

Infinity – system automatyki domowej

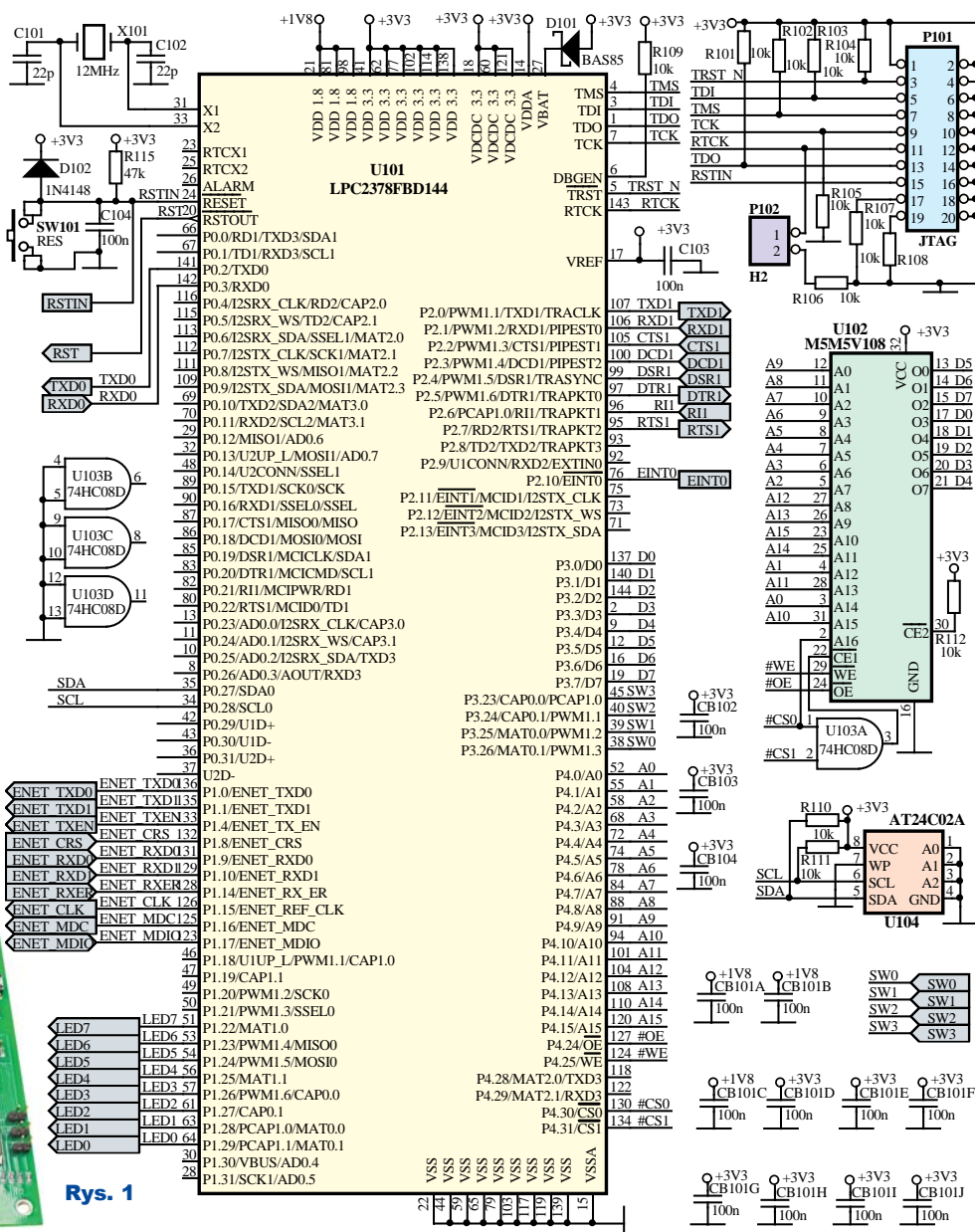
Mój własny serwer www



W grudniowym numerze EdW zamieszczony był ogólny opis systemu automatyki domowej, pracującego w moim domu. System jest rozbudowany, dlatego zgodnie z zapowiedzią poszczególne moduły zostaną opisane w kolejnych numerach EdW. Nie przejmuję się więc, jeśli nie wszystko zrozumiesz albo gdy nasuną Ci się pytania czy wątpliwości. Z czasem obraz się rozjaśni, a gdybyś pod koniec cyklu nadal miał jakieś pytania – napiszesz o tym do Redakcji lub wprost do mnie.

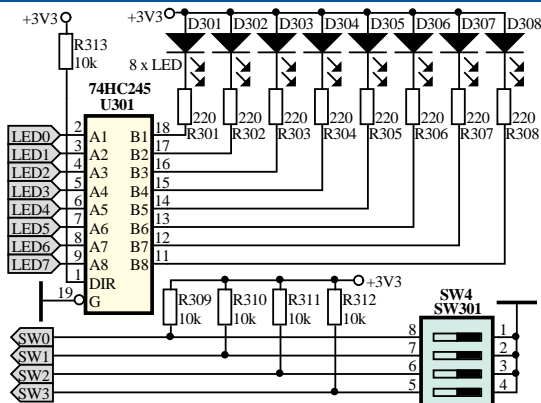
A teraz omówimy serce systemu: **serwer stron www**. Fotografia tytułowa pokazuje model, zmontowany specjalnie na potrzeby tego artykułu, natomiast **fotografia 1** przedstawia wcześniej zrealizowany egzemplarz. Serwer www oparty jest na mikrokontrolerze z rodziny ARM o symbolu LPC2378. Jego wybór jest podyktowany kilkoma względami, z których najważniejszym jest wbudowany blok do obsługi sieci ethernetowej. Dodatkowo bardzo istotną cechą tego mikrokontrolera jest w miarę proste wykorzystanie i pod tym względem posługiwanie się nim nie odbiega od obsługi popularnych mikrokontrolerów z rodziny AVR.

Jak widać, układ jest w sumie dość skomplikowany, ale łatwiej będzie zrozumieć jego budowę i działanie po podzieleniu



Fot. 1

Rys. 1

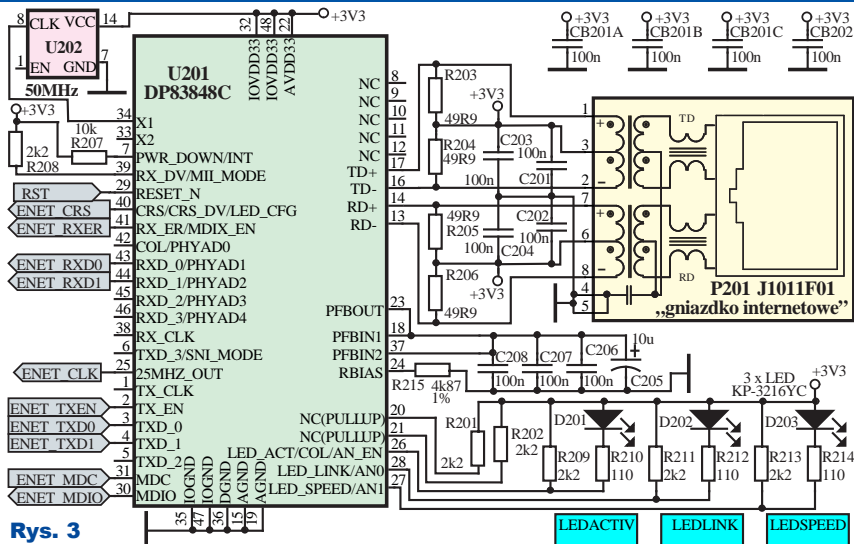


Rys. 2

schematu na części. Schemat jednostki centralnej serwera pokazuje rysunek 1. Rysunek 2 to blok kontrolek. Na rysunku 3 widać zespół współpracy z przewodową siecią komputerową LAN z typową skrętką komputerową. Na rysunkach 4, 5 pokazane są dwa interfejsy RS232. Rysunek 6 pokazuje schemat bloku zasilającego.

Wszystkie szczegóły działania urządzenia mogą wydawać się bardzo skomplikowane. Nawet sam opis konstrukcji może przestraszyć mniej zaawansowanych. Są jednak dobre wiadomości. Otóż wcale nie trzeba rozumieć wszystkich szczegółów, by taki układ wykonać (lub zakupić jako kit). Nie trzeba też być programistą. Wystarczy wgrać program (omówiony i udostępniony w Elportalu w kolejnym miesiącu). Potrzebny jest do tego program FLASHMAGIC i pełnomodemowy kabel RS232 (więcej informacji w dalszej części).

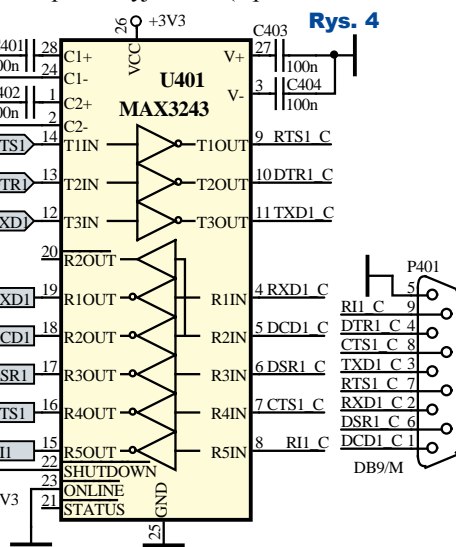
Omawianie układu zaczniemy od rysunku 1. Zastosowany mikrokontroler LPC2378 (U101) ma typową aplikację. Taktowany jest sygnałem zegarowym, używanym w wewnętrznym generatorze współpracującym z rezonatorem kwarcowym 12MHz. Jest to typowe w mikrokontrolerach rozwiązanie, gdzie obwód rezonansowy składa się z rezonatora (X101) oraz dwóch kondensatorów C101 oraz C102. W rzeczywistości częstotliwość taktowania mikrokontrolera jest większa, gdyż istnieje możliwość programowego włączenia wbudowanego układu PLL, którego zadaniem jest kilkukrotne zwiększenie częstotliwości sygnału zegarowego. Układ LPC2378 ma dodatkowy generator do odmierzania czasu rzeczywistego (RTC), ale w tym rozwiązaniu nie jest on używany. Elementy D102, R115 oraz C104 tworzą obwód, którego zadaniem jest wygenerowanie sygnału zerowania dla mikrokontrolera w momencie włączenia napięcia zasilającego. Dodany mikroprzełącznik SW101 pozwala na asynchroniczne wyzerowanie procesora w dowolnym momencie. Dodat-



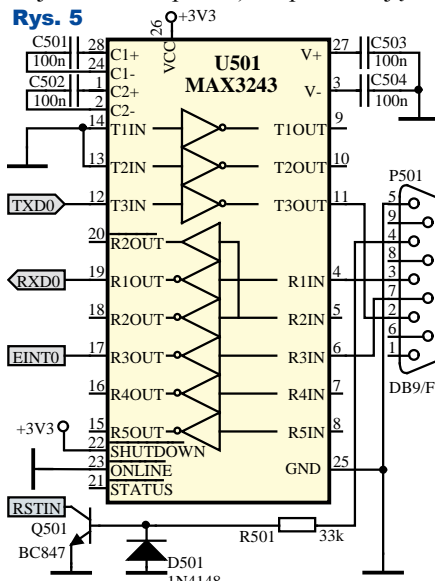
Rys. 3

kowo obwód ten (sygnał RSTIN) jest przyłączony do interfejsu RS232 związanego z portem UART0, którego schemat pokazuje rysunek 5. Wyjaśnienie dla dociekliwych i zaawansowanych: połączenie to jest niezbędne podczas operacji umieszczenia kodu programu w wewnętrznej pamięci FLASH. Taka możliwość jest określana jako programowanie w trybie IAP (programowanie w układzie aplikacyjnym). W tym trybie używany jest dodatkowo sygnał zewnętrznego przerwania EINT0, który również poprzez interfejs RS232 jest wyprowadzony na złącze szeregowe. Oba te sygnały (EINT0 oraz RSTIN) są sterowane przez wyjściowe (z punktu widzenia

komputera PC) linie modemowe (RTS i DTR). Te linie modemowe, obok transmitowanych danych TXD oraz RXD, to dodatkowe sygnały sterujące. Wykorzystywane są one przez wspomniany już wcześniej program FLASHMAGIC w operacji programowania (umieszczenia kodu programu w wewnętrznej pamięci FLASH). Program ten, dostępny w internecie, odpowiednio steruje liniami modemowymi, które poprzez układ interfejsu RS232 sterują liniami mikrokontrolera (sygnał RSTIN przyłączony do wejścia RESET, pin 24 oraz sygnał EINT0 przyłączony do wejścia EINT0 pin 76) i wprowadzają

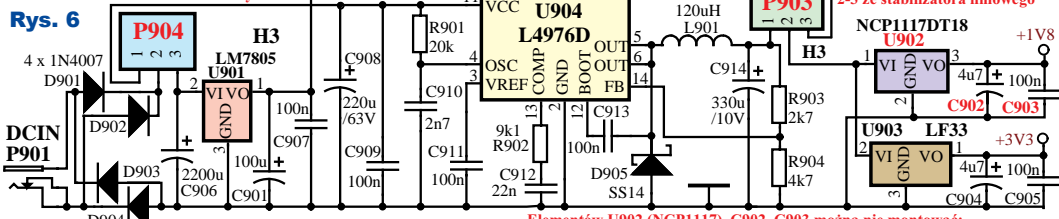


Rys. 4



Rys. 5

P904 Napięcie wejściowe podane: 1-2 na stabilizator impulsowy 2-3 na stabilizator liniowy



Rys. 6

P903 Zasilanie układu pochodzi: 1-2 ze stabilizatora impulsowego 2-3 ze stabilizatora liniowego

Elementów U902 (NCPI117), C902, C903 można nie montować; przy braku napięcia 1,8V mikrokontroler LPC2378 sam „wyprodukuje” potrzebne napięcie

mikrokontroler do trybu IAP. Kod programu, jako zawartość pamięci FLASH, jest przesyłany w dalszej kolejności jako dane szeregowo (sygnał TXD0 oraz RXD0). Wymienione sygnały tworzą kompletny interfejs niezbędny do zaprogramowania pamięci FLASH zawartej w strukturze mikrokontrolera. Tu potwierdzam sugestię nasuwającą się uważnym Czytelnikom, do zaprogramowania mikrokontrolera **nie jest potrzebny żaden programator** lub jakiegokolwiek inne urządzenie. Wymagany jest jedynie pełnomodemowy przewód do połączenia komputera z programowanym mikrokontrolerem ARM. Określenie „pełnomodemowy” oznacza, że zawiera on połączenie wszystkich styków złącza DB9 i jest to typowy element wyposażenia modemów telekomunikacyjnych. Jest zrozumiałe, że nie każdy sympatyk elektroniki musi mieć na wyposażeniu modem, by móc pożytyć z zestawu odpowiedni przewód, ale przewód taki można wykonać samodzielnie. Niezbędne materiały to dwa złącza typu DB9 (jedno męskie i jedno żeńskie) oraz kawałek wielożyłowego przewodu. W rzeczywistości, tę „pełnomodemowość” można okroić do przewodu 5-żyłowego i połączyć (bez przeplotów) jedynie te styki w złączach, które są używane przy programowaniu (jak na rysunku 5 styki o numerach 2, 3, 4, 5 i 7).

W fazie tworzenia oprogramowania przydatne jest złącze do interfejsu JTAG (P101 wraz z kilkoma rezystorami wymuszającymi właściwy stan pasywny na liniach JTAG w sytuacji, gdy nie jest on używany). Złącze to używa dużej liczby sygnałów, toteż do docelowego programowania mikrokontrolera ARM wykorzystywany jest tryb IAP. Nie można wykluczyć modyfikacji czy rozbudowy funkcjonalności programu realizowanego przez mikrokontroler już w eksploatowanym urządzeniu, toteż wyprowadzenie złącza kanału szeregowego wykorzystywanego w trybie IAP na panel przedni obudowy zwalnia nas z konieczności każdorazowego demontażu serwera lub otwierania jego obudowy. Zastosowany mikrokontroler ma możliwość dołączenia zewnętrznej równoległej pamięci RAM (U102). Do jej obsługi dedykowane są określone wyprowadzenia portu P3 i P4. W oparciu o sygnały używane przez mikrokontroler w chwili dostępu do zewnętrznej pamięci RAM wygenerowany jest dla niej sygnał wyboru w układzie kombinacyjnym, opartym na brankach logicznych (U103). W rozwiązaniu tym posługiwanie się zewnętrzną pamięcią RAM niczym nie różni się od używania wbudowanej pamięci operacyjnej. Obok pamięci statycznej, w układzie

przewidziana jest pamięć nieulotna (U104 jako AT24C02). Pamięć ta służy do przechowywania danych konfiguracyjnych. W przypadku popularnych mikrokontrolerów z rodziny AVR tego typu pamięć jest zintegrowana razem z mikrokontrolerem. Niestety w przypadku procesorów z rodziny LPC2000 tego typu pamięć nie występuje, toteż została dobudowana jako dodatkowy element. Zastosowany układ (U104, AT24C02) to popularna pamięć EEPROM z interfejsem I2C, stąd rezystory podciągające R110 oraz R111.

Kolejnym elementem wchodzącym w skład serwera www jest zespół do sygnalizacji stanów pracy procesora za pomocą diod LED. Ze względu na ograniczoną wydajność prądową wyprowadzeń portów mikrokontrolera diody LED są sterowane za pośrednictwem układu cyfrowego 74HC245 (U301, rysunek 2), który z kolei jest przyłączony do odpowiednich wyprowadzeń portu P1 mikrokontrolera LPC2378 (rysunek 1). Znaczenie poszczególnych diod wynika wyłącznie z oprogramowania, toteż ich funkcje będą opisane w kolejnej części. Dodatkowo do mikrokontrolera przyłączony jest zespół czterobitowego przełącznika (SW301) z rezystorami podciągającymi R309 do R312. Każda z linii niezależnie może być poprzez przełącznik (SW301) przyłączona do masy, co przy odczytach przez mikrokontroler daje wartość 0. Przy manipulowaniu poszczególnymi przełącznikami istnieje możliwość przekazania programowi serwera www określonych informacji, czy wręcz poleceń. Rozpoczynając pracę program mikrokontrolera ARM może przejść do realizacji swojej podstawowej funkcji lub zainicjować jakąś specjalną akcję jak przykładowo konfiguracja. Znaczenie poszczególnych przełączników wynika z oprogramowania i szerzej będzie omówione w kolejnej części.

Oprócz wielu typowych w mikrokontrolerach peryferii (jak porty, liczniki czy zegary), LPC2378 integruje w swojej strukturze zespół EMAC (odpowiednik komputerowej karty sieciowej) i wymaga jedynie dołączenia układu określanego jako PHY, realizującego dostęp do fizycznego łącza sieci komputerowej. Idea stosowania układów PHY sprowadza się do umożliwienia wykorzystania infrastruktury elektrycznej (połączenia kablowe) lub optycznej (połączenia światłowodowe) bez znaczących zmian w układzie. W moim rozwiązaniu komunikacja bazuje na przewodach miedzianych, gdyż dom jest okablowany skrętką ethernetową. Możliwość dostępu do serwera poprzez sieć Wi-Fi wynika wyłącznie z funkcjonalności routera. Sam serwer komunikuje się z otoczeniem siecio-

wym wyłącznie poprzez połączenie kablowe. Układ PHY, z punktu widzenia mikrokontrolera jako jego element zewnętrzny, komunikuje się z zespołem EMAC za pośrednictwem „znormalizowanego złącza”. Dla zaawansowanych i docieklivych: najczęściej ten styk mikrokontrolera z układem PHY występuje w dwóch wariantach określanych jako **MII** (ang. Media Independent Interface – interfejs niezależny od zastosowanego medium komunikacyjnego) lub **RMII** (ang. Reduced Media Independent Interface) jako zredukowany wariant MII (użyty w serwerze układ PHY umożliwia wariant interfejsu określanego jako **SNI**, który ma zastosowanie jedynie w sieci pracującej z prędkością 10MB, więc jest pominięty w opisie). W sytuacji, gdy ktoś chciałby zastąpić miedziane kable przykładowo interfejsem światłowodowym, to (poza oczywistą zmianą układu PHY na taki, który jest dopasowany do charakteru medium) nie wymaga to ingerencji w interfejs komunikacyjny pomiędzy układem PHY a zespołem EMAC. Różnica pomiędzy rozwiązaniem MII a RMII sprowadza się do liczby połączeń występujących na styku. Oprócz niezbędnych sygnałów taktujących i synchronizujących przesyłanie danych, w styku tym występują linie przesyłające transmitowane dane. W przypadku wariantu MII jednocześnie (jednym taktym zegarowym) przesyłane są cztery bity danych. W wariantcie RMII dane są przesyłane po dwa bity (dwa bity do nadawania oraz dwa bity do odbierania, co daje cztery linie magistrali danych pomiędzy EMAC a układem PHY, a w wariantcie MII konieczne jest użycie ośmiu linii).

Schemat ideowy przyłącza do sieci Ethernet pokazuje rysunek 3. Układem PHY jest popularny układ scalony o symbolu DP83848 (U201). Umożliwia on wariant MII, RMII i SNI, jednak ze względu na zastosowany mikrokontroler ARM, który oferuje jedynie wariant RMII, aplikacja układu PHY jest dostosowana do tego rozwiązania. Implikuje to użycie dwóch linii do przesyłania danych (sygnały ENET_TXD0 i ENET_TXD1 do nadawania oraz ENET_RXD0 i ENET_RXD1 do odbierania). Konsekwencją stosowania styku RMII jest również częstotliwość taktująca. By uzyskać prędkość transmisji sieci wynoszącą 100MB przy dwubitowej magistrali danych, wymagana częstotliwość generatora taktującego (U202) wynosi 50MHz. W układzie PHY wariant interfejsu może być wstępnie określony poprzez podanie odpowiedniego stanu na wyprowadzenie 39 układu DP83848. Gdyby to wejście pozostało nieprzyłączone (ma ono wbudowany rezystor do masy) lub

przylączone zostało do masy, zostałby włączony wariant MII. Wtedy wyprowadzenia o numerach 3, 4, 5 i 6 tworzą magistralę danych. W przypadku przylączenia rezystora R208 do napięcia zasilającego istotny jest dodatkowo stan na wyprowadzeniu 6 tego układu. Skoro nie jest to wariant MII (ze względu na sposób przylączenia R208), to wyprowadzenie 6 nie tworzy magistrali danych i może być użyte do sprzętowego określenia wariantu interfejsu (pozostawienie go jako nieprzylączonego oznacza skonfigurowanie interfejsu do wariantu RMII). Wybrany wariant (MII lub RMII) może być również zmieniany poprzez zapis do odpowiedniego rejestru układu.

Wszystkie operacje transmisyjne do i z sieci realizowane przez układ DP83848 są synchronizowane sygnałem zegarowym pochodzącym z generatora U202. Dotyczy to również synchronizacji wymiany danych z zespołem EMAC, toteż układ PHY niejako udostępnia własny sygnał zegarowy zespołowi EMAC (sygnał ENET_CLK). Sygnał ENET_TXEN aktywuje układ DP83848 do wysłania danych (stan wysoki na tej linii oznacza obecność ważnych danych). W operacjach odbierania danych z sieci Ethernet przez układ DP83848 wykorzystywane są dwie linie sygnalizacyjne. Sygnał ENET_CRS informuje o ważności odebranych danych (znajdujących się na dwubitowej magistrali danych) oraz ENET_RXER, który sygnalizuje wykrycie błędu transmisji. Sam układ interfejsu sieciowego jest konfigurowalny (ma wewnętrzne rejestry, których zawartość wpływa na jego działanie), toteż musi istnieć możliwość zapisu przez mikrokontroler odpowiednich danych do układu PHY. Jest to zrealizowane jako przesyłanie szeregowo w oparciu o dwa sygnały ENET_MDC (jako sygnał taktujący) oraz ENET_MDIO (jako sygnał danych). Tą samą drogą zrealizowany jest odczyt rejestrów statusowych z układu PHY. Tutaj mogą też uspokoić Czytelników dostrzegających konieczność złożonej programowej obsługi układu PHY. Nie, zespół EMAC w mikrokontrolerze realizuje to wszystko samodzielnie. Przykładowo zapis do określonego rejestru znajdującego się w układzie DP83848, a który jest zrealizowany jako komunikacja szeregowo bazująca na sygnałach ENET_MDC i ENET_MDIO, sprowadza się do zapisu odpowiednich kodów do właściwych rejestrów w zespole EMAC, który nakazaną czynność realizuje dalej samodzielnie. Podobnie przesłanie ramki ethernetowej jako bloku danych wymaga manipulacji rejestrami zespołu EMAC, który zlecone czynności wykona autonomicznie.

Jeśli nie rozumiesz szczegółów – nie przejmuj się – układ będzie działał nieza-

leżnie od stopnia Twojego zaawansowania. Więcej informacji na ten temat będzie z kolejnych częściach.

Istotnym elementem w obsłudze warstwy fizycznej jest zastosowanie odpowiedniego transformatora separującego. Nie występuje on jawnie na schemacie, gdyż jest zintegrowany jako jeden element z 8-pinowym gniazdem (P201) do sieci Ethernet, a sam transformator jest symbolicznie narysowany wewnątrz złącza. Pomiędzy układem PHY a transformatorem występuje sieć rezystorów oraz kondensatorów, która jest zgodna z wymaganiami opisanymi w dokumentacji układu DP83848. Istotnym elementem jest rezystor (R215) przylączone do wyprowadzenia RBIAS (pin 24). Wartość rezystancji jest precyzyjnie określona (4,87kΩ). Rezystor musi pochodzić z szeregu o tolerancji 1% lub lepszej. Zbyt duża odchyłka wartości rezystancji uniemożliwia komunikację. Zamiast tego rezystora można zastosować połączenie szeregowo dwóch rezystorów z szeregu o gorszej tolerancji (przykładowo 4,7kΩ i 170Ω), ale wtedy staje się konieczne wyselekcjonowanie rezystorów o mierze. Również bardzo istotnym elementem jest przylączenie kondensatorów C205...C208. Znaczenie ma nawet kształt ścieżek łączących wymienione elementy, toteż znalazło to swoje odbicie na schemacie w postaci „dziwnego” przylączenia kondensatorów (istotne szczegóły są opisane w odpowiedniej dokumentacji producenta układu DP83848). W mojej praktyce stosowania tego układu zdarzyło się raz, że układ nie uzyskał prędkości 100MB, a jedynie 10MB i jestem przekonany, że problem dotyczył topologii ścieżek (w opisywanym serwerze www sieć Ethernet działa bez problemów z prędkością 100MB).

Zadaniem układu PHY jest również sygnalizowanie poprzez diody LED aktualnego jego stanu pracy (D201, D202 i D203). Do dyspozycji są trzy sygnały, których znaczenie jest konfigurowalne (poprzez zapis do odpowiednich rejestrów w układzie DP83848). Wybrany wariant, determinowany w oprogramowaniu, ma następujące znaczenie: włączenie diody D201 oznacza wystąpienie aktywności sieci Ethernet (nastąpiło przesłanie danych), dioda D202 sygnalizuje stan LINK (fizyczne połączenie z inną stacją gotową do wymiany danych) oraz dioda D203 sygnalizuje prędkość transmisyjną (dioda włączona oznacza prędkość 100MB, wyłączona 10MB). Tu można zauważyć, że diody LED wraz z rezystorami ograniczającymi prąd płynący przez diody są połączone równolegle z dodatkowymi rezystorami (R209, R211 i R213). Rezystory te mogą być przy-

łączone do +3,3V albo do masy (GND) i w momencie wystąpienia aktywnego stanu sygnału zerującego dla układu PHY, ich stan determinuje wartość początkową niektórych rejestrów konfiguracyjnych (mogą być one później zmodyfikowane przez oprogramowanie mikrokontrolera ARM). Zastosowane rezystory wymuszają wstępnie tryb pracy układu PHY określonego jako autonegocjacja. Oznacza to, że układ ten musi się dostosować do istniejącej sieci (10MB lub 100MB). W ogólnym przypadku, poprzez zapis do odpowiednich rejestrów układu PHY, można go skonfigurować do pracy jedynie w sieci Ethernet pracującej z prędkością 10MB, do pracy jedynie w sieci pracującej z prędkością 100MB albo by się dostosował do pracy w sieci do jakiej został przylączone. Sieci 10MB powoli wychodzą z użycia i standardem stają się sieci Ethernet pracujące z prędkością 100MB.

Moduł serwera www wykorzystuje dwa kanały asynchronicznej transmisji szeregowo. Jeden z nich (rysunek 4), jako zawierający wszystkie linie modemowe, jest przeznaczony do wymiany danych z przylączoneymi modułami pomiarowymi lub wykonawczymi. Zastosowany jest tam układ o symbolu MAX3243 (U401) pozwalający na transmisję danych szeregowo oraz obsługę wszystkich linii modemowych. Wyjście układu jest przylączone do męskiego złącza DB9 (stanowi identyczne rozwiązanie funkcjonalne jakie występuje w komputerach PC). Umożliwia to przylączenie do serwera dowolnych urządzeń, które mogą wymagać wykorzystania linii modemowych. Współpraca na tym styku z procesorem komunikacyjnym, który nie wykorzystuje linii modemowych, należy traktować jako jedną z wielu możliwości. W przypadku rezygnacji z wariantu pełnomodemowego powyższe rozwiązanie może być uproszczone poprzez zastosowanie przykładowo układu MAX3232 (odpowiednik popularnego interfejsu MAX232 przystosowanego do pracy z zasilaniem 3,3V).

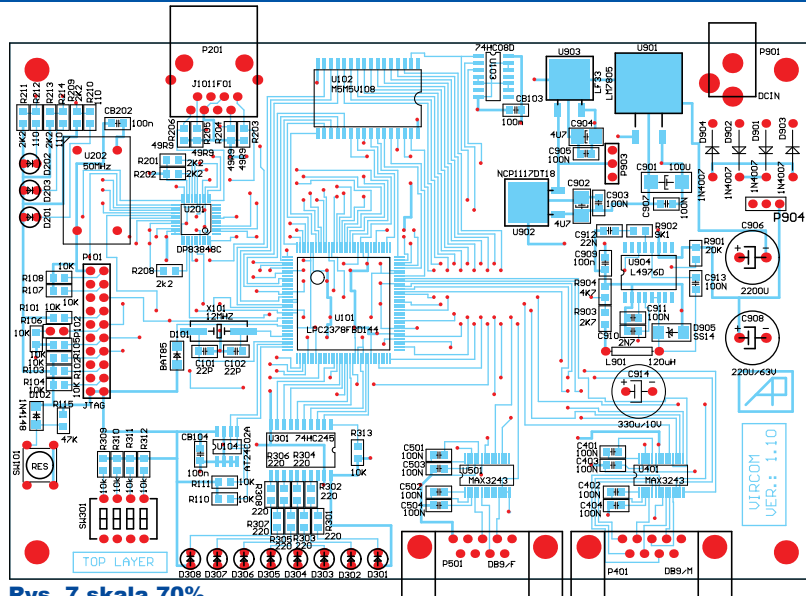
Drugi interfejs szeregowo (rysunek 5) przeznaczony do programowania mikrokontrolera (jako operacji ładowania kodu do jego pamięci FLASH) jest przystosowany do współpracy z programem FLASHMAGIC (specjalizowanym programem ładującym kod do pamięci FLASH mikrokontrolerów z rodziny LPC2000). Zastosowany jest tu identyczny układ scalony MAX3243 (U501) jak w interfejsie komunikacyjnym z rysunku 4. Jak było wspomniane wyżej, program FLASHMAGIC do ładowania zawartości pamięci FLASH oprócz danych szeregowo wykorzystuje

dwie linie modemowe. Dane szeregowe (nadawane oraz odbierane) są przetwarzane przez układ interfejsu w typowy sposób. Podobnie obsługiwana jest jedna z linii modemowych (sygnał EINT0). Druga linia modemowa, która ze złącza DB9 przekształca się na sygnał reset mikrokontrolera, ma odmienne rozwiązanie. Wynika to ze względu na potencjalne konflikty napięciowe w obwodzie generującym ten sygnał. Wymuszenie stanu aktywnego (zera logicznego) na tej linii może wynikać z ładowania kondensatora C104 przez rezystor R115 (rysunek 1), naciśnięcia przycisku SW101 (rysunek 1) lub wymuszenia wygenerowanego przez program FLASH-MAGIC jako odpowiednie wystereowanie linii modemowej (rysunek 5). Ponieważ układ interfejsu U501 nie ma wyjścia typu otwarty kolektor, funkcjonalność ta została odtworzona na bazie elementów dyskretnych (R501, D501, Q501). W interfejsie RS232 na liniach występują napięcia o wartościach +10...+12V lub -10...-12V. Układ interfejsu przetwarza sygnał o dodatnim napięciu na sygnał zera logicznego. Podobnie sygnały o ujemnym napięciu są konwertowane do stanu logicznej jedynki. W powyższym układzie tranzystorowym wystereowanie Q501 sygnałem o napięciu dodatnim powoduje nasycenie tranzystora, co sprawia, że na jego kolektorze wystąpi napięcie bliskie 0V. W sytuacji, gdy na linii wystąpi napięcie ujemne, tranzystor będzie zatkany a dioda D501 będzie chronić bazę tranzystora przed niewłaściwą polaryzacją. W tej sytuacji napięcie na kolektorze będzie ustalone przez elementy C101 i R115. Układ tranzystorowy również poprawnie się zachowa w sytuacji, gdy kabel RS232 będzie odłączony (tranzystor Q501 nie będzie wystereowany). Ten interfejs szeregowy zakończony jest żeńskim złączem DB9. Pozwala to na zastosowanie standardowych przewodów RS232, gdyż w trakcie ładowania kodu do pamięci FLASH moduł serwera logicznie odpowiada roli modemu podłączonego do komputera (jest urządzeniem podręcznym). Należy pamiętać, że interfejs RS232 jest niesymetryczny (zawiera trzy linie wyjściowe oraz pięć linii wejściowych, co implikuje, że występuje strona określana jako wyposażenie komputerowe oraz strona określana jako wyposażenie modemowe). Nie wyklucza to możliwości użycia tego kanału szeregowego do funkcji spełnianej przez wcześniej opisany interfejs (przewidywany do komunikacji z modułami pomiarowymi), jednak wymaga to wykonania właściwego przewodu połączeniowego. Kabel musi być bez linii modemowej (odpowiedni stan linii modemowej spowoduje reset mikrokontrolera) oraz wymaga-

ne jest skrzyżowanie linii transmitowanych danych.

Podobnie jak w przypadku interfejsu RS232 z rysunku 4, użyty układ MAX3243 może być zastąpiony układem MAX3232.

Ostatnim elementem serwera www jest zasilacz (rysunek 6). Rozwiązanie jest zaprojektowane w dwóch wariantach: z zastosowaniem liniowego i impulsowego stabilizatora napięcia typu 7805 (U901) oraz stabilizatora impulsowego o symbolu L4976 (U904). Jest wybór pomiędzy prostotą rozwiązania (stabilizator liniowy) a sprawnością (impulsowy). Mając na uwadze wymóg ciągłej pracy urządzenia, zalecane jest rozwiązanie impulsowe, ale mniej doświadczeni adepci elektroniki obawiający się kłopotów z uruchomieniem impulsowego stabilizatora mają wybór w postaci aplikacji znanego i popularnego rozwiązania (w takim przypadku montaż elementów związanych z układem U904 nie jest konieczny). Do odpowiedniego skonfigurowania przeznaczone są dwa zespoły zwojek (P902 oraz P903). Wstępnie napięcie zasilające urządzenie jest obniżone do wartości +5V (wszystkie moduły automatyki są zasilane przez zasilacz +12V). Finalne napięcie zasilające o wartości 3,3V jest wytwarzane przez układ U903. Sam mikrokontroler LPC2378 wymaga dwóch napięć zasilających (drugie ma wartość 1,8V), jednak w przypadku braku tego napięcia (1,8V) mikrokontroler może je sam wytworzyć.



(rysunek 6, **fotografia 2**), by zabezpieczyć cały serwer przed uszkodzeniem w przypadku, gdyby w wyniku pomyłki przy montażu stabilizator nie wytworzył właściwego napięcia zasilającego i zmierzyć napięcie na wyjściu stabilizatora impulsowego (bądź liniowego). Wartość napięcia powinna być w granicach 5...6V. Przy poprawnych wartościach napięcia można zasilić układ, zakładając odpowiednio zworek na złączu P903. Po przyłączeniu modułu do sieci (switcha 100MB), nawet bez załadowanego programu do pamięci FLASH mikrokontrolera, powinny zaświecić się dwie diody LED sygnalizujące prędkość transmisji oraz dioda sygnalizująca stan LINK (**fotografia 3**), co pokazuje, że układ PHY jest wstępnie skonfigurowany przez odpowiednie rezystory do trybu autonegociacji prędkości komunikacyjnej oraz został fizycznie połączony z innym urządzeniem sieciowym.

Mikrokontroler trzeba zaprogramować. Ja używam wspomnianego wyżej programu **FLASHMAGIC**. Program ten można bezpłatnie pobrać ze strony <http://www.flashmagictool.com/download.html&d=flashmagic>. Aktualnie jest dostępna wersja 11.20. Instalacja przebiega typowo i nie powinna sprawić jakiegokolwiek problemu. Po uruchomieniu (**rysunek 9**) należy pełnomodowym przewodem połączyć port COM komputera ze złączem P501 serwera. Uwaga! oryginały rysunków-zrzutów 9...15 dostępne są w Elportalu.

Program ten jest wstępnie właściwie skonfigurowany (w zakresie param-

trów sterowania liniami modemowymi) i nie należy tego modyfikować. Jedyne elementy do modyfikacji dotyczą przede wszystkim wyboru portu komunikacyjnego, wskazania pliku w formacie Intel-hex zawierającego kod programu do zapisania w pamięci FLASH mikrokontrolera. By zaprogramować mikrokontroler, należy kliknąć na przycisk **Select**, co spowoduje wyświetlenie listy obsługiwanych mikrokontrolerów, wybrać model użytego mikrokontrolera (**rysunek 10**). Rzecz oczywista, należy wybrać mikrokontroler LPC2378 (w przypadku wybrania niewłaściwego, oprogramowanie zwróci uwagę, że wybrana pozycja nie odpowiada użytemu w projekcie i należy dokonać korekty).

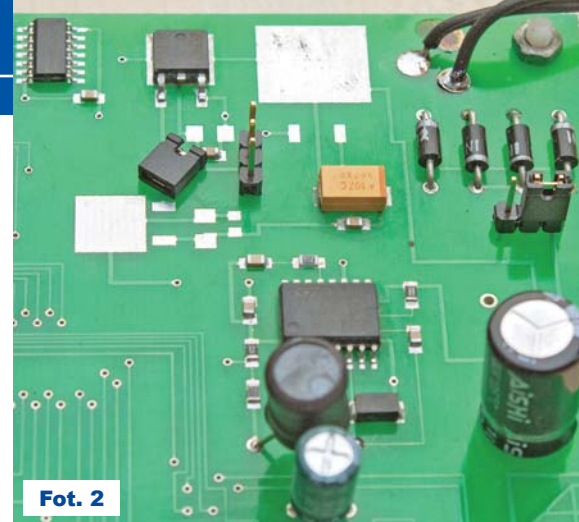
W kolejnym kroku program wymaga informacji określającej port COM użytego do komunikacji z procesorem. W moim przypadku jest to port COM1 i taką wybieram z zaproponowanej listy (**rysunek 11**).

W pozycji **Baud Rate** określam prędkość transmisji poprzez wybór jednej z zaproponowanej wartości. Naturalne jest, że wybór większej prędkości implikuje krótszy czas operacji, jednak nie jest zalecane „śrubowanie” parametrów prędkościowych, gdyż rośnie prawdopodobieństwo przekłamań. Z dotychczasowej praktyki wnioskuję, że prędkość 38 400 bps jest optymalna (**rysunek 12**). Pozycja **Interface** jest przeznaczona do określenia stosowanego interfejsu. W tym konkretnym przypadku nie jest stosowany żaden, więc pole to musi zawierać wartość **None**

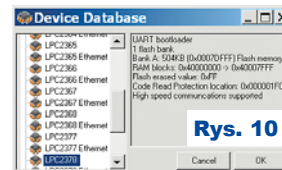
(ISP), **rysunek 13**.

W okienku **Oscillator** wpisuję częstotliwość rezonatora kwarcowego użytego do taktowania mikrokontrolera. W tym projekcie mikrokontroler jest taktowany sygnałem zegarowym o częstotliwości 12MHz, toteż wpisuję liczbę 12 000 000. W pozycji **File** jest możliwość wskazania pliku zawierającego kod programu do załadowania do pamięci FLASH, toteż po kliknięciu przycisku **Browse** powstaje możliwość wskazania właściwego pliku. Należy wyklikać właściwą kartotekę i wybrać wymagany plik w formacie hex (**rysunek 14**).

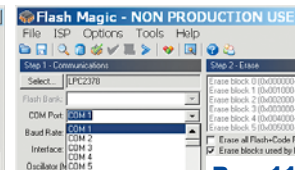
Warto mieć zaznaczone „ptaszki” w pozycjach **Erase block used by firmware**, co oznacza, że pamięć FLASH



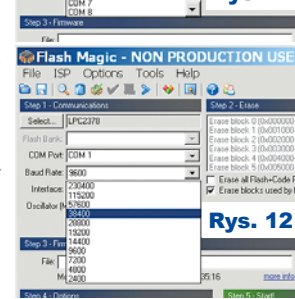
Fot. 2



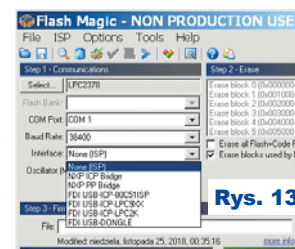
Rys. 10



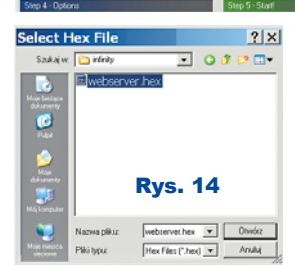
Rys. 11



Rys. 12

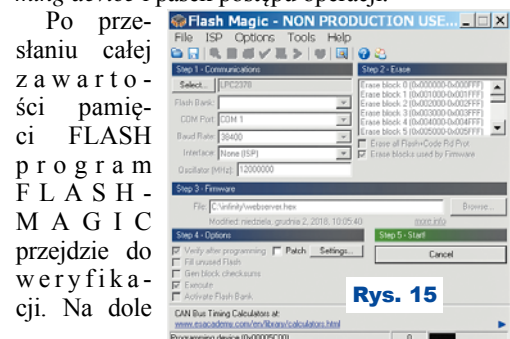


Rys. 13

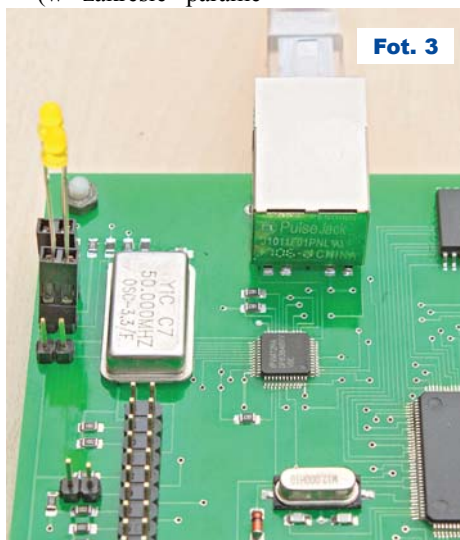


Rys. 14

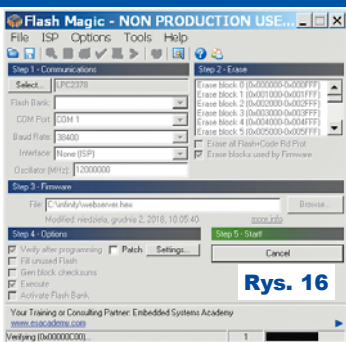
na mikrokontrolerze zostanie skasowana przed jej zaprogramowaniem w niezbędnym obszarze oraz **Verify after programming**, co implikuje, że po załadowaniu kodu programu do pamięci program FLASHMAGIC dokona weryfikacji poprawności operacji (odczyta zawartość pamięci FLASH i porówna z oryginałem). Proces programowania uruchamiany jest w wyniku kliknięcia na przycisk **Start**. W pierwszej kolejności zostanie skasowana pamięć FLASH i po jej zakończeniu nastąpi przesłanie kodu programu do mikrokontrolera (**rysunek 15**). Na dole okienka pojawi się komunikat **Programming device** i pasek postępu operacji.



Rys. 15



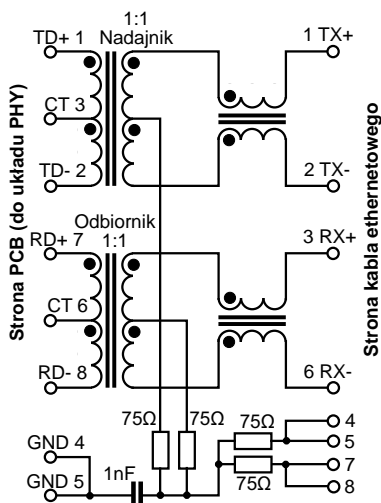
Fot. 3



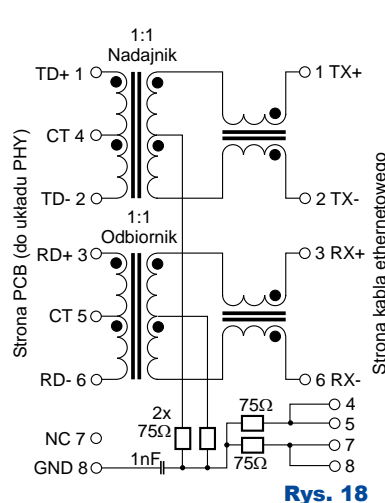
Rys. 16

okienka pojawi się napis *Verifying* wraz z paskiem postępu (rysunek 16). Po jej zakończeniu mikrokontroler jest zaprogramowany i gotów do działania z nowym programem.

Na zakończenie chcę poruszyć tematykę ewentualnych zamienników do użytych komponentów elektronicznych. W przypadku pamięci RAM nie ma bezwzględnej konieczności zastosowania układu wymienionego na liście użytych elementów. W trakcie opracowywania schematu dokładnie taki „znajdował się w mojej szufladzie” i z tego powodu trafił do schematu. Wspomniany układ jest statyczną pamięcią RAM o pojemno-



Rys. 17



Rys. 18

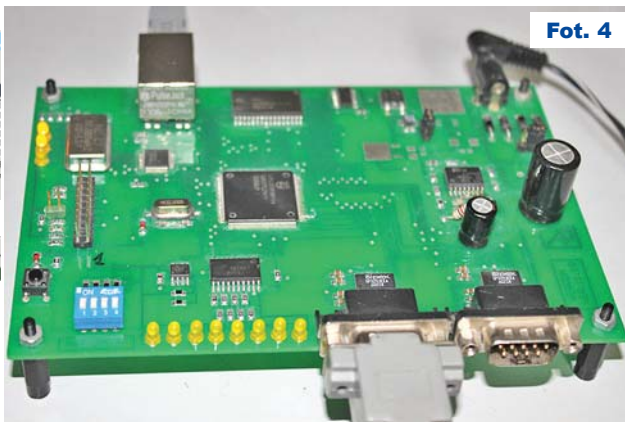
ści 128k*8 i w rzeczywistości może być zastąpiony dowolnym innym kompatybilnym układem (przykładowo: CY61228, AS6C1008). Pozornie gorzej przedstawia się zamiana gniazda do sieci Ethernet z zintegrowanym wewnątrz transformatorem impulsowym. Dystrybutorzy elementów elektronicznych mają w swoich ofertach wiele elementów, które mogą być zastosowane, jednak istnieje tu pewien problem, mianowicie występuje różny rozkład wyprowadzeń. Nawet w sytuacji, gdy złącze mechanicznie pasuje, musi zawierać identyczną funkcjonalność poszczególnych wyprowadzeń. W konstrukcji zastosowane jest złącze o symbolu J1011F01 i takie musi zostać użyte. W swojej dotychczasowej praktyce z powodzeniem wypróbowałem kilka innych, jak przykładowo J0011D21 lub HY91116C, jednak ich użycie wymaga przeprojektowania mozaiki ścieżek na płytce drukowanej. W przypadku złącza J1011F01 tor nadajnika

związany jest z wyprowadzeniami o numerach 1, 3, 2, a tor odbiornika z wyprowadzeniami 7, 6 i 8 (rysunek 17). Inne możliwe złącze (J0011D21, tego samego producenta) ma tor nadajnika na wyprowadzeniach 1, 4, 2 oraz tor odbiornika na wyprowadzeniach 3, 5, 6 (rysunek 18). Nie jest możliwe wymienienie wszystkich zamienników. W gruncie rzeczy złącze do sieci Ethernet z punktu widzenia układu PHY jest zestawem dwóch małych transformatorów impulsowych. Jednym z parametrów każdego transformatora jest jego przekładnia napięciowa. W dokumentacji do układu DP83848 znajduje się informacja, że wymagana przekładnia wynosi 1:1 zarówno dla toru nadajnika, jak i odbiornika. Poza oczywistym wymogiem, by transformator w złączu był przystosowany do transmisji z prędkością 10MB i 100MB, przekładnia jest praktycznie jedynym kryterium, jakie muszą spełniać zintegrowane ze złączem transformatory impulsowe.

Fotografia 4 pokazuje wcześniejszy model podczas testów. W Elportalu wśród materiałów dodatkowych do tego artykułu można znaleźć dokumentację płytki, rzuty ekranu oraz rysunki i fotografie.

W następnym miesiącu kolejna część będzie poświęcona wyjaśnieniu podstawowych pojęć związanych z siecią oraz będzie opisana implementacja programu serwera www (łącznie z udostępnieniem kodu programu).

Andrzej Pawluczuk
apawluczuk@vp.pl



Fot. 4

Wykaz elementów

R101-R112, R207, R309-R313	10kΩ (0805)
R115	47kΩ (0805)
R201, R202, R208, R209, R211, R213	2,2kΩ (0805)
R203, R204, R205, R206	49,9Ω (0805)
R210, R212, R214	110Ω (0805)
R215	4,87kΩ (tolerancja 1%, 0805)
R301-R308	220Ω (0805)
R501	33kΩ (0805)
R901	20kΩ (0805)
R902	9,1kΩ (0805)
R903	2,7kΩ (0805)
R904	4,7kΩ (0805)
C101, C102	22pF (0805)
C103, C104, C201-C204, C206-C208, C401-C404, C501-C504, C903 (opcja), C905, C907, C909, C911, C913, CB101A-J, CB102-CB104, CB201A-CB201C, CB202	100nF (0805)
C205	10μF

C901	100μF
C902 (opcja), C904	4,7μF
C906	2200μF
C908	220μF
C910	2,7nF (0805)
C912	22nF (0805)
C914	330μF (niskoimpedancyjny)
U101	LPC2378FBD144 (LQFP144)
U102	M5M5V108 (S032)
U103	74HC08D (SO14)
U104	AT24C02D (SO8)
U201	DP83848CVV (LQFP48)
U202	generator 50MHz + 3,3V
U301	HC74C245D (SO20)
U401, U501	SP3243EEA (SSOP28)
U901 (opcja)	LM7805CD2T (D2PAK)
U902 (opcja)	NCP1117DT18 (DPAK)
U904	L4976D (SOL16)
U903	LF33CDT (DPAK)

D101	BAS85 (MINIMELF)
D102, D501	1N4148 (MINIMELF)
D201-D203, D301-D308	diody LED
D901, D902, D903, D904	1N4007
D905	SS14 (DO214)
Q501	BC847 (SOT23)
X101	12MHz
L901	120μH
P201	J1011F01
SW101	mikroprzycisk
SW301	DIP switch czterobitowy
P401	złącze D-SUB 9 pin męskie, kątowe
P501	złącze D-SUB 9 pin żeńskie, kątowe
P101	GOLDPIN dwurzędowe 10x2
P102	GOLDPIN 2 pin + jumper
P903, P904	GOLDPIN 3 pin + jumper

Komplet podzespołów z płytką jest dostępny w Sklepie AVT jako zestaw AVT3232