9999 –ig számol. Az értékét eggyel növeljük, ha sikerült elküldeni az üzenetet. 9999 után 1 következik. A LifeSignalError akkor számol, ha nem sikerült elküldeni az üzenetet.

3.10 Mérlegelés

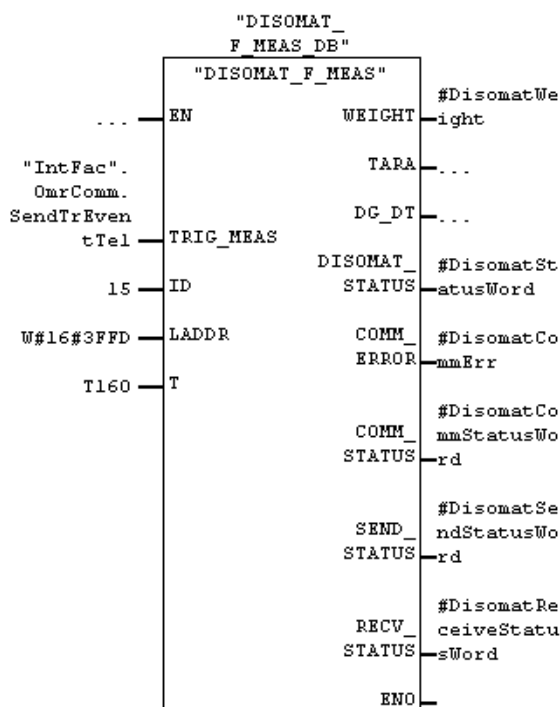
A mérlegelés a Disomat F típusú mérleggel történik. A PLC a Moxa Ethernet - RS232 Átalakítón keresztül kommunikál a mérleggel

3.10.1 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 5s –e 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.

3.10.2 DISOMAT_F_MEAS

A PLC –ben a súly lekérését a „DISOMAT_F_MEAS” Blokk végzi el.



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

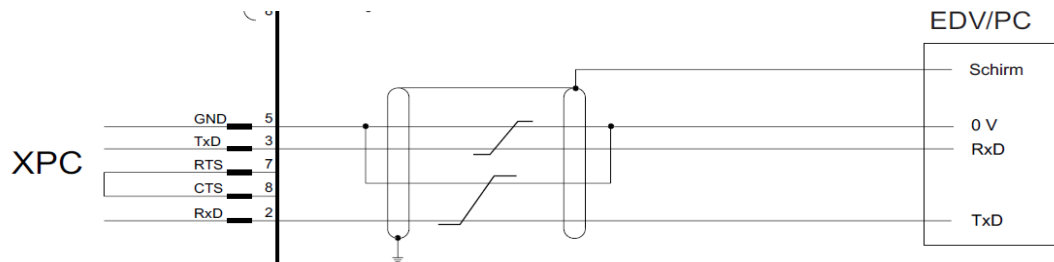
RECV_STATUS: ETH fogadás állapot

6. ábra - DISOMAT_F_MEAS blokk

A Disomat_F_Meas Blokk működésének a programját az 5.1 „Disomat_F_Meas SCL kódja” lehet megtekinteni.

3.10.3 Soros kommunikációs kábel

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

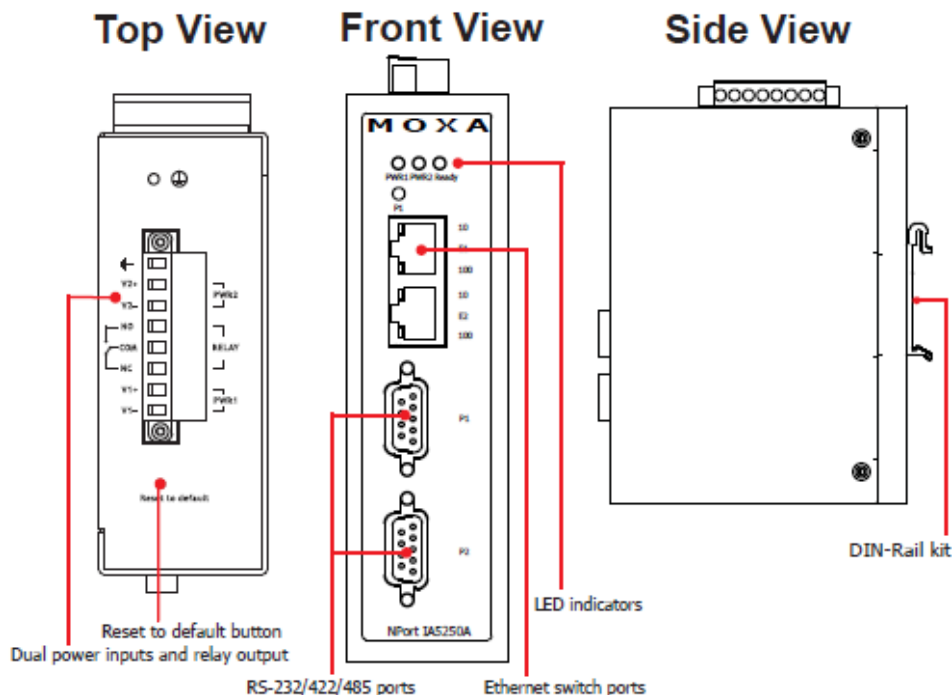


7. ábra - Mérleg kommunikációs kábel

A mérlegre a kommunikációs kábelt az XPC pontjára kell bekötni.

3.11 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.



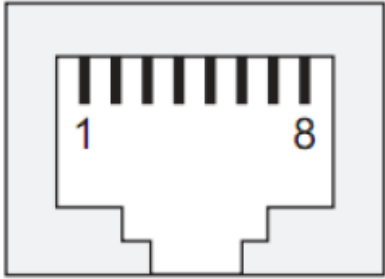
8. ábra - NPort IA5250AI

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:

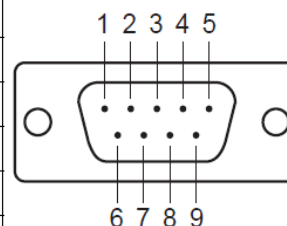
Pin	Signal
1	Tx+
2	Tx-
3	Rx+
6	Rx-



9. ábra - MOXA Ethernet port lábkiosztása

A soros port lábkiosztása:

Pin	RS-232	RS-422, 4-wire RS-485	2-wire RS-485
1	DCD	TxD-(A)	-
2	RxD	TxD+(B)	-
3	TxD	RxD+(B)	Data+(B)
4	DTR	RxD-(A)	Data-(A)
5	GND	GND	GND
6	DSR	-	-
7	RTS	-	-
8	CTS	-	-
9	-	-	-



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

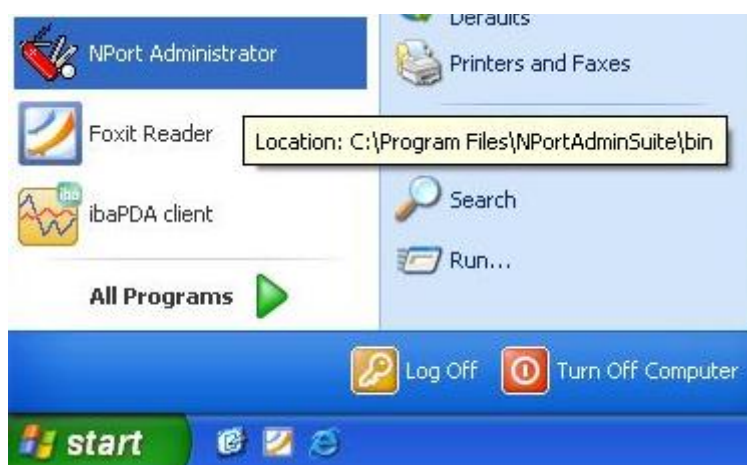
Az előlapi ledék jelentései:

Név	Szín	Jelentés
PWR1 PWR2	piros	A tápfeszültség rendben van a PWR1, PWR2 bemeneten.
Ready	Piros	Feszültség renben 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 aktyválta a "Locate" funkciót
	nem világít	Tápfeszültség kikapcsolva, vagy tápfeszültség hiba van.
E1, E2	narancs	10 Mbps Ethernet kapcsolat
	zöld	100 MBPS Ethernet kapcsolat
	nem világít	Ethernet kábel nincs bedugva, vagy meghibásodott.
P1,P2	narancs	Soros port fogadja az adatot
	zöld	Soros port küldi az adatot
	nem világít	Nincs adatátvitel a soros porton

10. táblázat - Ledék jelentései

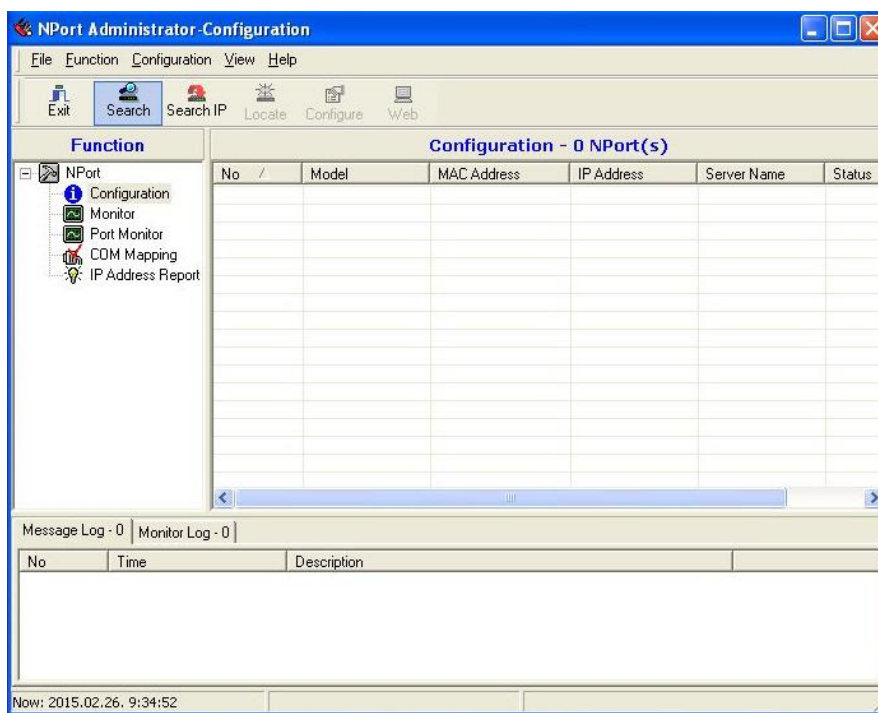
3.11.1 MOXA átalakító beállítása

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



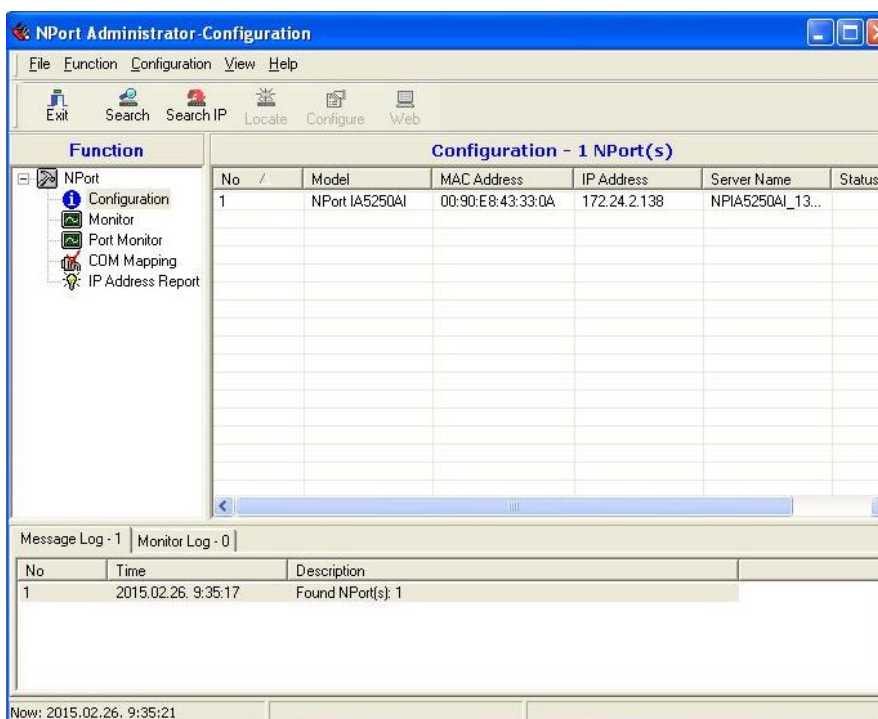
11. ábra - program elindítása

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



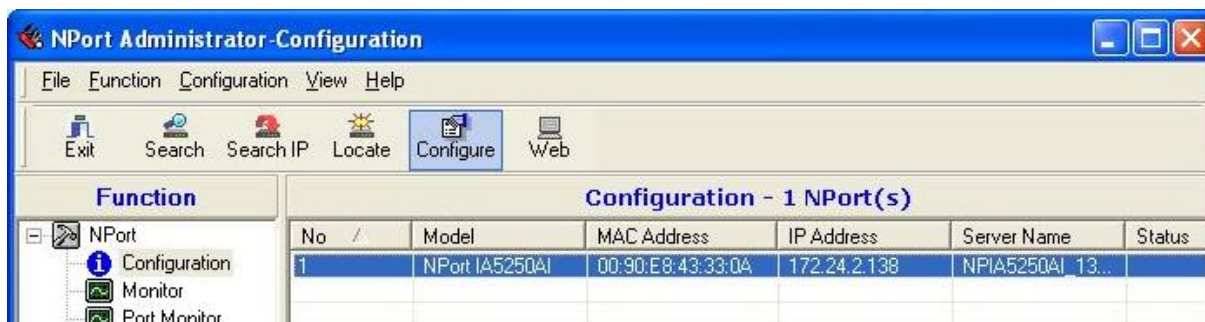
12. ábra - eszköz keresése

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



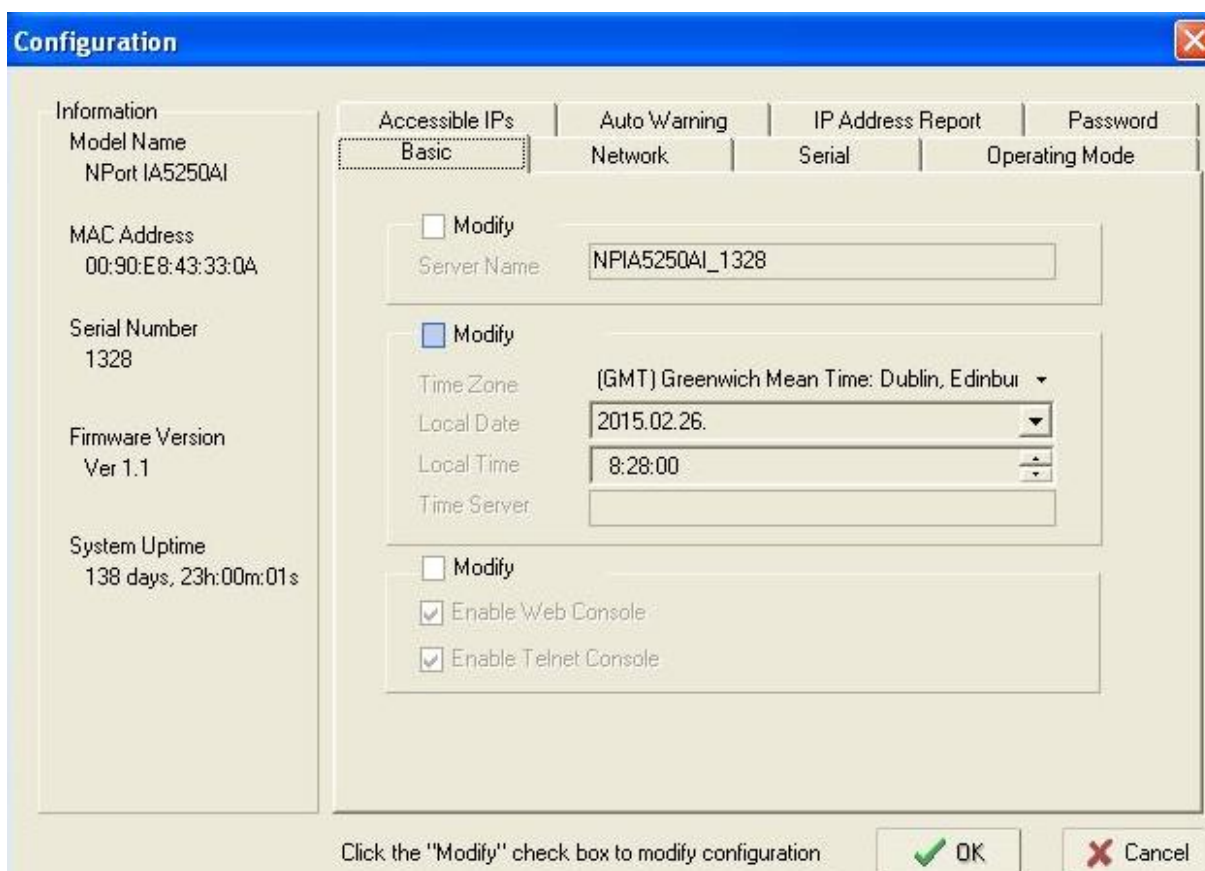
13. ábra - a keresés eredménye

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



14. ábra - konfigurálás

A „Basic” fül beállítása:



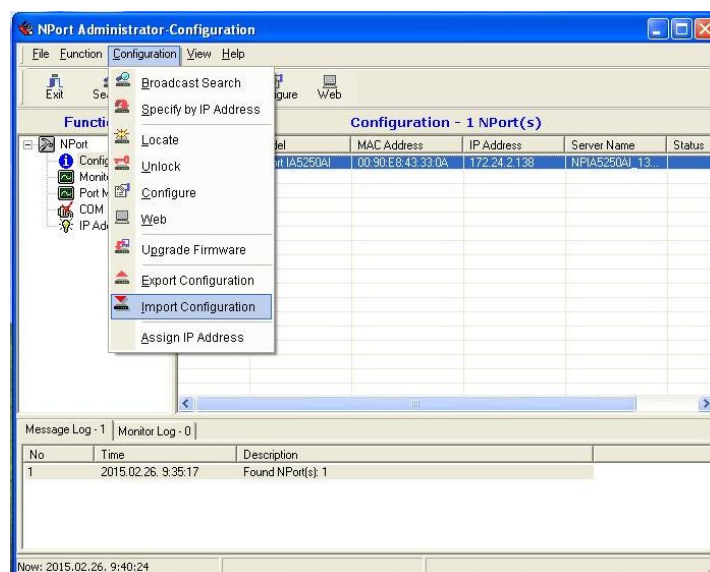
15. ábra- Basic beállítás

3.11.2 A beállítások:

- Basic:
 - o Server Name: „NP1A5250AI_1328” (Tetszőlegesen választható)”
 - o Time Zone: „(GMT) Greenwich Mean Time: Dublin...”
 - o „Enable Web console” kiválasztva
 - o „Enable Telnet Console” kiválasztva
- Network:
 - o IP Address: „172.24.2.138”
 - o Netmask: „255.255.255.192”
 - o „Enable SNMP” aktiválva
- Serial:
 - o 1. Settings: „9600,N,8,No flowctrl”
 - o 2. Settings: 115200,N,8,1,RTS/CTS (Nem használjuk)
- Operating Mode
 - o 1. OP Mode: „TCP Server Mode”
 - o 2. Disable (Nem használjuk)

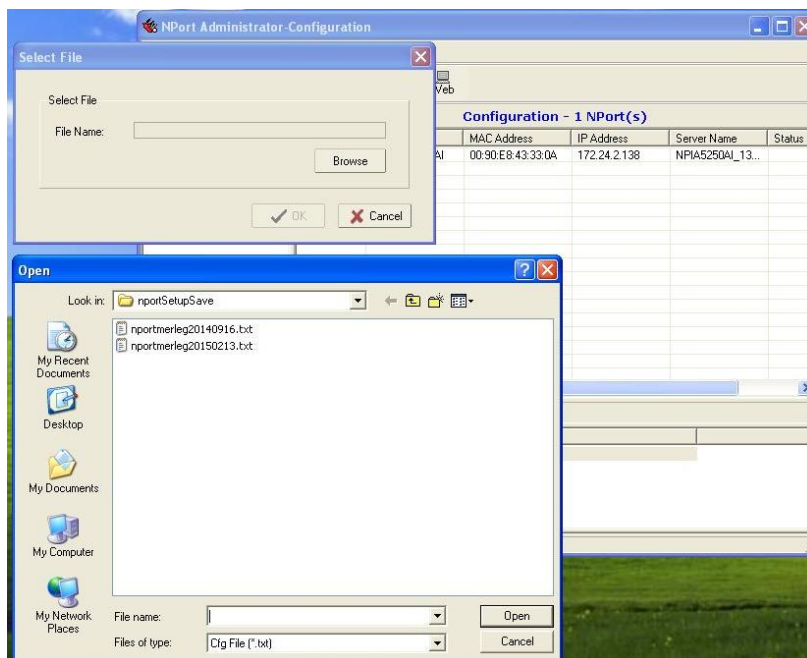
3.11.3 A beállítások visszaállítása

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.



16. 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.



17. ábra - konfiguráció importálása 2

4. DFIR – PLC Interfész

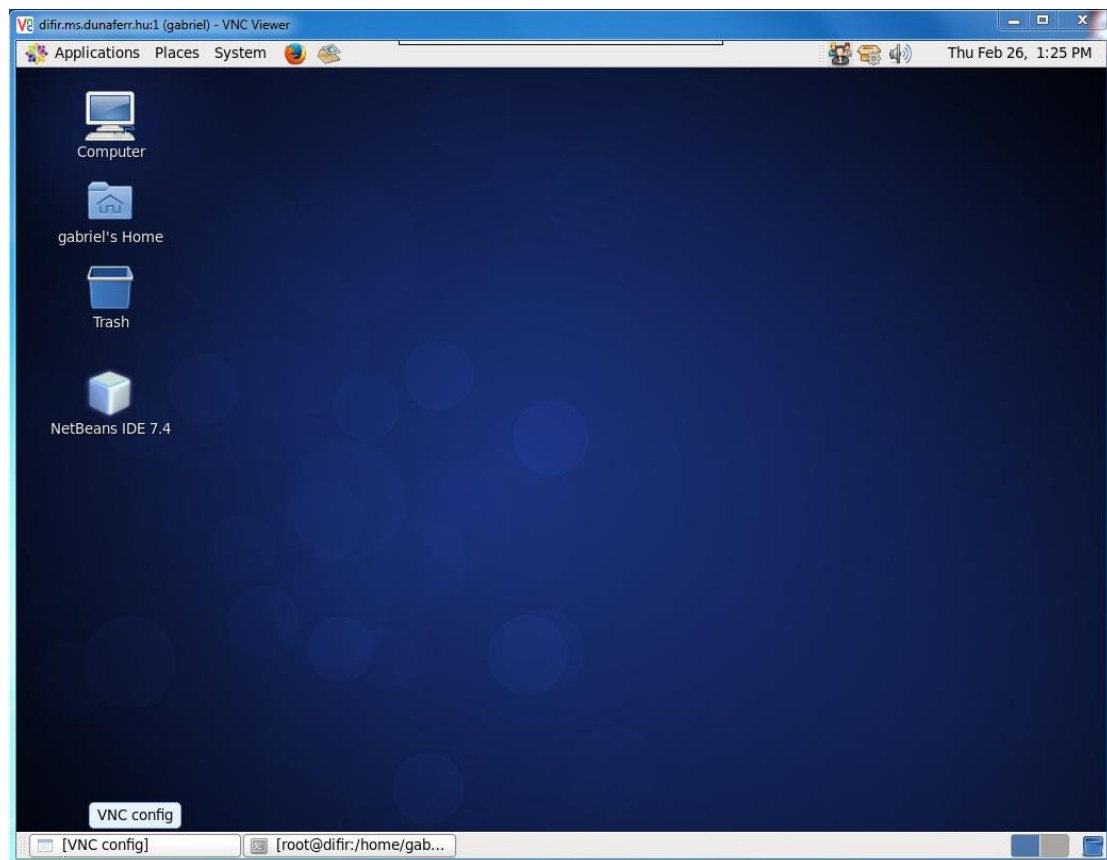
A PLC és DFIR rendszer között nincs közvetlen kapcsolat. A kommunikációt egy DFIRPLC nevű java nyelven írt alkalmazással oldottuk meg. Ez az alkalmazás Etherneten kommunikál a PLC-vel, és MySQL adatbázison keresztül adja át az adatokat a DFIR-nek. UDP üzeneteken keresztül küldi el a sor termel, vagy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

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

4.1 Virtuális számítógép

A számítógépet a Dunafer 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.



18. 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
- Apache 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 forgalom figyelő

4.2 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](#). A DFIR – hez adatbázishoz tartozó adatbázis neve: [dfir](#).

Az adatbázis a következő táblákból áll:

- regf: A hengerlés során regisztrált adatok kerülnek ide letárolásra. Az id: automatikusan növekedő futósorszám, htek_id: hidegtekerics szám, start_time: a dresszírozás kezdete, stop_time: a dresszírozás vége, record: Az összegyűjtött rekordok blob adatba. Az egy db rekord megegyezik az „5. táblázat - DB882 hengerlési adatok” táblázatban látható adatokkal, de a tekericsazonosító nem szerepel benne. A status mező 1 ha rendben van minden adat. coil_width a PLC által számolt lemez szélesség, a coil_meas_weight a mérleg által mért tekerics súly, a coil_cal_weight a PLC által számított tekerics súly, a coil_thickness a lemez vastagsága.
- regf_log: Az eredeti terv szerint a regf táblából az adatok törlésre kerülnek, miután a DFIR rendszer feldolgozta az adatokat. A DFIR rendszer miután feldolgozta az adatokat azért hogy a folyamatok nyomon követhetők legyenek készít egy másolatot a regf tábláról a record mező nélkül. A length mezőbe a record hossza kerül, a hash mezőbe a rekord hash-elt értéke kerül. Illetve van két státusz mező is.
- szurasterv: a DFIR –ből leadott tekercesk szúrasterv adatai találhatóak itt. Csak olyan tekercesk szerepelnek a táblában, amelyek még nem lettek elküldve a PLC-nek. Ha minden jól működik, akkor a tábla üres. Az psssschtelid: automatikusan növekedő futósorszám, coolid: hidegtekerics szám, steelgrade: acélminőség, width: lemez szélesség, thickness: lemez vastagság, weight: tekericsúly, corediameter: Tekercsmag átmérő, ts_küldes: A dátum, amikor le lett adva a szúrasterv az adatbázisba, exitcoilsno: a gyártandó tekercesk száma, processtype: a dresszírozás típusa, elongation:

nyúlás, elonglowlim: nyúlás minimum előírás, elonguplim: nyúlás maximum előírás.

- szurasterv_log: Ha a DFIRPLC program feldolgozta a szurástervet, akkor leküldi a PLC-be és a leküldött adatok megjelennek a szurasterv_log táblába. A szurasterv táblából 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 be új mezőt tároltunk le. Innen fogja tudni a DFIR hogy új adat került a regf táblába. table_name: „dfir”, table_status: „0”, time_stamp: a bejegyzés ideje.

Az alábbi ábrán a **dfir** adatbázis táblái láthatóak:

Name	Datatype	Comment
id	bigint(20) unsigned	
htek_id	varchar(14)	Tekercsazonosító (tekercsszám)
start_time	datetime	Dresszírozás kezdete
stop_time	datetime	Dresszírozás vége
record	longblob	Dresszírozás alatt rögzített rekordok blob fájlban
status	int(11)	
coil_part_number	int(11)	A vágás után képzett tekercsrész-szám [0..999]
coil_width	int(11)	Lemez szélesség (mm)
coil_meas_weight	int(11)	Tekercs súly (mért)
coil_cal_weight	int(11)	Tekercs súly (számított)
coil_thickness	int(11)	Lemez vastagság

Name	Datatype	Comment
id	bigint(20) unsigned	
htek_id	varchar(14)	Tekercsazonosító (tekercsszám)
start_time	datetime	Dresszírozás kezdete
stop_time	datetime	Dresszírozás vége
length	decimal(10,0)	
hash	varchar(32)	
status	varchar(100)	
sapstatus	varchar(100)	
coil_part_number	int(11)	
coil_width	int(11)	
coil_meas_weight	int(11)	
coil_cal_weight	int(11)	

Name	Datatype	Comment
psschtelid	bigint(20)	Szurásterv azonosító
coilid	varchar(14)	Tekercsazonosító (tekercsszám)
steelgrade	varchar(20)	Acélminőség
width	double	Lemez szélesség
thickness	double	Lemez vastagság
weight	double	Tekercssúly
corediameter	double	Tekercsmag átmérete
ts_kuldes	timestamp	
exitcoilsno	smallint(6)	Gyártandó tekercsek száma (hány részre osztjuk)
processtype	smallint(6)	Dresszírozás típusa (0: száraz, 1: nedves)
elongation	double	Előírt nyúlás [%]
elonglowlim	double	Nyúlás minimum előírás
elonguplim	double	Nyúlás maximum előírás

Name	Datatype	Comment
table_name	varchar(100)	
table_status	varchar(100)	
time_stamp	timestamp	

Name	Datatype	Comment
psschtelid	bigint(20)	Szurásterv azonosító
coilid	varchar(14)	Tekercsazonosító (tekercsszám)
steelgrade	varchar(20)	Acélminőség
width	double	Lemez szélesség
thickness	double	Lemez vastagság
weight	double	Tekercssúly
length	double	Lemez hossz
exitcoilsno	smallint(6)	Gyártandó tekercsek száma (hány részre osztjuk)
exitcoil1length	double	1. Kész tekercs hossza [m]
exitcoil2length	double	2. Kész tekercs hossza [m]
processtype	smallint(6)	Dresszírozás típusa (0: száraz, 1: nedves)
elongation	double	Előírt Nyúlás [%]
rollforce	double	Előírt hengerlési erő
bendingforce	double	Előírt hengerhajlítás
linespeed	double	Előírt sebesség
basicsprayamount	double	Előírt dresszírozó folyadék mennyiség
tensionporesbr	double	Előírt lecsévélofi feszítés
tensionesbrstd	double	Előírt belépo oldali Sgörgo feszítés
tensionstdsbr	double	Előírt kilépo oldali Sgörgo feszítés
tensionxsbir	double	Előírt felcsévélofi feszítés
corediameter	double	Tekercsmag átmérete
elonglowlim	double	Nyúlás minimum előírás
elonguplim	double	Nyúlás maximum előírás
ts_kuldes	timestamp	Időbélyeg az SAP-ból
ts_feldolgozas	timestamp	Időbélyeg a VATECH felé

19. ábra - dfir adatbázis

4.3 A DFIRPLC program

A DFIRPLC program java programozási nyelven 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 szűréstervet megkapja adatbázison keresztül a DFIR-től, és továbbküldi a PLC-nek a Dunaferres hálózaton keresztül. A PLC-vel a kommunikáció TCP/IP. A szűrésterv sikeres leküldése után a szurasterv_log tábla feltöltésével tudatja a DFIR-nek, hogy a PLC megkapta a szűréstervet.

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

4.3.1 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 címe
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 kiírásának engedélyezése
CENTRALOGGRAFMESSENGEABLE	boolean	true	Centralograf terminál programnak üzenet küldés engedélyezése
CENTRALOGGRAFIADDRESS	String	"10.1.39.154"	Centrál terminál ip címe
LOGMODE	int	1	0:=Minen log új fájlba, 1:= 31 napos naptár szerinti loggolás
PASSCHEDULEENABLE	boolean	true	Szűrősterv leküldés engedélyezése

11. táblázat - Program futását vezérlő jelek

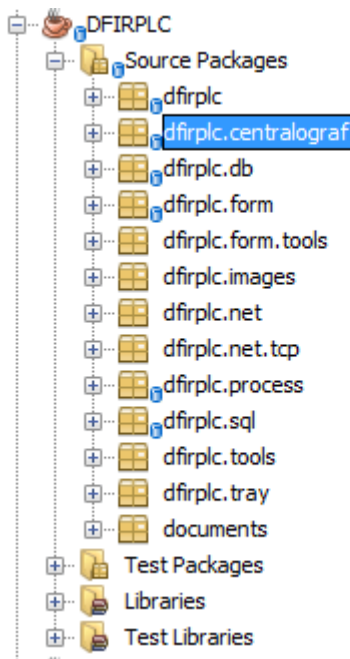
A változók megváltoztatása, csak a program lefordítása, és újraindítása után lép érvénybe.

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: Struktúra 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.

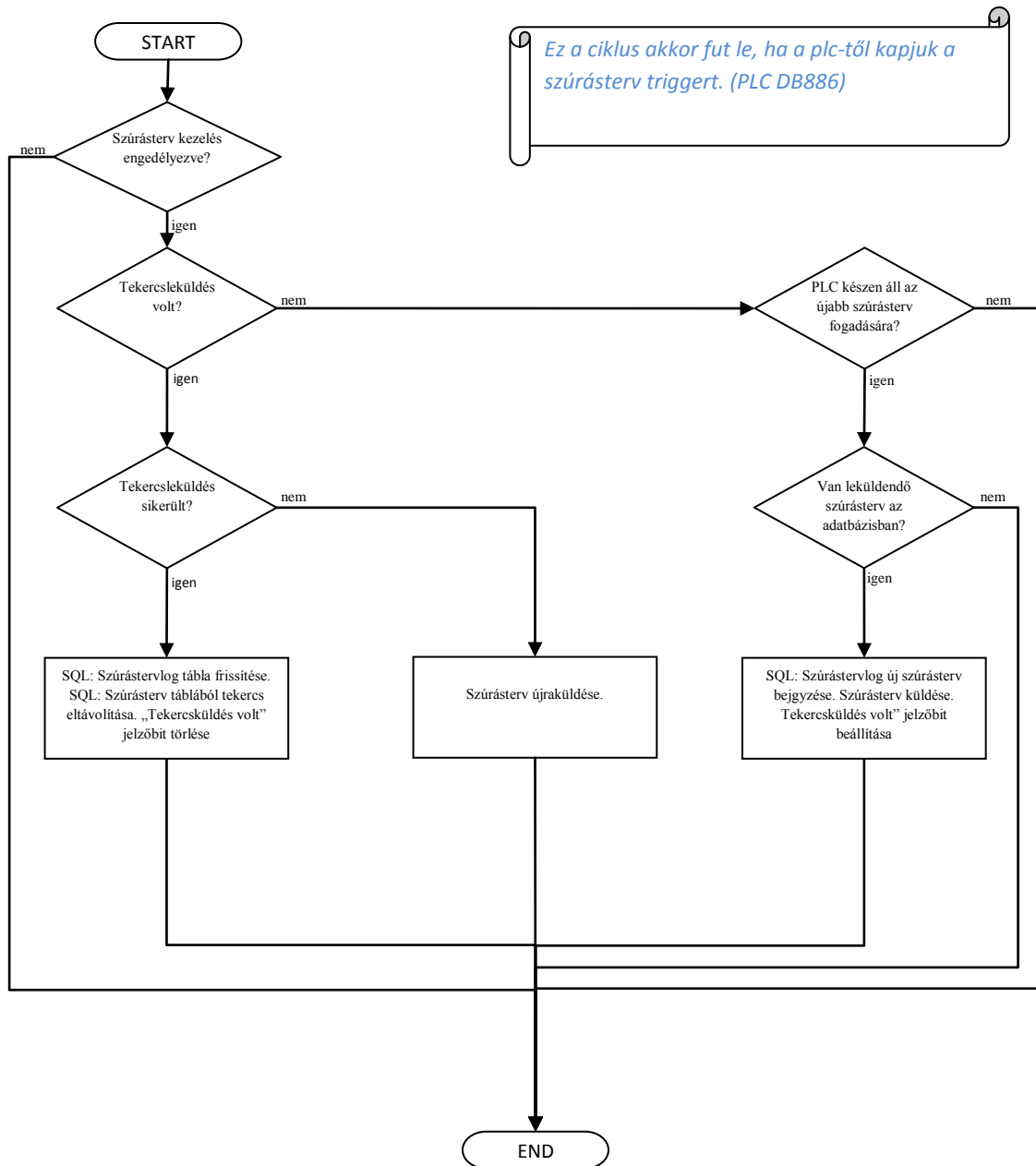


20. ábra - program strukturálása

4.3.2 Szűrősterv kezelése

A PLC 100 ms-onként küldi a DB886 adatblokkban található üzeneteket. Az üzenetben a legutoljára leadott szűrősterv azonosítója (nyugta), és egy jelzőbit található, ami azt tartalmazza, hogy a HMI képes fogadni az újabb szűrőstervet, vagy nem. Ha a szűrősterv kezelés engedélyezve van a programban, akkor a program (ProcessDB886), ellenőrzi, hogy van e leküldendő szűrősterv az adatbázisban, ellenőrzi, hogy a leküldés sikerült-e, és szükség szerint újraküldi. A sikeresen leadott

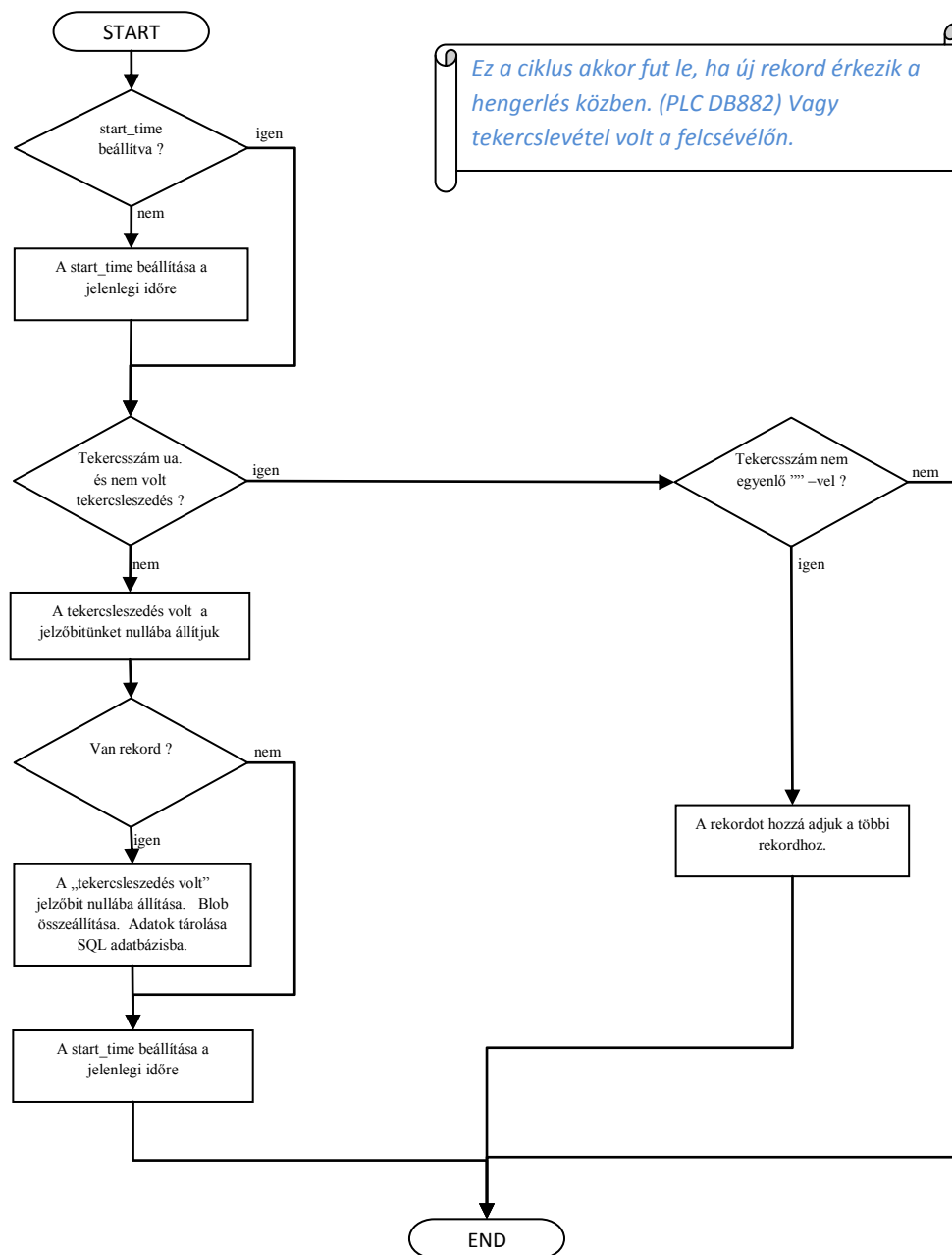
tekercs adatait kitörli az adatbázis **szuraster** táblájából és kibővített információkkal bemásolja a **szuraster_log** táblába. Az alábbi ábrán a szűrasterv küldés folyamat ábrája látható:



21. ábra - szűrasterv küldés folyamatábrája

4.3.3 Hengerlési adatok

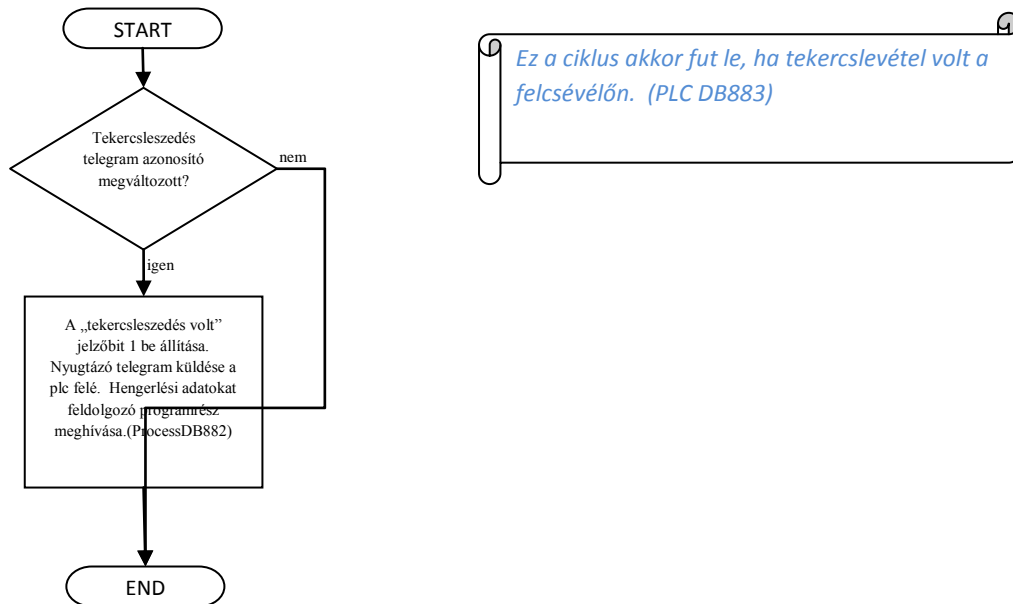
A PLC hengerlés közben 100 ms-onként küldi a hengerlési adatokat. A program a hengerlési adatokat addig gyűjti, míg a hengerlés közben meg nem változik a tekercs száma, vagy a tekercset le nem vesszük a felcsévélőről. Ha a tekercset levettük a felcsévélőről, akkor a tekercshez tartozó összegyűjtött hengerlési 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

4.3.4 Tekercslevétel

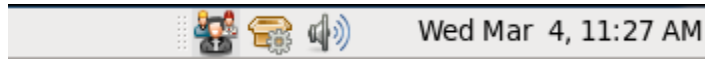
A PLC minden tekercslevételnél elküldi a tekercs azonosítót, a súlyt, a vastagságot, és a szélességet. A tekercslevétel üzenetből tudja a program, hogy a tekercs elkészült, és letárolja a hengerlési adatokat. A felcsévéző tekercslevételt feldolgozó program (ProcessDB883) folyamat ábrája:



23. ábra - tekercslevétel folyamatábrája

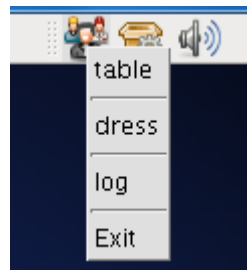
4.3.5 A DFIRPLC használata

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.

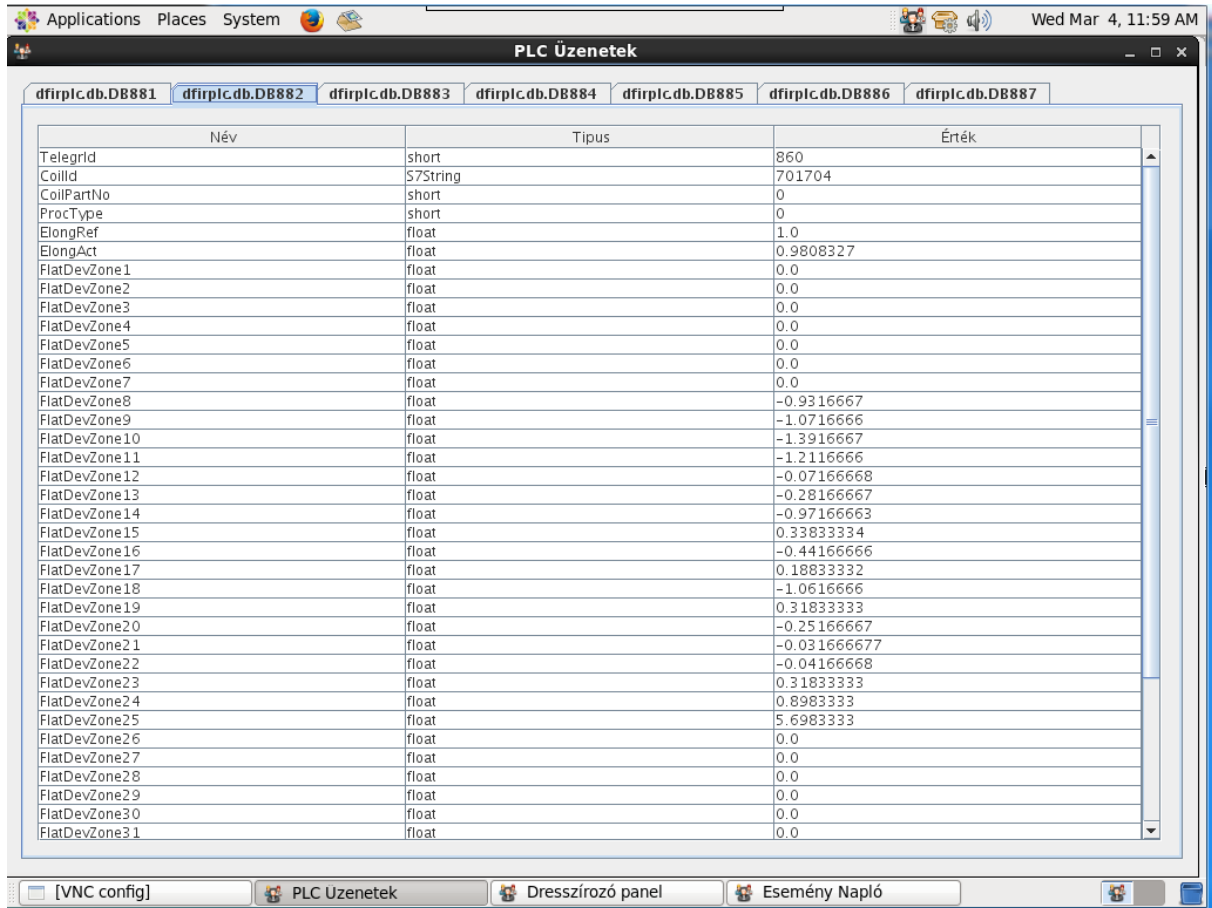
Figyelem! A programot csak akkor zárjuk be, ha a berendezés nem megy és a program letárolta a levett tekercs adatait. Egyéb esetben a program el fogja veszíteni az utoljára hengerelt tekercs adatait.

Mivel fut a háttérben a startDFIRPLC.sh szkript 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.

4.3.6 PLC Üzenetek panel

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



The screenshot shows a window titled "PLC Üzenetek" with a tabbed interface. The active tab is "dfirplc.db.DB882". The table displays the following data:

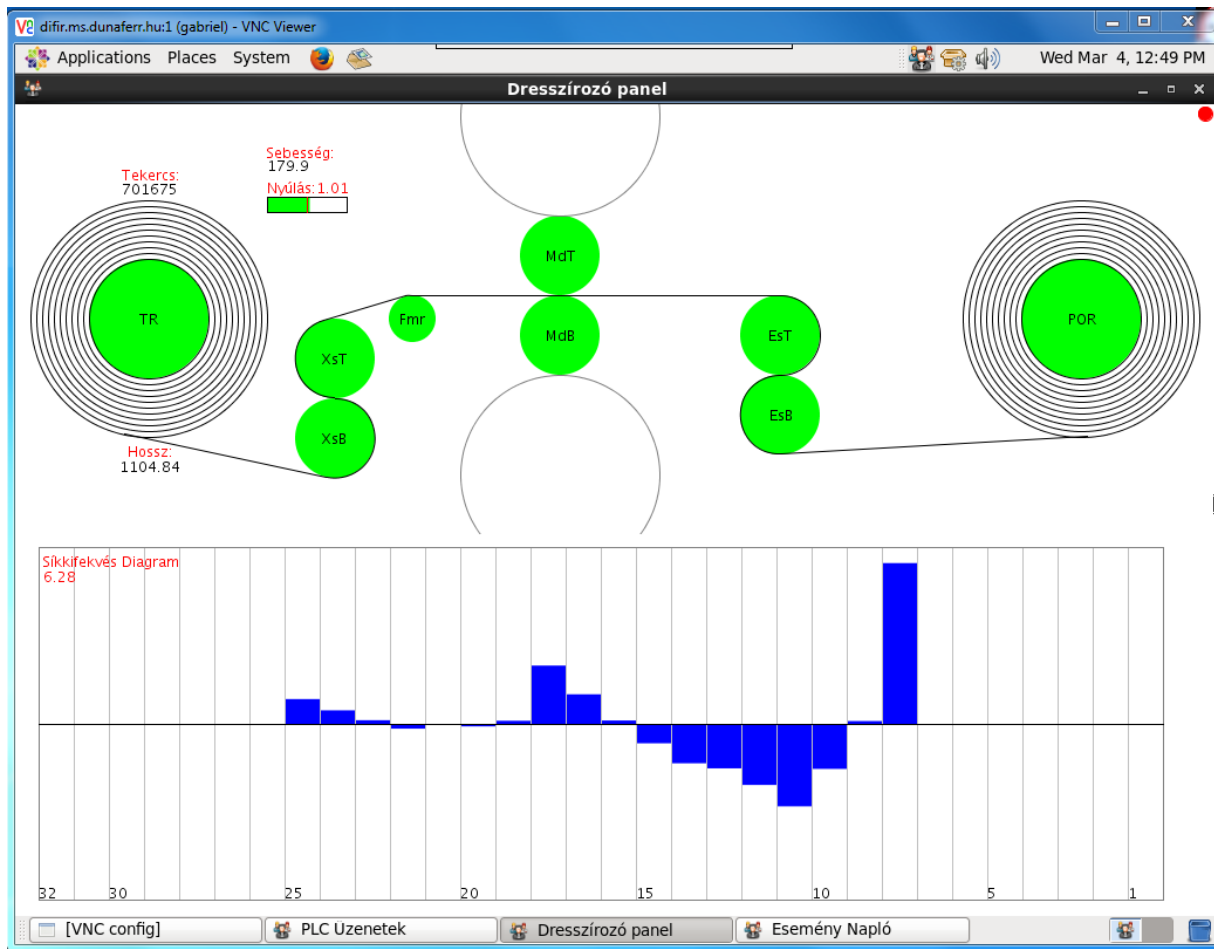
Név	Tipus	Érték
TelegrId	short	860
CoilId	S7String	701704
CoilPartNo	short	0
ProcType	short	0
ElongRef	float	1.0
ElongAct	float	0.9808327
FlatDevZone1	float	0.0
FlatDevZone2	float	0.0
FlatDevZone3	float	0.0
FlatDevZone4	float	0.0
FlatDevZone5	float	0.0
FlatDevZone6	float	0.0
FlatDevZone7	float	0.0
FlatDevZone8	float	-0.9316667
FlatDevZone9	float	-1.0716666
FlatDevZone10	float	-1.3916667
FlatDevZone11	float	-1.2116666
FlatDevZone12	float	-0.07166668
FlatDevZone13	float	-0.28166667
FlatDevZone14	float	-0.97166663
FlatDevZone15	float	0.33833334
FlatDevZone16	float	-0.44166666
FlatDevZone17	float	0.18833332
FlatDevZone18	float	-1.0616666
FlatDevZone19	float	0.31833333
FlatDevZone20	float	-0.25166667
FlatDevZone21	float	-0.031666677
FlatDevZone22	float	-0.04166668
FlatDevZone23	float	0.31833333
FlatDevZone24	float	0.8983333
FlatDevZone25	float	5.6983333
FlatDevZone26	float	0.0
FlatDevZone27	float	0.0
FlatDevZone28	float	0.0
FlatDevZone29	float	0.0
FlatDevZone30	float	0.0
FlatDevZone31	float	0.0

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

4.3.7 Dresszírozó panel

A dresszírozó panel az aktuális hengerlés ábráját mutatja:

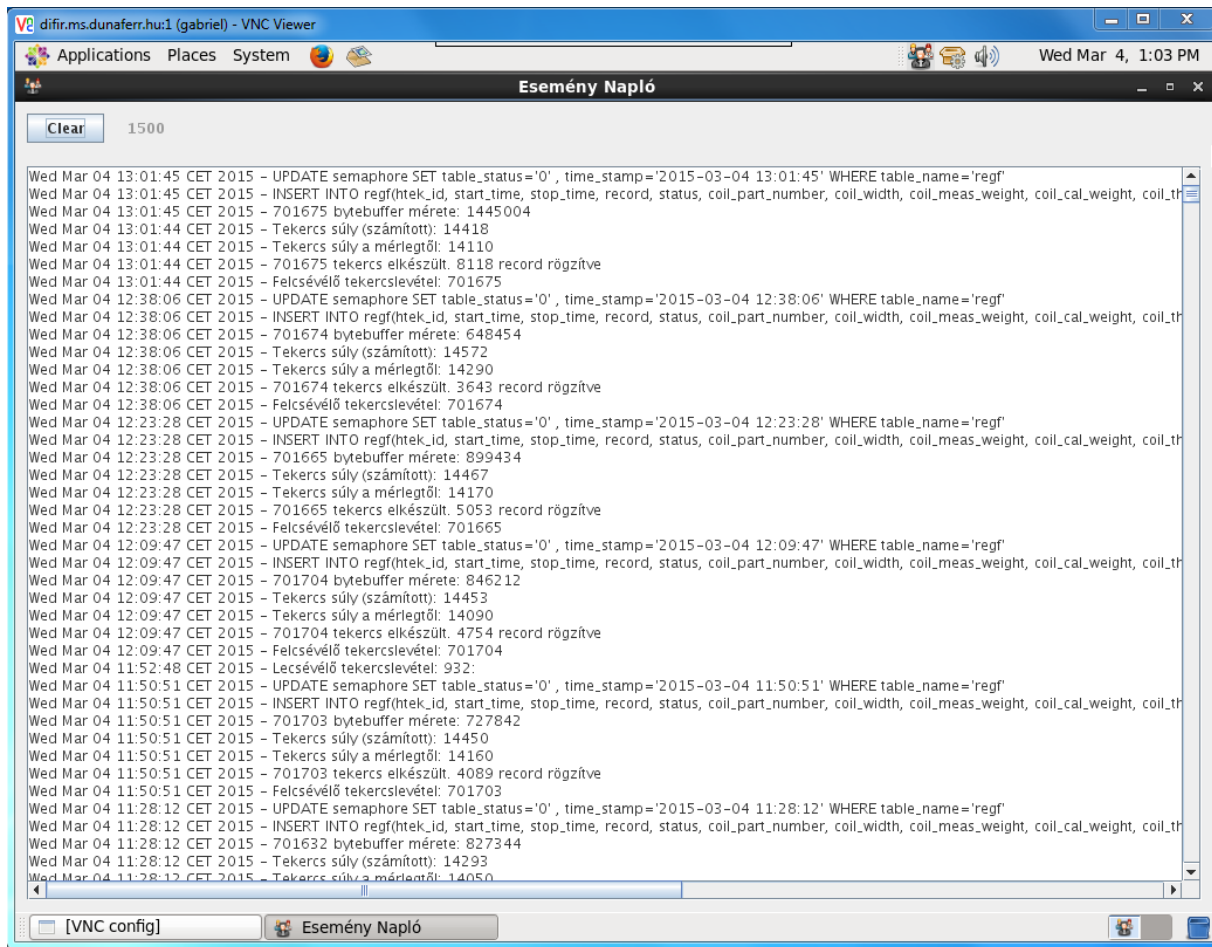


27. ábra - Dresszírozó panel

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 síkkihívás diagram látható.

4.3.8 – 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.



28. ábra - Esemény napló

4.3.9 Esemény napló fájlok

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

Példa a log fájlra:

```

1 [2015-02-28 00:03:23.493] [dfirplc.process.ProcessDB883] SelectDB883 : Felcsévéld tekercslevétel: 7010570000
2 [2015-02-28 00:03:23.497] [dfirplc.process.ProcessDB882] SelectDB882 :7010570000 tekercs elkészült. 1891 record rögzít
3 [2015-02-28 00:03:23.518] [dfirplc.process.ProcessDB882] SelectDB882 :7010570000 bytearray mérete: 336598
4 [2015-02-28 00:03:23.547] [dfirplc.sql.SQL] INSERT INTO regf(htek_id, start_time, stop_time, record, status, coil_part
5 [2015-02-28 00:03:23.549] [dfirplc.sql.SQL] UPDATE semaphore SET table_status='0' , time_stamp='2015-02-28 00:03:23' W
6 [2015-02-28 00:13:50.522] [dfirplc.process.ProcessDB883] SelectDB883 : Felcsévéld tekercslevétel: 7010590000
7 [2015-02-28 00:13:50.526] [dfirplc.process.ProcessDB882] SelectDB882 :7010590000 tekercs elkészült. 2373 record rögzít
8 [2015-02-28 00:13:50.554] [dfirplc.process.ProcessDB882] SelectDB882 :7010590000 bytearray mérete: 422394
9 [2015-02-28 00:13:50.588] [dfirplc.sql.SQL] INSERT INTO regf(htek_id, start_time, stop_time, record, status, coil_part
10 [2015-02-28 00:13:50.590] [dfirplc.sql.SQL] UPDATE semaphore SET table_status='0' , time_stamp='2015-02-28 00:13:50' W
11 [2015-02-28 00:21:36.898] [dfirplc.sql.SQL] SELECT * FROM 'dfir'. 'szurasterv' WHERE 1 order by pssschtelid limit 1
12 [2015-02-28 00:21:37.939] [dfirplc.process.ProcessDB886] SelectDB886 :7010820000 tekercs szurásterve leadva.
13 [2015-02-28 00:21:37.947] [dfirplc.sql.SQL] SELECT * FROM 'dfir'. 'szurasterv_log' WHERE 'pssschtelid' = 222
14 [2015-02-28 00:21:37.950] [dfirplc.sql.SQL] UPDATE 'szurasterv_log' SET 'ts_feldolgozas'='2015-02-28 00:21:36.0' WHERE
15 [2015-02-28 00:21:37.957] [dfirplc.sql.SQL] 1 sor törölve a szurasterv táblából. tekerccszám: 7010820000
16 [2015-02-28 00:22:19.582] [dfirplc.process.ProcessDB883] SelectDB883 : Felcsévéld tekercslevétel: 7010580000
17 [2015-02-28 00:22:19.586] [dfirplc.process.ProcessDB882] SelectDB882 :7010580000 tekercs elkészült. 1782 record rögzít
18 [2015-02-28 00:22:19.603] [dfirplc.process.ProcessDB882] SelectDB882 :7010580000 bytearray mérete: 317196
19 [2015-02-28 00:22:19.630] [dfirplc.sql.SQL] INSERT INTO regf(htek_id, start_time, stop_time, record, status, coil_part
20 [2015-02-28 00:22:19.632] [dfirplc.sql.SQL] UPDATE semaphore SET table_status='0' , time_stamp='2015-02-28 00:22:19' W
21 [2015-02-28 00:30:33.597] [dfirplc.process.ProcessDB883] SelectDB883 : Felcsévéld tekercslevétel: 7010600000
22 [2015-02-28 00:30:33.601] [dfirplc.process.ProcessDB882] SelectDB882 :7010600000 tekercs elkészült. 1884 record rögzít
23 [2015-02-28 00:30:33.624] [dfirplc.process.ProcessDB882] SelectDB882 :7010600000 bytearray mérete: 335352
24 [2015-02-28 00:30:33.653] [dfirplc.sql.SQL] INSERT INTO regf(htek_id, start_time, stop_time, record, status, coil_part
25 [2015-02-28 00:30:33.655] [dfirplc.sql.SQL] UPDATE semaphore SET table_status='0' , time_stamp='2015-02-28 00:30:33' W
26 [2015-02-28 00:30:48.928] [dfirplc.sql.SQL] SELECT * FROM 'dfir'. 'szurasterv' WHERE 1 order by pssschtelid limit 1
27 [2015-02-28 00:30:49.979] [dfirplc.process.ProcessDB886] SelectDB886 :7010830000 tekercs szurásterve leadva.
28 [2015-02-28 00:30:49.992] [dfirplc.sql.SQL] SELECT * FROM 'dfir'. 'szurasterv_log' WHERE 'pssschtelid' = 223
29 [2015-02-28 00:30:49.995] [dfirplc.sql.SQL] UPDATE 'szurasterv_log' SET 'ts_feldolgozas'='2015-02-28 00:30:48.0' WHERE
30 [2015-02-28 00:30:50.003] [dfirplc.sql.SQL] 1 sor törölve a szurasterv táblából. tekerccszám: 7010830000

```

29. 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.

5. Mellékletek

5.1 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 mérlegtől 5 sec-en belül (idotúllépés)
//     22 - A fogadott adatok hossza nem megfelelő (nem 39 karakter)
//     23 - A fogadott adatcsomag nem értelmezhető

AUTHOR: 'FD'
VERSION: '1.0'

// Bemeneti változók
VAR_INPUT
    TRIG_MEAS: BOOL;      // Súly lekérdezése a mérlegtől
    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 küldendő
    adatok

```



```

DISOMAT_RECV_DATA: ARRAY[0..35] OF BYTE;    // Mérlegtől érkezett
adatok

COMM_ENABLE: BOOL;        // Kommunikáció engedély

SEND_REQ: BOOL;           // Küldés parancs
SEND_LEN: INT;            // Küldendő adatok hossza
SEND_DONE: BOOL;         // Küldés kész

RECV_REQ: BOOL;           // Fogadás parancs
RECV_NDR: BOOL;           // Fogadás - új adatok érkeztek
RECV_LEN: INT;            // Érzékelt 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 időtúllépés

TIMERTIME: S5TIME;        // Időtúllépés hátralévő idő
TIMEOUT_TIME: S5TIME;     // Időtúllépés időtartam
TIMEOUT_TIMER_ON: BOOL;   // Időtúllépés figyelő timer engedélyezése
TIMEOUT_TIMER_RESET: BOOL; // Időtúllépés figyelő 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;    // Mérlegtől érkezett
adatok
PTR: INT;                                     // Mérlegtől érkezett
adat eltolódás figyeléshez használt mutató
END_VAR

// Kommunikáció 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 megfelelő művelet 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 kezdő STX
            kivételével a csomag minden bájta 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 mérlegtől (<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 ellenőrzése

            // 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
    // Értelmezhető 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;
    END_FOR;
```

```

// 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 kisbetűvel 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
```

5.2 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=NPJA5250AI_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=NPJA5250AI_1328@NPJA5250AI
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
```