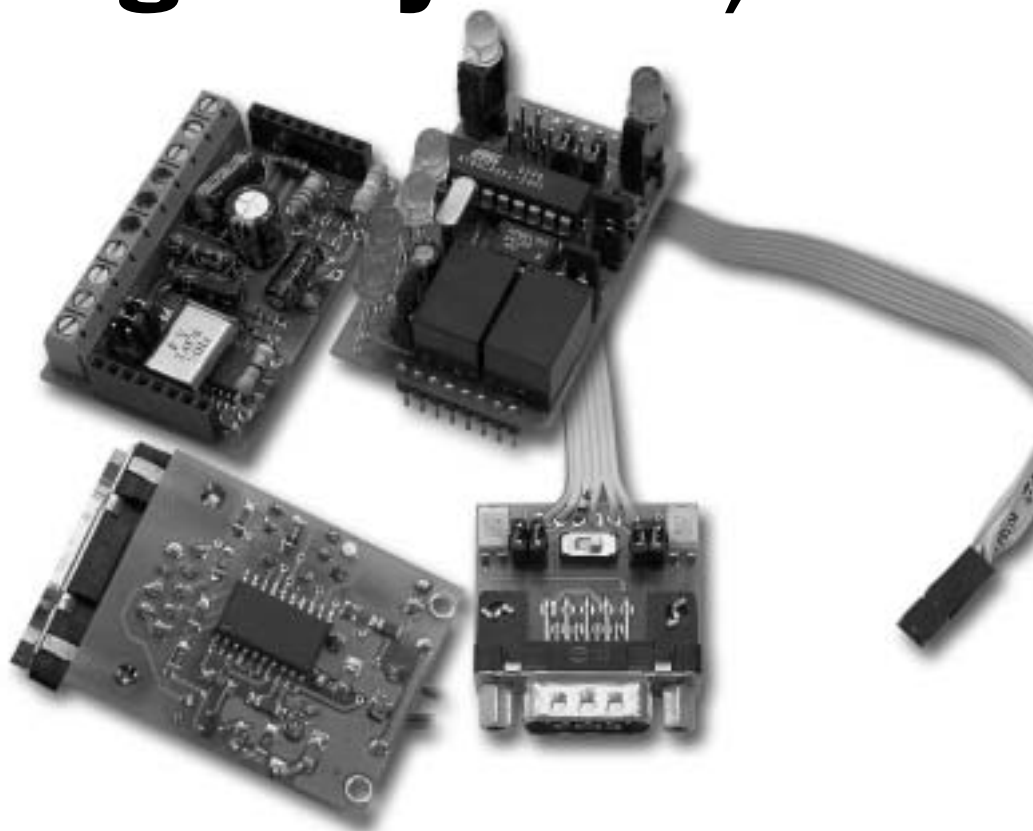


EIB - inteligentny dom, część 2



W drugiej części artykułu przedstawiamy pozostałe interfejsy przystosowane do współpracy w systemie EIB oraz oprogramowanie sterujące prostym systemem z magistralą EIB.

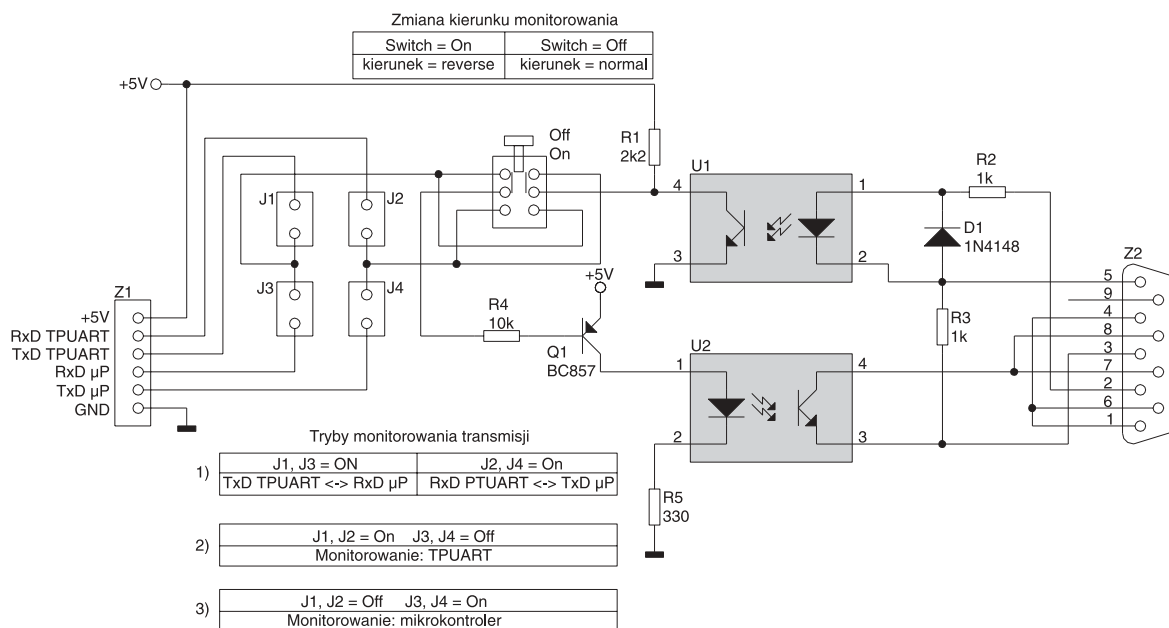
Rekomendacje: artykuł polecamy nie tylko Czytelnikom zainteresowanym szybką automatyzacją swojego domu, lecz także wszystkim pragnącym rozszerzyć swoją wiedzę o działaniu EIB.

Interfejs RS232/UART

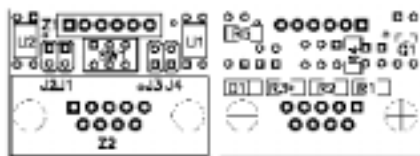
Dzięki użyciu interfejsu RS232/UART (schemat na rys. 8) uproszczona staje się diagnostyka układu. Umożliwia on monitorowanie za pośrednictwem komputera PC ruchu na styku mikrokontroler/TP-UART. Ponadto, interfejs umożliwia testowanie układów dołączonych do magistrali i wszelkich

innych urządzeń wykorzystujących podobne łącze szeregowe.

Przełącznik SW1 służy do zmiany rodzaju współpracujących łączy szeregowych. W pozycji *off* interfejs współpracuje z urządzeniami typu DTE (*Data Terminal Equipment*). Natomiast w pozycji



Rys. 8. Schemat elektryczny interfejsu RS232/UART



Rys. 9. Rozmieszczenie elementów na płytce interfejsu RS232/UART

on interfejs realizuje funkcje umożliwiające współpracę z urządzeniami typu DCE (*Data Communication Equipment*). Użycie zwor J1..J4 opisano na schemacie. Rozmieszczenie elementów na płytce drukowanej interfejsu RS232/UART pokazano na **rys. 9**.

Interfejs RS232/EIB

Chociaż system EIB ma architekturę zdecentralizowaną (brak sterowników nadrzędnych), powinien być wyposażony w przynajmniej jeden interfejs umożliwiający zdalną realizację dostępu do sieci, np. zewnętrznemu komputerowi PC. Dzięki temu jest możliwe dokonanie pierwotnej konfiguracji systemu i późniejszych rekonfiguracji elementów wchodzących w jego skład bądź np. monitorowanie działania poszczególnych elementów magistralnych systemu czy też sterowanie nimi. Interfejs RS232/EIB umożliwia przesyłanie komunikatów do sieci EIB i odbiór przepływających w medium tejże sieci. Najprostszym sposobem re-

WYKAZ ELEMENTÓW

Interfejs RS232/UART

Rezystory

- R1: 2,2kΩ 1206
- R2, R3: 1kΩ 1206
- R4: 10kΩ 1206
- R5: 330Ω 1206

Półprzewodniki

- D1: 1N4148 SMD
- Q1: BC857 SOT-23
- U1, U2: TLP621

Różne

- J1...J4: jumper
- SW1: mikroswitch 2-biegunowy
- Z1: DIP6
- Z2: gniazdo kątowe DB9M

WYKAZ ELEMENTÓW

Interfejs RS232/EIB

Rezystory

- R1: 2,2kΩ 1206
- R2...R4: 1kΩ 1206
- R5: 10kΩ 1206
- R6: 68Ω 0,5W
- R7: 330Ω 1206

Kondensatory

- C1, C2: 47nF 1206
- C3, C6: 10nF 1206
- C4: 6,8μF/16V
- C5: 100μF/16V

Półprzewodniki

- D1: 1N4148
- D2: 1N5819
- DZ1, DZ2: BZX85C39
- Q1: BC857
- U1, U2: TLP621
- U3: TP-UART01

Różne

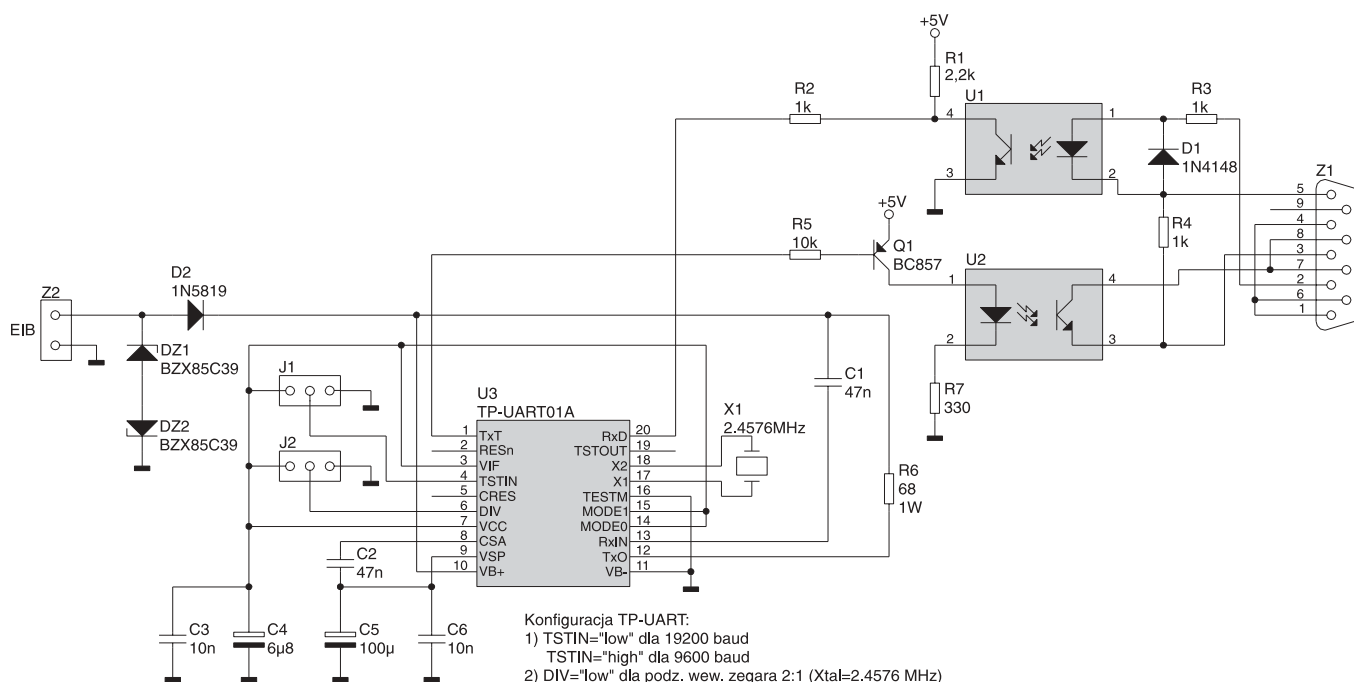
- J1, J2: jumper
- X1: kwarc 2,4576MHz
- Z1: gniazdo kątowe DB9M

alizacji takiego interfejsu jest wykorzystanie dedykowanego układu realizującego dostęp do sieci EIB. Takim układem jest na przykład TP-UART, wykorzystany również do budowy prezentowanego w artykule urządzenia typu EIS1 (załącz/wyłłącz).

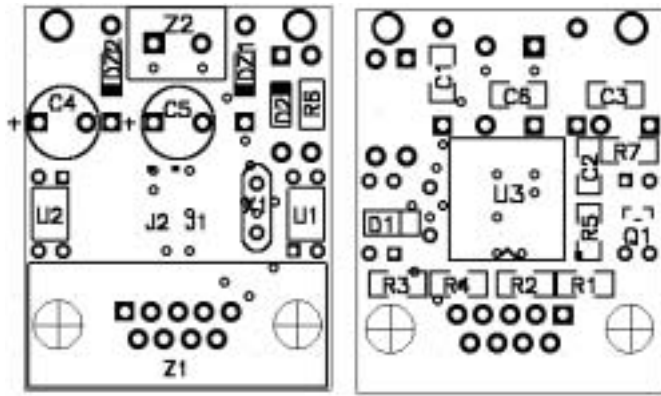
Na **rys. 10** pokazano schemat elektryczny interfejsu RS232/EIB. Rozmieszczenie elementów na płytce drukowanej interfejsu przedstawiono na **rys. 11**.

Za pomocą złączy śrubowych Z6, Z7 interfejs jest dołączany do magistrali EIB. Część układu związana z układem TP-UART jest zrealizowana w ten sam sposób, jak w prezentowanym w poprzedniej części artykułu urządzeniu

typu EIS1. Tym razem łącznik Z4 służy do określania prędkości transmisji obowiązującej na styku interfejsu szeregowego RS232/TP-UART. W przypadku, gdy za pomocą tego łącznika linia TSTIN połączona zostaje z GND, prędkość ta wynosi 19200 kbd, a w przeciwnym razie 9600 kbd. Niestety, na tym styku poziomy napięciowe sygnałów nie są zgod-



Rys. 10. Schemat elektryczny interfejsu RS232/EIB



Rys. 11. Rozmieszczenie elementów na płytce drukowanej interfejsu RS232/EIB

ne ze specyfikacją RS232. W celu konwersji poziomów TTL na poziomy RS232 wykorzystany został prosty układ z transoptorami U1, U2, które oprócz tej funkcji izolują zarazem interfejs galwanicznie od niekorzystnych oddziaływań zewnętrznych. Transmisja danych w kierunku do TP-UART-a odbywa się poprzez wyprowadzenie numer 3 złącza Z1 (DB9).

„Cisza“ na łączu RS232 charakteryzuje się poziomem wysokim. W takim przypadku tranzystor transoptora U1 zostaje wystawiany, a poziom na wejściu RxD TP-UART-a jest niski. Gdy transmitowana jest logiczna „1“, na linii TxD znajduje się w poziom niski. W takiej sytuacji dioda D1 zabezpiecza U1 przed uszkodzeniem spowodowanym polaryzacją w kierunku wstecznym. Rezystor R3 ogranicza prąd diody optoizolatora U1.

W przypadku transmisji w kierunku od TP-UART-a, dane przesyłane są za pośrednictwem tranzystora w transoptorze U2. Rezystor R4 powoduje podanie na emiter tego tranzystora potencjału masy. Natomiast kolektor fototranzystora U2 jest zasilany napięciem pochodzącym z linii RTS złącza RS232. Do prawidłowego działania układu konieczne jest podanie sygnału RTS od strony zewnętrznego interfejsu RS232 (od strony komputera klasy PC).

Kabel łączący interfejs z portem szeregowym PC-ta powinien być standardowym kablem typu DTE-DCE.

Dla zapewnienia prawidłowej wymiany informacji wystarczające jest wykonanie następujących po-

łączeń pomiędzy obu złączami łącza komunikacyjnego: [2-2][3-3][5-5][7-7][8-8].

Zasilacz

W systemie EIB napięcie znamionowe wynosi 24 VDC i jest dostarczane dwoma przewodami miedzianymi do wszystkich terminali. Są zarazem medium transmisyjnym sieci informacyjnej. Z tego powodu zasilacz (schemat pokazano na rys. 12) musi być wyposażony w elementy zabezpieczające przesyłane siecią dane przed zniekształceniami. W tym celu w obwód zasilania włączona została indukcyjność L1, która separuje układy wewnętrzne zasilacza od sygnału zmiennego przesyłanego tą samą linią. Wraz ze wzrostem obciążenia sieci napięcie zasilające maleje. Z tego powodu napięcie wyjściowe zasilacza przy całkowitym braku obciążenia jest wyższe od znamionowego i wynosi 28 V. Dzięki temu minimalne napięcie zasilające, przy którym pracują jeszcze prawidłowo elementy magistralne, jest równe 20 V. Rosnąca liczba elementów magistralnych powoduje niekorzystne obciążenie zarówno stałego napięcia zasilającego, jak i sygnału informacyjnego. Wartości obu tych napięć maleją w funkcji wzrastającej liczby urządzeń (większe obciążenie),

WYKAZ ELEMENTÓW

Zasilacz

Kondensatory

C1: 100nF/400V

C2...C5: 1nF/400V

C6: 100nF/63V

C7, C8: 1000MF/35V

Półprzewodniki

D1...D4: 1N4007

Różne

L1: 120mH

TR1: transformator sieciowy, np. TS40/027 24V/1,5A

Z1, Z2: złącze typu Terminal Block

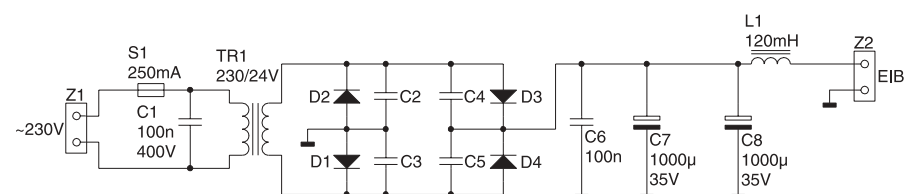
S1: bezp. bezwłocznym 250mA/250V

jak również w funkcji rosnącego wraz z odległością tłumienia toru transmisyjnego. Z powyższego wynika, że prąd zasilania nie może być tak duży, aby spadek napięcia zasilania był większy od dopuszczalnego, a poziom względny sygnału informacyjnego musi mieścić się w granicach umożliwiających jego poprawną interpretację przez najdalej oddalone terminale.

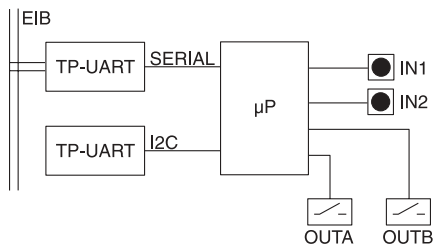
Opis programu mikrokontrolera

Sposób realizacji modułu aplikacyjnego oraz sposób interpretacji odebranych z sieci EIB danych jest zależny wyłącznie od wykonawcy danego urządzenia. Musi on jednak pamiętać o zachowaniu zasad związanych z określonym dla sieci EIB protokołem wymiany informacji, ponieważ jest on unormowany. W programie mikrokontrolera zrealizowano podstawowe funkcje potrzebne do zaprezentowania działania urządzenia z interfejsem do sieci EIB. Funkcje realizowane przez urządzenie są zależne od konkretnego zapotrzebowania.

Mikroprocesor (w module pokazanym na rys. 4, EP1/2004) obsługuje dwa wejścia binarne i dwa wyjścia binarne. Jest to więc urządzenie typu włącz/wy-



Rys. 11. Schemat elektryczny zasilacza



Rys. 13. Uproszczony schemat blokowy sterownika z interfejsem EIB

łącz. Jest zarazem zarówno podwójnym sensorem, jak i podwójnym elementem wykonawczym. Odpowiada więc jednemu ze zdefiniowanych w sieci EIB typów EIS (EIB *Internetworking Standards*). W tym przypadku jest to EIS1. Mikrokontroler za pośrednictwem magistrali I²C umieszcza w pamięci EEPROM dane niezbędne do poprawnej pracy urządzenia po zaniku zasilania. Są to: adres fizyczny, adres(y) grupowe i parametry, w zależności od których, zróżnicowane staje się działanie urządzenia. Urządzenie obsługuje funkcje takie jak zapis/odczyt adresu fizycznego w trybie programowania (po naciśnięciu przycisku programowania). Poza tym program mikrokontrolera umożliwia zapis/odczyt zawartości komórek pamięci spod wskazanego adresu oraz sterowanie stanem wskazanego obiektu. Przez obiekt należy rozumieć w tym przypadku jedno z wyjść binarnych bądź jedno z wejść binarnych (omawiane urządzenie obsługuje 4 zależne lub niezależne od siebie obiekty). To właśnie wyjście, jako niezależny obiekt może być załączone („1“) lub wyłączone („0“). Może ono należeć do jednej lub kilku grup i poza tym może reagować na komunikaty odebrane z sieci w niezależny od pozostałych obiektów sposób. Dla przykładu, jeżeli jedno z wyjść zostanie skonfigurowane jako element grupy dajmy na to 3/3/1 i zostanie odebrany telegram zaadresowany do tej właśnie grupy, to urządzenie to wykona program zależny od wcześniej określonej wartości parametru.

W szczególności, jeśli wartość parametru zapisanego w 1-bajtowej komórce pamięci, mieszczącej się pod adresem 0x0000, będzie równa 0, to dane wyjście zostanie

odpowiednio wysterowane (załączone). Jeśli natomiast wartość parametru będzie równa 1, to dane wyjście zostanie załączone dopiero po 5 sekundach od czasu odebrania polecenia pod warunkiem, że w międzyczasie nie został odebrany telegram wyłączający grupę, do której dany obiekt należy.

W przypadku, gdy wartość parametru będzie równa 2, wyjście natychmiast się załączy, lecz zostanie wyłączone dopiero po upływie 5 sekund. W omówionym przykładzie jednak dwa obiekty typu wyjściowego posiadają identyczny program z tym, jaki opisano powyżej.

Oba obiekty typu wejściowego natomiast wykonują jeden tylko program (nie został tu wykorzystany parametr), mianowicie jedno i drugie z wejść mono lub bistabilnych (zależnie od wartości parametru umieszczonego w komórce pamięci o adresie 0xff) wysyła po prostu sukcesywnie telegramy załączające/wyłączające grupę bądź grupy, do których należą. W szczególności mogą to być te same grupy. Jeden obiekt wejściowy (przełącznik mono lub bistabilny) może np. załączać/wyłączać jedną grupę, a drugi obiekt wejściowy może załączać/wyłączać wszystkie użyte w systemie grupy jednocześnie (każde z urządzeń zawierających obiekty należące do tych grup indywidualnie reaguje wykonaniem specyficznego programu).

Przy założeniu, że tablica adresów grupowych w pamięci EEPROM rozpoczyna się od adresu 0x0020 i posiada założoną długość 16 bajtów, dla jednego obiektu można określić 2 adresy grupowe, określające grupy, do których obiekt ten należy. Natomiast parametr określający wybór jednego z predefiniowanych programów dla pierwszego obiektu wyjściowego zapisywany jest pod adresem 0x0000, a dla drugiego pod adresem 0x0001.

Oczywiście, element magistralny powinien raczej umożliwiać zdalne załadowanie całego nowego programu, a nie jedynie dokonywanie wyboru jednego z możliwych programów do wykonania za pośrednictwem parametru. Oczywiście, taki program zawsze

musiałby odpowiadać dostępnej platformie sprzętowej, czyli takiej, na której możliwe byłoby jego wykonanie. W prezentowanej aplikacji wykorzystano sposób umożliwiający najprostsze wykonanie urządzenia, a więc wybór parametru. Uproszczony schemat blokowy przedstawiono na rys. 13.

Opis programu konfiguracyjnego

Komputer klasy PC wraz z zainstalowanym odpowiednim oprogramowaniem spełnia rolę stanowiska centralnego w sieci EIB. Możliwość oddziaływania na system z centralnego stanowiska nie oznacza, że w przypadku awarii komputera PC przestanie on funkcjonować prawidłowo. Taki komputer może być wykorzystywany w zależności od potrzeb, np. jako interfejs użytkownika z siecią, obejmujący funkcje wizualizacji i sterowania z centralnego stanowiska. Może być wykorzystywany np. przez stróża, jako interfejs przekazujący mu informacje o stanie systemu i pozwalający na reagowanie w zależności od sytuacji. Programem dedykowanym do zastosowania w urządzeniach objętych niniejszym projektem jest specjalnie do tego celu opracowany EIB_2IN2OUT_OnOff (dostępny na CD-EP2/2004B). Umożliwia on zdalne konfigurowanie elementów magistralnych.

Każdy terminal sieciowy EIB posiada unikalny adres fizyczny. Adres zapisywany jest w 16-bitowym słowie, przy czym najstarsze 4 bity określają adres strefy (obszaru), młodsze 4 bity określają adres linii (segmentu) danej strefy, a najmłodsze 8 bitów służą do określania poszczególnych terminali.

Sytuacja wynikająca z zastosowania w jednym, otwartym systemie EIB komponentów wytworzonych przez różnych producentów wymusza stosowanie określonych standardów umożliwiających ich wspólną pracę w jednym systemie. W związku z powyższym opracowano wspomniane wcześniej standardy EIS. Odpowiadają one poszczególnym typom obiektów automatyki systemu EIB. Obecnie istnieje 15 podstawowych typów EIS. Typy te definiują sposoby interpretacji danych przesyłanych

siecią i związane z tym sterowanie takimi obiektami, jak np.:

1. Element wykonawczy ZAŁĄCZ/WYŁĄCZ (najprostszym przykładem jest tu zwykła żarówka).
2. Sterowanie ściemniaczem lub falownikiem przy określaniu wartości procentowej (0...100%) zakresu bądź zmianie stanu o wartość względną np. + 1/128 część zakresu.
3. Obiekty sterowane w zależności od czasu rzeczywistego (dzień tygodnia, godzina, minuta, sekunda lub dzień miesiąca, miesiąc, rok).
4. Obiekty sterowane za pośrednictwem przesyłanych wartości zmiennopozycyjnych lub całkowitych.
5. Liczniki.
6. Obiekty wymagające sterowania zależnego od zadanego priorytetu.
7. Napędy (typowe wykorzystanie to sterowanie żaluzjami LEWO-STOP-PRAWO).
8. Zabezpieczenia dostępu.
9. Obiekty sterowane za pośrednictwem tekstów (przesyłanie rozkazów bądź informacji do wyświetlenia jako łańcuchów znaków).

Program EIB_2IN2OUT_OnOff pozwala na sterowanie urządzeniami typu EIS1 pochodzącymi od różnych producentów. Za pomocą tego oprogramowania można konfigurować adresy fizyczne dowolnych urządzeń w trybie rozgłoszeniowym. Natomiast konfigurowanie adresów grupowych i konkretnych parametrów wiąże się z zapisem do pamięci EEPROM odpowiednich wartości pod adresy wybrane przez programistę, z uwzględnieniem charakterystyki wykorzystanego układu 24C02 i ograniczonego zakresu dostępnych adresów.

Program EIB_2IN2OUT_OnOff (rys. 14) umożliwia odczyt/zapis adresu fizycznego urządzenia znajdującego się w trybie programowania. Ponadto, program ten umożliwia załączanie/wyłączanie dowolnej grupy obiektów. Komunikacja z terminalami sieci EIB odbywa się za pośrednictwem, także opisanego w artykule, interfejsu RS232/EIB poprzez jeden z portów szeregowych komputera PC. Bezpośrednio po uruchomieniu program sprawdza zajętość

portów COM1...COM4 przez pracujące w systemie Windows aplikacje i przejmuje pierwszy wolny. Zmiany wybranego portu można dokonać w dowolnym momencie z wykorzystaniem menu *Konfiguracja*. Praca z interaktywnym programem EIB_2IN2OUT_OnOff nie wymaga szczegółowego opisu, gdyż nieprawidłowości uniemożliwiające wykonanie wybranych funkcji są natychmiast sygnalizowane, np. w sposób przedstawiony na rys. 13. Konfigurację połączeń modelu przedstawiono na rys. 15.

Przykładowo, dla urządzenia skonfigurowanego zgodnie z rys. 13, każda zmiana stanu wejścia WE1 wpływa bezpośrednio na stan grup 0/0/1 i 0/0/2. Dzieje się tak za pośrednictwem telegramów wysyłanych przez element magistralny o adresie fizycznym 3.11.34 z rozkazem załączenia bądź wyłączenia tych grup. Obiekt wyjściowy WY1 reaguje jedynie na telegramy zaadresowane do grupy 0/0/2, także na wysyłane przez lokalne urządzenie, np. sukcesywnie wysyłane zależnie od stanu WE1, natomiast WY2 reaguje na telegramy zaadresowane do grupy 0/0/1, której elementem jest także obiekt WE1, jak również na telegramy zaadresowane do grupy 12/4/3. Takie telegramy mogą pochodzić jedynie od urządzenia zdalnego, jeśli takie istnieje. Obiekt WE2 wpływa na stan grupy 0/0/2 oraz na stan grupy 10/1/5, do której należeć powinny zdalne elementy magistralne.

Reasumując, każdy z obiektów zaimplementowanych w elemencie magistralnym jest niezależny od innych. Obiekty te łączą ewentualna przynależność do tej samej lub tych samych grup. Dla WY1 obiekty WE1, WE2 oraz inne, zaimplementowane w zdalnych urządzeniach i przynależne do grupy 0/0/2 niczym się funkcjonalnie nie różnią i są traktowane identycznie. W ten sposób skonfigurowane WE1 mogłyby być zaimplementowane w odrębnym urządzeniu i wpływałyby na stan WY1 w identyczny sposób jak w obrębie tego samego elementu magistralnego.

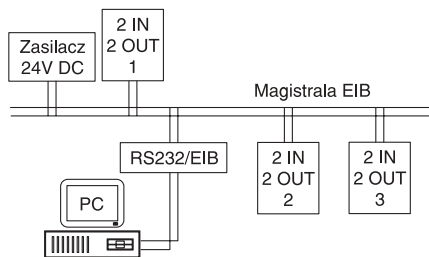
Jeżeli do WE2 podłączony byłby termostat, to po przekroczeniu zadanej temperatury obiekt ten



Rys. 14. Widok okna programu EIB_2IN2OUT_OnOff

transmitowałby do sieci telegram załączający grupy 0/0/2 i 10/1/5. W takiej sytuacji wyjście WY1 (przełącznik) w urządzeniu o adresie fizycznym 3.11.34 załączyłoby się z opóźnieniem, gdyż taki właśnie wykonuje ono program w reakcji na załączenie grupy, do której należy. Wyłączenie natomiast następuje natychmiast po otrzymaniu telegramu. Do tego wyjścia mogłaby zostać podłączona, np. syrena uruchamiana po przekroczeniu, w dopuszczalnym czasie trwania, wartości temperatury. Zanim jednak syrena załączyłaby się, obiekt wyjściowy należący do grupy 10/1/5, zaimplementowany w zupełnie odrębnym elemencie magistralnym, spowodowałby już, dajmy na to, załączenie pompy cieczy chłodzącej. Zastosowanie urządzeń z interfejsem EIB do tego typu sterowania nie nastęrcza żadnych trudności.

W aplikacji EIB_2IN2OUT_OnOff zaimplementowano jedynie niektóre z istniejących funkcji ACPI (*Application Control Programming Interface*) udostępnianych przez standard EIB. Są to głównie rozgłoszeniowy zapis/odczyt adresu fizycznego i zapis/odczyt wartości pod wskazany adres pamięci oraz załączanie/wyłączanie obiektów typu EIS1 (załącz/wyłącz) należących do określonej grupy. Standard EIB określa wiele innych funkcji konfiguracyjnych oraz przeznaczonych do sterowania i współpracy z obiektami różnych typów, np. analogowymi. Ważna

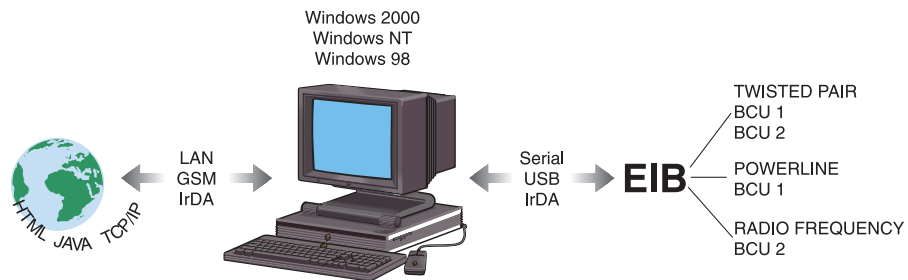


Rys. 15. Konfiguracja systemu modelowego

jest np. funkcja będąca zapytaniem o stan grupy. Jej zastosowanie łatwo znaleźć, np. w systemie wizualizacyjnym. Ponadto, standard EIB oddaje w ręce projektanta możliwość tworzenia własnych funkcji APCI, przeznaczonych do szczególnych zastosowań.

Zakończenie

Do systemu EIB można uzyskać dostęp poprzez wykorzystanie sieci teleinformatycznych innych standardów, w szczególności z wy-



Rys. 16. Interfejs EIB można łatwo zintegrować z innymi systemami sieciowymi (także Internetem)

korzystaniem linii telefonicznej i modemu. Z tego wynika możliwość uzyskiwania dostępu do sieci EIB i możliwość dokonywania zarówno odczytu żądanych wartości, jak i wpływanie na stan urządzeń za pośrednictwem sieci Internet o zasięgu globalnym, np. wykorzystując komputer klasy PC (rys. 16).

Robert Zduński i Witold Tonderski
robert.zdunski@wp.pl
witold.tonderski@wp.pl

Literatura:

[1] - „Interfacing the EIB Bus with Windows Computers“ G.Westermeier, T. Weinzierl, F. Schneider

Wzory płytek drukowanych w formacie PDF są dostępne w Internecie pod adresem: pcb.ep.com.pl oraz na płycie CD-EP2/2004B w katalogu PCB.