

Displej UNI TXT(ND)

PARAMETRIZÁCIA KOMUNIKAČNÉHO ROZHRANIA PROFIBUS-DP

Obsah

1. ÚVOD	3
2. INŠTALÁCIA GSD SÚBORU	3
3. VZOROVÝ PROJECT ELEN_UNI-TXT(ND)	4
4. RIADIACE PRÍKAZY A FORMÁT TEXTU.....	5
4.1. PRÍKAZY PRE ZÁPIS TEXTU.....	6
4.1.1. Priamy zápis textu.....	6
4.1.2. Zmazanie displeja	6
4.1.3. Zápis textu na konkrétnu pozíciu	6
4.2. PRÍKAZY PRE FORMÁTOVANIE TEXTU	6
4.2.1. Riadenie blikania textu	6
4.2.2. Zmena farby textu	6
4.3. PRÍKAZY PRE GLOBÁLNE FUNKCIE DISPLEJA	7
4.3.1. Riadenie jasu displeja.....	7
4.3.2. Nastavenie komunikačného Timeout -u zobrazovania.....	7
4.3.3. Synchronizačný paket reálneho času	7
4.3.4. Parametre blikania textu	7
5. VAT TABUĽKA	8

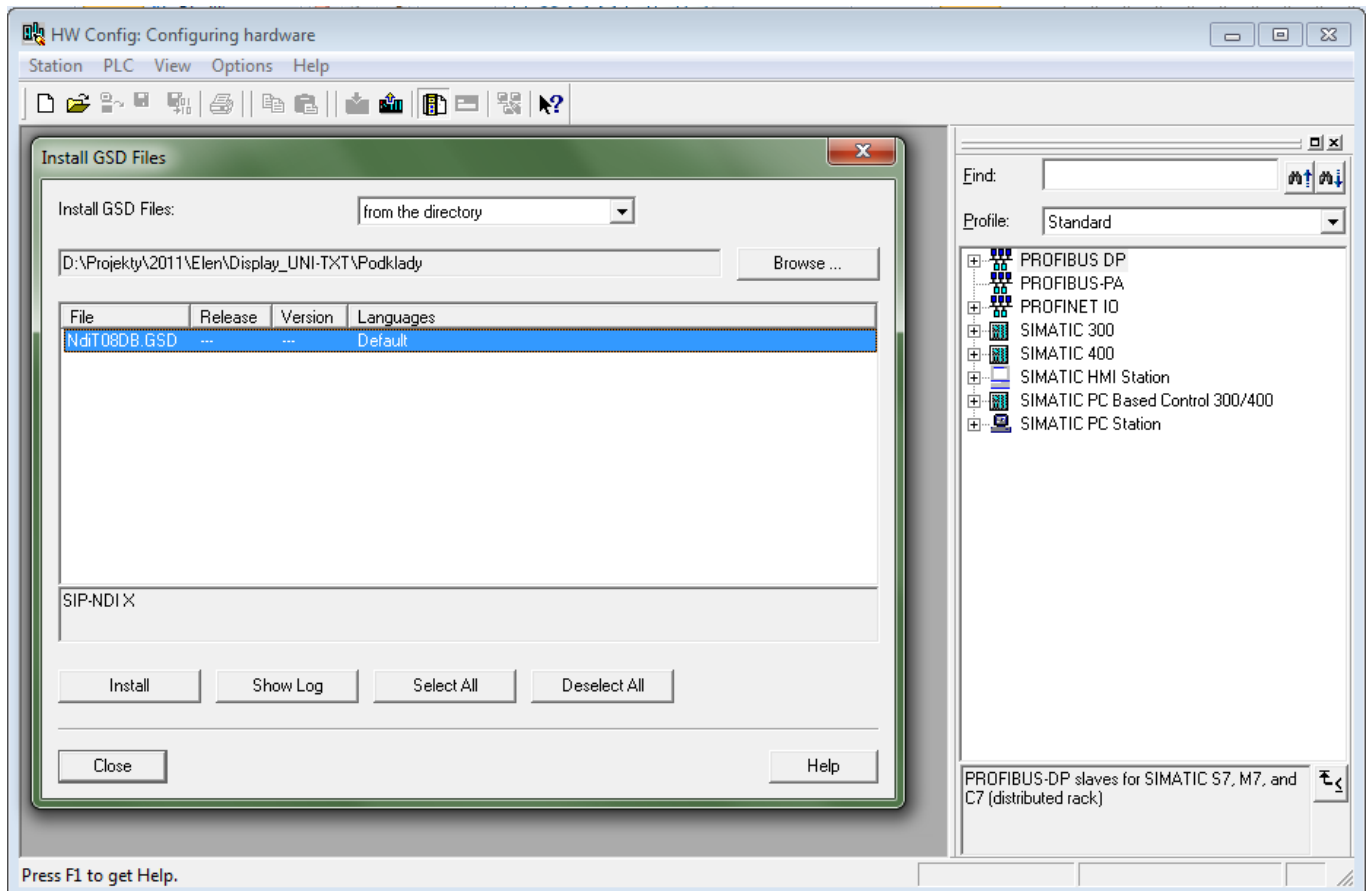
1. Úvod

Tento manuál je určený pre programátorov riadiacich systémov SIEMENS, SIMATIC S7. Prílohou tohoto manuálu je vzorový Step7 projekt „ELEN_UNI-TXT“, spracovaný v prostredí STEP7 ver. 5.5, ktorý je vývojovým prostredím riadiacich systémov SIMATIC S7-300 a S7-400. Vzorový Step7 projekt obsahuje jednoduchú konfiguráciu riadiaceho systému SIMATIC S7-300, pozostávajúceho z CPU 315-2DP ako Profibus-DP Master a textového displeja s komunikačným protokolom UNI TXT a s rozhraním PROFIBUS-DP ako slave.

2. Inštalácia GSD súboru

V prípade, že chceme doplniť displej UNI TXT do existujúceho Step7 projektu, ako prvý krok je potrebné tento .GSD súbor downloadovať a následne pridať displej UNI TXT do HW catalogu STEP7. V prostredí Step7 sa otvorí HW config editor a v menu „Options – Install GSD file ...“ pomocou funkcie „Browse“ nastaví adresár, v ktorom je uložený spomínaný súbor NdiT08DB.GSD, vid' Obr. 1 – Inštalácia GSD súboru. Následne tlačidlom Install previesť samotnú inštaláciu GSD súboru do HW katalógu vývojového prostredia Step7. Súbor NdiT08DB.GSD je dostupný na WEB stránkach firmy ELEN, s.r.o. (www.elen.sk).

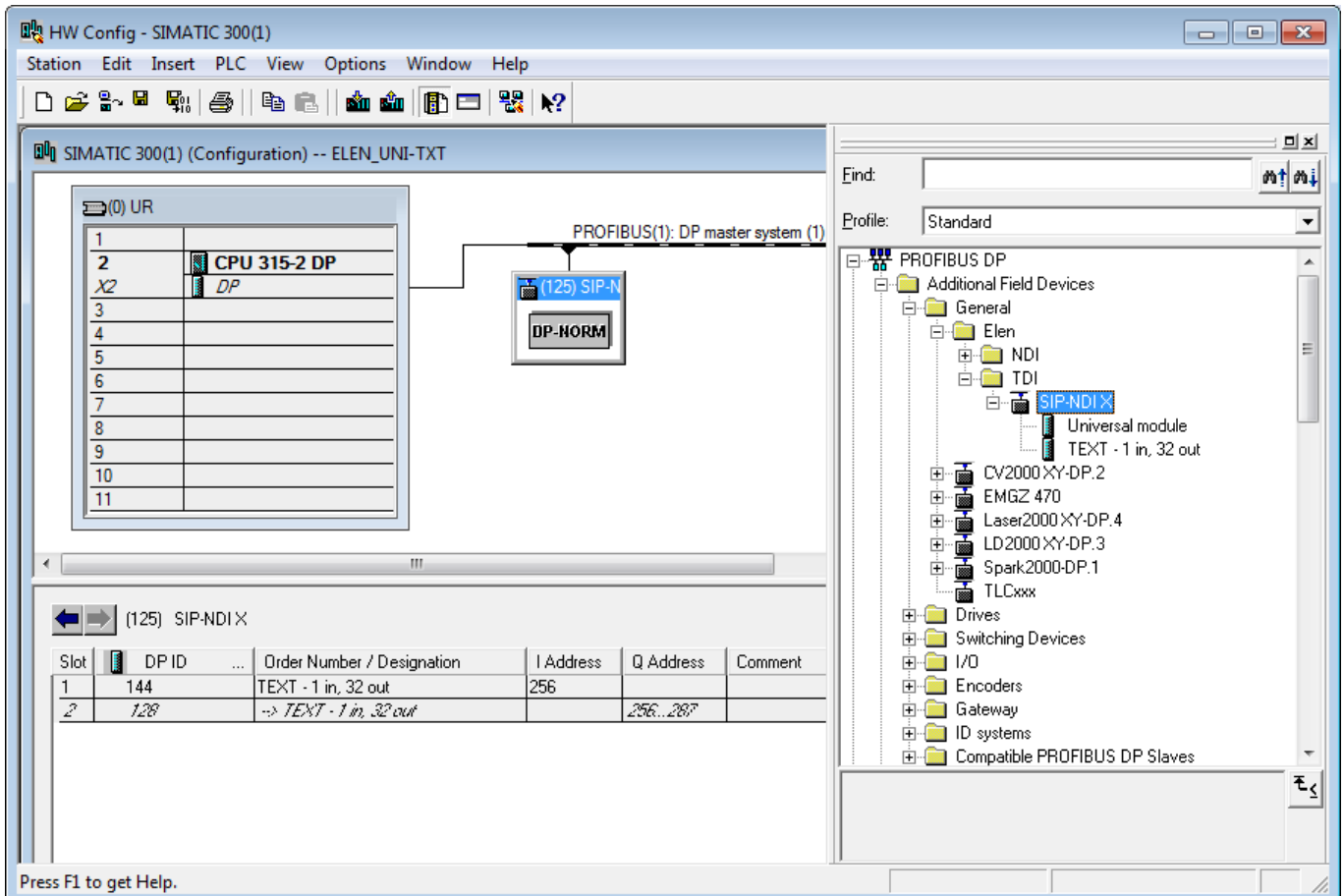
Ak pracujeme s displejom UNI TXT po prvý krát, doporučujeme preštudovať si vzorový Step7 projekt „ELEN_UNI-TXT“, popri prípade zrealizovať testovacie pracovisko t.j. CPU s Profibus DP rozhraním a displej NDI, vzorový Step7 projekt downloadovať do CPU a odskúšať funkcie displeja. Vo vzorovom projekte sú na sieti Profibus-DP dvaja účastníci. Ako PROFIBUS-DP Master je použitý CPU 315-2DP, profibus adresa 2. K nemu je pripojený PROFIBUS-DP Slave displej UNI TXT s továrensky preddefinovanou profibus adresou 125.



Obr. 1 – Inštalácia GSD súboru

Po úspešnom vložení GSD súboru je do HW katalogu doplnený nový Profibus slave s názvom „SIP-NDI X“, vid' Obr. 2 – HW katalog. V jeho virtuálnych slotoch sa nachádzajú dva bloky:

1. Vstup PLC „**TEXT 1 in**“ – 1 byte
2. Výstup PLC „**TEXT 32 out**“ – 32 byte



Obr. 2 – HW katalog

3. Vzorový projekt ELEN UNI-TXT (ND)

Po inštalácii GSD súboru do HW katalogu prostredia Step7 môžeme nový DP slave „SIP-NDI X“ vložiť do HW konfigurácie existujúceho Step7 projektu, alebo otvoríme vzorový Step7 projekt s názvom ELEN_NDI. V tomto vzorovom Step7 projekte sa v programovom adresári „Blocks“ nachádzajú dva vzorové funkčné bloky pre obsluhu displeja. Prvý je FB1, pri volaní je potrebné vyplniť nasledovné vstupno-výstupné parametre:

```
CALL "UNI-TXT_full" , "IDB_FB1"
PerAdr :=256                //Periférna adresa z HW konfigurácie
Blocks :=5                  //Počet prenášaných 30 bytových blokov
DataDisp:="DATA_DISP"     //Dátový blok s textami pre displej
Write := "Start_disp"      //Štartovací bit pre zápis z DataDisp do buffera displeja
```

Tento funkčný blok prostredníctvom rozhrania Profibus-DP prenáša žiadaný počet (v rozsahu 1 až 34) 30 bytových blokov zo Simaticu do vyrovnávacej pamäte displeja. Textové údaje pre displej sú

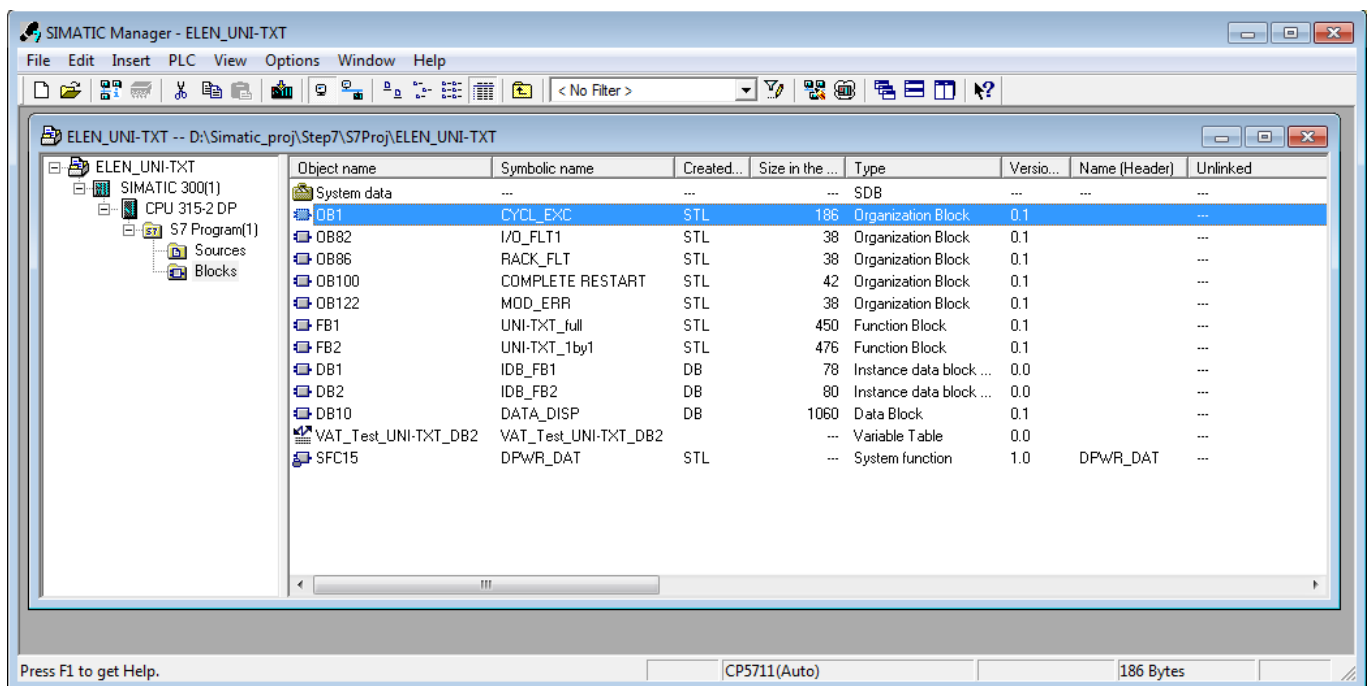
v Simaticu uložené v dátovom bloku DB10 ("DATA_DISP"). Vyrovnávací pamäť displeja má veľkosť 1024 byte a obsahuje textové reťazce, ktoré sa majú zobrazovať na displeji a riadiace znaky protokolu UNI-TXT. Nastavením štartovacieho bitu „Write“ sa spustí samotný prenos stanoveného počtu 30 bytových blokov a na konci sa odošle FFFFHex pre zobrazenie obsahu buffera na displej. Po ukončení prenosu sa bit Write automaticky resetuje. Samotný profibus zápis je realizovaný volaním systémovej funkcie SFC15 (DPWR_DAT). Celá komunikácia t.j. Prenos max. 34 blokov prebieha v 1 PLC cykle, čo by mohlo pri časovo kritických PLC programoch spôsobiť neprípustné predĺženie času cyklu.

Preto je v príklade uvedený aj druhý blok FB2, ktorého funkcia je analogická ale v jednom PLC cykle sa prenáša do vyrovnávacej pamäti displeja iba 1 blok o veľkosti 30 byte. Dole je uvedený príklad volania bloku FB2 s príslušným instant data blokom DB2.

```
CALL "UNI-TXT_1by1" , "IDB_FB2"
    PerAdr :=256                //Periférna adresa z HW konfigurácie
    Blocks :=5                  //Počet prenášaných 30 bytových blokov
    DataDisp:="DATA_DISP"      //Datový blok s textami pre displej
    Write :="Start_disp"       //Štartovací bit pre zápis údajov z DataDisp do buffera disp.
    Active :="Start_disp_active" //Príznakový bit s významom prenos údajov prebieha
```

Volanie FB1 s príslušným instant data blokom DB1 (popr. FB2,DB2) je v hlavnom programovom cykle OB1. V nábehovom bloku OB100 je inštrukcia pre set bitu „Write“, ktorá slúži na automatické zobrazenie textu po nábehu PLC.

Zoznam všetkých blokov vzorového projektu je v Obr. 3 – Step7 projekt ELEN_NDI.



Obr. 3 – Step7 projekt ELEN_NDI

4. Riadiace príkazy a formát textu

Ako bolo spomenuté v časti 2, komunikácia prebieha cez 1 vstupný byte a 32 výstupných byte. Výstupný blok má v prvých 2 byte pozíciu umiestnenia vo vyrovnávacej pamäti a v nasledujúcich 30 bytoch je textový obsah včítane riadiacich znakov. Zápis do vyrovnávacej pamäti teda prebieha po 30 bytových blokoch. Po odoslaní všetkých blokov s textovými údajmi sa obsah vyrovnávacej pamäte zobrazí na displeji zaslaním správy, ktorá má v prvých 2 bytoch obsah FFFF Hex. V prípade výpadku

komunikácie, napr. prerušenie komunikačnej linky profibus na displeji zostávajú posledne odoslané texty.

Informačný obsah pozostáva z príkazov pre prácu displeja s konkrétnym textom a tie sú oddelené znakom \$.

4.1. Príkazy pre zápis textu

4.1.1. Priamy zápis textu

Displej priamo zobrazí požadovaný text, pričom vymaže obsah celého riadku. Pri prekročení dĺžky riadku pokračuje zápis automaticky v nasledujúcom riadku.

abcdef jednoduchý zápis ASCII znakov textu, ktorý môže obsahovať aj príkazy na formátovanie textu popísané v kapitole 4.2.

4.1.2. Zmazanie displeja

\$0 Vymazanie užívateľskej časti displeja a všetkých atribútov textu

4.1.3. Zápis premennej na vyhradené miesto

\$P<p10X><p1X><n10X><n1X>text

<p10X><p1X> – pozícia začiatku textu v ASCII formáte
rozsah 0. – 64. pozícia
<n10X><n1X> – počet bajtov na zobrazenie
rozsah 0. – 64. pozícia
text – znakový reťazec, ktorý sa má zobraziť

Znakový reťazec môže obsahovať čísla, niektoré vybrané alfabetické znaky a desatinnú bodku. Na zobrazenie desatinnej bodky je možné použiť tak kód 0x2C, ako aj kód 0x2D. Znak desatinnej bodky je považovaný za samostatný znak. Pre zobrazenie samostatnej desatinnej bodky je možné použiť postupnosť prázdneho znaku a znaku desatinnej bodky.

4.2. Príkazy pre formátovanie textu

V jednom vysielanom rámci môže byť viacej príkazov tohto typu, príkazy môžu byť vnorené len do príkazu priameho zápisu textu vid'. Kapitola 4.1.1.

4.2.1. Riadenie blikania textu

\$F1 nasledujúce znaky budú blikať
\$F0 nasledujúce znaky neblikajú
 ASCII formát, default 0

Platnosť príkazu \$F je do ďalšej zmeny hodnoty príkazom \$F, resp. do vypnutia displeja, resp. do nastavenia parametrov z výroby príkazom \$R.

4.2.2. Zmena farby textu

Platí len pre viacfarebné LED displeje. Farba textu (default je C1):

\$C1 červená
\$C2 zelená
\$C3 žltá

4.3. Príkazy pre globálne funkcie displeja

Po resete sú default hodnoty a príkaz trvá až do prepísania novým príkazom. Nastavenia týchto parametrov ostáva zachované aj po reštarte displeja. V jednom vysielanom rámci môže byť len jeden príkaz.

4.3.1. Riadenie jasú displeja

\$B<typ riadenie jasú><hodnota jasú>

<typ riadenie jasú> – je znak typu nastavenia jasú

0 – riadenie jasú zadaním priamej hodnoty PWM bez automatickej regulácie

1 – riadenie jasú zadaním strmosti regulačnej krivky ASCII formát default 0

<hodnota jasú> – hodnota, kde bit D7 je 1, rozsah 0 – 100%, default 100

4.3.2. Nastavenie komunikačného Timeout-u zobrazovania

\$T<hodnota timeoutu>

<hodnota timeoutu> - hodnota od 0 do 127 kde bit D7 je 1

rozsah 0 – 127 sekúnd

4.3.3. Synchronizačný paket reálneho času

Displej nastaví interný reálny čas podľa obsahu synchronizačného paketu. Displej očakáva informáciu o čase a dátume vo formáte, ktorý obsahuje korekciu pre časové pásmo a DST (letný/zimný čas).

\$S<Y10X><Y1X><M10X><M1X><D10X><D1X>H10X><H1X><MIN10X><MIN1X><SEC10X><SEC1X>

<Y10X> desiatky roka v ASCII tvare

<Y1X> jednotky roka v ASCII tvare

<M10X> desiatky mesiaca v ASCII tvare

<M1X> jednotky mesiaca v ASCII tvare

<D10X> desiatky dňa v ASCII tvare

<D1X> jednotky dňa v ASCII tvare

<H10X> desiatky hodín v ASCII tvare

<H1X> jednotky hodín v ASCII tvare

<MIN10X> desiatky minút v ASCII tvare

<MIN1X> jednotky minút v ASCII tvare

<SEC10X> desiatky sekúnd v ASCII tvare

<SEC1X> jednotky sekúnd v ASCII tvare

4.3.4. Parametre blikania textu

\$G<perióda><plnenie>

<perióda> – perióda blikania x100 ms

rozsah 0 – 127 (t.j. 0,1 – 12,7 sekundy), hodnota, bit D7 je 1, default 5

<plnenie> – koľko percent z periódy blikania svieti displej

rozsah 0 – 100%, bit D7 je 1, default 50

5. VAT tabuľka

Na Obr. 4 – Vat_Test_UNI-TXT je VAT tabuľka pre prepisovanie premenných, deklarovaných pri volaní FB1 v hlavnom bloku OB1, je vhodná na odskúšanie všetkých funkcií displeja NDI.

	Address	Symbol	Symbol comment	Display format	Status value	Modify value
1	DB2.DBW 40	"IDB_FB2".Cycl_a	Actual 30bytes blo...	DEC	30	
2	M 10.0	"Start_disp"	Rewrite data in bu...	BOOL	false	
3	M 10.1	"Start_disp_active"	Rewriting data in b...	BOOL	false	
4	DB10.DBB 0	"DATA_DISP".Dis	Red color	CHARACTER	'\$\$'	'\$\$'
5	DB10.DBB 1	"DATA_DISP".Dis	Red color	CHARACTER	'C'	'C'
6	DB10.DBB 2	"DATA_DISP".Dis	Red color	CHARACTER	'1'	'1'
7	DB10.DBB 3	"DATA_DISP".Dis		CHARACTER	'a'	'a'
8	DB10.DBB 4	"DATA_DISP".Dis		CHARACTER	'b'	'b'
9	DB10.DBB 5	"DATA_DISP".Dis	Green color	CHARACTER	'\$\$'	'\$\$'
10	DB10.DBB 6	"DATA_DISP".Dis	Green color	CHARACTER	'C'	'C'
11	DB10.DBB 7	"DATA_DISP".Dis	Green color	CHARACTER	'2'	'2'
12	DB10.DBB 8	"DATA_DISP".Dis		CHARACTER	'c'	'c'
13	DB10.DBB 9	"DATA_DISP".Dis		CHARACTER	'd'	'd'
14	DB10.DBB 10	"DATA_DISP".Dis	Yellow color	CHARACTER	'\$\$'	'\$\$'
15	DB10.DBB 11	"DATA_DISP".Dis	Yellow color	CHARACTER	'C'	'C'
16	DB10.DBB 12	"DATA_DISP".Dis	Yellow color	CHARACTER	'3'	'3'
17	DB10.DBB 13	"DATA_DISP".Dis		CHARACTER	'e'	'e'
18	DB10.DBB 14	"DATA_DISP".Dis		CHARACTER	'f'	'f'
19	DB10.DBB 15	"DATA_DISP".Dis	Red color	CHARACTER	'\$\$'	'\$\$'
20	DB10.DBB 16	"DATA_DISP".Dis	Red color	CHARACTER	'C'	'C'
21	DB10.DBB 17	"DATA_DISP".Dis	Red color	CHARACTER	'1'	'1'
22	DB10.DBB 18	"DATA_DISP".Dis	New line	CHARACTER	'\$\$'	'\$\$'
23	DB10.DBB 19	"DATA_DISP".Dis	New line	CHARACTER	'C'	'C'
24	DB10.DBB 20	"DATA_DISP".Dis	New line	CHARACTER	'R'	'R'
25	DB10.DBB 21	"DATA_DISP".Dis	Time	CHARACTER	'\$\$'	'\$\$'
26	DB10.DBB 22	"DATA_DISP".Dis	Time	CHARACTER	'H'	'H'
27	DB10.DBB 23	"DATA_DISP".Dis	Time	CHARACTER	'0'	'0'
28	DB10.DBB 24	"DATA_DISP".Dis	New line	CHARACTER	'\$\$'	'\$\$'
29	DB10.DBB 25	"DATA_DISP".Dis	New line	CHARACTER	'C'	'C'
30	DB10.DBB 26	"DATA_DISP".Dis	New line	CHARACTER	'R'	'R'

Obr. 4 – Vat_Test_NDI