



Instrukcja

Programowanie panelu HMI35 z wyświetlaczem TFT 3,5"

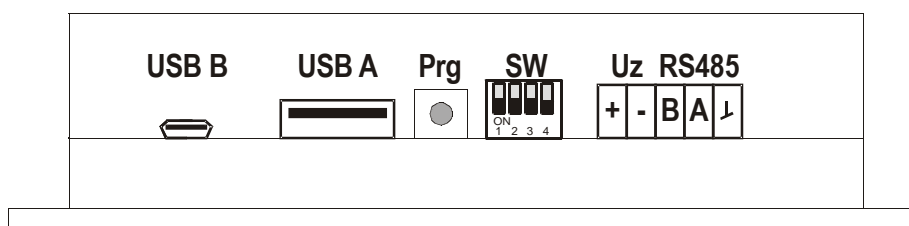
wersja dla kompilatora 7.xx

OPIS PANELU

Panel operatorski HMI35 jest swobodnie programowalnym panelem HMI z 3,5" ekranem dotykowym. Rozdzielczość ekranu wynosi 480x320 pikseli. Panel umożliwia komunikację z maksymalnie ośmioma urządzeniami typu SLAVE. Komunikacja może odbywać się poprzez magistralę szeregową RS485 z wykorzystaniem protokołu Modbus RTU lub poprzez sieć Wi-Fi (LAN) z wykorzystaniem protokołu Modbus TCP.

WEJŚCIA I WYJŚCIA

Widok złączy panelu




Oznaczenia	Opis
Uz	Napięcie zasilania 12...24VDC: + - biegun dodatni (+) napięcia zasilania, - - biegun ujemny (masa) napięcia zasilania.
RS485	Port RS485.
SW	Przełącznik funkcji panelu. Opis: SW1 – nieobstugiwany, SW2=ON – wejście do trybu programowania, SW3=ON – wejście do trybu konfiguracji modułu Wi-Fi, SW4=ON – wejście do trybu kalibracji ekranu dotykowego. W trybie normalnej pracy wszystkie przełączniki powinny być ustawione na OFF.
Prg	Przycisk trybu aktualizacji firmware.
USB A	Złącze USB A przeznaczone do podłączenia pendrive z oprogramowaniem użytkownika.
USB B	Złącze mikro USB B przeznaczone do aktualizacji firmware.

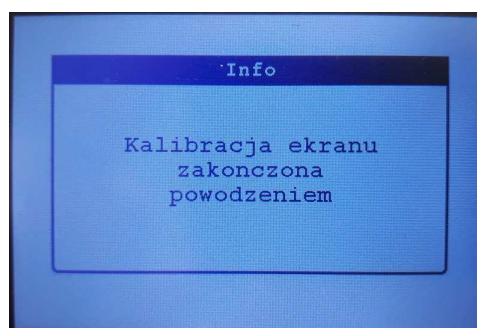
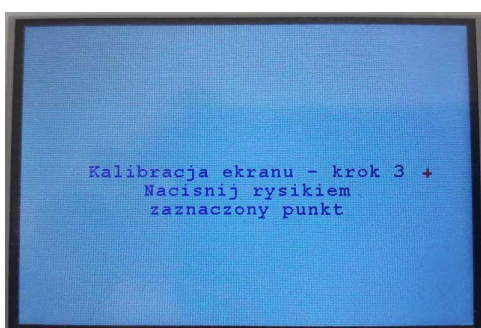
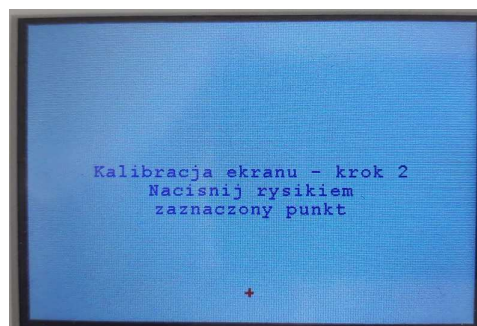
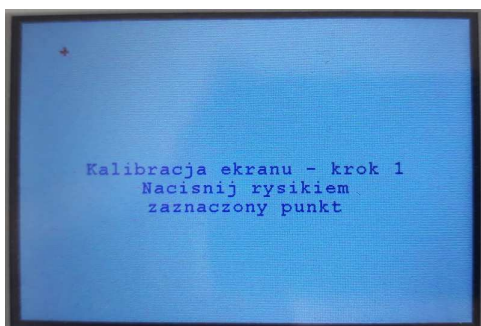
DIODY


Panel posiada w prawym górnym rogu trzy diody.

Oznaczenia	Opis
PWR	Świeci na zielono, jeśli panel ma podłączone zasilanie.
STATUS	Miga na zielono, jeśli nie ma żadnej awarii. Wszystkie rejestry statusowe zadeklarowane w projekcie mają wartość równą „0”. Miga na czerwono, jeśli wykryto awarię lub brak komunikacji z zadeklarowanym sterownikiem. Awaria oznacza, że przynajmniej jeden rejestr statusowy zadeklarowany w projekcie ma wartość różną od „0”. Rejestry statusowe deklaruje się w plikach predefiniowanych o nazwach ster0.txt ... ster7.txt w parametrze ADR STATUS .
LAN	Świeci na zielono, gdy aktywne jest połączenie przez Wi-Fi (niezależnie od trybu pracy wbudowanego modułu).

KALIBRACJA EKRANU DOTYKOWEGO

W celu skalibrowania ekranu dotykowego panelu trzeba wyłączyć jego zasilanie, przełączyć SW4 na ON  i włączyć zasilanie. Następnie naciskać na punkty pojawiające się na ekranie wyświetlacza.



Po poprawnym przejściu procedury kalibracji wyświetli się komunikat informacyjny: „Kalibracja ekranu zakończona powodzeniem”. Następnie trzeba wyłączyć zasilanie, przełączyć SW4 w pozycję OFF  i włączyć ponownie zasilanie.

PROGRAMOWANIE

Programowanie panelu odbywa się poprzez proste skrypty zapisywane w plikach tekstowych. Do napisania skryptów wystarczy standardowy notatnik Windows. Do kompilacji służy dedykowany, dostarczany bezpłatnie, kompilator „Kompilator5”. Wgranie gotowego programu do panelu HMI35 ogranicza się do zapisania skompilowanego wynikowego pliku na pendrive i uruchomieniu procedury programowania na panelu. Kompilator i przykładowy projekt można ściągnąć ze strony <https://frisko.com.pl> z działu Do Pobrania.

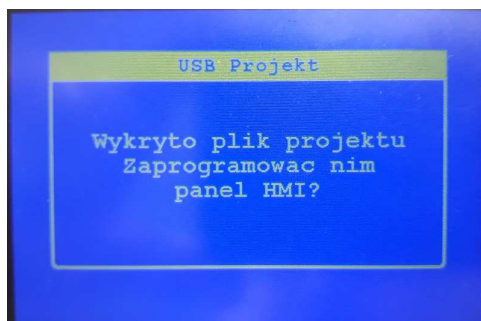
Projekt składa się plików tekstowych (z rozszerzeniem .txt), które mogą być napisane za pomocą zwykłego notatnika lub innego programu tekstowego. Umożliwia to stworzenie programu na panel HMI35 przez zwykłego użytkownika po przeczytaniu tej instrukcji. Nie jest wymagany żaden specjalistyczny program. Po napisaniu wszystkich potrzebnych plików projektowych trzeba uruchomić kompilator, który zweryfikuje i połączy wszystkie pliki, a następnie utworzy plik wynikowy **hmi35.bin**. Gotowy plik z projektem należy skopiować na pendrive - nośnik pamięci z USB typu A.

W celu zaprogramowania panelu trzeba wyłączyć jego zasilanie, przełączyć SW2 na ON




i włożyć pendrive do gniazda USB. Po włączeniu zasilania postępować zgodnie z instrukcjami wyświetlanymi na ekranie.

Jeśli na nośniku pamięci wykryty zostanie plik z programem o nazwie: **hmi35.bin** to wyświetli się następujący ekran:

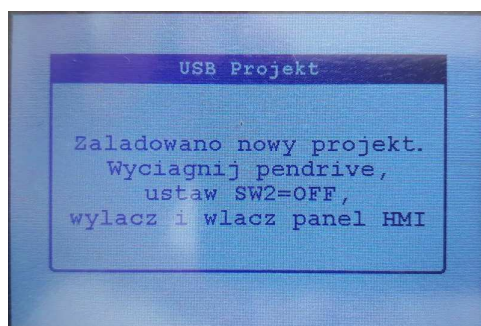


Jeśli chcemy zaprogramować nasz panel HMI35 nowym programem to wystarczy dotknąć ekran. Jeśli nie chcemy nowego programu trzeba wyłączyć zasilanie, wyjąć nośnik

pamięci i przełączyć SW2 w pozycję OFF .

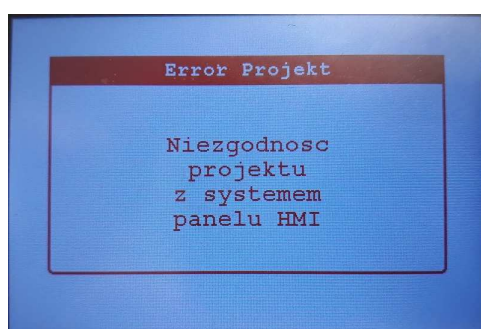
Podczas wgrывania programu następuje weryfikacja pamięci i w przypadku wystąpienia błędów wyświetlony zostanie odpowiedni komunikat, a nowy program nie będzie zaakceptowany.

Jeśli program zostanie poprawnie wgrany to wyświetli się następujący komunikat:



Po wgraniu należy wyłączyć zasilanie, wyciągnąć nośnik pamięci i przełączyć SW2 na OFF a następnie włączyć zasilanie.


Jeśli wyświetli się następujący komunikat:

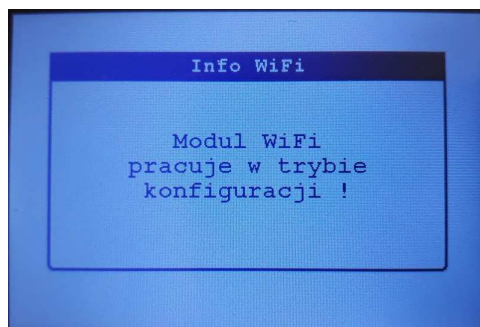


oznacza to, że system w panelu i kompilator użyty do stworzenia programu mają różne wersje.

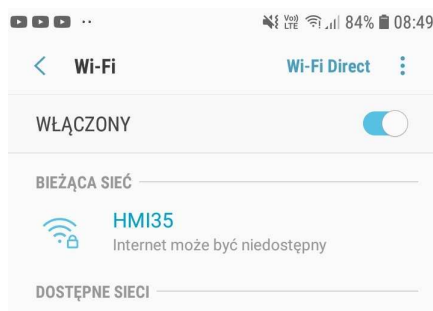
KONFIGURACJA MODUŁU WI-FI

Konfiguracja wewnętrznego modułu Wi-Fi w panelu HMI35 odbywa się poprzez sieć Wi-Fi z poziomu przeglądarki WWW. Najlepiej do tego celu użyć smartfona lub tabletu. W celu

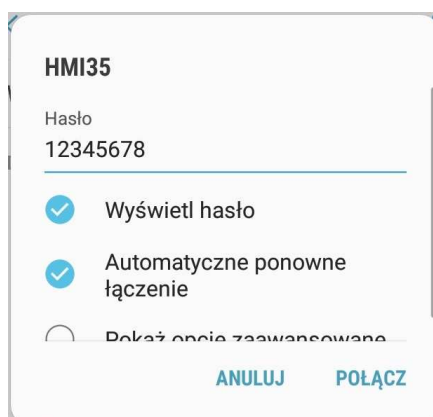
włączenia trybu konfiguracji trzeba wyłączyć zasilanie, przełączyć SW3 na ON  i włączyć zasilanie. Dioda 'LAN' zacznie migać na zielono a na ekranie powinien się wyświetlić następujący ekran informacyjny:



Teraz panel HMI35 pracuje jako access point oraz serwer WWW. Musisz połączyć się z nim poprzez Wi-Fi. Włącz w swoim urządzeniu (smartfonie, tablecie) obsługę sieci Wi-Fi. Znajdź sieć o nazwie "HMI35" i połącz się z nią.



Hasło dostępu to "12345678".



Następnie uruchom przeglądarkę, w polu adresu wpisz "192.168.4.1" i naciśnij klawisz Enter (lub OK).




Na ekranie smartfona (tabletu) zostanie wyświetlona strona umożliwiająca konfigurację czujnika.


Znaczenie poszczególnych parametrów przedstawia poniższa tabela.

Parametr	Opis
Nazwa WiFi	Nazwa sieci Wi-Fi (ssid) do której ma być podłączony panel HMI35.
Hasło WiFi	Hasło dostępu do sieci Wi-Fi do której ma być podłączony panel HMI35.
DHCP	Obsługa funkcji DHCP. Wybór opcji "Tak" oznacza, że adres IP panelu HMI35 zostanie przydzielony automatycznie poprzez usługę DHCP. Wybór opcji "Nie" oznacza wyłączenie funkcji DHCP. Adres IP czujnika, maskę i bramę należy skonfigurować ręcznie.
Adres IP	Adres IP czujnika, gdy DHCP=Nie.
Maska	Maska sieci, gdy DHCP=Nie.
Brama	Brama sieci, gdy DHCP=Nie.

Po poprawnym wypełnieniu listy parametrów naciśnij klawisz "Zapisz" znajdujący się na dole strony WWW. Po tym w przeglądarce zostanie wyświetlony poniższy komunikat.



Następnie trzeba wyłączyć zasilanie, przełączyć SW3 w pozycję OFF  i włączyć ponownie zasilanie.

 W przypadku nieprawidłowego ustawienia parametrów nie zostaną one zapisane, a po naciśnięciu klawisza "Zapisz" zostanie ponownie wyświetlona strona z parametrami.

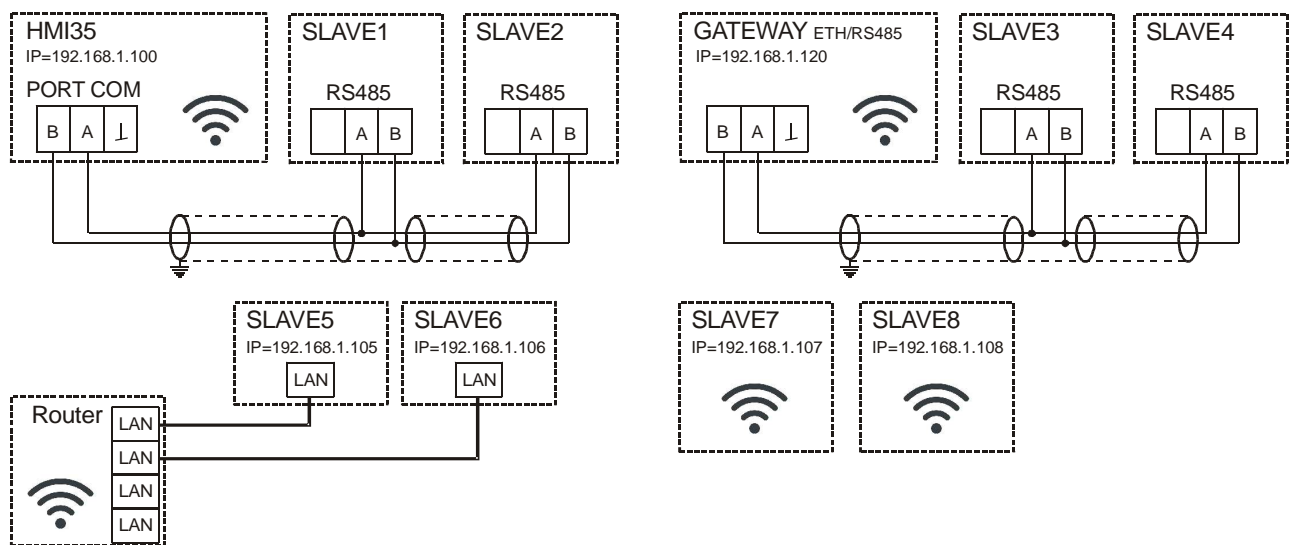
KOMUNIKACJA ZE STEROWNIKAMI

Panel umożliwia komunikację maksymalnie z ośmioma sterownikami typu SLAVE (w projekcie oznaczonymi numerami od 0 do 7). Komunikacja ze sterownikami może odbywać się:

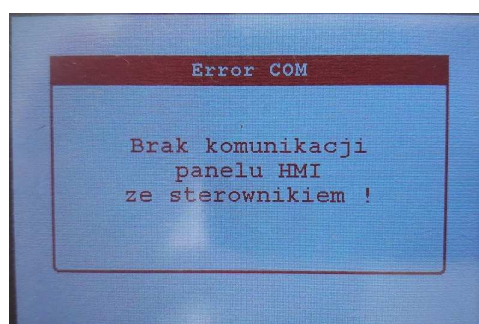
- poprzez magistralę szeregową RS485 z wykorzystaniem protokołu Modbus RTU,
- poprzez sieć Wi-Fi (LAN) z wykorzystaniem protokołu Modbus TCP.

Wyboru trybu komunikacji z danym sterownikiem dokonuje się na etapie tworzenia deklaracji tego sterownika (pliku **ster.txt**) wpisując odpowiednią wartość parametru "MAG". Prędkość oraz format danych magistrali RS485 jest wspólny dla wszystkich sterowników z niej korzystających i definiowany jest w pliku **projekt.txt**.

Przykładowy schemat podłączenia przedstawia poniższy rysunek.



W przypadku wykrycia braku komunikacji z danym sterownikiem SLAVE zostanie wyświetlony poniższy komunikat o błędzie:



JAK ZACZAĆ PROJEKT

Najprostszy projekt posiada jeden ekran i komunikuje się z jednym sterownikiem. Każdy projekt musi posiadać pliki **projekt.txt**, **ster0.txt** oraz **ekran0.txt**. Wszystkie pliki projektu muszą być w jednym katalogu, w którym znajduje się również kompilator. Pliki projektu tworzymy i edytujemy za pomocą dowolnego edytora tekstu np. dobrze znanego z Windows Notatnika. Schemat blokowy przykładowego projektu, znajduje się poniżej.

PROJEKT

Projekt jest zapisany w pliku **projekt.txt**. Każdy parametr projektu musi być zapisany w osobnej linii pliku.

Wszystkie wymagane parametry projektu przedstawia poniższa tabela.

Parametr	Opis
ID=	Numer ID projektu.
NAZWA=	Nazwa projektu. Maksymalna długość nazwy to 16 znaków. Nazwa projektu jest wyświetlana na ekranie startowym.
WERSJA=	Wersja projektu: np. data. Maksymalna długość wersji to 16 znaków. Wersja projektu jest wyświetlana na ekranie startowym.
PASS1=	Hasło dostępowe - poziom 1. Możliwe są 3 poziomy dostępu. Wszystkie obiekty, które mają możliwość edycji mogą być chronione hasłami określonymi przez parametry PASS1, PASS2 i PASS3. Dostęp z wykorzystaniem haseł realizowany jest przez obiekt PASS. Obiekt ten w wyniku swojego działania ustawia rejestr systemowy HMI_PASS_LEVEL (patrz opis sterowników oraz obiektu PASS). Podane tutaj hasło jest stałe nie podlegające edycji. Jeśli podamy PASS1=0, PASS2=0 oraz PASS3=0 to cały projekt nie będzie korzystał z funkcji hasła i wszystkie obiekty będą normalnie dostępne do edycji przez wszystkich użytkowników.
PASS2=	Hasło dostępowe - poziom 2.
PASS3=	Hasło dostępowe - poziom 3.
KOLOR_TLO=	Kolor tła w formacie RGB 24b (24 bitowym). Liczby muszą być rozdzielone przecinkiem.
KOLOR_SYS=	Kolor systemowy w formacie RGB 24b. W tym kolorze będą generowane komunikaty systemowe, klawiatura i listy wyboru.
EKRANY=	Liczba zdefiniowanych ekranów w ramach projektu. Maksymalna liczba ekranów, jaką można zdefiniować to 32. Ekran musi być opisane w plikach o predefiniowanych nazwach ekran0.txt ... ekran31.txt . Pierwszy ekran projektu ma numer zero i jest opisany w pliku ekran0.txt . Drugi ekran opisany jest w pliku ekran1.txt itd. Liczba plików opisujących ekran musi się zgadzać z liczbą przypisaną do parametru EKRANY . <i>Przykład:</i> <i>Dla EKRANY=1 tworzymy tylko plik ekran0.txt.</i> <i>Dla EKRANY=3 tworzymy pliki ekran0.txt, ekran1.txt, ekran2.txt.</i> Ekran startowy (z nazwą projektu oraz logo FRISKO) nie jest wliczany do liczby ekranów.
STER=	Liczba sterowników, z którymi ma się komunikować panel w ramach projektu. Maksymalna liczba sterowników to 8. Sterowniki muszą być opisane w plikach o predefiniowanych nazwach ster0.txt ... ster7.txt . Pierwszy sterownik projektu ma numer zero i jest opisany w pliku ster0.txt . Drugi sterownik opisany jest w pliku ster1.txt itd. Liczba plików opisujących sterowniki musi się zgadzać z liczbą przypisaną do parametru STER . <i>Na Przykład:</i> <i>Dla STER=1 tworzymy tylko plik ster0.txt.</i> <i>Dla STER=3 tworzymy pliki ster0.txt, ster1.txt, ster2.txt.</i>
COM_DATA=	Konfiguracja portu COM (magistrali szeregowej RS485) - format danych. Opcje: <ul style="list-style-type: none"> ▪ 8n1 - 8 bitów danych, brak kontroli parzystości, 1 bit stopu, ▪ 8e1 - 8 bitów danych, kontrola parzystości parzystej EVEN, 1 bit stopu, ▪ 8o1 - 8 bitów danych, kontrola parzystości nieparzystej ODD, 1 bit stopu.

COM_V=	Konfiguracja portu COM (magistrali szeregowej RS485) - prędkość transmisji. Opcje: <ul style="list-style-type: none">▪ 1200 - 1200 bps,▪ 2400 - 2400 bps,▪ 4800 - 4800 bps,▪ 9600 - 9600 bps,▪ 19200 - 19200 bps.
---------------	--

Przykład

Pokazany poniżej plik **projekt.txt** opisuje projekt o nazwie "MyProjekt1" (NAZWA=MyProjekt1) i wersji "20.10.2023" (WERSJA=20.10.2023). Dostęp do parametrów nie będzie chroniony hasłem (PASS=0). Projekt zawiera 4 ekrany (EKRANY=4). Ekrany muszą być opisane w plikach ekran0.txt, ekran1.txt, ekran2.txt oraz ekran3.txt. Kolorem tła ekranów będzie kolor biały (KOLOR_TLO=255,255,255). Kolorem podstawowym, systemowym będzie kolor zielony (KOLOR_SYS=0,255,0). Projekt będzie obsługiwał dane z jednego sterownika (STER=1). Właściwości sterownika muszą być opisane w pliku ster0.txt. Parametry portu RS485 (magistrali com): prędkość 9600bps (COM_V=9600), format znaku 8N1 (COM_DATA=8N1).

Plik **projekt.txt**

```
ID=2
NAZWA=MyProjekt1
WERSJA=20.10.2023
PASS1=0
PASS2=0
PASS3=0
KOLOR_TLO=255,255,255
KOLOR_SYS=0,255,0
EKRANY=4
STER=1
COM_DATA=8n1
COM_V=9600
```

STEROWNIKI

Liczbę sterowników, z którymi ma się komunikować panel w ramach projektu określa parametr **STER** w pliku „projekt.txt”. Maksymalna liczba sterowników to 8. Sterowniki numerowane są od 0 do 7. Sterowniki muszą być opisane w plikach o predefiniowanych nazwach **ster0.txt...ster7.txt**.

Pierwszy sterownik projektu ma numer zero i jest opisany w pliku **ster0.txt**. Drugi sterownik opisany jest w pliku **ster1.txt** itd. Liczba plików opisujących sterowniki musi się zgadzać z liczbą przypisaną do parametru **STER**.

Każdy parametr sterownika musi być zapisany w osobnej linii pliku.

Wszystkie wymagane parametry sterownika przedstawia poniższa tabela.

Parametr	Opis
SL_ADR=	Adres SLAVE sterownika, z którym komunikuje się panel.
SL_DFFORMAT=	Format zapisu liczb 16 bitowych (rejestrów) w sterowniku. Opcje: <ul style="list-style-type: none">▪ zm - format: znak-moduł. Format używany między innymi w sterownikach firmy FRISKO serii: MR65, MR208, MR210, RX910, RX911, S20.▪ u2 - format: U2 uzupełnienia do dwóch. Format używany w większości sterowników dostępnych na rynku, między innymi w sterownikach firmy FRISKO serii: ATTO i S1000.
SL_DADR=	Adres pierwszego 16 bitowego rejestru (holding register) dostępnego do doczytu/zapisu z wykorzystaniem protokołu Modbus RTU lub Modbus TCP.
SL_DSIZE=	Liczba dostępnych rejestrów w sterowniku. Panel HMI umożliwia odczyt/zapis rejestrów 16 bitowych o adresach od SL_DADR do SL_DADR+SL_DSIZE-1.
SL_FADR=	Adres pierwszej flagi (coil) dostępnej do doczytu/zapisu z wykorzystaniem protokołu Modbus RTU lub Modbus TCP.
SL_FSIZE=	Liczba dostępnych flag. Panel HMI umożliwia odczyt/zapis flag o adresach od SL_FADR do SL_FADR+SL_FSIZE-1.
ADR_STATUS=	Adres rejestru statusowego w sterowniku. Panel HMI wykorzystuje rejestr statusowy do sterowania diodą "STATUS". W przypadku, gdy odczytane wartości rejestrów statusowych ze wszystkich zadeklarowanych sterowników są równe 0 dioda miga kolorem zielonym. Odczytanie z rejestru statusowego dowolnego sterownika wartości różnej od 0 powoduje mruganie diody kolorem czerwonym.
MAG=	Typ magistrali wykorzystywanej do komunikacji ze sterownikiem SLAVE. Opcje: <ul style="list-style-type: none">▪ com - komunikacja poprzez RS485 (port COM), z wykorzystaniem Modbus RTU,▪ lan - komunikacja poprzez sieć LAN (połączenie HMI z siecią realizowane przez Wi-Fi) z wykorzystaniem protokołu Modbus TCP.
TOUT=	Czas timeout odpowiedzi (maksymalny czas na odpowiedź sterownika) wyrażony w sekundach. Zalecana nastawa TOUT dla MAG=com to 500...1000, dla MAG=lan 2000...5000.
IP=	Adres IP sterownika. Parametr wymagany, gdy ustawiono MAG=lan.

Przykład

Pokazany poniżej plik **ster0.txt** opisuje sterownik o adresie SLAVE równym 1 (SL_ADR=1). Sterownik posiada rejestry 16 bitowe dostępne w przestrzeni adresowej 4000...4999 (SL_DADR=4000, SL_DSIZE=1000) oraz flagi dostępne w przestrzeni adresowej 7000...7999 (SL_FADR=7000, SL_FSIZE=1000). Rejestry zapisywane są w formacie znak-moduł (SL_DFFORMAT=zm). Adres rejestru statusowego wynosi 4011 (ADR_STATUS=4011). Komunikacja ze sterownikiem będzie realizowana poprzez magistralę RS485 z wykorzystaniem protokołu Modbus RTU (MAG=com). Timeout odpowiedzi ustawiony na 1 sekundę.

Plik ster0.txt

```
SL_ADR=1
SL_DFFORMAT=zm
SL_DADR=4000
SL_DSIZE=1000
SL_FADR=7000
SL_FSIZE=1000
ADR_STATUS=4011
MAG=com
TOUT=1000
```

Predefiniowany sterownik

Niezależnie od zadeklarowanej liczby sterowników panel obsługuje wirtualny predefiniowany sterownik o numerze 8. Sterownika tego nie definiujemy w pliku sterX.txt. Sterownik odnosi się do zasobów samego panelu HMI. Zdefiniowany jest następującymi parametrami:

```
SL_DFFORMAT=u2
SL_DADR=0
SL_DSIZE=4095
SL_FADR=7000
SL_FSIZE=1023
```

co oznacza, że dla użytkownika dostępne jest obszar rejestrów o adresach 0...4094 oraz obszar flag o adresach 7000...8022.

Rejestry o adresach 0...256 oraz flagi o adresach 0...256 mogą zostać zapamiętane w pamięci Flash umożliwiając konfigurowanie działania panelu HMI.

W celu zapisu rejestrów i flag do pamięci należy ustawić flagę PAR_HMI_WRITE_FLASH na 1. Po zapisie flaga zostanie automatycznie wyzerowana.

Część rejestrów i flag ma już systemowo przypisane funkcje z których można korzystać. Poniżej przedstawiono ich zestawienie wraz z krótkim opisem. Zestawienie to można bezpośrednio skopiować do pliku rejestry.txt i wykorzystywać we własnym projekcie.

```
//REG_FLAGA  ADRES      FUNKCJA
HMI_ADR_STER0;10;    //adres slave sterownika 0
HMI_ADR_STER1;11;    //adres slave sterownika 1
HMI_ADR_STER2;12;    //adres slave sterownika 2
HMI_ADR_STER3;13;    //adres slave sterownika 3
HMI_ADR_STER4;14;    //adres slave sterownika 4
HMI_ADR_STER5;15;    //adres slave sterownika 5
HMI_ADR_STER6;16;    //adres slave sterownika 6
HMI_ADR_STER7;17;    //adres slave sterownika 7
HMI_STER_IP_B1;18;   //adres IP, bajt B1.  IP= B1.B2.B3.B4
HMI_STER_IP_B2;19;   //adres IP, bajt B2.
HMI_STER_IP_B3;20;   //adres IP, bajt B3.
HMI_STER0_IP_B4;21;  //adres IP sterownika 0, bajt B4. IP= B1.B2.B3.HMI_STER0_IP_B4
HMI_STER1_IP_B4;22;  //adres IP sterownika 1, bajt B4. IP= B1.B2.B3.HMI_STER1_IP_B4
HMI_STER2_IP_B4;23;  //adres IP sterownika 2, bajt B4.
HMI_STER3_IP_B4;24;  //adres IP sterownika 3, bajt B4.
HMI_STER4_IP_B4;25;  //adres IP sterownika 4, bajt B4.
HMI_STER5_IP_B4;26;  //adres IP sterownika 5, bajt B4.
```

```

HMI_STER6_IP_B4;27; //adres IP sterownika 6, bajt B4.
HMI_STER7_IP_B4;28; //adres IP sterownika 7, bajt B4.
HMI_PASS_LEVEL;257; //aktualny poziom dostępu level=0,1,2,3 ustawiany przez komponent PASS
HMI_STATUS;256; //status hmi=0
HMI_RTC_SEKUNDY;260; //rtc hmi.Sekundy
HMI_RTC_MINUTY;261; //rtc hmi.Minuty
HMI_RTC_GODZINY;262; //rtc hmi.Godziny
HMI_RTC_DZIEEN_TYG;263; //rtc hmi.Dzień tygodnia 1...7
HMI_RTC_DZIEEN;264; //rtc hmi.DZien 1...31
HMI_RTC_MIESIAC;265; //rtc hmi.Miesiac 1...12
HMI_RTC_ROK;266; //rtc hmi.Rok 0...99
HMI_STER0_STATKOM;270; //status komunikacji sterownik 0. statkom=0...10, 0 to ERROR
HMI_STER1_STATKOM;271; //status komunikacji sterownik 1
HMI_STER2_STATKOM;272; //status komunikacji sterownik 2
HMI_STER3_STATKOM;273; //status komunikacji sterownik 3
HMI_STER4_STATKOM;274; //status komunikacji sterownik 4
HMI_STER5_STATKOM;275; //status komunikacji sterownik 5
HMI_STER6_STATKOM;276; //status komunikacji sterownik 6
HMI_STER7_STATKOM;277; //status komunikacji sterownik 7
HMI_FLASH_WRITE;7000; //polecenie zapisu REG0...255 i FLAG0...255 do pamięci FLASH
HMI_BUZER;7001; //konfiguracja buzera 0-off, 1-on
HMI_LCD_LED;7002; //konfiguracja podświetlania lcd 0-brak wygaszania, 1-wygaszanie aktywne

```

Dynamiczna zmiana adresu slave sterownika

Modyfikując z poziomu panelu rejestry HMI_ADR_STER0...HMI_ADR_STER7 możemy zmienić adresy slave sterowników 0...7 (opisanych plikami ster0.txt...ster7.txt).

Przykład

Sterownik opisany plikiem ster0.txt ma adres 1 (SL_ADR=1). Chcąc wykorzystać ten sam projekt do obsługi sterownika np. o adresie slave 4 nie musimy kompilować od nowa projektu i wgrywać go na HMI. Wystarczy, że zrobimy ekran konfiguracyjny w którym będzie możliwa zmiana rejestru HMI_ADR_STER0. Ustawiając rejestr HMI_ADR_STER0 na 4 panel będzie komunikował się ze sterownikiem o adresie 4. Wpisana wartość nie zostanie zapamiętana. Po resecie zasilana znów zostanie przywrócony adres 1. Taki mechanizm można wykorzystać do obsługi kilku sterowników o różnych adresach na raz. W przypadku, gdy chcemy na stałe zmienić adres należy po zmianie konfiguracji (tu rejestru HMI_ADR_STER0) ustawić jeszcze flagę HMI_FLASH_WRITE na 1 aktywując tym samym zapis do pamięci nieulotnej typu Flash (np. poprzez klawisz „zapis” powiązany z funkcją SET).

Dynamiczna zmiana adresu IP sterownika

Adres IP sterownika 0 (opisanego plikiem ster0.txt) wyznaczają rejestry:

```
HMI_STER_IP_B1, HMI_STER_IP_B2, HMI_STER_IP_B3, HMI_STER0_IP_B4
```

Adres IP sterownika 1 wyznaczają odpowiednio rejestry:

```
HMI_STER_IP_B1, HMI_STER_IP_B2, HMI_STER_IP_B3, HMI_STER1_IP_B4
```

Sterowniki muszą pracować w tej samej podsieci stąd 3 pierwsze bajty adresu IP są wspólne (HMI_STER_IP_B1, HMI_STER_IP_B2, HMI_STER_IP_B3).

Modyfikując z poziomu panelu rejestry HMI_STER_IP_B1, HMI_STER_IP_B2, HMI_STER_IP_B3 oraz HMI_STER0_IP_B4...HMI_STER7_IP_B4 możemy zmienić adresy IP sterowników 0...7 tym samym wprowadzić możliwość konfiguracji projektu to danej adresacji sieci LAN bez konieczności jego ponownego kompilowania.

W celu trwałej zmiany adresów IP należy po zmianie nastaw powyższych rejestrów ustawić flagę HMI_FLASH_WRITE na 1 aktywując zapis do pamięci nieulotnej typu Flash.

Odczyt statusu komunikacji panelu ze sterownikami

Rejestry HMI_STER0_STATKOM...HMI_STER7_STATKOM zawierają aktualny status komunikacji panelu z danym sterownikiem. Wartość danego rejestru =0 oznacza brak komunikacji. Wartość 10 to poprawna komunikacja bez błędów. Stany pośrednie oznaczają występowanie błędów w komunikacji.

Poziomy dostępu do parametrów panelu HMI

Definiując projekt określamy trzy hasła dostępowe PASS1, PASS2 i PASS3 wyznaczające trzy poziomy dostępu. Hasło PASS1 określa dostęp do poziomu 1. Hasło PASS2 określa dostęp do poziomu 2. Hasło PASS3 określa dostęp do poziomu 3. Za odblokowanie dostępu odpowiada obiekt PASS. To on umożliwia wprowadzenie hasła dostępowego przez użytkownika. Po wprowadzeniu hasła ustawiany jest rejestr systemowy HMI_PASS_LEVEL. Rejestr może przyjmować stany 0, 1, 2 i 3. stany te wyznaczają poziom dostępu. Wartość 0 to brak dostępu (blokada edycji). Wartości 1, 2 i 3 to aktualny poziom dostępu oraz odblokowanie edycji parametrów. Korzystając z właściwości ENABLE obiektów projektu powiązanej z rejestrem HMI_PASS_LEVEL możemy tworzyć grupy parametrów lub ekranów dostępne w zależności od posiadanych uprawnień.

W przypadku, gdy w projekcie zdefiniowano PASS1=0, PASS2=0 oraz PASS3=0 mechanizm kontroli dostępu jest nieaktywny.

Buzer

Panel umożliwia włączenie lub wyłączenie obsługi wbudowanego buzera. Konfiguracja pracy buzera realizowana jest przez flagę systemową HMI_BUZER. Ustawienie flagi HMI_BUZER na wartość 1 oznacza aktywny buzer. Ustawienie flagi HMI_BUZER na wartość 0 to wyłączenie buzera. W celu trwałej zmiany konfiguracji pracy buzera należy ustawić flagę HMI_FLASH_WRITE na 1 aktywując zapis do pamięci nieulotnej typu Flash.

Wygaszanie ekranu

Panel umożliwia włączenie lub wyłączenie wygaszania ekranu. Konfiguracja wygaszania realizowana jest przez flagę systemową HMI_LCD_LED. Ustawienie flagi HMI_LCD_LED na wartość 1 to aktywne wygaszanie ekranu. Ustawienie flagi HMI_LCD_LED na wartość 0 powoduje, że ekran nie będzie się wygaszał całkowicie a jedynie ściemniał. W celu trwałej zmiany konfiguracji wygaszania ekranu należy ustawić flagę HMI_FLASH_WRITE na 1 aktywując zapis do pamięci nieulotnej typu Flash.

EKRANY

Maksymalna liczba ekranów, jaką można zdefiniować to 32.

Ekranu muszą być opisane w plikach o predefiniowanych nazwach **ekran0.txt ... ekran31.txt**.

Pierwszy ekran projektu ma numer zero i jest opisany w pliku **ekran0.txt**. Drugi ekran opisany jest w pliku **ekran1.txt** itd. Liczba plików opisujących ekranu musi się zgadzać z liczbą przypisaną do parametru **EKRANY** w pliku „projekt.txt”.

Plik **ekran0.txt** zawiera wszystkie obiekty umieszczone na pierwszym ekranie.

Każdy opis obiektu zaczyna się znacznikiem otwierającym, czyli nagłówkiem z nazwą w ostrych nawiasach: <nazwa> i kończy znacznikiem zamykającym: </>.

Przykład:

Przykład pliku **ekran0.txt** z jednym elementem – nagłówkiem.

```
<nazwa>
TYP=NAGLOWEK
TXT=Menu
KOLOR=25,25,112
L_ICON=0
P_ICON=-1
</>
```

Maksymalna liczba obiektów, jaką można zdefiniować na jednym ekranie to 48.

Dla przejrzystości projektu możemy stosować puste linie pomiędzy kolejnymi dodawanymi obiektami.

Przykład pliku **ekran0.txt**

```
<nazwa>
TYP=NAGLOWEK
TXT=Menu
KOLOR=25,25,112
L_ICON=0
P_ICON=-1
</>

<klaw1>
TYP=KLAWISZ_EKRAN
NAZWA=Demo1
X=240
Y=55
W=300
H=50
ROWNAJ=centralnie
KOLOR=25,25,112
KOLOR_ON=192,192,192
EKARAN=2
STER=0
</>
```

```
<klaw2>
TYP=KLAWISZ_EKRAN
NAZWA=Demo2
X=240
Y=105
W=300
H=50
ROWNAJ=centralnie
KOLOR=25,25,112
KOLOR_ON=192,192,192
EKARAN=3
STER=0
</>
```

```
<klaw3>
TYP=KLAWISZ_EKRAN
NAZWA=Demo3
X=240
Y=155
W=300
H=50
ROWNAJ=centralnie
KOLOR=25,25,112
KOLOR_ON=192,192,192
EKARAN=4
STER=0
</>
```

Komentarze

W plikach opisujących ekran możliwe jest szybkie wyłączenie z kompilacji danej linii poprzez użycie symbolu „/”. Jest to znana metoda komentarzowania z języka C.

LISTA REJESTRÓW

W celu wygodniejszego pisania programów na panel HMI35 można wcześniej przygotować sobie listę rejestrów i flag ze zdefiniowanymi nazwami tych rejestrów i flag. Taka mapa pamięci zapisana w pliku rejestry.txt wygląda następująco:

Przykład pliku rejestry.txt

```
STER_MINUTY;4001
STER_GODZINY;4002
STER_DZIEN_TYG;4006
STER_AWARIA_TAZ;7187
STER_TZEW;4094
STER_OUT_NG;4211
STER_STATUS;4112
NAST_PRACA_CTR;4086
```

Po zdefiniowaniu takiej listy możemy w projekcie stosować w parametrach ADR lub ENABLE zamiast liczby 4002 oznaczającej rejestr, nazwę GODZINY, lub zamiast wartości 4211 nazwę OUT_NG.

Przykład: ADR=GODZINY

Predefiniowane rejestry panelu HMI – sterownika o numerze 8

Poniższe rejestry możesz skopiować do swojego pliku rejestry.txt.

```
//REG_FLAGA  ADRES  FUNKCJA
HMI_ADR_STER0;10;    //adres slave sterownika 0
HMI_ADR_STER1;11;    //adres slave sterownika 1
HMI_ADR_STER2;12;    //adres slave sterownika 2
HMI_ADR_STER3;13;    //adres slave sterownika 3
HMI_ADR_STER4;14;    //adres slave sterownika 4
HMI_ADR_STER5;15;    //adres slave sterownika 5
HMI_ADR_STER6;16;    //adres slave sterownika 6
HMI_ADR_STER7;17;    //adres slave sterownika 7
HMI_STER_IP_B1;18;    //adres IP, bajt B1. IP= B1.B2.B3.B4
HMI_STER_IP_B2;19;    //adres IP, bajt B2.
HMI_STER_IP_B3;20;    //adres IP, bajt B3.
HMI_STER0_IP_B4;21;    //adres IP sterownika 0, bajt B4. IP= B1.B2.B3.HMI_STER0_IP_B4
HMI_STER1_IP_B4;22;    //adres IP sterownika 1, bajt B4. IP= B1.B2.B3.HMI_STER1_IP_B4
HMI_STER2_IP_B4;23;    //adres IP sterownika 2, bajt B4.
HMI_STER3_IP_B4;24;    //adres IP sterownika 3, bajt B4.
HMI_STER4_IP_B4;25;    //adres IP sterownika 4, bajt B4.
HMI_STER5_IP_B4;26;    //adres IP sterownika 5, bajt B4.
HMI_STER6_IP_B4;27;    //adres IP sterownika 6, bajt B4.
HMI_STER7_IP_B4;28;    //adres IP sterownika 7, bajt B4.
HMI_PASS_LEVEL;257;    //aktualny poziom dostępu level=0,1,2,3 ustawiany przez komponent PASS
HMI_STATUS;256;    //status hmi=0
HMI_RTC_SEKUNDY;260;    //rtc hmi.Sekundy
HMI_RTC_MINUTY;261;    //rtc hmi.Minuty
HMI_RTC_GODZINY;262;    //rtc hmi.Godziny
HMI_RTC_DZIEEN_TYG;263;    //rtc hmi.Dzień tygodnia 1...7
HMI_RTC_DZIEEN;264;    //rtc hmi.DZien 1...31
HMI_RTC_MIESIAC;265;    //rtc hmi.Miesiac 1...12
HMI_RTC_ROK;266;    //rtc hmi.Rok 0...99
HMI_STER0_STATKOM;270;    //status komunikacji sterownik 0. statkom=0...10, 0 to ERROR
HMI_STER1_STATKOM;271;    //status komunikacji sterownik 1
HMI_STER2_STATKOM;272;    //status komunikacji sterownik 2
HMI_STER3_STATKOM;273;    //status komunikacji sterownik 3
HMI_STER4_STATKOM;274;    //status komunikacji sterownik 4
HMI_STER5_STATKOM;275;    //status komunikacji sterownik 5
HMI_STER6_STATKOM;276;    //status komunikacji sterownik 6
HMI_STER7_STATKOM;277;    //status komunikacji sterownik 7
HMI_FLASH_WRITE;7000;    //polecenie zapisu REG0...255 i FLAG0...255 do pamięci FLASH
HMI_BUZER;7001;    //konfiguracja buzera 0-off, 1-on
HMI_LCD_LED;7002;    //konfiguracja podświetlania lcd 0-brak wygaszania, 1-wygaszanie aktywne
```


OBIEKTY

Obiekty umieszczone na ekranach mogą być różnego typu i mają oczywiście różne parametry.

Typy obiektów, jakie mogą wystąpić w projekcji HMI:

- NAGLOWEK – nagłówek ekranu,
- VAR16 – wyświetlacz liczby 16-bitowej,
- TXT – wyświetlacz tekstu,
- AKCJA_EKRAN – akcja zmiany ekranu,
- AKCJA_NC_EKRAN – akcja zmiany ekranu bez odświeżania,
- AKCJA_DEC – akcja dekrementacji,
- AKCJA_INC – akcja inkrementacji,
- AKCJA_SET – akcja ustawienia zmiennej,
- AKCJA_FLAG_SET – akcja ustawienia flagi,
- KLAWISZ_EKRAN – przycisk zmiany ekranu,
- KLAWISZ_NC_EKRAN – przycisk zmiany ekranu bez odświeżania,
- KLAWISZ_DEC – przycisk dekrementacji,
- KLAWISZ_INC – przycisk inkrementacji,
- KLAWISZ_SET – przycisk ustawienia zmiennej,
- KLAWISZ_FLAG_SET – przycisk ustawienia flagi,
- BMP – obrazek typu RGB 24-bitowy
- IKONA – obrazek typu mono,
- ONOFF_IKONA – migająca ikona,
- LISTA_TXT – lista tekstów,
- LISTA_ALERT – lista alarmów,
- LISTA_BMP – lista obrazków typu RGB 24-bitowy,
- DYNLISTA_BMP – dynamiczna lista obrazków typu RGB 24-bitowy,
- LISTA_IKON – lista obrazków typu mono,
- PASS – obiekt umożliwiający wpisanie hasła,
- PROGRAM – interpretacja graficzna programu tygodniowego
- RAMKA – prostokątna ramka,
- BOX_ALERT – obiekt wyświetlający na całym ekranie alarmy,
- STATUS_COM – status komunikacji ze sterownikiem.

Poniżej zostały szczegółowo opisane wszystkie możliwe typy obiektów wraz ze wszystkimi ich parametrami.

Obiekt typu NAGLOWEK – nagłówek ekranu

Obiekt NAGLOWEK zawiera nazwę danego ekranu i umożliwia wyświetlenie dwóch systemowych klawiszy w lewym i prawym górnym rogu. Klawisze te umożliwiają przejście do innych ekranów.

zmienna	opis
TYP=	Typ danego obiektu.
TXT=	Nazwa danego ekranu. Maksymalna długość nazwy to 16 znaków. HMI nie obsługuje polskich znaków.
KOLOR=	Kolor całego nagłówka (nazwy i przycisków) w formacie RGB 24b (24-bitowym). Liczby muszą być rozdzielone przecinkiem.
L_ICON=	Lewy klawisz. Wartość tego parametru określa numer ekranu, do którego ma skoczyć program. Przypisanie wartości -1 powoduje, że dany klawisz nie będzie wyświetlany.
P_ICON=	Prawy klawisz. Opis analogiczny jak powyżej.

Przykład

Pokazany poniżej obiekt <nazwa> opisuje nagłówek ekranu (TYP=NAGLOWEK) o nazwie „Ekran startowy” (TXT=Ekran startowy). Nagłówek ten posiada jeden klawisz po prawej stronie po dotknięciu, którego zostaje wyświetlony ekran o numerze 1 (P_ICON=1). Cały nagłówek ma kolor niebieski (KOLOR=0,0,255).

```
<nazwa>  
TYP=NAGLOWEK  
TXT=Ekran startowy  
KOLOR=0,0,255  
L_ICON=-1  
P_ICON=1  
</>
```

Obiekt typu VAR16 – wyświetlacz liczby 16-bitowej

Obiekt VAR16 umożliwia wyświetlenie lub edycję parametrów liczbowych 16-bitowych, czyli rejestrów w sterowniku.

zmienna	opis																												
TYP=	Typ danego obiektu.																												
NAZWA=	Nazwa obiektu, będzie wyświetlana tylko podczas edycji tego parametru. Maksymalna długość nazwy to 16 znaków.																												
X=	Współrzędna x lewego górnego rogu wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy lewej krawędzi ekranu.																												
Y=	Współrzędna y lewego górnego rogu wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy górnej krawędzi ekranu.																												
ROWNAJ=	<p>Parametr określa, od której strony ma być narysowany obiekt. Opcje:</p> <ul style="list-style-type: none"> ▪ do lewej – obiekt będzie narysowany od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu, ▪ centralnie – obiekt będzie narysowany tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu, ▪ do prawej – obiekt będzie narysowany od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu. 																												
MIN=	Minimalna wartość liczby, jaką można wprowadzić.																												
MAX=	Maksymalna wartość liczby, jaką można wprowadzić.																												
FORMAT=	<p>Format wyświetlania liczby. Opcje:</p> <ul style="list-style-type: none"> ▪ fx1 – X – jedna cyfra, ▪ fx2 – XX – dwie cyfry, ... ▪ fx9 – XXXXXXXXX – dziewięć cyfr, ▪ fx02 – 0X – dwie cyfry z zerem wiodącym, ▪ fx03 – 00X – trzy cyfry z zerami wiodącymi, ... ▪ fx09 – 00000000X – dziewięć cyfr z zerami wiodącymi, ▪ fx1.1 – X.X – jedna cyfra i jedna cyfra po przecinku, ▪ fx2.1 – XX.X – dwie cyfry i jedna cyfra po przecinku, ... ▪ fx9.1 – XXXXXXXXX.X – dziewięć cyfr i jedna cyfra po przecinku, ▪ fx02.1 – 0X.X – dwie cyfry z zerem wiodącym i jedna cyfra po przecinku, ▪ fx03.1 – 00X.X – trzy cyfry z zerami wiodącymi i jedna cyfra po przecinku, ... ▪ fx09.1 – 00000000X.X – dziewięć cyfr z zerami wiodącymi i jedna cyfra po przecinku. 																												
FONT=	<p>Wielkość czcionki, Opcje: 12, 16, 20, 24, 36, 45.</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>FONT</th> <th>wysokość</th> <th>szerokość</th> <th>szerokość małych elementów (., : ;)</th> </tr> </thead> <tbody> <tr> <td>12</td> <td>12</td> <td>8</td> <td>4</td> </tr> <tr> <td>16</td> <td>16</td> <td>11</td> <td>5</td> </tr> <tr> <td>20</td> <td>20</td> <td>13</td> <td>6</td> </tr> <tr> <td>24</td> <td>24</td> <td>15</td> <td>7</td> </tr> <tr> <td>36</td> <td>36</td> <td>24</td> <td>10</td> </tr> <tr> <td>45</td> <td>45</td> <td>32</td> <td>14</td> </tr> </tbody> </table> <p>Powyższa tabela może być przydatna, jeśli chcemy na ekranie dokładnie rozmieścić kilka tekstów lub wyświetlaczy numerycznych o różnych rozmiarach czcionek i różnej długości znaków. Wszystkie litery i cyfry mają taką samą szerokość z wyjątkiem kilku znaków interpunkcyjnych: . , ; ; .itp.</p>	FONT	wysokość	szerokość	szerokość małych elementów (., : ;)	12	12	8	4	16	16	11	5	20	20	13	6	24	24	15	7	36	36	24	10	45	45	32	14
FONT	wysokość	szerokość	szerokość małych elementów (., : ;)																										
12	12	8	4																										
16	16	11	5																										
20	20	13	6																										
24	24	15	7																										
36	36	24	10																										
45	45	32	14																										

KOLOR=	Kolor wyświetlanej liczby w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.
AKCJA=	Rodzaj wyświetlacza. Opcje: <ul style="list-style-type: none"> ▪ edit – liczba edytowalna, ▪ brak – tylko wyświetlanie.
STER=	Numer sterownika, którego dotyczy dana zmienna VAR16. W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika. <i>Przykład:</i> <i>STER=0 oznacza, że zmienna dotyczy sterownika opisanego w pliku ster0.txt.</i>
ADR=	Adres 16-bitowego rejestru (holding register) dostępnego do odczytu/zapisu z wykorzystaniem protokołu Modbus RTU lub Modbus TCP. Adres musi być z zakresu zadeklarowanego w pliku sterX.txt gdzie X oznacza numer sterownika (SL_DADR do SL_DADR+SL_DSIZE-1). W tym parametrze jako adres rejestru zamiast wartości liczbowej można podać nazwę rejestru. W tym celu należy najpierw przygotować plik (rejstry.txt) z taką listą nazw rejestrów.
ENABLE=	Warunek wyświetlania danego obiektu. Jeśli chcemy żeby dany obiekt był wyświetlany zawsze to możemy całkowicie pominąć linijkę z tym warunkiem ENABLE. Jeśli chcemy, aby dany obiekt był wyświetlany w zależności od jakiegoś warunku to podajemy go w tej linijce po słowie ENABLE. Składnia ENABLE jest następująca: $\text{ENABLE}=\text{STER},\text{ADR}.\text{WARUNEK}.\text{WARTOŚĆ}$ <p>STER - numer sterownika 0..7 oraz 8 (panel HMI), ADR - adres rejestru lub flagi w danym sterowniku, argument 1 warunku, WARUNEK -warunek logiczny: =,>,< lub ! WARTOŚĆ - wartość stała (liczba), argument 2 warunku.</p> <i>Przykłady:</i> <i>ENABLE=0,4165=1</i> - obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 0 jest wartość równa 1. <i>ENABLE=0,4165>5</i> - obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 0 jest wartość większa niż 5. <i>ENABLE=1,4165<3</i> - obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 1 jest wartość mniejsza niż 3. <i>ENABLE=8,4165!0</i> - obiekt będzie wyświetlany, jeśli w rejestrze 4165 panelu HMI jest wartość różna od 0. <i>ENABLE=8,HMI_PASS_LEVEL!0</i> - obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp (użytkownik podał poprawnie jedno z trzech haseł projektu, zapisanych w pliku projektu jako PASS1, PASS2 lub PASS3). <i>ENABLE=8,HMI_PASS_LEVEL=3</i> - o obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp na poziomie 3 (użytkownik podał poprawnie hasło zadeklarowane w pliku projektu jako PASS3). W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejstry.txt) z taką listą nazw rejestrów.

Przykład

Pokazany poniżej obiekt <godziny> opisuje wyświetlacz 16-bitowego rejestru (TYP=VAR16) o nazwie „Godziny” (NAZWA=Godziny). Obiekt będzie wyświetlany na współrzędnych X=5; Y=60 (x=5, Y=60) podajemy współrzędne lewego górnego rogu obiektu (ROWNAJ=do lewej). Minimalna (MIN=0) i maksymalna (MAX=23) wartość, jaką możemy wprowadzić podczas edycji tego obiektu. Wyświetlana liczba będzie miała dwie cyfry w tym zero wiodące (FORMAT=fx02) i wielkość czcionki (FONT=24) w kolorze czerwonym (KOLOR=255,0,0). Obiekt ten będzie edytowalny (AKCJA=edit) i będzie można zmienić wartość w zadeklarowanym w pliku ster0.txt sterowniku (STER=0) w rejestrze o adresie (ADR=4002). Dodatkowo ten obiekt będzie wyświetlany jeśli w rejestrze o adresie 4165 sterownika 0 będzie wartość 1 (ENABLE=0,4165=1).

```
<godziny>
TYP=VAR16
NAZWA=Godziny
X=5
Y=60
ROWNAJ=do lewej
MIN=0
MAX=23
FORMAT=fx02
FONT=24
KOLOR=255,0,0
AKCJA=edit
STER=0
ADR=4002
ENABLE=0,4165=1
</>
```

Obiekt typu TXT – wyświetlacz tekstu

Obiekt typu TXT wyświetla zwykły tekst w dowolnym miejscu na ekranie.

zmienna	Opis																								
TYP=	Typ danego obiektu.																								
TXT=	Tekst do wyświetlenia. Maksymalna długość to 32 znaki. HMI nie obsługuje polskich znaków. Aby wyświetlić znak stopni Celsjusza '°' trzeba użyć znaku kratki (hash) '#'. <i>Przykład:</i> <i>Jeśli w tej linijce wpisujemy TXT=#C to na ekranie wyświetli się: °C.</i>																								
X=	Współrzędna x pozycji wyświetlanego obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.																								
Y=	Współrzędna y pozycji wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy górnej krawędzi ekranu.																								
ROWNAJ=	Parametr określa, od której strony ma być narysowany obiekt. Opcje: <ul style="list-style-type: none"> ▪ do lewej – obiekt będzie narysowany od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu, ▪ centralnie – obiekt będzie narysowany tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu, ▪ do prawej – obiekt będzie narysowany od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu. 																								
FONT=	<p>Wielkość czcionki, Opcje: 12, 16, 20, 24, 36.</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>FONT</th> <th>wysokość</th> <th>szerokość</th> <th>szerokość małych elementów (. , : ;)</th> </tr> </thead> <tbody> <tr> <td>12</td> <td>12</td> <td>8</td> <td>4</td> </tr> <tr> <td>16</td> <td>16</td> <td>11</td> <td>5</td> </tr> <tr> <td>20</td> <td>20</td> <td>13</td> <td>6</td> </tr> <tr> <td>24</td> <td>24</td> <td>15</td> <td>7</td> </tr> <tr> <td>36</td> <td>36</td> <td>24</td> <td>10</td> </tr> </tbody> </table> <p>Powyższa tabela może być przydatna, jeśli chcemy na ekranie dokładnie rozmieścić kilka tekstów lub wyświetlaczy numerycznych o różnych rozmiarach czcionek i różnej długości znaków. Wszystkie litery i cyfry mają taką samą szerokość z wyjątkiem kilku znaków interpunkcyjnych: . , : ; itp.</p>	FONT	wysokość	szerokość	szerokość małych elementów (. , : ;)	12	12	8	4	16	16	11	5	20	20	13	6	24	24	15	7	36	36	24	10
FONT	wysokość	szerokość	szerokość małych elementów (. , : ;)																						
12	12	8	4																						
16	16	11	5																						
20	20	13	6																						
24	24	15	7																						
36	36	24	10																						
KOLOR=	Kolor wyświetlanego tekstu w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.																								
STER=	Numer sterownika, którego dotyczy dany obiekt. W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika. Numer sterownika jest potrzebny tylko, jeśli korzystamy z warunkowego wyświetlania podanego w następnym linijce – ENABLE. <i>Przykład:</i> <i>STER=0 oznacza, że obiekt dotyczy sterownika opisanego w pliku ster0.txt.</i>																								
ENABLE=	Warunek wyświetlania danego obiektu. Jeśli chcemy żeby dany obiekt był wyświetlany zawsze to możemy całkowicie pominąć linijkę z tym warunkiem ENABLE. Jeśli chcemy, aby dany obiekt był wyświetlany w zależności od jakiegoś warunku to podajemy go w tej linijce po słowie ENABLE. <i>Przykłady:</i> <i>ENABLE=0,4165=1 - obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 0 jest wartość równa 1.</i> <i>ENABLE=0,4165>5 - obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 0 jest wartość większa niż 5.</i> <i>ENABLE=0,4165<3 - obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 0 jest wartość mniejsza niż 3.</i>																								

<p><i>ENABLE=8,HMI_PASS_LEVEL!0 - obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp (użytkownik podał poprawnie jedno z trzech haseł projektu).</i></p> <p>W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.</p>

Przykład

Pokazany poniżej obiekt <tekst1> opisuje wyświetlacz tekstu (TYP=TXT) z tekstem „°C” (TXT=#C). Obiekt będzie wyświetlany na współrzędnych X=240; Y=200, podajemy współrzędne lewego górnego rogu obiektu (ROWNAJ=do lewej). Wyświetlany tekst będzie miał wielkość czcionki 36 (FONT=36) w kolorze niebieskim (KOLOR=0,0,255). Tekst ten będzie wyświetlany pod warunkiem, że w zadeklarowanym w pliku sterowniku (STER=0) w rejestrze o adresie 4165 będzie wartość równa 1 (ENABLE=0,4165=1).

<pre><tekst1> TYP=TXT TXT=#C X=240 Y=200 ROWNAJ=do lewej FONT=36 KOLOR=0,0,255 STER=0 ENABLE=0,4165=1 </></pre>

Obiekt typu AKCJA_EKRAN i AKCJA_NC_EKRAN – akcja zmiany ekranu

Obiekt AKCJA_EKRAN jest niewidocznym na ekranie prostokątem, który umożliwia przejście do innego ekranu.

zmienna	opis
TYP=	Typ danego obiektu. <ul style="list-style-type: none"> ▪ AKCJA_EKRAN – przy zmianie ekranu nastąpi wyczyszczenie całego ekranu na białą. ▪ AKCJA_NC_EKRAN – przy zmianie ekranu nie będzie czyszczony ekran na białą.
X=	Współrzędna x pozycji obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.
Y=	Współrzędna y pozycji obiektu. 0 oznacza, że obiekt będzie umieszczony maksymalnie przy górnej krawędzi ekranu.
W=	Szerokość obiektu.
H=	Wysokość obiektu.
ROWNAJ=	Parametr określa, od której strony ma być interpretowana pozycja obiektu. Opcje: <ul style="list-style-type: none"> ▪ do lewej – obiekt będzie umieszczony od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu, ▪ centralnie – obiekt będzie umieszczony tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu, ▪ do prawej – obiekt będzie umieszczony od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu.
EKRAN=	Parametr określa numer ekranu, do którego ma skoczyć dany obiekt po jego dotknięciu.
STER=	Numer sterownika, którego dotyczy dany obiekt. W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika. Numer sterownika jest potrzebny tylko, jeśli korzystamy z warunkowego wyświetlania podanego w następniej linijce – ENABLE. <i>Przykład:</i> <i>STER=0 oznacza, że obiekt dotyczy sterownika opisanego w pliku ster0.txt.</i>
ENABLE=	Warunek wyświetlania danego obiektu. Jeśli chcemy żeby dany obiekt był wyświetlany zawsze to możemy całkowicie pominąć linijkę z tym warunkiem ENABLE. Jeśli chcemy, aby dany obiekt był wyświetlany w zależności od jakiegoś warunku to podajemy go w tej linijce po słowie ENABLE. <i>Przykłady:</i> <i>ENABLE=0,4165=1 - obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 0 jest wartość równa 1.</i> <i>ENABLE=0,4165>5 - obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownik 0 jest wartość większa niż 5.</i> <i>ENABLE=0,4165<3 - obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 0 jest wartość mniejsza niż 3.</i> <i>ENABLE=8,HMI_PASS_LEVEL!0 - obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp (użytkownik podał poprawnie jedno z trzech haseł projektu).</i> W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejstry.txt) z taką listą nazw rejestrów.

Przykład

Pokazany poniżej obiekt <akcja1> opisuje niewidzialny przycisk, który umożliwia przejście do innego ekranu (TYP=AKCJA_EKRAN) o numerze trzecim (EKTRAN=3). Obiekt ten jest nie widocznym prostokątem o wymiarach 100 na 50 pikseli (W=100, H=50) umieszczonym centralnie (ROWNAJ=centralnie) na współrzędnych x:50 i y:80 (X=50, Y=80). Dodatkowo obiekt będzie aktywny pod warunkiem, że w zadeklarowanym w pliku sterowniku (STER=0) w rejestrze o adresie 4165 będzie wartość równa 1 (ENABLE=0,4165=1).

```
<akcja1>
TYP=AKCJA_EKRAN
X=50
Y=80
W=100
H=50
ROWNAJ=centralnie
EKTRAN=3
STER=0
ENABLE=0,4165=1
</>
```

Obiekt typu AKCJA_DEC – akcja dekrementacji

Obiekt AKCJA_DEC umożliwia odjęcie 1 od wartości wybranego rejestru. Obiekt jest niewidocznym na ekranie prostokątem.

zmienna	opis
TYP=	Typ danego obiektu.
X=	Współrzędna x pozycji obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.
Y=	Współrzędna y pozycji obiektu. 0 oznacza, że obiekt będzie umieszczony maksymalnie przy górnej krawędzi ekranu.
W=	Szerokość obiektu.
H=	Wysokość obiektu.
ROWNAJ=	<p>Parametr określa, od której strony ma być interpretowana pozycja obiektu. Opcje:</p> <ul style="list-style-type: none"> ▪ do lewej – obiekt będzie umieszczony od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu, ▪ centralnie – obiekt będzie umieszczony tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu, ▪ do prawej – obiekt będzie umieszczony od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu.
MIN=	Minimalna wartość, do której można obniżyć dany rejestr za pomocą tej akcji dekrementacji.
STER=	<p>Numer sterownika, którego dotyczy dany obiekt. W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika. <i>Przykład:</i> <i>STER=0 oznacza, że obiekt dotyczy sterownika opisanego w pliku ster0.txt.</i></p>
ADR=	<p>Adres 16-bitowego rejestru (holding register) dostępnego do odczytu/zapisu z wykorzystaniem protokołu Modbus RTU lub Modbus TCP. Adres musi być z zakresu zadeklarowanego w pliku sterX.txt gdzie X oznacza numer sterownika (SL_DADR do SL_DADR+SL_DSIZE-1). W tym parametrze jako adres rejestru zamiast wartości liczbowej można podać nazwę rejestru. W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.</p>
ENABLE=	<p>Warunek aktywności danego obiektu. Jeśli chcemy żeby dany obiekt był aktywny zawsze to możemy całkowicie pominąć linijkę z tym warunkiem ENABLE. Jeśli chcemy, aby dany obiekt był aktywny w zależności od jakiegoś warunku to podajemy go w tej linijce po słowie ENABLE. <i>Przykłady:</i> <i>ENABLE=0,4165=1</i> – obiekt będzie aktywny, jeśli w rejestrze 4165 sterownika 0 jest wartość równa 1. <i>ENABLE=0,4165>5</i> – obiekt będzie aktywny, jeśli w rejestrze 4165 sterownika 0 jest wartość większa niż 5. <i>ENABLE=0,4165<3</i> – obiekt będzie aktywny, jeśli w rejestrze 4165 sterownika 0 jest wartość mniejsza niż 3. <i>ENABLE=8,HMI_PASS_LEVEL!0</i> – obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp (użytkownik podał poprawnie jedno z trzech haseł projektu). W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.</p>

Przykład

Pokazany poniżej obiekt <akcja2> opisuje niewidzialny przycisk, który umożliwia dekrementację rejestru (TYP=AKCJA_DEC) czyli odjęcie 1 od wartości w rejestrze 4103 (ADR=4103) w sterowniku opisanym w pliku ster0.txt (STER=0). Minimalna wartość, do której możemy dekrementować to 0 (MIN=0). Obiekt ten jest nie widocznym prostokątem o wymiarach 100 na 50 pikseli (W=100, H=50) umieszczonym na współrzędnych x:50 i y:200 (X=50, Y=200) gdzie te współrzędne określają lewy górny róg prostokąta (ROWNAJ=do lewej).

```
<akcja2>
TYP=AKCJA_DEC
X=50
Y=200
W=100
H=50
ROWNAJ=do lewej
MIN=0
STER=0
ADR=4103
</>
```

Obiekt typu AKCJA_INC – akcja inkrementacji

Obiekt AKCJA_INC umożliwia dodanie 1 do wartości wybranego rejestru. Obiekt jest niewidocznym na ekranie prostokątem.

zmienna	opis
TYP=	Typ danego obiektu.
X=	Współrzędna x pozycji obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.
Y=	Współrzędna y pozycji obiektu. 0 oznacza, że obiekt będzie umieszczony maksymalnie przy górnej krawędzi ekranu.
W=	Szerokość obiektu.
H=	Wysokość obiektu.
ROWNAJ=	<p>Parametr określa, od której strony ma być interpretowana pozycja obiektu. Opcje:</p> <ul style="list-style-type: none"> ▪ do lewej – obiekt będzie umieszczony od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu, ▪ centralnie – obiekt będzie umieszczony tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu, ▪ do prawej – obiekt będzie umieszczony od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu.
MAX=	Maksymalna wartość, do której można zwiększyć dany rejestr za pomocą tej akcji inkrementacji.
STER=	<p>Numer sterownika, którego dotyczy dany obiekt. W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika. <i>Przykład:</i> <i>STER=0 oznacza, że obiekt dotyczy sterownika opisanego w pliku ster0.txt.</i></p>
ADR=	<p>Adres 16-bitowego rejestru (holding register) dostępnego do odczytu/zapisu z wykorzystaniem protokołu Modbus RTU lub Modbus TCP. Adres musi być z zakresu zadeklarowanego w pliku sterX.txt gdzie X oznacza numer sterownika (SL_DADR do SL_DADR+SL_DSIZE-1). W tym parametrze jako adres rejestru zamiast wartości liczbowej można podać nazwę rejestru. W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.</p>
ENABLE=	<p>Warunek aktywności danego obiektu. Jeśli chcemy żeby dany obiekt był aktywny zawsze to możemy całkowicie pominąć linijkę z tym warunkiem ENABLE. Jeśli chcemy, aby dany obiekt był aktywny w zależności od jakiegoś warunku to podajemy go w tej linijce po słowie ENABLE. <i>Przykłady:</i> <i>ENABLE=0,4165=1</i> – obiekt będzie aktywny, jeśli w rejestrze 4165 sterownika 0 jest wartość równa 1. <i>ENABLE=0,4165>5</i> – obiekt będzie aktywny, jeśli w rejestrze 4165 sterownika 0 jest wartość większa niż 5. <i>ENABLE=0,4165<3</i> – obiekt będzie aktywny, jeśli w rejestrze 4165 sterownika 0 jest wartość mniejsza niż 3. <i>ENABLE=8,HMI_PASS_LEVEL!0</i> – obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp (użytkownik podał poprawnie jedno z trzech haseł projektu). W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.</p>

Przykład

Pokazany poniżej obiekt <akcja3> opisuje niewidzialny przycisk, który umożliwia inkrementację rejestru (TYP=AKCJA_INC) czyli dodanie 1 do wartości w rejestrze 4103 (ADR=4103) w sterowniku opisanym w piku ster0.txt (STER=0). Maksymalna wartość, do której możemy inkrementować do 15 (MAX=15). Obiekt ten jest nie widocznym prostokątem o wymiarach 100 na 100 pikseli (W=100, H=100) umieszczonym na współrzędnych x:300 i y:200 (X=300, Y=200) gdzie te współrzędne określają prawy górny róg prostokąta (ROWNAJ=do prawej).

```
<akcja3>
TYP=AKCJA_INC
X=300
Y=200
W=100
H=100
ROWNAJ=do prawej
MAX=15
STER=0
ADR=4103
</>
```

Obiekt typu AKCJA_SET – akcja ustawienia zmiennej

Obiekt AKCJA_SET umożliwia ustawienie wartości wybranego rejestru. Obiekt jest niewidocznym na ekranie prostokątem.

zmienna	opis
TYP=	Typ danego obiektu.
X=	Współrzędna x pozycji obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.
Y=	Współrzędna y pozycji obiektu. 0 oznacza, że obiekt będzie umieszczony maksymalnie przy górnej krawędzi ekranu.
W=	Szerokość obiektu.
H=	Wysokość obiektu.
ROWNAJ=	Parametr określa, od której strony ma być interpretowana pozycja obiektu. Opcje: <ul style="list-style-type: none">▪ do lewej – obiekt będzie umieszczony od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu,▪ centralnie – obiekt będzie umieszczony tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu,▪ do prawej – obiekt będzie umieszczony od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu.
SET=	Wartość, która będzie wpisana do danego rejestru po dotknięciu tego obiektu.
STER=	Numer sterownika, którego dotyczy dany obiekt. W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika. <i>Przykład:</i> <i>STER=0 oznacza, że obiekt dotyczy sterownika opisanego w pliku ster0.txt.</i>
ADR=	Adres 16-bitowego rejestru (holding register) dostępnego do zapisu z wykorzystaniem protokołu Modbus RTU lub Modbus TCP. Adres musi być z zakresu zadeklarowanego w pliku sterX.txt gdzie X oznacza numer sterownika (SL_DADR do SL_DADR+SL_DSIZE-1). W tym parametrze jako adres rejestru zamiast wartości liczbowej można podać nazwę rejestru. W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.
ENABLE=	Warunek aktywności danego obiektu. Jeśli chcemy żeby dany obiekt był aktywny zawsze to możemy całkowicie pominąć linijkę z tym warunkiem ENABLE. Jeśli chcemy, aby dany obiekt był aktywny w zależności od jakiegoś warunku to podajemy go w tej linijce po słowie ENABLE. <i>Przykłady:</i> <i>ENABLE=0,4165!1 - obiekt będzie aktywny, jeśli w rejestrze 4165 sterownika 0 jest wartość różna od 1.</i> <i>ENABLE=8,HMI_PASS_LEVEL!0 - obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp (użytkownik podał poprawnie jedno z trzech haseł projektu).</i> <i>ENABLE=8,HMI_PASS_LEVEL=1 - obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp do poziomu 1 (użytkownik podał poprawnie hasło projektu PASS1).</i> <i>ENABLE=8,HMI_PASS_LEVEL=2 - obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp do poziomu 2 (użytkownik podał poprawnie hasło projektu PASS2).</i> W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.

Przykład

Pokazany poniżej obiekt <akcja4> opisuje niewidzialny przycisk, który umożliwia wpisanie (TYP=AKCJA_SET) wartości 8 (SET=8) do rejestru o adresie (ADR=4103) w sterowniku opisanym w piku ster0.txt (STER=0). Obiekt ten jest nie widocznym prostokątem o wymiarach 100 na 50 pikseli (W=100, H=50) umieszczonym na współrzędnych x:175 i y:250 (X=175, Y=250) gdzie te współrzędne określają lewy górny róg prostokąta (ROWNAJ=do lewej).

```
<akcja4>
TYP=AKCJA_SET
X=175
Y=250
W=100
H=50
ROWNAJ=do lewej
SET=8
STER=0
ADR=4002
</>
```

Obiekt typu AKCJA_FLAG_SET – akcja ustawienia flagi

Obiekt AKCJA_FLAG_SET umożliwia ustawienie stanu wybranej flagi. Obiekt jest niewidocznym na ekranie prostokątem.

zmienna	opis
TYP=	Typ danego obiektu.
X=	Współrzędna x pozycji obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.
Y=	Współrzędna y pozycji obiektu. 0 oznacza, że obiekt będzie umieszczony maksymalnie przy górnej krawędzi ekranu.
W=	Szerokość obiektu.
H=	Wysokość obiektu.
ROWNAJ=	<p>Parametr określa, od której strony ma być interpretowana pozycja obiektu. Opcje:</p> <ul style="list-style-type: none"> ▪ do lewej – obiekt będzie umieszczony od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu, ▪ centralnie – obiekt będzie umieszczony tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu, ▪ do prawej – obiekt będzie umieszczony od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu.
SET=	Wartość '0' lub '1', która będzie wpisana do danej flagi po dotknięciu tego obiektu.
STER=	<p>Numer sterownika, którego dotyczy dany obiekt. W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika. <i>Przykład:</i> <i>STER=0 oznacza, że obiekt dotyczy sterownika opisanego w pliku ster0.txt.</i></p>
ADR=	<p>Adres flagi (coil) dostępnej do zapisu z wykorzystaniem protokołu Modbus RTU lub Modbus TCP. Adres musi być z zakresu zadeklarowanego w pliku sterX.txt gdzie X oznacza numer sterownika (SL_FADR do SL_FADR+SL_FSIZE-1). W tym parametrze jako adres flagi zamiast wartości liczbowej można podać nazwę flagi. W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.</p>
ENABLE=	<p>Warunek aktywności danego obiektu. Jeśli chcemy żeby dany obiekt był aktywny zawsze to możemy całkowicie pominąć linijkę z tym warunkiem ENABLE. Jeśli chcemy, aby dany obiekt był aktywny w zależności od jakiegoś warunku to podajemy go w tej liniжке po słowie ENABLE. <i>Przykłady:</i> <i>ENABLE=0,7165=0</i> - obiekt będzie aktywny, jeśli w sterowniku 0 flaga o adresie 7165 ma wartość 0. <i>ENABLE=0,7165!0</i> - obiekt będzie aktywny, jeśli w sterowniku 0 flaga o adresie 7165 ma wartość różna od 0 (równą 1). <i>ENABLE=8,HMI_PASS_LEVEL=1</i> - obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp do poziomu 1 (użytkownik podał poprawnie hasło projektu PASS1). W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.</p>

Przykład

Pokazany poniżej obiekt <akcja5> opisuje niewidzialny przycisk, który umożliwia ustawienie flagi (TYP=AKCJA_FLAG_SET). Do flagi o adresie (ADR=7083) w sterowniku opisanym w pliku ster0.txt (STER=0) zostanie wpisana wartość 1 (SET=1). Obiekt ten jest nie widocznym prostokątem o wymiarach 100 na 50 pikseli (W=100, H=50) umieszczonym na współrzędnych x:300 i y:80 (X=300, Y=80) gdzie te współrzędne określają lewy górny róg prostokąta (ROWNAJ=do lewej).

```
<akcja5>
TYP=AKCJA_FLAG_SET
X=300
Y=80
W=100
H=50
ROWNAJ=do lewej
SET=1
STER=0
ADR=7083
</>
```

Obiekt typu KLAWSZ_EKRAN i KLAWSZ_NC_EKRAN – przycisk zmiany ekranu

Obiekt KLAWSZ_EKRAN jest przyciskiem, który umożliwia przejście do innego ekranu.

zmienna	Opis
TYP=	Typ danego obiektu. Typ danego obiektu. <ul style="list-style-type: none"> ▪ KLAWSZ_EKRAN – przy zmianie ekranu nastąpi wyczyszczenie całego ekranu na białą, ▪ KLAWSZ_NC_EKRAN – przy zmianie ekranu nie będzie czyszczony ekran na białą.
NAZWA=	Wyświetlana nazwa dla danego przycisku. Maksymalna długość nazwy to 16 znaków. HMI nie obsługuje polskich znaków.
X=	Współrzędna x pozycji wyświetlanego obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.
Y=	Współrzędna y pozycji wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy górnej krawędzi ekranu.
W=	Szerokość obiektu.
H=	Wysokość obiektu.
ROWNAJ=	Parametr określa, od której strony ma być narysowany obiekt. Opcje: <ul style="list-style-type: none"> ▪ do lewej – obiekt będzie narysowany od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu, ▪ centralnie – obiekt będzie narysowany tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu, ▪ do prawej – obiekt będzie narysowany od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu.
KOLOR=	Kolor przycisku w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.
KOLOR_ON=	Kolor dotkniętego przycisku w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.
EKRAN=	Parametr określa numer ekranu, do którego ma skoczyć dany obiekt po jego dotknięciu.
STER=	Numer sterownika, którego dotyczy dany obiekt. W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika. Numer sterownika jest potrzebny tylko, jeśli korzystamy z warunkowego wyświetlania podanego w następnym wierszu – ENABLE. <i>Przykład:</i> <i>STER=0 oznacza, że obiekt dotyczy sterownika opisanego w pliku ster0.txt.</i>
ENABLE=	Warunek wyświetlania danego obiektu. Jeśli chcemy żeby dany obiekt był wyświetlany zawsze to możemy całkowicie pominąć linię z tym warunkiem ENABLE. Jeśli chcemy, aby dany obiekt był wyświetlany w zależności od jakiegoś warunku to podajemy go w tej linii po słowie ENABLE. <i>Przykłady:</i> <i>ENABLE=0,4165=1 - obiekt będzie wyświetlany, jeśli w sterowniku 0 rejestr o adresie 4165 ma wartość równą 1.</i> <i>ENABLE=8,HMI_PASS_LEVEL=2 - obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp do poziomu 2 (użytkownik podał poprawnie hasło projektu PASS2).</i> W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestr.txt) z taką listą nazw rejestrów.

Przykład

Pokazany poniżej obiekt <klaw1> opisuje przycisk (klawisz), który umożliwia przejście do innego ekranu (TYP=KLAWISZ_EKRAN) o numerze drugim (EKRAN=2). Obiekt ten jest widocznym prostokątem z zaokrąglonymi rogami i napisem „Ekran2” (NAZWA=Ekran2) w kolorze zielonym (KOLOR=0,255,0) i wymiarach 100 na 50 pikseli (W=100, H=50) umieszczonym na współrzędnych x:50 i y:80 (X=50, Y=80) gdzie są to współrzędne lewego górnego rogu (ROWNAJ=do lewej). Po wciśnięciu przycisku zmieni on kolor na ciemno szary (KOLOR_ON=25,25,25). Dodatkowo obiekt będzie aktywny pod warunkiem, że w panelu odblokowano dostęp na poziomie 2 (ENABLE=8,HMI_PASS_LEVEL=2).

```
<klaw1>
TYP=KLAWISZ_EKRAN
NAZWA=Ekran2
X=50
Y=80
W=100
H=50
ROWNAJ=do lewej
KOLOR=0,255,0
KOLOR_ON=25,25,25
EKRAN=2
STER=0
ENABLE=8,HMI_PASS_LEVEL=2
</>
```

Obiekt typu KLAWSZ_DEC – przycisk dekrementacji

Obiekt KLAWSZ_DEC jest przyciskiem, który odejmuje 1 od zadeklarowanej zmiennej.

zmienna	Opis
TYP=	Typ danego obiektu.
NAZWA=	Wyświetlana nazwa dla danego przycisku. Maksymalna długość nazwy to 16 znaków. HMI nie obsługuje polskich znaków.
X=	Współrzędna x pozycji wyświetlanego obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.
Y=	Współrzędna y pozycji wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy górnej krawędzi ekranu.
W=	Szerokość obiektu.
H=	Wysokość obiektu.
ROWNAJ=	Parametr określa, od której strony ma być narysowany obiekt. Opcje: <ul style="list-style-type: none"> ▪ do lewej – obiekt będzie narysowany od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu, ▪ centralnie – obiekt będzie narysowany tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu, ▪ do prawej – obiekt będzie narysowany od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu.
KOLOR=	Kolor przycisku w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.
KOLOR_ON=	Kolor dotkniętego przycisku w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.
MIN=	Minimalna wartość, do której można obniżyć dany rejestr za pomocą tej akcji dekrementacji.
STER=	Numer sterownika, którego dotyczy dany obiekt. W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika. <i>Przykład:</i> <i>STER=0 oznacza, że obiekt dotyczy sterownika opisanego w pliku ster0.txt.</i>
ADR=	Adres 16-bitowego rejestru (holding register) dostępnego do odczytu/zapisu z wykorzystaniem protokołu Modbus RTU lub Modbus TCP. Adres musi być z zakresu zadeklarowanego w pliku sterX.txt gdzie X oznacza numer sterownika (SL_DADR do SL_DADR+SL_DSIZE-1). W tym parametrze jako adres rejestru zamiast wartości liczbowej można podać nazwę rejestru. W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.
ENABLE=	Warunek wyświetlania danego obiektu. Jeśli chcemy żeby dany obiekt był wyświetlany zawsze to możemy całkowicie pominąć liniijkę z tym warunkiem ENABLE. Jeśli chcemy, aby dany obiekt był wyświetlany w zależności od jakiegoś warunku to podajemy go w tej liniijkce po słowie ENABLE. <i>Przykłady:</i> <i>ENABLE=4165=1 - obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 0 jest wartość równa 1.</i> <i>ENABLE=8,HMI_PASS_LEVEL=3 - obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp do poziomu 3 (użytkownik podał poprawnie hasło projektu PASS3).</i> W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.

Przykład

Pokazany poniżej obiekt <klaw2> opisuje przycisk (klawisz), który umożliwia dekrementację rejestru (TYP=KLAW_DEC), czyli odjęcie 1 od wartości w rejestrze 4103 (ADR=4103) w sterowniku opisanym w pikie ster0.txt (STER=0). Minimalna wartość, do której możemy dekrementować to 0 (MIN=0). Obiekt ten jest widocznym prostokątem z zaokrąglonymi rogami i napisem „Dec” (NAZWA=Dec) w kolorze niebieskim (KOLOR=0,0,255) i wymiarach 100 na 100 pikseli (W=100, H=100) umieszczonym na współrzędnych x:50 i y:200 (X=50, Y=200) gdzie są to współrzędne lewego górnego rogu (ROWNAJ=do lewej). Po wciśnięciu przycisku zmieni on kolor na ciemno szary (KOLOR_ON=25,25,25).

```
<klaw2>
TYP=KLAWISZ_DEC
NAZWA=Dec
X=50
Y=200
W=100
H=100
ROWNAJ=do lewej
KOLOR=0,0,255
KOLOR_ON=25,25,25
MIN=0
STER=0
ADR=4103
</>
```

Obiekt typu KLAWISZ_INC – przycisk inkrementacji

Obiekt KLAWISZ_INC jest przyciskiem, który dodaje 1 do zadeklarowanej zmiennej.

zmienna	Opis
TYP=	Typ danego obiektu.
NAZWA=	Wyświetlana nazwa dla danego przycisku. Maksymalna długość nazwy to 16 znaków. HMI nie obsługuje polskich znaków.
X=	Współrzędna x pozycji wyświetlanego obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.
Y=	Współrzędna y pozycji wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy górnej krawędzi ekranu.
W=	Szerokość obiektu.
H=	Wysokość obiektu.
ROWNAJ=	Parametr określa, od której strony ma być narysowany obiekt. Opcje: <ul style="list-style-type: none"> ▪ do lewej – obiekt będzie narysowany od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu, ▪ centralnie – obiekt będzie narysowany tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu, ▪ do prawej – obiekt będzie narysowany od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu.
KOLOR=	Kolor przycisku w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.
KOLOR_ON=	Kolor dotkniętego przycisku w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.
MAX=	Maksymalna wartość, do której można zwiększyć dany rejestr za pomocą tej akcji inkrementacji.
STER=	Numer sterownika, którego dotyczy dany obiekt. W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika. <i>Przykład:</i> <i>STER=0</i> oznacza, że obiekt dotyczy sterownika opisanego w pliku <i>ster0.txt</i> .
ADR=	Adres 16-bitowego rejestru (holding register) dostępnego do odczytu/zapisu z wykorzystaniem protokołu Modbus RTU lub Modbus TCP. Adres musi być z zakresu zadeklarowanego w pliku <i>sterX.txt</i> gdzie X oznacza numer sterownika (SL_DADR do SL_DADR+SL_DSIZE-1). W tym parametrze jako adres rejestru zamiast wartości liczbowej można podać nazwę rejestru. W tym celu należy najpierw przygotować plik (<i>rejestry.txt</i>) z taką listą nazw rejestrów.
ENABLE=	Warunek wyświetlania danego obiektu. Jeśli chcemy żeby dany obiekt był wyświetlany zawsze to możemy całkowicie pominąć liniijkę z tym warunkiem ENABLE. Jeśli chcemy, aby dany obiekt był wyświetlany w zależności od jakiegoś warunku to podajemy go w tej liniijke po słowie ENABLE. <i>Przykłady:</i> <i>ENABLE=0,4165=1</i> – obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 0 jest wartość równa 1. <i>ENABLE=8,HMI_PASS_LEVEL=2</i> – obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp do poziomu 2 (użytkownik podał poprawnie hasło projektu PASS2). W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (<i>rejestry.txt</i>) z taką listą nazw rejestrów.

Przykład

Pokazany poniżej obiekt <klaw3> opisuje przycisk (klawisz), który umożliwia inkrementację rejestru (TYP=KLAWISZ_INC), czyli dodanie 1 do wartości w rejestrze 4103 (ADR=4103) w sterowniku opisanym w piku ster0.txt (STER=0). Maksymalna wartość, do której możemy inkrementować to 15 (MAX=15). Obiekt ten jest widocznym prostokątem z zaokrąglonymi rogami i napisem „Inc” (NAZWA=Inc) w kolorze czerwonym (KOLOR=255,0,0) i wymiarach 100 na 100 pikseli (W=100, H=100) umieszczonym na współrzędnych x:300 i y:200 (X=300, Y=200) gdzie są to współrzędne lewego górnego rogu (ROWNAJ=do lewej). Po wciśnięciu przycisku zmieni on kolor na ciemno szary (KOLOR_ON=25,25,25).

```
<klaw3>
TYP=KLAWISZ_INC
NAZWA=Inc
X=300
Y=200
W=100
H=100
ROWNAJ=do lewej
KOLOR=250,25,25
KOLOR_ON=25,25,25
MAX=15
STER=0
ADR=4103
</>
```

Obiekt typu KLAWISZ_SET – przycisk ustawienia zmiennej

Obiekt KLAWISZ_SET jest przyciskiem, który ustawia wartość zadeklarowanej zmiennej.

zmienna	opis
TYP=	Typ danego obiektu.
NAZWA=	Wyświetlana nazwa dla danego przycisku. Maksymalna długość nazwy to 16 znaków. HMI nie obsługuje polskich znaków.
X=	Współrzędna x pozycji wyświetlanego obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.
Y=	Współrzędna y pozycji wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy górnej krawędzi ekranu.
W=	Szerokość obiektu.
H=	Wysokość obiektu.
ROWNAJ=	Parametr określa, od której strony ma być narysowany obiekt. Opcje: <ul style="list-style-type: none">▪ do lewej – obiekt będzie narysowany od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu,▪ centralnie – obiekt będzie narysowany tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu,▪ do prawej – obiekt będzie narysowany od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu.
KOLOR=	Kolor przycisku w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.
KOLOR_ON=	Kolor dotkniętego przycisku w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.
SET=	Wartość, która będzie wpisana do danego rejestru po dotknięciu tego przycisku.
STER=	Numer sterownika, którego dotyczy dany obiekt. W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika. <i>Przykład:</i> <i>STER=0 oznacza, że obiekt dotyczy sterownika opisanego w pliku ster0.txt.</i>
ADR=	Adres 16-bitowego rejestru (holding register) dostępnego do zapisu z wykorzystaniem protokołu Modbus RTU lub Modbus TCP. Adres musi być z zakresu zadeklarowanego w pliku sterX.txt gdzie X oznacza numer sterownika (SL_DADR do SL_DADR+SL_DSIZE-1). W tym parametrze jako adres rejestru zamiast wartości liczbowej można podać nazwę rejestru. W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.
ENABLE=	Warunek wyświetlania danego obiektu. Jeśli chcemy żeby dany obiekt był wyświetlany zawsze to możemy całkowicie pominąć linijkę z tym warunkiem ENABLE. Jeśli chcemy, aby dany obiekt był wyświetlany w zależności od jakiegoś warunku to podajemy go w tej linijce po słowie ENABLE. <i>Przykłady:</i> <i>ENABLE=0,4165=1 - obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 0 jest wartość równa 1.</i> <i>ENABLE=8,HMI_PASS_LEVEL=1 - obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp do poziomu 1 (użytkownik podał poprawnie hasło projektu PASS1).</i> W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.

Przykład

Pokazany poniżej obiekt <klaw4> opisuje przycisk (klawisz), który umożliwia wpisanie (TYP=KLAWISZ_SET) wartości 8 (SET=8) do rejestru o adresie (ADR=4103) w sterowniku opisanym w piku ster0.txt (STER=0). Obiekt ten jest widocznym prostokątem z zaokrąglonymi rogami i napisem „Set:8” (NAZWA=Set:8) w kolorze zielonym (KOLOR=0,255,0) i wymiarach 100 na 50 pikseli (W=100, H=50) umieszczonym na współrzędnych x:175 i y:250 (X=175, Y=250) gdzie te współrzędne określają lewy górny róg prostokąta (ROWNAJ=do lewej). Po wciśnięci przycisku zmieni on kolor na ciemno szary (KOLOR_ON=25,25,25).

```
<klaw4>
TYP=KLAWISZ_SET
NAZWA=Set:8
X=175
Y=250
W=100
H=50
ROWNAJ=do lewej
KOLOR=0,255,0
KOLOR_ON=25,25,25
SET=8
STER=0
ADR=4103
</>
```

Obiekt typu KLAWISZ_FLAG_SET – przycisk ustawienia flagi

Obiekt KLAWISZ_FLAG_SET jest przyciskiem, który ustawia wartość zadeklarowanej flagi.

zmienna	opis
TYP=	Typ danego obiektu.
NAZWA=	Wyświetlana nazwa dla danego przycisku. Maksymalna długość nazwy to 16 znaków. HMI nie obsługuje polskich znaków.
X=	Współrzędna x pozycji wyświetlanego obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.
Y=	Współrzędna y pozycji wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy górnej krawędzi ekranu.
W=	Szerokość obiektu.
H=	Wysokość obiektu.
ROWNAJ=	Parametr określa, od której strony ma być narysowany obiekt. Opcje: <ul style="list-style-type: none"> ▪ do lewej – obiekt będzie narysowany od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu, ▪ centralnie – obiekt będzie narysowany tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu, ▪ do prawej – obiekt będzie narysowany od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu.
KOLOR=	Kolor przycisku w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.
KOLOR_ON=	Kolor dotkniętego przycisku w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.
SET=	Stan, który będzie wpisany do danej flagi po dotknięciu tego przycisku. Możliwe stany 0 albo 1.
STER=	Numer sterownika, którego dotyczy dany obiekt. W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika. <i>Przykład:</i> <i>STER=0</i> oznacza, że obiekt dotyczy sterownika opisanego w pliku <i>ster0.txt</i> .
ADR=	Adres flagi (coil) dostępnej do zapisu z wykorzystaniem protokołu Modbus RTU lub Modbus TCP. Adres musi być z zakresu zadeklarowanego w pliku <i>sterX.txt</i> gdzie X oznacza numer sterownika (SL_FADR do SL_FADR+SL_FSIZE-1). W tym parametrze jako adres flagi zamiast wartości liczbowej można podać nazwę flagi. W tym celu należy najpierw przygotować plik (<i>rejstry.txt</i>) z taką listą nazw rejestrów.
ENABLE=	Warunek wyświetlania danego obiektu. Jeśli chcemy żeby dany obiekt był wyświetlany zawsze to możemy całkowicie pominąć linijkę z tym warunkiem ENABLE. Jeśli chcemy, aby dany obiekt był wyświetlany w zależności od jakiegoś warunku to podajemy go w tej linijce po słowie ENABLE. <i>Przykłady:</i> <i>ENABLE=0,4165=1</i> – obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 0 jest wartość równa 1. <i>ENABLE=8,HMI_PASS_LEVEL!0</i> – obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp (użytkownik podał poprawnie hasło projektu <i>PASS1</i> , <i>PASS2</i> lub <i>PASS3</i>). W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (<i>rejstry.txt</i>) z taką listą nazw rejestrów.

Przykład

Pokazany poniżej obiekt <klaw5> opisuje przycisk (klawisz), który umożliwia ustawienie flagi (TYP=KLAWISZ_FLAG_SET). Do flagi o adresie (ADR=7083) w sterowniku opisanym w pliku ster0.txt (STER=0) zostanie wpisana wartość 1 (SET=1). Obiekt ten jest widocznym prostokątem zaokrąglonymi rogami i napisem „Serwis” (NAZWA=Serwis) w kolorze zielonym (KOLOR=0,255,0) i wymiarach 100 na 50 pikseli (W=100, H=50) umieszczonym na współrzędnych x:300 i y:80 (X=300, Y=80) gdzie te współrzędne określają lewy górny róg prostokąta (ROWNAJ=do lewej). Po wciśnięciu przycisku zmieni on kolor na ciemno szary (KOLOR_ON=25,25,25).

```
<klaw5>
TYP=KLAWISZ_FLAG_SET
NAZWA=Serwis
X=300
Y=80
W=100
H=50
ROWNAJ=do lewej
KOLOR=0,255,0
KOLOR_ON=25,25,25
SET=1
STER=0
ADR=7083
</>
```

Obiekt typu BMP – bitmapa typu RGB 24-bitowy

Obiekt typu **BMP** umożliwia wyświetlenie bitmapy na podstawie pliku graficznego 24-bitowego o rozszerzeniu .bmp.

zmienna	opis
TYP=	Typ danego obiektu.
X=	Współrzędna x pozycji wyświetlanego obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.
Y=	Współrzędna y pozycji wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy górnej krawędzi ekranu.
W=	Szerokość obiektu. Trzeba podać rzeczywisty rozmiar bitmapy. Jeśli obraz ma mieć inne wymiary trzeba go przyciąć w programie graficznym i podać tutaj nowe wymiary. Kompilator nie przycina oryginalnych rozmiarów bitmapy.
H=	Wysokość obiektu. Trzeba podać rzeczywisty rozmiar bitmapy. Jeśli obraz ma mieć inne wymiary trzeba go przyciąć w programie graficznym i podać tutaj nowe wymiary. Kompilator nie przycina oryginalnych rozmiarów bitmapy.
ROWNAJ=	<p>Parametr określa, od której strony ma być narysowany obiekt. Opcje:</p> <ul style="list-style-type: none"> ▪ do lewej – obiekt będzie narysowany od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu, ▪ centralnie – obiekt będzie narysowany tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu, ▪ do prawej – obiekt będzie narysowany od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu.
BMP=	Nazwa pliku graficznego 24-bitowego z rozszerzeniem .bmp. Plik ten musi być umieszczony w tym samym katalogu co cały projekt.
STER=	<p>Numer sterownika, którego dotyczy dany obiekt.</p> <p>W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika.</p> <p>Numer sterownika jest potrzebny tylko, jeśli korzystamy z warunkowego wyświetlania podanego w następnej linijce – ENABLE.</p> <p><i>Przykład:</i> STER=0 oznacza, że obiekt dotyczy sterownika opisanego w pliku ster0.txt.</p>
ENABLE=	<p>Warunek wyświetlania danego obiektu. Jeśli chcemy żeby dany obiekt był wyświetlany zawsze to możemy całkowicie pominąć linijkę z tym warunkiem ENABLE.</p> <p>Jeśli chcemy, aby dany obiekt był wyświetlany w zależności od jakiegoś warunku to podajemy go w tej linijce po słowie ENABLE.</p> <p><i>Przykłady:</i> ENABLE=0,4165=1 – obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 0 jest wartość równa 1. ENABLE=8,HMI_PASS_LEVEL!0 – obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp (użytkownik podał poprawnie hasło projektu PASS1, PASS2 lub PASS3).</p> <p>W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestr.txt) z taką listą nazw rejestrów.</p>

Przykład

Pokazany poniżej obiekt <obraz1> opisuje wyświetlany obrazek (TYP=BMP) na podstawie 24-bitowego pliku graficznego o nazwie „grafika24.bmp” (BMP=grafika24) o wymiarach 50 na 50 pikseli (W=50, H=50). Będzie on wyświetlany na współrzędnych x:80 i y:80 (X=80, Y=80), podane współrzędne dotyczą środka górnej krawędzi (ROWNAJ=centralnie). Obrazek ten będzie wyświetlany pod warunkiem, że w zadeklarowanym w pliku sterowniku (STER=0) w rejestrze o adresie 4103 będzie wartość większa od 5 (ENABLE=0,4103>5).

```
<obraz1>  
TYP=BMP  
X=80  
Y=80  
W=50  
H=50  
ROWNAJ=centralnie  
BMP=grafika24  
STER=0  
ENABLE=0,4103>5  
</>
```

Obiekt typu IKONA – grafika typu mono

Obiekt typu **IKONA** jest obiektem mono czyli jednokolorowym. Umożliwia wyświetlenie obrazka na podstawie pliku graficznego 24-bitowego o rozszerzeniu .bmp (bitmapy 24b). Każdy bit w kolorze innym niż biały traktowany jest jako „1” i rysowany jest w kolorze zdefiniowanym w obiekcie. Plik kolorowy w czasie kompilacji przekonwertowany będzie do postaci mono, czyli jednokolorowej ikony.

zmienna	opis
TYP=	Typ danego obiektu.
X=	Współrzędna x pozycji wyświetlanego obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.
Y=	Współrzędna y pozycji wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy górnej krawędzi ekranu.
W=	Szerokość obiektu. Trzeba podać rzeczywisty rozmiar obrazka. Jeśli obrazek ma mieć inne wymiary trzeba go przyciąć w programie graficznym i podać tutaj nowe wymiary. Kompilator nie przycina oryginalnych rozmiarów obrazu.
H=	Wysokość obiektu. Trzeba podać rzeczywisty rozmiar obrazka. Jeśli obrazek ma mieć inne wymiary trzeba go przyciąć w programie graficznym i podać tutaj nowe wymiary. Kompilator nie przycina oryginalnych rozmiarów obrazu.
ROWNAJ=	Parametr określa, od której strony ma być narysowany obiekt. Opcje: <ul style="list-style-type: none">▪ do lewej – obiekt będzie narysowany od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu,▪ centralnie – obiekt będzie narysowany tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu,▪ do prawej – obiekt będzie narysowany od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu.
KOLOR=	Kolor wyświetlanego obrazka (ikony) w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.
IKONA=	Nazwa pliku graficznego 24-bitowego z rozszerzeniem .bmp, który będzie przekonwertowany do postaci mono czyli jednokolorowej ikony.
STER=	Numer sterownika, którego dotyczy dany obiekt. W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika. Numer sterownika jest potrzebny tylko, jeśli korzystamy z warunkowego wyświetlania podanego w następnej linijce – ENABLE. <i>Przykład:</i> <i>STER=0 oznacza, że obiekt dotyczy sterownika opisanego w pliku ster0.txt.</i>
ENABLE=	Warunek wyświetlania danego obiektu. Jeśli chcemy żeby dany obiekt był wyświetlany zawsze to możemy całkowicie pominąć linijkę z tym warunkiem ENABLE. Jeśli chcemy, aby dany obiekt był wyświetlany w zależności od jakiegoś warunku to podajemy go w tej linijce po słowie ENABLE. <i>Przykłady:</i> <i>ENABLE=0,4165=1 - obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 0 jest wartość równa 1.</i> <i>ENABLE=8,HMI_PASS_LEVEL!0 - obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp (użytkownik podał poprawnie hasło projektu PASS1, PASS2 lub PASS3).</i> W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.

Przykład

Pokazany poniżej obiekt **<ikona1>** opisuje wyświetlany monochromatyczny obrazek (TYP=IKONA) na podstawie pliku „icon_dom.bmp” (IKONA=icon_dom) o wymiarach 70 na 32 pikseli (W=70, H=32). Będzie on wyświetlany na współrzędnych x:10 i y:200 (X=10, Y=200), podane współrzędne dotyczą lewego górnego narożnika (ROWNAJ=do lewej). Ikona będzie wyświetlana w niebieskim kolorze (KOLOR=0,0,255). Ikona ta będzie wyświetlana pod warunkiem, że w zadeklarowanym w pliku sterowniku (STER=0) w rejestrze o adresie 4103 będzie wartość mniejsza od 4 (ENABLE=0,4103<4).

```
<ikona1>  
TYP=IKONA  
X=10  
Y=200  
W=70  
H=32  
ROWNAJ=do lewej  
KOLOR=255,0,0  
IKONA=icon_dom  
STER=0  
ENABLE=0,4103<4  
</>
```

Obiekt typu LISTA_TXT – lista tekstów

Obiekt typu LISTA_TXT umożliwia wyświetlenie różnych tekstów w zależności od wartości danego rejestru lub flagi. Np. dzień tygodnia.

zmienna	opis																								
TYP=	Typ danego obiektu.																								
NAZWA=	Nazwa obiektu, będzie wyświetlana tylko podczas edycji tego parametru. Maksymalna długość nazwy to 16 znaków.																								
X=	Współrzędna x pozycji wyświetlanego obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.																								
Y=	Współrzędna y pozycji wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy górnej krawędzi ekranu.																								
ROWNAJ=	<p>Parametr określa, od której strony ma być narysowany obiekt. Opcje:</p> <ul style="list-style-type: none"> ▪ do lewej – obiekt będzie narysowany od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu, ▪ centralnie – obiekt będzie narysowany tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu, ▪ do prawej – obiekt będzie narysowany od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu. 																								
FONT=	<p>Wielkość czcionki, Opcje: 12, 16, 20, 24, 36.</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>FONT</th> <th>wysokość</th> <th>szerokość</th> <th>szerokość małych elementów (. , ;)</th> </tr> </thead> <tbody> <tr> <td>12</td> <td>12</td> <td>8</td> <td>4</td> </tr> <tr> <td>16</td> <td>16</td> <td>11</td> <td>5</td> </tr> <tr> <td>20</td> <td>20</td> <td>13</td> <td>6</td> </tr> <tr> <td>24</td> <td>24</td> <td>15</td> <td>7</td> </tr> <tr> <td>36</td> <td>36</td> <td>24</td> <td>10</td> </tr> </tbody> </table> <p>Powyższa tabela może być przydatna, jeśli chcemy na ekranie dokładnie rozmieścić kilka tekstów lub wyświetlaczy numerycznych o różnych rozmiarach czcionek i różnej długości znaków. Wszystkie litery i cyfry mają taką samą szerokość z wyjątkiem kilku znaków interpunkcyjnych: . , ; . itp.</p>	FONT	wysokość	szerokość	szerokość małych elementów (. , ;)	12	12	8	4	16	16	11	5	20	20	13	6	24	24	15	7	36	36	24	10
FONT	wysokość	szerokość	szerokość małych elementów (. , ;)																						
12	12	8	4																						
16	16	11	5																						
20	20	13	6																						
24	24	15	7																						
36	36	24	10																						
KOLOR=	Kolor wyświetlanej listy w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.																								
AKCJA=	<p>Rodzaj listy. Opcje:</p> <ul style="list-style-type: none"> ▪ edit – lista edytowalna, pojawia się okienko wyboru, maksymalna długość listy edytowalnej to 7 elementów (tekstów po 19 znaków), ▪ cedit – lista cyklicznie przewijana, po dotknięciu obiektu następuje zmiana na kolejny tekst z listy, maksymalna długość listy cyklicznej to 15 elementów (tekstów po 19 znaków), ▪ brak – lista tylko do wyświetlania, maksymalna długość tej listy to 15 elementów (tekstów po 19 znaków). 																								
LISTA=	Nazwa pliku z listą tekstów do wyświetlania. Wartości i teksty im przypisane zawarte są w dodatkowym pliku.																								
LISTA_EDIT=	Nazwa pliku z listą tekstów, które się pojawia podczas edycji tego parametru, czyli tylko dla list z ustawionym parametrem <i>AKCJA=edit</i> . Wartości i teksty im przypisane są w dodatkowym pliku. Parametry LISTA i LISTA_EDIT umożliwiają przygotowanie dwóch list, np. jednej wyrównanej do lewej strony, a drugiej wyrównanej do prawej. Możliwe jest też zastosowanie w jednej liście skróconych tekstów. Poniżej przykłady takich plików.																								
STER=	<p>Numer sterownika, którego dotyczy dany obiekt.</p> <p>W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika. <i>Przykład:</i> <i>STER=0</i> oznacza, że obiekt dotyczy sterownika opisanego w pliku <i>ster0.txt</i>.</p>																								

ADR=	Adres 16-bitowego rejestru (holding register) lub flagi (coil) dostępnego do odczytu/zapisu z wykorzystaniem protokołu Modbus RTU lub Modbus TCP. Adres musi być z zakresu zadeklarowanego w pliku sterX.txt gdzie X oznacza numer sterownika (SL_DADR do SL_DADR+SL_DSIZE-1) lub z zakresu (SL_FADR do SL_FADR+SL_FSIZE-1). W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.
ENABLE=	Warunek wyświetlania danego obiektu. Jeśli chcemy żeby dany obiekt był wyświetlany zawsze to możemy całkowicie pominąć linijkę z tym warunkiem ENABLE. Jeśli chcemy, aby dany obiekt był wyświetlany w zależności od jakiegoś warunku to podajemy go w tej linijce po słowie ENABLE. <i>Przykłady:</i> <i>ENABLE=4165!0</i> - obiekt będzie wyświetlany, jeśli w rejestrze 4165 jest wartość różna od 0. <i>ENABLE=8,HMI_PASS_LEVEL!0</i> - obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp (użytkownik podał poprawnie hasło projektu PASS1, PASS2 lub PASS3). W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.

Przykład

Pokazany poniżej obiekt <dzien> opisuje edytowalną (AKCJA=edit) listę dni tygodnia w postaci tekstowej (TYP=LISTA_TXT). Obiekt będzie wyświetlany na współrzędnych x:475 i y:60 (X=475, Y=60), podajemy współrzędne prawego górnego rogu obiektu (ROWNAJ=do prawej). Wyświetlany tekst będzie miał wielkość czcionki 36 (FONT=36) w kolorze niebieskim (KOLOR=0,0,255). Teksty będą wyświetlane w zależności od wartości rejestru 4006 (ADR=4006) w sterowniku zadeklarowanym w pliku ster0.txt (STER=0).

Lista wyświetlanych tekstów i wartości do nich przypisanych będzie zadeklarowana w pliku „l_dni_skrot.txt” (LISTA=l_dni_skrot), poniżej przykład takiego pliku. Podczas edycji tego parametru będzie wyświetlone dodatkowe okienko wyskakujące zawierające listę tekstów zadeklarowanych w pliku „l_dni_lewa.txt” (LISTA=l_dni_lewa), poniżej przykład takiego pliku.

```

<dzien>
TYP=LISTA_TXT
NAZWA=Dzien tygodnia
X=475
Y=60
ROWNAJ=do prawej
FONT=36
KOLOR=0,0,255
AKCJA=edit
LISTA=l_dni_skrot
LISTA_EDIT=l_dni_lewa
STER=0
ADR=4006
</>

```

W poniższym pliku w każdej linijce mamy dwa parametry oddzielone średnikiem. Pierwszy to wartość rejestru, dla którego ma być wyświetlony dany tekst, drugi to tekst o maksymalnej długości 19 znaków. Możemy zadeklarować do 15 elementów.

Plik I_dni_skrot.txt

```
0;Po
1;Wt
2;Sr
3;Cz
4;Pi
5;So
6;Ni
```

W poniższym pliku w każdej linijce mamy dwa parametry oddzielone średnikiem. Pierwszy to wartość rejestru, drugi to tekst o maksymalnej długości 19 znaków. Trzeba pamiętać, aby teksty miały tą samą długość, w razie potrzeby trzeba uzupełnić brakujące litery „SPACJAMI”.

Ten plik opisuje listę tekstów w okienku wyskakującym podczas edycji, możemy w nim zadeklarować maksymalnie do 7 elementów.

Plik I_dni_lewa.txt

```
0;Poniedzialek
1;Wtorek
2;Sroda
3;Czwartek
4;Piatek
5;Sobota
6;Niedziela
```

Obiekt typu LISTA_ALERT – lista alarmów

Obiekt typu LISTA_ALERT umożliwia wyświetlenie naprzemiennie wszystkich aktualnie występujących awarii.

zmienna	opis																								
TYP=	Typ danego obiektu.																								
X=	Współrzędna x pozycji wyświetlanego obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.																								
Y=	Współrzędna y pozycji wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy górnej krawędzi ekranu.																								
ROWNAJ=	Parametr określa, od której strony ma być narysowany obiekt. Opcje: <ul style="list-style-type: none">▪ do lewej – obiekt będzie narysowany od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu,▪ centralnie – obiekt będzie narysowany tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu,▪ do prawej – obiekt będzie narysowany od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu.																								
FONT=	Wielkość czcionki, Opcje: 12, 16, 20, 24, 36. <table border="1" data-bbox="584 853 1318 1061"><thead><tr><th>FONT</th><th>wysokość</th><th>szerokość</th><th>szerokość małych elementów (. , : ;)</th></tr></thead><tbody><tr><td>12</td><td>12</td><td>8</td><td>4</td></tr><tr><td>16</td><td>16</td><td>11</td><td>5</td></tr><tr><td>20</td><td>20</td><td>13</td><td>6</td></tr><tr><td>24</td><td>24</td><td>15</td><td>7</td></tr><tr><td>36</td><td>36</td><td>24</td><td>10</td></tr></tbody></table> <p>Powyższa tabela może być przydatna, jeśli chcemy na ekranie dokładnie rozmieścić kilka tekstów lub wyświetlaczy numerycznych o różnych rozmiarach czcionek i różnej długości znaków. Wszystkie litery i cyfry mają taką samą szerokość z wyjątkiem kilku znaków interpunkcyjnych: . , : ; .itp.</p>	FONT	wysokość	szerokość	szerokość małych elementów (. , : ;)	12	12	8	4	16	16	11	5	20	20	13	6	24	24	15	7	36	36	24	10
FONT	wysokość	szerokość	szerokość małych elementów (. , : ;)																						
12	12	8	4																						
16	16	11	5																						
20	20	13	6																						
24	24	15	7																						
36	36	24	10																						
KOLOR=	Kolor wyświetlanej listy z alarmami w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.																								
BLINK=	Parametr określa czy dany obiekt będzie migał. <ul style="list-style-type: none">▪ 0 – obiekt będzie wyświetlany ciągle,▪ 1 – obiekt będzie migał z częstotliwością 1s.																								
LISTA=	Nazwa pliku z listą alarmów do wyświetlania. W tym dodatkowym pliku zapisane są następujące informacje: numer sterownika, numer rejestru, wartość tego rejestru i treść komunikatu. Tej samej listy można użyć w obiekcie BOX_ALERT.																								

Przykład

Pokazany poniżej obiekt <alert> opisuje listę wyświetlanych awarii (TYP=LISTA_ALERT). Obiekt będzie wyświetlany na współrzędnych x:240 i y:280 (X=240, Y=280), podajemy współrzędne środka górnej krawędzi obiektu (ROWNAJ=centralnie). Wyświetlany tekst będzie miał wielkość czcionki 24 (FONT=24) w kolorze czerwonym (KOLOR=250,0,0) i będzie migać (BLINK=1). Lista wyświetlanych alarmów będzie zadeklarowana w pliku „l_alert.txt” (LISTA=l_alert), poniżej przykład takiego pliku.

```
<alert>
TYP=LISTA_ALERT
X=240
Y=280
ROWNAJ=centralnie
FONT=24
KOLOR=250,0,0
BLINK=1
LISTA=l_alert
</>
```

W poniższym pliku w każdej linijce mamy cztery parametry oddzielone średnikiem:

- numer sterownika, (np. 0 oznacza, że zmienna dotyczy sterownika opisanego w pliku ster0.txt),
- numer rejestru,
- wartość tego rejestru, dla którego występuje dana awaria,
- wyświetlany tekst dla danej awarii,

Możemy zadeklarować do 64 elementów (różnych awarii) o maksymalnej długości tekstu 32 znaki.

W poniższym przykładzie z listą alarmów widzimy, że jeśli w sterowniku o numerze 0 opisanym w pliku ster0.txt w rejestrze 4101 będzie wartość 1 to zostanie wyświetlony komunikat „Awaria czujnika Tkan!”.

Plik l_alert.txt

```
0;4101;1;Awaria czujnika Tkan!
0;4102;1;Awaria czujnika Tzew!
0;4111;1;Brudny filtr czerpni!
0;4112;1;Brudny filtr nawiewu!
1;4201;1;Awaria czujnika Twel!
1;4202;1;Awaria czujnika Twe2!
```

W liście alrtów jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.

Obiekt typu LISTA_BMP – lista bitmap typu RGB 24-bitowy

Obiekt typu LISTA_BMP umożliwia wyświetlenie różnych bitmap w zależności od wartości danego rejestru lub flagi. Obrazki te muszą być 24-bitowymi bitmapami o rozszerzeniu .bmp.

zmienna	opis
TYP=	Typ danego obiektu.
X=	Współrzędna x pozycji wyświetlanego obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.
Y=	Współrzędna y pozycji wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy górnej krawędzi ekranu.
W=	Szerokość obiektu.
H=	Wysokość obiektu.
ROWNAJ=	<p>Parametr określa, od której strony ma być narysowany obiekt. Opcje:</p> <ul style="list-style-type: none"> ▪ do lewej – obiekt będzie narysowany od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu, ▪ centralnie – obiekt będzie narysowany tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu, ▪ do prawej – obiekt będzie narysowany od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu.
BLINK=	<p>Parametr określa czy dany obiekt będzie migał.</p> <ul style="list-style-type: none"> ▪ 0 – obiekt będzie wyświetlany ciągle, ▪ 1 – obiekt będzie migał z częstotliwością 1s.
AKCJA=	<p>Rodzaj listy. Opcje:</p> <ul style="list-style-type: none"> ▪ cedit – lista cyklicznie przewijana, po dotknięciu obiektu następuje zmiana na następną w kolejności grafikę z listy, ▪ brak – lista tylko do wyświetlania.
LISTA=	Nazwa pliku z listą wartości i przypisanymi do nich plikami graficznymi. Pliki te muszą być 24-bitowymi bitmapami o rozszerzeniu .bmp. Poniżej przykład takiego pliku.
STER=	<p>Numer sterownika, którego dotyczy dany obiekt.</p> <p>W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika. <i>Przykład:</i> <i>STER=0 oznacza, że obiekt dotyczy sterownika opisanego w pliku ster0.txt.</i></p>
ADR=	<p>Adres 16-bitowego rejestru (holding register) lub flagi (coil) dostępnego do odczytu/zapisu z wykorzystaniem protokołu Modbus RTU lub Modbus TCP. Adres musi być z zakresu zadeklarowanego w pliku sterX.txt gdzie X oznacza numer sterownika (SL_DADR do SL_DADR+SL_DSIZE-1) lub z zakresu (SL_FADR do SL_FADR+SL_FSIZE-1). W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejstry.txt) z taką listą nazw rejestrów.</p>
ENABLE=	<p>Warunek wyświetlania danego obiektu. Jeśli chcemy żeby dany obiekt był wyświetlany zawsze to możemy całkowicie pominąć liniijkę z tym warunkiem ENABLE.</p> <p>Jeśli chcemy, aby dany obiekt był wyświetlany w zależności od jakiegoś warunku to podajemy go w tej linijce po słowie ENABLE.</p> <p><i>Przykłady:</i> <i>ENABLE=0,4165=1</i> – obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 0 jest wartość równa 1. <i>ENABLE=8,HMI_PASS_LEVEL!0</i> – obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp (użytkownik podał poprawnie hasło projektu PASS1, PASS2 lub PASS3).</p>

	W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.
--	---

Przykład

Pokazany poniżej obiekt <bmp_dni> opisuje cyklicznie edytowalną (AKCJA=cedit) listę dni tygodnia w postaci graficznej czyli obrazków z 24-bitową grafiką (TYP=LISTA_BMP) o wymiarach 135 na 90 pikseli (W=135, H=90). Obiekt będzie wyświetlany na współrzędnych x:5 i y:230 (X=5, Y=230), podajemy współrzędne lewego górnego rogu obiektu (ROWNAJ=do lewej).

Grafiki będą wyświetlane w zależności od wartości rejestru 4006 (ADR=4006) w sterowniku zadeklarowanym w pliku ster0.txt (STER=0) i nie będą migać (BLINK=0).

Lista wyświetlanych grafik i wartości do nich przypisanych będzie zadeklarowana w pliku „l_dni_bmp.txt” (LISTA=l_dni_bmp), poniżej przykład takiego pliku.

```
<bmp_dni>
TYP=LISTA_BMP
X=5
Y=230
W=135
H=90
ROWNAJ=do lewej
BLINK=0
AKCJA=cedit
LISTA=l_dni_bmp
STER=0
ADR=4006
</>
```

W poniższym pliku w każdej linijce mamy dwa parametry oddzielone średnikiem. Pierwszy to wartość rejestru, dla którego ma być wyświetlona dana grafika, drugi to nazwa pliku graficznego 24-bitowego o rozszerzeniu .bmp.

Plik l_dni_bmp.txt

```
0;dt_pn
1;dt_wt
2;dt_sr
3;dt_cz
4;dt_pt
5;dt_so
6;dt_ni
```

Obiekt typu DYNLISTA_BMP – dynamiczna lista bitmap typu RGB 24-bitowy

Obiekt typu DYNLISTA_BMP umożliwia wyświetlenie różnych bitmap w zależności od zadeklarowanej listy z możliwymi zdarzeniami. W danym momencie wyświetlana będzie jedna bitmapa, ta o najwyższym priorytecie. Priorytet ustalany jest w zależności od kolejności wpisanych zdarzeń. Najwyższy priorytet ma zdarzenie na samej górze listy. Lista ta jest zapisana w osobnym pliku.

zmienna	opis
TYP=	Typ danego obiektu.
X=	Współrzędna x pozycji wyświetlanego obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.
Y=	Współrzędna y pozycji wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy górnej krawędzi ekranu.
W=	Szerokość obiektu.
H=	Wysokość obiektu.
ROWNAJ=	Parametr określa, od której strony ma być narysowany obiekt. Opcje: <ul style="list-style-type: none">▪ do lewej – obiekt będzie narysowany od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu,▪ centralnie – obiekt będzie narysowany tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu,▪ do prawej – obiekt będzie narysowany od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu.
LISTA=	Nazwa pliku z listą możliwych zdarzeń i przypisanymi do nich plikami graficznymi. Pliki te muszą być 24-bitowymi bitmapami o rozszerzeniu .bmp. Poniżej przykład takiego pliku.
STER=	Numer sterownika, którego dotyczy dany obiekt. W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika. <i>Przykład:</i> <i>STER=0</i> oznacza, że obiekt dotyczy sterownika opisanego w pliku <i>ster0.txt</i> .
ENABLE=	Warunek wyświetlania danego obiektu. Jeśli chcemy żeby dany obiekt był wyświetlany zawsze to możemy całkowicie pominąć linijkę z tym warunkiem ENABLE. Jeśli chcemy, aby dany obiekt był wyświetlany w zależności od jakiegoś warunku to podajemy go w tej linijce po słowie ENABLE. <i>Przykłady:</i> <i>ENABLE=8,HMI_PASS_LEVEL!0</i> – obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp (użytkownik podał poprawnie hasło projektu PASS1, PASS2 lub PASS3). <i>ENABLE=0,4165!0</i> – obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 0 jest wartość różna od 0. W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestr.txt) z taką listą nazw rejestrów.

Przykład

Pokazany poniżej obiekt <dynlist> opisuje dynamiczną listę bitmap (TYP=LISTA_BMP) o wymiarach 36 na 96 pikseli (W=36, H=96). Obiekt będzie wyświetlany na współrzędnych x:240 i y:100 (X=240, Y=100), podajemy współrzędne środka górnej krawędzi obiektu (ROWNAJ=centralnie). Grafiki będą wyświetlane w zależności od zadeklarowanej listy zapisanej w pliku „l_dynlista.txt” (LISTA=l_dynlista). Obrazek ten będzie wyświetlany pod warunkiem, że w zadeklarowanym w pliku sterowniku „ster0.txt”(STER=0) w rejestrze o adresie 4103 będzie wartość większa od 0 (ENABLE=0,4103>0).

```
<dynlist>
TYP=DYNLISTA_BMP
X=240
Y=100
W=36
H=96
ROWNAJ=centralnie
LISTA=l_dynlista
STER=0
ENABLE=0,4103>0
</>
```

W poniższym pliku „l_dynlista.txt”, w każdej linijce mamy cztery parametry oddzielone średnikiem i są to:

- numer sterownika,
- adre rejestru lub flagi i warunek logiczny, dla którego wyświetlana będzie dana grafika, Możliwe logiczne warunki są takie same jak dla parametru ENABLE:
 - równe 7187=1,
 - różne 7188!1,
 - większe 4212>2,
 - mniejsze 4212<1,
- informacje o sposobie wyświetlania danej grafiki, jeśli spełniony jest warunek zdarzenia (0-wyświetlanie ciągłe, 1-miganie obrazu),
- nazwa pliku graficznego 24-bitowego o rozszerzeniu .bmp. który ma być wyświetlony jeśli spełniony jest warunek zdarzenia.

Plik l_dynlista.txt

```
0;7187=1;1;ng_err
0;7188!1;0;ng_war
0;4212>2;0;ng1
0;4212<1;0;ng0
```

W danym momencie będzie wyświetlany tylko jeden obraz, ten o najwyższym priorytecie (czyli w kolejności na liście od góry do dołu). Najpierw sprawdzany jest pierwszy warunek: czy flaga 7187 jest równa 1 (7187=1). Jeśli tak to na ekranie będzie migał obraz zapisany w pliku „ng_err.bmp”. Jeśli nie to sprawdzany jest drugi warunek czy flaga 7188 jest różna od 1 (7188!1). Jeśli tak to na ekranie będzie wyświetlany obraz zapisany w pliku „ng_war.bmp”. Jeśli nie to sprawdzany jest kolejny warunek czy rejestr 4212 jest większy niż 2 (4212>2). Jeśli tak to na ekranie będzie wyświetlany obraz zapisany w pliku „ng1.bmp”. Jeśli nie to sprawdzany jest kolejny warunek czy rejestr 4212 jest mniejszy niż 1 (4212<1). Jeśli tak to na ekranie będzie wyświetlany obraz zapisany w pliku „ng0.bmp”. Jeśli nie to kolejnych warunków do sprawdzania z listy w tym przykładzie już nie ma i nie będzie wyświetlana żadna grafika.

Obiekt typu LISTA_IKON – lista obrazów typu mono

Obiekt typu LISTA_IKON umożliwia wyświetlenie różnych obrazów w zależności od wartości danego rejestru lub flagi. Obrazy te muszą być 24-bitowymi bitmapami o rozszerzeniu .bmp, które podczas kompilacji będą przekonwertowane do obrazków mono czyli jednokolorowych. Każdy bit w kolorze innym niż biały traktowany jest jako „1” i rysowany jest w kolorze zdefiniowanym w obiekcie. Dla każdej wartości rejestru możemy zdefiniować inny kolor wyświetlania ikony.

zmienna	opis
TYP=	Typ danego obiektu.
X=	Współrzędna x pozycji wyświetlanego obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.
Y=	Współrzędna y pozycji wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy górnej krawędzi ekranu.
W=	Szerokość obiektu.
H=	Wysokość obiektu.
BLINK=	Parametr określa czy dany obiekt będzie migał. <ul style="list-style-type: none"> ▪ 0 – obiekt będzie wyświetlany ciągle, ▪ 1 – obiekt będzie migał z częstotliwością 1s.
ROWNAJ=	Parametr określa, od której strony ma być narysowany obiekt. Opcje: <ul style="list-style-type: none"> ▪ do lewej – obiekt będzie narysowany od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu, ▪ centralnie – obiekt będzie narysowany tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu, ▪ do prawej – obiekt będzie narysowany od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu.
AKCJA=	Rodzaj listy. Opcje: <ul style="list-style-type: none"> ▪ cedit – lista cyklicznie przewijana, po dotknięciu obiektu następuje zmiana na następną w kolejności grafikę z listy, ▪ brak – lista tylko do wyświetlania.
LISTA=	Nazwa pliku z listą wartości, kolorów i przypisanymi do nich plikami graficznymi. Pliki te muszą być 24-bitowymi bitmapami o rozszerzeniu .bmp. Poniżej przykład takiego pliku.
STER=	Numer sterownika, którego dotyczy dany obiekt. W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika. <i>Przykład:</i> <i>STER=0 oznacza, że obiekt dotyczy sterownika opisanego w pliku ster0.txt.</i>
ADR=	Adres 16-bitowego rejestru (holding register) lub flagi (coil) dostępnego do odczytu/zapisu z wykorzystaniem protokołu Modbus RTU lub Modbus TCP. Adres musi być z zakresu zadeklarowanego w pliku sterX.txt gdzie X oznacza numer sterownika (SL_DADR do SL_DADR+SL_DSIZE-1) lub z zakresu (SL_FADR do SL_FADR+SL_FSIZE-1). W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestr.txt) z taką listą nazw rejestrów.
ENABLE=	Warunek wyświetlania danego obiektu. Jeśli chcemy żeby dany obiekt był wyświetlany zawsze to możemy całkowicie pominąć linijkę z tym warunkiem ENABLE. Jeśli chcemy, aby dany obiekt był wyświetlany w zależności od jakiegoś warunku to podajemy go w tej linijce po słowie ENABLE. <i>Przykłady:</i> <i>ENABLE=4165=1 - obiekt będzie wyświetlany, jeśli w rejestrze 4165 jest wartość równa 1.</i> <i>ENABLE=8,HMI_PASS_LEVEL!0 - obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp</i>

(użytkownik podał poprawnie hasło projektu PASS1, PASS2 lub PASS3).
W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.

Przykład

Pokazany poniżej obiekt <ikona1> opisuje wyświetlaną listę ikon (TYP=LISTA_IKON), które mają sygnalizować różne stany awaryjne w sterowniku opisanym w pliku (STER=0) i nie będą migać (BLINK=0). Adres rejestru z informacją o awariach to 4011 (ADR=4011). Obiekt będzie wyświetlał ikony zadeklarowane w pliku „l_ierror.txt” (LISTA=l_ierror) bez możliwości edycji (AKCJA=brak). Ikony będą miały rozmiar 180 na 120 pikseli (W=180, H=120), które będą wyświetlane na współrzędnych x:100 i y:200 (x=100, y=200), podane współrzędne dotyczą lewego górnego narożnika (ROWNAJ=do lewej).

```
<ikona1>
TYP=LISTA_IKON
X=100
Y=200
W=180
H=120
ROWNAJ=do lewej
AKCJA=brak
LISTA=l_ierror
STER=0
ADR=4011
</>
```

Pokazany poniżej plik **I_ierror.txt** opisuje dokładnie listę ikon wyświetlaną w powyższym obiekcie. W każdej linijce są trzy parametry oddzielone średnikiem:

- wartość rejestru, dla której dana ikona jest wyświetlana (2),
- kolor tej ikony, dla każdej wartości może być inny kolor (255, 0, 0),
- nazwa pliku z daną ikoną (icon_t_error).

Kolor wyświetlanej ikony jest zapisany w formacie RGB 24b (24-bitowym). Liczby muszą być rozdzielone przecinkiem (np. dla koloru czerwonego to będzie następujący zapis 255,0,0). Każdy plik ikony musi być plikiem graficznym 24-bitowym o rozszerzeniu .bmp, który w czasie kompilacji przekonwertowany będzie do postaci mono czyli jednokolorowej ikony. Każdy bit w kolorze innym niż biały traktowany jest jako „1” i rysowany jest w kolorze zdefiniowanym osobno dla każdej ikony.

Plik I_ierror.txt

```
2;255,0,0;icon_t_error
3;255,0,0;icon_t_error
4;255,0,0;icon_t_error
5;255,0,0;icon_t_error
9;255,0,0;icon_t_error
203;255,165,0;icon_t_max
209;255,165,0;icon_t_max
```

Pliki „icon_t_error.bmp” i „icon_t_max.bmp” są plikami graficznymi zapisanymi w formacie 24-bitowej bitmapy.

Obiekt typu ONOFF_IKONA – migająca ikona

Obiekt typu ONOFF_IKONA jest specyficznym obiektem. Posiada dwa stany: wyłączenia i załączenia. W stanie załączenia obiekt cyklicznie zmienia swój kolor, dzięki czemu można zrobić efekt migania. Stan załączenia można uwarunkować przez dodatkowy rejestr. Plik graficzny musi być 24-bitowa bitmapą o rozszerzeniu .bmp, która podczas kompilacji będzie przekonwertowana do obrazka mono czyli jednokolorowej ikony.

zmienna	opis
TYP=	Typ danego obiektu.
X=	Współrzędna x pozycji wyświetlanego obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.
Y=	Współrzędna y pozycji wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy górnej krawędzi ekranu.
W=	Szerokość obiektu.
H=	Wysokość obiektu.
ROWNAJ=	Parametr określa, od której strony ma być narysowany obiekt. Opcje: <ul style="list-style-type: none"> ▪ do lewej – obiekt będzie narysowany od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu, ▪ centralnie – obiekt będzie narysowany tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu, ▪ do prawej – obiekt będzie narysowany od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu.
KOLOR_OFF=	Kolor ikony w stanie wyłączenia w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.
KOLOR_ON1=	Pierwszy (początkowy) kolor ikony w stanie załączenia w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem. Ikona w stanie załączenia będzie cyklicznie zmieniać kolor pomiędzy KOLOR_ON1 a KOLOR_ON2.
KOLOR_ON2=	Drugi (końcowy) kolor ikony w stanie załączenia w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.
IKONA=	Nazwa pliku graficznego 24-bitowego z rozszerzeniem .bmp, który będzie przekonwertowany do postaci mono czyli jednokolorowej ikony.
ON=	Warunek załączenia. Jeśli wartość odczytana z rejestru zadeklarowanego w parametrze ADR jest równa temu parametrowi to ikona jest w stanie załączenia. <i>Przykłady:</i> <i>Jeśli ON=0 i ADR=7093 to</i> <i>R7093=0 to ikona załączona,</i> <i>R7093≠1 to ikona wyłączona.</i> <i>Jeśli ON=100 i ADR=4094 to</i> <i>R4094=100 to ikona załączona,</i> <i>R4094≠100 to ikona wyłączona.</i>
STER=	Numer sterownika, którego dotyczy dany obiekt. W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika. <i>Przykład:</i> <i>STER=0 oznacza, że obiekt dotyczy sterownika opisanego w pliku ster0.txt.</i>
ADR=	Adres 16-bitowego rejestru (holding register) lub flagi (coil) dostępnego do odczytu/zapisu z wykorzystaniem protokołu Modbus RTU lub Modbus TCP. Adres musi być z zakresu zadeklarowanego w pliku sterX.txt gdzie X oznacza numer sterownika (SL_DADR do SL_DADR+SL_DSIZE-1) lub z zakresu (SL_FADR do SL_FADR+SL_FSIZE-1). W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.

IF_ADR=	<p>Dodatkowy warunek załączenia. Jeśli wartość odczytana z rejestru lub flagi IF_ADR jest większa od 0 to stan ikony zależy od rejestru (lub flagi) określonego w parametrze ADR. Jeśli wartość jest równa 0 to stan ikony jest wyłączony. Jeśli IF_ADR= -1 to warunek jest nieaktywny.</p> <p><i>Przykład:</i> <i>Jeśli ON=0, ADR=4094 i IF_ADR=4095 to R4094=0 i R4095>0 to ikona jest załączona, R4094=1 i R4095>0 to ikona wyłączona, R4094=0 i R4095=0 to ikona wyłączona.</i></p>
ENABLE=	<p>Warunek wyświetlania danego obiektu. Jeśli chcemy żeby dany obiekt był wyświetlany zawsze to możemy całkowicie pominąć linijkę z tym warunkiem ENABLE.</p> <p>Jeśli chcemy, aby dany obiekt był wyświetlany w zależności od jakiegoś warunku to podajemy go w tej linijce po słowie ENABLE.</p> <p><i>Przykłady:</i> <i>ENABLE=4165=1 - obiekt będzie wyświetlany, jeśli w rejestrze 4165 jest wartość równa 1.</i> <i>ENABLE=8,HMI_PASS_LEVEL!0 - obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp (użytkownik podał poprawnie hasło projektu PASS1, PASS2 lub PASS3).</i></p> <p>W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestr.txt) z taką listą nazw rejestrów.</p>

Przykład

Pokazany poniżej obiekt <ikona_cwu> opisuje wyświetlaną dwustanową ikonę (TYP=ONOFF_IKONA). Ikona pobrana z pliku graficznego 24-bitowego o nazwie „icon_cwu.bmp” (IKONA=icon_cwu) zmienia swój stan w zależności od rejestru 4074 (ADR=4074) w sterowniku opisanym w pliku „ster.txt” (STER=0). W stanie wyłączenia ma kolor (KOLOR_OFF=25,25,112), natomiast w stanie załączenia, jeśli rejestr równa się 100 (ON=100), płynnie zmienia swój kolor pomiędzy dwoma zdefiniowanymi kolorami (KOLOR_ON1=32,128,32, KOLOR_ON2=0,255,0). Ikona w rozmiarze 36 na 30 pikseli (W=36, H=30), będzie wyświetlana na współrzędnych x:250 i y:200 (X=250, Y=200), podane współrzędne dotyczą lewego górnego narożnika (ROWNAJ=do lewej). Ikona ta będzie wyświetlana zawsze, nie posiada dodatkowego warunku wyświetlania (IF_ADR=-1).

```

<ikona_cwu>
TYP=ONOFF_IKONA
X=250
Y=200
W=36
H=30
ROWNAJ=do lewej
KOLOR_OFF=25,25,112
KOLOR_ON1=32,128,32
KOLOR_ON2=0,255,0
IKONA=icon_cwu
ON=100
STER=0
ADR=4074
IF_ADR=-1
</>

```

Obiekt typu PASS – hasło do trybu edycji

Obiekt typu PASS umożliwia odblokowanie dostępu do akcji edycji parametrów edytowalnych oraz ustawienie poziomu dostępu w rejestrze HMI_PASS_LEVEL.

Obiekt ten to obrazek 24-bitowy o rozszerzeniu .bmp, który po dotknięciu otwiera okna z klawiaturą umożliwiającą wpisanie hasła. Po poprawnym wpisaniu jednego z haseł zadeklarowanych w pliku projekt.txt w linijce PASS1, PASS2 lub PASS3 panel HMI pracuje w trybie odblokowanego dostępu do akcji edycja parametrów, co umożliwia zmianę nastaw wszystkich parametrów, które mają zadeklarowaną taką możliwość w swoich parametrach (AKCAJ=edit lub AKCJA=cedit). Dotyczy to następujących obiektów:

- VAR16 – wyświetlacz liczby 16-bitowej,
- AKCJA_DEC – akcja dekrementacji,
- AKCJA_INC – akcja inkrementacji,
- AKCJA_SET – akcja ustawienia zmiennej,
- AKCJA_FLAG_SET – akcja ustawienia flagi,
- KLAWISZ_DEC – przycisk dekrementacji,
- KLAWISZ_INC – przycisk inkrementacji,
- KLAWISZ_SET – przycisk ustawienia zmiennej,
- KLAWISZ_FLAG_SET – przycisk ustawienia flagi,
- LISTA_TXT – lista tekstów,
- LISTA_BMP – lista obrazków typu RGB 24-bitowy,
- LISTA_IKON – lista obrazków typu mono.

Dodatkowo w rejestrze HMI_PASS_LEVEL ustawiony jest poziom dostępu (1, 2 lub 3) uzależniony od wprowadzonego poprawnie hasła (PASS1, PASS2 lub PASS3).

Rejestr HMI_PASS_LEVEL można wykorzystać w definicji ENABLE uzależniając wyświetlanie parametrów od posiadanych przez użytkownika praw dostępowych.

Przykład

ENABLE=8,HMI_PASS_LEVEL=1 - obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp do poziomu 1 (użytkownik podał poprawnie hasło projektu PASS1).

ENABLE=8,HMI_PASS_LEVEL=2 - obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp do poziomu 2 (użytkownik podał poprawnie hasło projektu PASS2).

ENABLE=8,HMI_PASS_LEVEL=3 - obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp do poziomu 3 (użytkownik podał poprawnie hasło projektu PASS3).

ENABLE=8,HMI_PASS_LEVEL>1 - obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp do poziomu 2 lub 3 (użytkownik podał poprawnie hasło projektu PASS2 lub PASS3).

ENABLE=8,HMI_PASS_LEVEL!0 - obiekt będzie wyświetlany, jeśli w panelu odblokowano dostęp (użytkownik podał poprawnie hasło projektu PASS1, PASS2 lub PASS3).

Do czasu podania poprawnego hasła żaden obiekt nie podlega edycji i jest możliwe tylko ich przeglądanie i przechodzenie pomiędzy wszystkimi ekranami.

Funkcja zabezpieczenia hasłem działa tylko wtedy, gdy podamy hasła różne od zera w pliku projekt.txt w parametrach PASS1, PASS2 i PASS3.

zmienna	opis
TYP=	Typ danego obiektu.
X=	Współrzędna x pozycji wyświetlanego obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.
Y=	Współrzędna y lewego górnego rogu wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy górnej krawędzi ekranu.
W=	Szerokość obiektu.
H=	Wysokość obiektu.
ROWNAJ=	Parametr określa, od której strony ma być narysowany obiekt (nie dotyczy to napisu określonego w parametrze NAZWA). Opcje: <ul style="list-style-type: none">▪ do lewej – obiekt będzie narysowany od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu,▪ centralnie – obiekt będzie narysowany tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu,▪ do prawej – obiekt będzie narysowany od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu.
BMP_ON=	Nazwa pliku graficznego 24-bitowego z rozszerzeniem .bmp. Obrazek ten jest wyświetlany w trybie edycji, czyli po poprawnym podaniu hasła. Plik ten musi być umieszczony w tym samym katalogu co cały projekt. Może to być np. obrazek z otwartą kłódką.
BMP_OFF=	Nazwa pliku graficznego 24-bitowego z rozszerzeniem .bmp. Obrazek ten jest wyświetlany w trybie blokady edycji, czyli bez poprawnie wpisanego hasła. Plik ten musi być umieszczony w tym samym katalogu co cały projekt. Może to być np. obrazek z zamkniętą kłódką.

Przykład

Pokazany poniżej obiekt `<pass>` umożliwia wejście do trybu edycji. Obrazek o wymiarach 64 na 64 pikseli ($W=64$, $H=64$) będzie wyświetlany na współrzędnych $x:0$ i $y:97$ ($x=0$, $y=97$), podane współrzędne dotyczą lewego górnego narożnika (`ROWNAJ=do lewej`). Grafika wyświetlana w trybie edycji jest w pliku `icon_pass_on.bmp` (`BMP_ON=icon_pass_on`) natomiast grafika wyświetlana w trybie blokady edycji jest w pliku `icon_pass_off.bmp` (`BMP_ON=icon_pass_off`).

```
<pass>
TYP=PASS
X=0
Y=97
W=64
H=64
ROWNAJ=do lewej
BMP_ON=icon_pass_on
BMP_OFF=icon_pass_off
</>
```

Obiekt typu PROGRAM – interpretacja graficzna programu dobowego.

Obiekt typu PROGRAM jest specyficznym obiektem, który umożliwia graficzne przedstawienie dwóch przedziałów czasowych programu dobowego w formie wykresu słupkowego. Jest to obiekt współpracującym z programami w sterownikach Frisko.

zmienna	opis
TYP=	Typ danego obiektu.
X=	Współrzędna x pozycji wyświetlanego obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.
Y=	Współrzędna y pozycji wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy górnej krawędzi ekranu.
ROWNAJ=	Parametr określa, od której strony ma być narysowany obiekt. Opcje: <ul style="list-style-type: none"> ▪ do lewej – obiekt będzie narysowany od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu, ▪ centralnie – obiekt będzie narysowany tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu, ▪ do prawej – obiekt będzie narysowany od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu.
KOLOR_HI=	Kolor, który reprezentuje aktywny program dla dwóch zadeklarowanych odcinków czasu: od G11:M11 do G12:M12, od G21:M21 do G22:M22. Kolor zapisany jest w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.
KOLOR_LO=	Kolor, który reprezentuje nieaktywny program, czyli pokazuje odcinki poza dwoma zadeklarowanymi poniżej przedziałami czasowymi. Kolor zapisany jest w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.
KOLOR_TXT=	Kolor czcionki w tym obiekcie w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.
STER=	Numer sterownika, którego dotyczy dany obiekt. W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika. <i>Przykład:</i> <i>STER=0</i> oznacza, że obiekt dotyczy sterownika opisanego w pliku <i>ster0.txt</i> .
ADR_G11=	Adres 16-bitowego rejestru (holding register) z godziną początku pierwszego przedziału czasowego. Rejestr dostępny do odczytu/zapisu z wykorzystaniem protokołu Modbus RTU lub Modbus TCP. Adres musi być z zakresu zadeklarowanego w pliku <i>sterX.txt</i> gdzie X oznacza numer sterownika (SL_DADR do SL_DADR+SL_DSIZE-1). W tym parametrze jako adres rejestru zamiast wartości liczbowej można podać nazwę rejestru. W tym celu należy najpierw przygotować plik (<i>rejestry.txt</i>) z taką listą nazw rejestrów.
ADR_M11=	Adres 16-bitowego rejestru (holding register) z minutami początku pierwszego przedziału czasowego.
ADR_G12=	Adres 16-bitowego rejestru (holding register) z godziną końca pierwszego przedziału czasowego
ADR_M12=	Adres 16-bitowego rejestru (holding register) z minutami końca pierwszego przedziału czasowego.
ADR_G21=	Adres 16-bitowego rejestru (holding register) z godziną początku drugiego przedziału.
ADR_M21=	Adres 16-bitowego rejestru (holding register) z minutami początku drugiego przedziału czasowego.
ADR_G22=	Adres 16-bitowego rejestru (holding register) z godziną końca drugiego przedziału czasowego.
ADR_M22=	Adres 16-bitowego rejestru (holding register) z minutami końca drugiego przedziału czasowego.

Przykład

Pokazany poniżej obiekt <programbar> opisuje wyświetlany program tygodniowy (TYP=PROGRAM) w postaci graficznej. Jest to program tygodniowy zwykle używany w sterownikach firmy Frisko. Obiekt będzie wyświetlany na współrzędnych x:240 i y:50 (X=240, Y=50), podane współrzędne dotyczą środka górnej krawędzi (ROWNAJ=centralnie). Okresy zawierające się w zadeklarowanym przedziale czasowym będą wyświetlane w kolorze czerwonym (KOLOR_HI=255,0,0), natomiast okresy z poza zadeklarowanego przedziału w kolorze niebieskim (KOLOR_LO=0,0,205). Wszystkie teksty w tym obiekcie będą w kolorze szarym (KOLOR_TXT=105,105,105). Wszystkie dane potrzebne do wyświetlenia tej graficznej reprezentacji programu są odczytywane ze sterownika zdefiniowanego w pliku ster0.txt (STER=0).

```
<programbar>
TYP=PROGRAM
X=240
Y=50
ROWNAJ=centralnie
KOLOR_HI=255,0,0
KOLOR_LO=0,0,205
KOLOR_TXT=105,105,105
STER=0
ADR_G11=4181
ADR_M11=4185
ADR_G12=4182
ADR_M12=4186
ADR_G21=4183
ADR_M21=4187
ADR_G22=4184
ADR_M22=4188
</>
```


Obiekt typu RAMKA – kolorowy prostokąt

Obiekt typu RAMKA jest obiektem, który umożliwia narysowanie kolorowego prostokąta z zaokrąglonymi rogami z napisem. Tekst wyświetlany jest na górnej krawędzi ramki i zaczyna się zawsze z lewej strony. Ten obiekt często stosujemy się z obiektami typu VAR16.

zmienna	opis																								
TYP=	Typ danego obiektu.																								
NAZWA=	Wyświetlana nazwa na górnej krawędzi ramki. Maksymalna długość nazwy to 16 znaków. Należy zadbać, aby napis nie był zbyt długi i nie wykraczał poza szerokość ramki. Napis zawsze zaczyna się od lewej strony.																								
X=	Współrzędna x pozycji wyświetlanego obiektu. W zależności od parametru ROWNAJ może to być lewa lub prawa krawędź obiektu albo środek.																								
Y=	Współrzędna y lewego górnego rogu wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy górnej krawędzi ekranu.																								
W=	Szerokość obiektu.																								
H=	Wysokość obiektu.																								
ROWNAJ=	<p>Parametr określa, od której strony ma być narysowany obiekt (nie dotyczy to napisu określonego w parametrze NAZWA). Opcje:</p> <ul style="list-style-type: none"> ▪ do lewej – obiekt będzie narysowany od lewej strony czyli współrzędne x i y oznaczają lewy górny róg obiektu, ▪ centralnie – obiekt będzie narysowany tak, że współrzędne x i y oznaczają środek górnej krawędzi obiektu, ▪ do prawej – obiekt będzie narysowany od prawej strony czyli współrzędne x i y oznaczają prawy górny róg obiektu. 																								
FONT=	<p>Wielkość czcionki, Opcje: 12, 16, 20, 24, 36.</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>FONT</th> <th>wysokość</th> <th>szerokość</th> <th>szerokość małych elementów (.,:;)</th> </tr> </thead> <tbody> <tr> <td>12</td> <td>12</td> <td>8</td> <td>4</td> </tr> <tr> <td>16</td> <td>16</td> <td>11</td> <td>5</td> </tr> <tr> <td>20</td> <td>20</td> <td>13</td> <td>6</td> </tr> <tr> <td>24</td> <td>24</td> <td>15</td> <td>7</td> </tr> <tr> <td>36</td> <td>36</td> <td>24</td> <td>10</td> </tr> </tbody> </table> <p>Powyższa tabela może być przydatna, jeśli chcemy na ekranie dokładnie rozmieścić kilka tekstów lub wyświetlaczy numerycznych o różnych rozmiarach czcionek i różnej długości znaków. Wszystkie litery i cyfry mają taką samą szerokość z wyjątkiem kilku znaków interpunkcyjnych: . , : ; . itp. Zbyt duża czcionka może spowodować że napis wyjdzie w prawo poza zarys ramki.</p>	FONT	wysokość	szerokość	szerokość małych elementów (.,:;)	12	12	8	4	16	16	11	5	20	20	13	6	24	24	15	7	36	36	24	10
FONT	wysokość	szerokość	szerokość małych elementów (.,:;)																						
12	12	8	4																						
16	16	11	5																						
20	20	13	6																						
24	24	15	7																						
36	36	24	10																						
KOLOR=	Kolor ramki w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.																								
STER=	<p>Numer sterownika, którego dotyczy dany obiekt.</p> <p>W pliku o podanym numerze zapisane są wszystkie parametry transmisji w tym adres Modbus sterownika.</p> <p>Numer sterownika jest potrzebny tylko, jeśli korzystamy z warunkowego wyświetlania podanego w następnej linijce – ENABLE.</p> <p><i>Przykład:</i> STER=0 oznacza, że obiekt dotyczy sterownika opisanego w pliku ster0.txt.</p>																								
ENABLE=	<p>Warunek wyświetlania danego obiektu. Jeśli chcemy żeby dany obiekt był wyświetlany zawsze to możemy całkowicie pominąć linijkę z tym warunkiem ENABLE.</p> <p>Jeśli chcemy, aby dany obiekt był wyświetlany w zależności od jakiegoś warunku to podajemy go w tej linijce po słowie ENABLE.</p>																								

Przykłady:

*ENABLE=0,4165=1 - obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 0 jest wartość równa 1.
ENABLE=0,4165>5 - obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 0 jest wartość większa niż 5.
ENABLE=0,4165<3 - obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 0 jest wartość mniejsza niż 3.
ENABLE=0,4165!0 - obiekt będzie wyświetlany, jeśli w rejestrze 4165 sterownika 0 jest wartość różna od 0.*

W warunkach równy '=' lub różny '!', zamiast rejestru może być użyta flaga. W tym parametrze jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.

Przykład

Pokazany poniżej obiekt <ramka1> opisuje prostokątną ramkę (TYP=RAMKA) z napisem „Tzadana” (NAZWA=Tzadana). Ramka o wymiarach 80 na 50 pikseli (W=80, H=50) będzie wyświetlana na współrzędnych x:200 i y:100 (X=200, Y=100), podane współrzędne dotyczą lewego górnego narożnika (ROWNAJ=do lewej).

Ramka i napis będzie w kolorze czerwonym (KOLOR=255,0,0), a czcionka będzie w rozmiarze 20 (FONT=20).

```
<ramka1>
TYP=RAMKA
NAZWA=Tzadana
X=200
Y=100
W=80
H=50
ROWNAJ=do lewej
FONT=20
KOLOR=255,0,0
STER=0
</>
```

Obiekt typu BOX_ALERT – obiekt wyświetlający na całym ekranie alarmy

Obiekt typu BOX_ALERT wykorzystuje cały ekran i umożliwia wyświetlenie w wierszach do 8 aktualnie występujących awarii. Wielkość czcionki jest stała i wynosi 24.

zmienna	opis
TYP=	Typ danego obiektu.
KOLOR=	Kolor wyświetlanej listy z alarmami w formacie RGB 24b (24-bitowym). Wagi poszczególnych kolorów muszą być rozdzielone przecinkiem.
LISTA=	Nazwa pliku z listą alarmów do wyświetlania. W tym dodatkowym pliku zapisane są następujące informacje: numer sterownika, numer rejestru, wartość tego rejestru i treść komunikatu. Tej samej listy można użyć w obiekcie LISTA_ALERT.

Przykład

Pokazany poniżej obiekt <alert> opisuje listę wyświetlanych awarii (TYP=BOX_ALERT). Wyświetlane teksty będą w wierszach na całym ekranie w kolorze czerwonym (KOLOR=250,0,0). Lista wyświetlanych alarmów będzie zadeklarowana w pliku „l_alert.txt” (LISTA=l_alert), poniżej przykład takiego pliku.

```
<boxalert>
TYP=BOX_ALERT
KOLOR=250,0,0
LISTA=l_alert
</>
```

W poniższym pliku w każdej linijce mamy cztery parametry oddzielone średnikiem:

- numer sterownika, (np. 0 oznacza, że zmienna dotyczy sterownika opisanego w pliku ster0.txt),
- numer rejestru,
- wartość tego rejestru, dla którego występuje dana awaria,
- wyświetlany tekst dla danej awarii,

Możemy zadeklarować do 64 elementów (różnych awarii) o maksymalnej długości tekstu 32 znaki.

W poniższym przykładzie z listą alarmów widzimy, że jeśli w sterowniku o numerze 0 opisanym w pliku ster0.txt w rejestrze 4101 będzie wartość 1 to zostanie wyświetlony komunikat „Awaria czujnika Tkan!”.

Plik l_alert.txt

```
0;4101;1;Awaria czujnika Tkan!
0;4102;1;Awaria czujnika Tzew!
0;4111;1;Brudny filtr czerpni!
0;4112;1;Brudny filtr nawiewu!
1;4201;1;Awaria czujnika Twe1!
1;4202;1;Awaria czujnika Twe2!
```

W liście alertów jako adres rejestru (flagi) zamiast wartości liczbowej można podać nazwę rejestru (flagi). W tym celu należy najpierw przygotować plik (rejestry.txt) z taką listą nazw rejestrów.

Obiekt typu STATUS_COM – status komunikacji ze sterownikiem

Obiekt typu STATUS_COM wyświetla symbol statusu komunikacji ze sterownikiem.

zmienna	opis
TYP=	Typ danego obiektu.
X=	Współrzędna x lewego górnego rogu wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy lewej krawędzi ekranu.
Y=	Współrzędna y lewego górnego rogu wyświetlanego obiektu. 0 oznacza, że obiekt będzie narysowany maksymalnie przy górnej krawędzi ekranu.
STER=	Parametr określa, którego zadeklarowanego sterownika dotyczy status komunikacji. <i>Przykład:</i> <i>STER=0 oznacza, że obiekt dotyczy sterownika opisanego w pliku ster0.txt.</i>

Przykład

Pokazany poniżej obiekt <com> opisuje status komunikacji (TYP=STATUS_COM) ze sterownikiem opisanym w pliku ster0 (STER=0).

```
<com>
TYP=STATUS_COM
X=5
Y=300
STER=0
</>
```

Status komunikacji ze sterownikami można też wyświetlić korzystając z rejestrów systemowych HMI_STER0_STATKOM...HMI_STER7_STATKOM.

Specyfikacja techniczna

Wyświetlacz	Typ	3,5" TFT
	Aktywny obszar	73,44 x 48,96 mm (SxW)
	Rozdzielczość	480 x 320 pikseli
	Rozmiar pixela	0,153 x 0,153 mm
	Liczba kolorów	65K RGB 16bit
	Jasność (cd/m2)	300
	Rekomendowany kierunek patrzenia	12 godzina
	Kąty widzenia	50° , 60° , 60° , 60°
	Typ podświetlania	LED
	Żywotność podświetlanie	>30000 godzin
Matryca dotykowa	Typ	Rezystancyjna
	Precyzja	(X)±2%, (Y)±2% (danej długości boku obszaru aktywnego)
Jednostka centralna	Procesor	32 Bitowy ARM Cortex-M7 @ 96MHz
	Pamięć	1,4MB RAM
		2MB Flash (system)
		8MB ExFlash (program i dane)
RTC	Wbudowany	
Porty	USB Host	USB 2.0 typ A (programowanie z wykorzystaniem Pendrive)
	USB Klient	USB 2.0 typ micro-B (złącze systemowe)
	COM	RS485 2W obsługa Modbus RTU MASTER, prędkości transmisji: 1200, 2400, 4800, 9600, 19200, 38400 bps, format: 8-NONE-1, 8-EVEN-1, 8-ODD-1.
	Ehternet	Wi-Fi (moduł ESP8266) obsługa Modbus TCP MASTER
Zasilanie	Napięcie zasilania	12...24VDC
	Pobór prądu (tryb aktywny)	55mA@12VDC...30mA@24VDC
	Pobór prądu (tryb uśpienia)	16mA@12VDC...10mA@24VDC
Obudowa	Typ	Tablicowa, ABS
	Wymiary	119 x 84 x 33,5 mm (SxWxG)
	Wymiary otworu montażowego	109 x 74 mm (SxW)
	Masa	165 g
Środowisko	Stopień ochrony	IP65 (front panelu), IP20 (reszta obudowy)
	Temperatura przechowywania	+5°C...+40°C
	Temperatura pracy	+5°C...+40°C
	Wilgotność względna	10%...90% (bez kondensacji)