

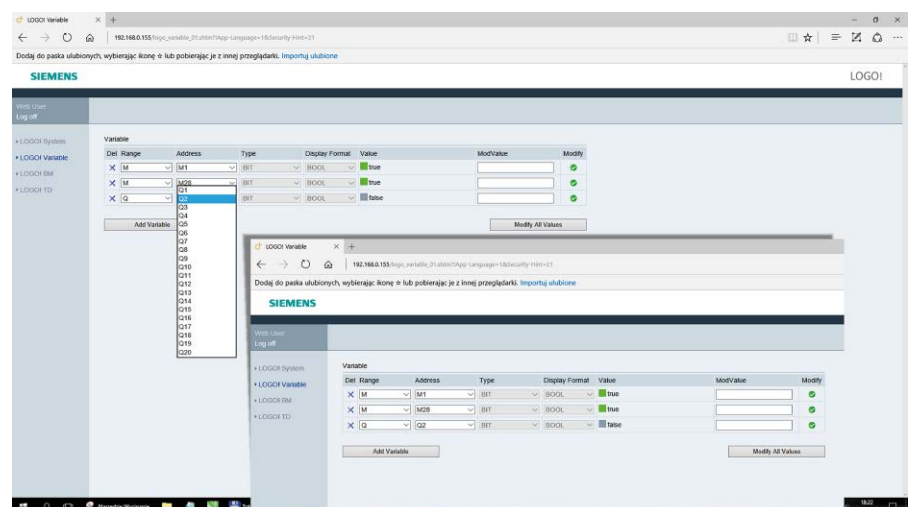
Internetowy sterownik podlewania ogrodowego na LOGO!

Zdalna parametryzacja programu

Artykuł jest odpowiedzią na pytania Czytelników, którzy szukają rozwiązań powiększających standardowe możliwości LOGO! Jednym z tematów, zgłaszanych do redakcji jako istotny do opracowania, była możliwość zdalnego wpływu na stany wyjść sterownika (np. ich zdalnego blokowania) niezależnie od aktualnego stanu programu. Rozwiązanie takie – pozornie – jest dostępne we wbudowanym w sterownik serwerze http, bowiem po wyświetleniu strony generowanej przez LOGO! w oknie przeglądarki dostępna jest możliwość zmiany stanu wybranych wyjść (**rysunek 1**).

Pozorność tej możliwości wynika z ochrony użytych w projekcie obwodów wyjściowych sterowników przed zewnętrzną manipulacją, bowiem może ona grozić nieprawidłowym działaniem urządzenia sterowanego przez LOGO! i wynikającymi z tego niebezpieczeństwami dla użytkownika. Tak więc, pomimo, tego że wszystkie wyjścia są dostępne na stronie LOGO! Variable (**rysunek 1**) ze statusem „Modify” (zielony znaczek z prawej strony ekranu), to próby

Przedstawione w poprzednich artykułach sposoby zdalnego dostępu do zasobów LOGO! 8 nie wyczerpują oferowanych przez ten sterownik możliwości. W artykule przedstawiamy sposoby zdalnej parametryzacji programów, którą można wykorzystać także do zarządzania pracą urządzenia z poziomu serwera http wbudowanego w LOGO!



Rysunek 1. Okno umożliwiające zmianę stanu wybranych wyjść

zdalnej zmiany stanu wyjść użytych w programie nie będą niczym skutkowały. W ten sposób użytkownik może wpływać wyłącznie na stan wyjść niewykorzystanych w programie.

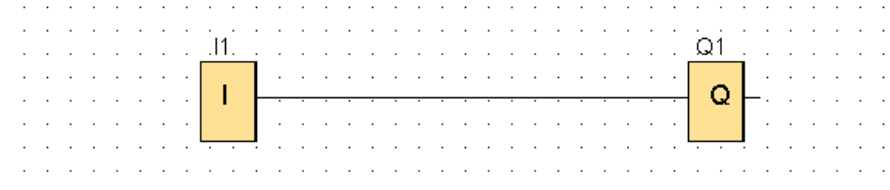
Jeżeli z jakichś przyczyn użytkownikowi zależy na zdalnym wpływaniu na zachowanie wyjść użytych w programie (np. zdalna blokada wyjść niezależnie od stanu sterownika i wykonywanego przez niego programu), to można taki efekt uzyskać poprzez niewielką rozbudowę programu.

Jak to zrobić, pokażemy na prostym przykładzie – program, który będziemy wyposażać w możliwość zdalnego blokowania wyjść, pokazano na **rysunku 2**. Jest to – jak widać – prosty monitor stanu wejścia, powielający stan wejścia I1 na wyjście Q1.

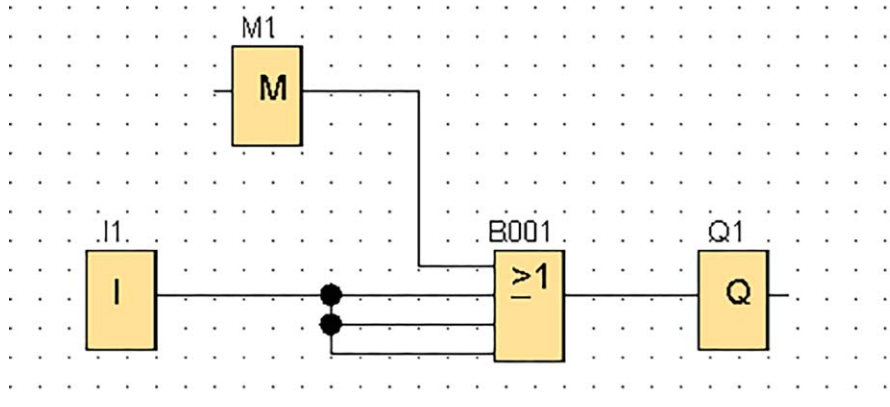
Zgodnie z tym, co pisaliśmy wcześniej, po zapisaniu w sterowniku tego programu, próby modyfikowania stanu wyjścia Q1 z poziomu wbudowanego serwera http spełzną na niczym – będzie on zależał wyłącznie od stanu na wejściu I1. Jeżeli wprowadzimy do programu pokazanego na rysunku 2 drobną modyfikację – dodając bramkę OR i flagę M – co pokazano na **rysunku 3**, uzyskamy możliwość modyfikowania stanu wyjściowego. W tej konfiguracji zapisanie do wybranej flagi M (w przykładzie była to flaga M1, lista dostępnych flag jest szersza – **rysunek 4**) logicznej „1” powoduje zwarcie styków przekaźnika Q1 do czasu, aż wartość wybranej flagi M zmieni się na logiczne „0”, co przywróci funkcję kopiowania stanu wejścia I1 na wyjście Q1 (czyli funkcjonowanie sterownika zgodnie z programem zapisanym w jego pamięci). W przypadku, gdy chcemy uzyskać efekt zdalnego blokowania możliwości zmiany stanu wyjścia przy rozłączonych stykach przekaźnika, program z rysunku 1 trzeba zmodyfikować w nieco inny sposób – bramkę OR zastępujemy bramką AND (**rysunek 5**).

W obydwu przedstawionych sytuacjach użytkownik uzyskuje zdalny, bezpośredni dostęp wyjść użytych w programie z poziomu wbudowanego serwera http, mogąc zmieniać dowolnie ich stan (stany, w przypadku większej liczby wyjść) niezależnie od wykonywanego programu. Podobnym modyfikacjom można poddać każdy program dla LOGO!, należy jedynie pamiętać, że jeżeli zależy nam na niezależnym od programu blokowaniu lub aktywowaniu wyjść sterownika, realizujące te funkcje bramki AND lub OR należy włączać jako ostatnie w łańcuchu bloków FB, tuż przed zaciskami wyjściowymi Q.

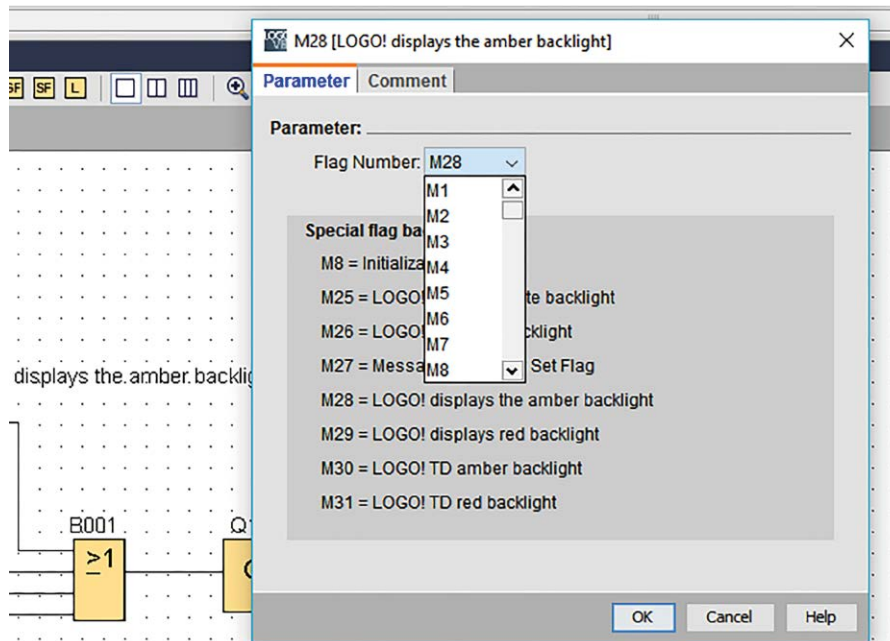
Serwer http wbudowany w LOGO! 8 można wykorzystać także do zdalnej parametryzacji programu, czyli modyfikowania np. wartości odmierzanych czasów przez timery użyte w programie. Do zilustrowania takiego przykładu użyjemy programu



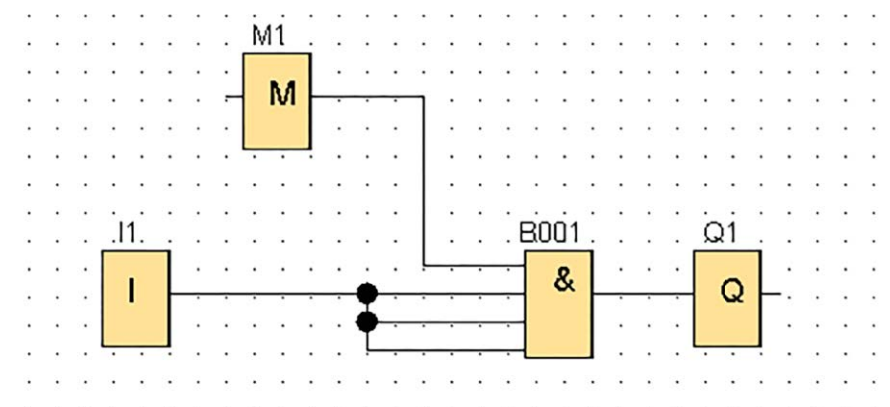
Rysunek 2. Przykładowy program umożliwiający zdalne blokowanie zmiany stanu wyjść



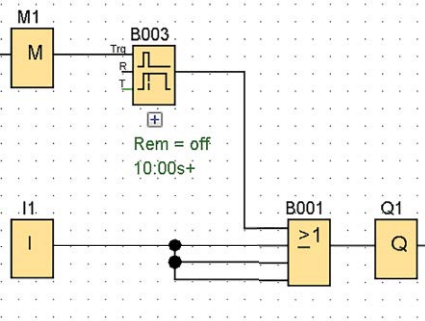
Rysunek 3. Przykładowy program po modyfikacji



Rysunek 4. Lista dostępnych flag



Rysunek 5. Przykładowy program po kolejnej modyfikacji



Rysunek 6. Program używany do demonstracji zdalnej parametryzacji

pokazanego na **rysunku 6**. Stan linii wyjściowej w prezentowanym rozwiązaniu jest zależny od stanu wejścia I1 i stanu wyjścia timera B003 z opóźnionym wyłączeniem. Jeżeli na wejściu I1 jest logiczna „1” lub wyjście timera jest aktywne, to Q1 przyjmuje wartość „1” – realizację funkcji OR zapewnia nam bramka B001. Timer B003 jest wyzwalany za pomocą flagi M1, która jest dostępna z serwera http, jak w poprzednim przykładzie. Z poziomu serwera http będziemy nie tylko wyzwalać odmierzanie czasu, ale także zmieniać jego wartość i odczytywać wartość czasu pozostałą do odmierzania.

W tym celu w projekcie została zdefiniowana tabela zmiennych dostępnych zdalnie (**rysunek 7**), w której zostały zawarte czasy

ID	Block	Parameter	Type	Address
1	B003 [Off-Delay]	Off-Delay	Word	0
2	B003 [Off-Delay]	Remaining Time	Word	2
3				

Rysunek 7. Tabela zmiennych dostępnych zdalnie

REKLAMA

Wszystko, co lubisz,
w jednym miejscu



UlubionyKiosk.pl

Oferuje papierowe i elektroniczne wydania czasopism z najważniejszych segmentów rynku:

budownictwo i wnętrza, muzyka i dźwięk, elektronika i automatyka, edukacja i hi-tech, rodzina.

Przesyłka
GRATIS

Off-Delay (domyślna wartość czasu podtrzymania „1” na wyjściu timera po zaniku sygnału wyzwalającego) oraz *Remaining Time* (ilość czasu pozostałego do końca podtrzymania „1” na wyjściu timera po zaniku sygnału wyzwalającego).

W oknie przeglądarki internetowej – po zalogowaniu się w serwerze http sterownika LOGO! – wybieramy LOGO! Variable i definiujemy listę dostępnych zdalnie zmiennych zgodnie z **rysunkiem 8**. Po ich zdefiniowaniu wykonujemy krok po kroku następujący test (**rysunek 9**, od góry do dołu):

- wpisujemy wartość M1=1 (flaga przyjmuje stan *true*) – wartości zmiennych VM0 (*Off-Delay*) i VM2 (*Remaining Time*) zwrótną się, stan wyjścia Q1=1,
 - wpisujemy wartość M1=0 (flaga przyjmuje stan *false*) – wartość zmiennej VM2 (*Remaining Time*) zmniejsza się do zera, po osiągnięciu „0” stan wyjścia Q1=0.
- Teraz zdalnie zmienimy wartość odmierzanego czasu z 222hex na 3FFhex. W tym celu wykonujemy kolejno czynności (**rysunek 10**, od góry do dołu):

- w polu *ModValue* zmiennej VM0 (*Remaining Time*) wpisujemy 3FF i naciskamy *Modify All Values*,
- zmienna VM0 przyjmuje wprowadzoną wartość,
- zainicjowanie odmierzenia czasu (poprzez wpisanie sekwencji „1” i następnie „0” do zmiennej M1) spowoduje (po wpisaniu „1”) przepisanie wartości z VM0 do VM2, następnie odmierzenie czasu zgodnego z nowo wprowadzoną wartością i adekwatną zmianą stanu wyjścia Q1.

W ten sposób można dokonywać zdalnej parametryzacji dowolnych parametrów bloków „wystawionych” na zdalny dostęp poprzez umieszczenie ich w tablicy VM Mapping (w programie LOGO! Soft Comfort).

Piotr Zbysiński, EP

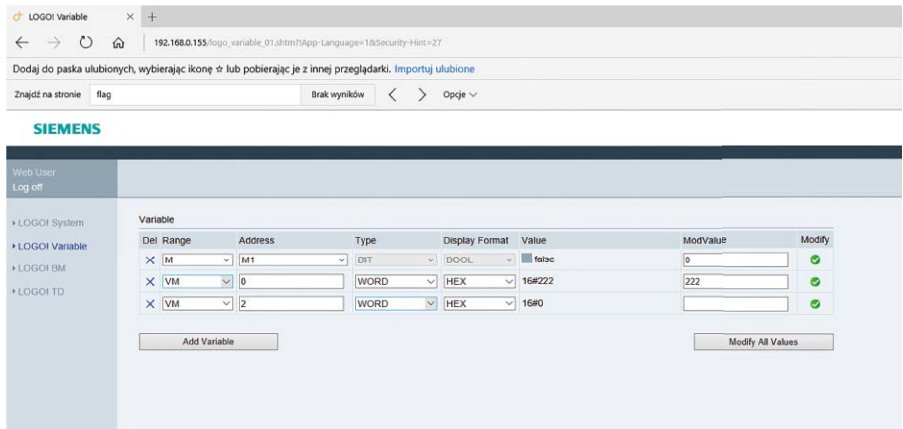
REKLAMA

<http://m.ep.com.pl>
Najlepszy Mobilny Adres w Sieci

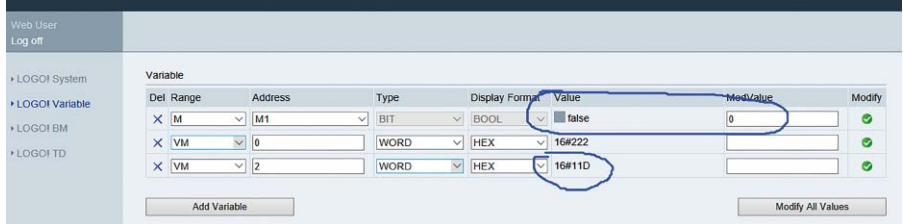
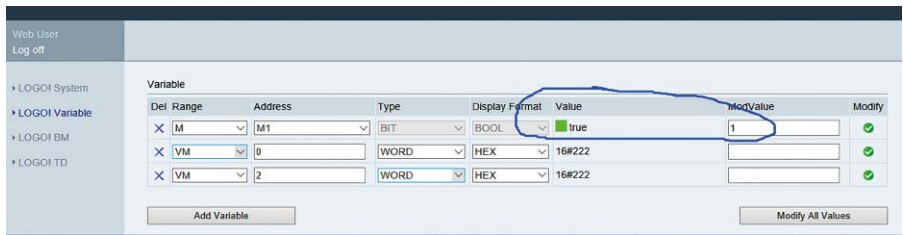
PODZESPOŁY

MAX14827 - nowy transceiver IO-Link

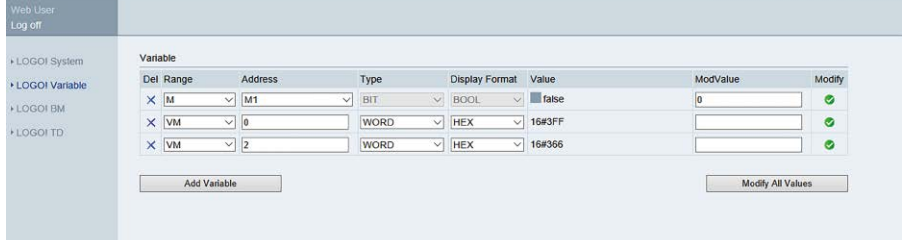
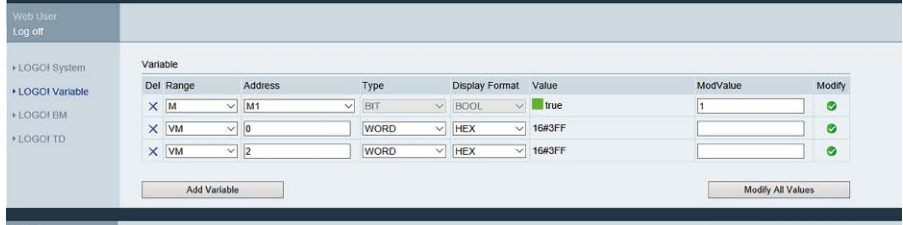
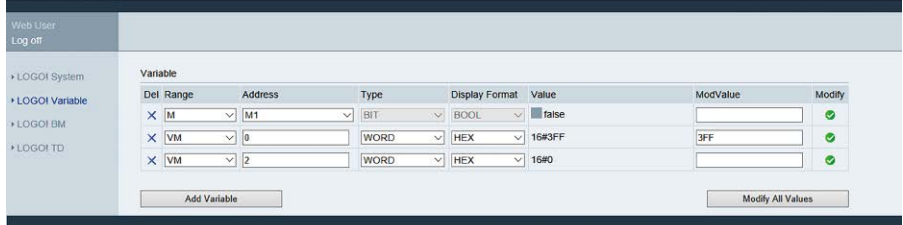
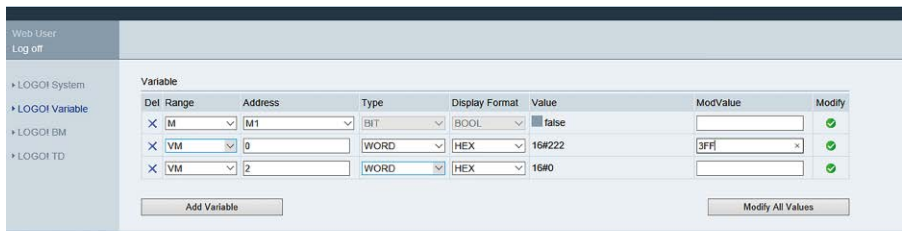
Maxim Integrated oferuje transceiver IO-Link w obudowie o powierzchni 2,5 mmx2,5 mm.



Rysunek 8. Definicja listy zmiennych



Rysunek 9. Test stanu zmiennych



Rysunek 10. Czynności wykonywane w trakcie testu