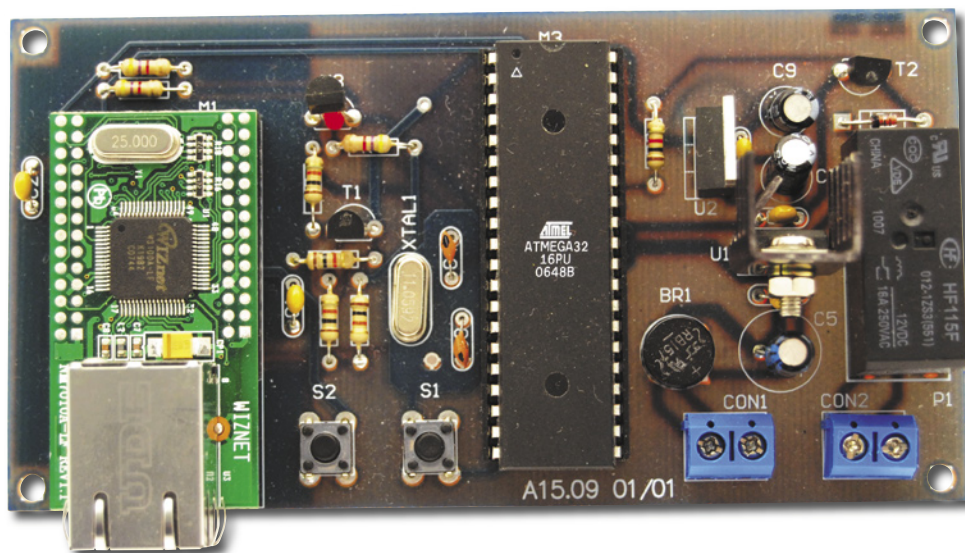




Serwer HTTP

Opisywany projekt może posłużyć jako bazy do realizacji wielu różnych urządzeń kontrolowanych poprzez sieć Internet lub Ethernet. Jest klasycznym przykładem użycia prostych i ogólnie dostępnych narzędzi do realizacji ambitnego zadania. Autor zaimplementował **pomiar temperatury i sterowanie funkcją włącz/wyłącz** za pomocą strony WWW, jednak korzystając z opisywanego rozwiązania, można zbudować urządzenie służące do realizacji znacznie bardziej zaawansowanych zadań i kontroli wielu różnych parametrów.

Rekomendacje: serwer HTTP może stanowić inspirację dla konstruktorów projektujących urządzenia i instalacje kontrolowane zdalnie, np. dla inteligentnych budynków.



Sieć Internet jest dziś prawie w każdym domu, a większość z nas posiada stały dostęp do Internetu. Dlaczego zatem nie wykorzystać tego ogólnie dostępnego medium do komunikacji z mikrokontrolerem? Podłączając go do sieci uzyskuje się wygodną, dwukierunkową komunikację niemalże z dowolnego miejsca na świecie. W prezentowanym urządzeniu mikrokontroler pełni rolę serwera http. Wartości zmierzonej temperatury i przyciski sterujące wyświetlane są na stronie WWW. Zastosowanie przeglądarki stron internetowych jako interfejsu użytkownika, zwalnia użytkownika z konieczności instalowania i poznawania dodatkowych programów komunikacyjnych. Dzięki łatwej, intuicyjnej obsłudze, z układu może korzystać każdy, kto umie poruszać się po Internecie. Na wygenerowanej przez urządzenie stronie internetowej wskazywane są temperatury minimalna i maksymalna oraz wyświetlane są przyciski umożliwiające wyzerowanie temperatur, i sterowanie obciążeniem dołączonym do przekaźnika.

Budowa

Schemat serwera HTTP przedstawiono na rys. 1. Jego sercem jest mikrokontroler ATmega32 firmy Atmel, pracujący z zewnętrznym rezonatorem kwarcowym. Do połączenia mikroprocesora z siecią Ethernet zastosowano moduł NM7010A firmy Wiznet, zawierający wszystko to, co jest potrzebne do umożliwienia Atmedze komunikowania się za pomocą protokołu TCP/IP. Producent wyposażył go w gniazdo RJ-45 oraz 2 diody LED informujące o stanie interfejsu. Moduł komunikuje się z otoczeniem za pomocą interfejsu I²C, co nie jest bez znaczenia, ponieważ konieczne jest użycie tylko dwóch doprowadzeń portów. Napięcie zasilania modułu jest równe 3,3 V, jednak na doprowadzeniach interfejsu I²C dopuszczalne jest używanie napięć o poziomach TTL, bez konieczności stosowania dodatkowych układów dopasowujących poziomy napięć. Układ scalony W3150A umożliwia otwarcie 4 niezależnych gniazd komunikacyjnych (sockets). Gniazda dzielą pomiędzy siebie bufor danych o wielkości 16 kB. Powód, dla którego wybrano właśnie ten moduł komunikacyjny był taki, że kompilator Bascom ma gotowe do użycia funkcje jego obsługi, co znacznie skraca czas od pomysłu do realizacji.

Komunikacja przy pomocy I²C wymaga, aby każde urządzenie podłączone do interfejsu posiadało swój adres. Tak jest również w przypadku użytego modułu komunikacyjnego. Jego adres można ustalić za pomocą doprowadzeń oznaczonych jako A8...A14 (A14 to bit bardziej znaczący), na które należy podać albo stan niski zwierając je do masy, albo stan wysoki podając poprzez rezystor zasilający o wartości 10 kΩ napięcie 3,3 V. W opisywanym urządzeniu adres modułu ustalono na 80H, czyli binarnie 10000000. Oczywiście w zależności od po-

AVT-5166

W ofercie AVT:
AVT-5166A – płytką drukowaną • AVT-5166B – płytką drukowaną + elementy

PODSTAWOWE PARAMETRY

- Napięcie zasilania: 12 VDC
- płytka dwustronna, z metalizacją, o wymiarach (118×69) mm
- komunikacja: Ethernet (moduł NM701A)
- oprogramowanie w Bascom AVR
- mikrokontroler ATmega 32

PROJEKTY POKREWNE

| Tytuł artykułu | Nr EP/EdW | Kit |
|---|-------------------|----------|
| Internetowy interfejs dla mikrokontrolera | EP 3-5/2002 | AVT-5055 |
| „Wyświetlacz” ethernetowy | EP 12/2007-1/2008 | AVT-5118 |
| Uniwersalny interfejs internetowy | EP 4-5/2006 | AVT-927 |
| Karta przekaźników z interfejsem Ethernet | EP 2/2007 | AVT-966 |
| Uniwersalny interfejs ethernetowy | EP 1/2007 | AVT-1443 |
| Karta wejść z interfejsem Ethernet | EP 10/2006 | AVT-953 |

resu przez przeglądarkę, tylko jest przesyłana w sposób niewidoczny dla użytkownika. Dzięki temu nie ma możliwości podejrzenia hasła przez osoby postronne oraz nie jest ono zapisywane w historii przeglądanych stron. Przykład zapytania metody POST podano niżej (może on nieznacznie różnić się w przypadku różnych przeglądarek):

POST /index.htm HTTP/1.1

Host: 127.0.0.1

User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows 98; BCD2000)

Content-Length: 7

Content-Type: application/x-www-form-urlencoded

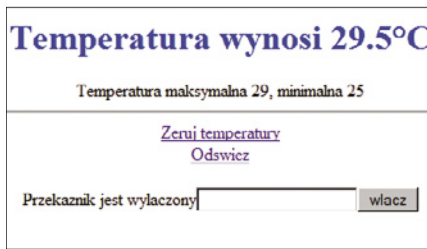
Pwd=123

Linia *Content-Length* zawiera liczbę informującą o długości w bajtach przesyłanej zawartości (w tym przypadku hasła) z wyłączeniem części nagłówkowej. Linia z hasłem nie zawiera znaku końca wiersza, dlatego należy ją pobierać bajt po bajcie. W zależności od przeglądarki, po linii *Content-Length*, wysyłana jest pusta linia, a po niej zmienna lub jeszcze kilka linii innych informacji. Problem kompatybilności z różnymi przeglądarkami rozwiązano w ten sposób, że serwer pobiera kolejne wiersze czekając na linię *Content-Length*. Z niej pozyskuje informacje o liczbie bajtów do pobrania. Następnie sprawdzana jest liczba bajtów w buforze odbiorczym, a linie pobierane są tak długo, aż zawartość bufora będzie o 6 bajtów większa, niż długość zmiennej *Pwd*. Zależnie od przeglądarki, długość linii zawierającej zmienną może różnić się i stąd te dodatkowo pobierane 6 bajtów, co doskonale sprawdza się w praktyce. Gdy w buforze znajdzie się pożądana liczba bajtów, są one kolejno pobierane i zamieniane na zmienną typu *string* (tekst). Linie pomiędzy *Content-Length* a zmienną *Pwd* nie zawierają informacji ważnych z punktu widzenia aplikacji i są przezeń ignorowane.

Opis programu

Na początku programu umieszczono deklarację stałych, które będą używane przez program np. do sterowania otwarciem gniazda komunikacyjnego. Jako kolejne umieszczono instrukcje wprowadzające nastawy interfejsu TCP/IP. Są to `Config Tcpip = Int0 , Mac = 12.128.12.34.56.78 , Ip = 192.168.0.50 , Submask = 255.255.255.0 , Gateway = 192.168.2.1 , Localport = 5000 , Tx = $55 , Rx = $55 , Twi = &H80 , Clock = 400000.`

Pierwszy parametr określa, z którego przerwania będzie korzystał mikrokontroler podczas



Rys. 2. Strona WWW opisywanego serwera

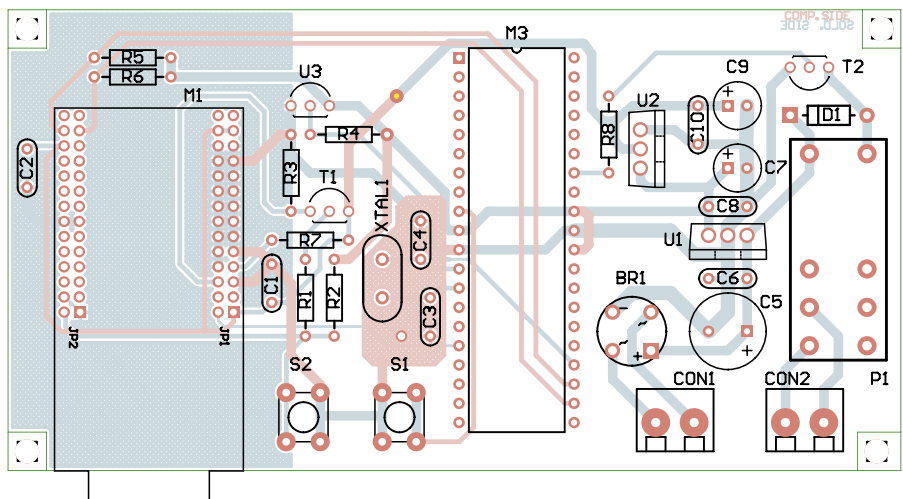
współpracy z modułem. Kolejne parametry określają: numer MAC układu W3100A i jego adres IP, maskę podsięci i adres bramy. Parametr *Localport* zawiera informację o domyślnym numerze portu, który posłuży do otwierania gniazd komunikacyjnych. Dopuszczalna jest zmiana parametru poza częścią inicjującą programu, w której można podać dowolną jego wartość, którą można później zmienić. Parametry Tx i Rx definiują rozmiary buforów transmisyjnego i odbiorczego. Układ W3100A jest wyposażony w 4 gniazda komunikacyjne i pamięć o wielkości 16 kB, po 8 kB na transmisję i odbiór. Do programisty należy właściwy podział tej przestrzeni. Każde gniazdo komunikacyjne zdefiniowane jest przez 2 bity parametru. Dwa młodsze bity dotyczą gniazda nr 0, a najstarsze gniazda nr 3. Wartość 00 ustala rozmiar bufora na 1 kB, 01 – 2 kB, 10 – 4 kB a 11 – 8 kB. W tym ostatnim przypadku możliwe jest użycie tylko pojedynczego gniazda o numerze 0. Ten sposób podziału dotyczy zarówno bufora odbiorczego, jak i transmisyjnego. Przypomnijmy, że każdy z nich ma rozmiar 8 kB.

Jako kolejne ustawiane są parametry interfejsu I²C, to jest: adres modułu Ethernet i szybkość transmisji (parametr *clock* ustalający częstotliwość zegara SCL). Standardowa

częstotliwość zegarowa szybkiej transmisji I²C wynosi 0,4 MHz i taką częstotliwość podano jako parametr w programie.

Aby interfejs pracował poprawnie wymagane jest zezwolenie na przerwania. W Bascomie służy do tego instrukcja `Enable Interrupts`. Po załączeniu przerwań program przechodzi do pętli głównej. Na początku wykonywane jest sprawdzenie, czy gniazdo komunikacyjne jest zamknięte. Jeśli tak, to jest ono przez program otwierane z użyciem portu 80 i ustawiane w tryb nasłuchu. Pierwszy parametr komendy `Getsocket` określa numer gniazda, kolejny numer portu, a ostatni opcje dodatkowe. Wartość 0 oznacza brak dodatkowych opcji.

W kolejnym kroku program sprawdza, czy w buforze odbiorczym znajdują się jakieś dane. Jeśli jest on pusty, rozpoczynany jest pomiar temperatury. Układ DS18B20 zastosowany w urządzeniu był już wielokrotnie opisywany na łamach Elektroniki Praktycznej i dlatego pominięto opis części programu zawierający obsługę odczytu temperatury. Jeśli bufor zawiera ważne dane, to rozpoczyna się analiza zapytania wysłanego przez klienta. Program sprawdza w jaki sposób rozpoczyna się zapytanie i odnajduje ciąg znaków HTTP. Pomiędzy ciągiem a nagłówkiem odnajdywana jest nazwa żądanego zasobu. Jeśli zostanie rozpoznana metoda GET, rozpoczyna się przesyłanie strony internetowej. W przypadku metody POST, zanim klient otrzyma kod strony, zostanie pobrane hasło i porównane ze zmienną programu o nazwie *haslo*. Gdy wynik porównania będzie pomyślny, to następuje zmiana stanu przekaźnika i powrót do strony głównej. W przeciwnym przypadku użytkownik zostanie poinformowany o błędnym hasle przy pomocy



Rys. 3. Schemat montażowy

List.1. Kod źródłowy strony głównej w języku HTML

```
<head><meta http-equiv=refresh content=40 /></head>
<hr><center><b><font size=6 color=blue>Temperatura wynosi 21°C</b></font></center>
<br><center>Temperatura maksymalna 23 minimalna 20</center>
<hr><center><a href=zeruj>Zeruj temperatury</a></center>
<center><a href=index.htm>Odswiez</a></center>
<br><form action=index.htm method=post >Przekaznik jest Wlaczony
<input type=password name=pwd>
<input type=submit value= włącz</form>
```


komunikatu wyświetlanego na odrębnej stronie internetowej.

Strona główna

Kod źródłowy strony głównej, który jest wysyłany do przeglądarki przedstawiono na list. 1.

Pierwsza linia jest odpowiedzialna za automatyczne odświeżanie strony. Parametr `content` określa częstotliwość odświeżania w sekundach. Kod źródłowy może być zmieniony, jednak wymagane jest pozostawienie bez zmian następujących wartości:

- Adres wywoływany w celu wyzerowania musi pozostać `zeruj`.
- Adres, do którego odwołuje się formularz, musi pozostać `index.htm`, a nazwa pola do wprowadzania hasła `pwd`.

Inaczej może być konieczna modyfikacja programu sterującego.

Montaż urządzenia

Urządzenie zmontowano na dwustronnej płycie drukowanej (rys. 3). Montaż rozpoczyna się od rezystorów i kondensatorów, a kończy

na podstawie mikrokontrolera. Wraz z modulem dobrze jest go umieścić w podstawce, by uniknąć uszkodzenia podczas lutowania. Mikrokontroler należy skonfigurować tak, aby pracował z zewnętrznym rezonatorem kwarcowym. W tym celu, przy pomocy programatora, należy nadać bitom `cksel3...cksel0` wartość logiczną 1.

Jan Jaczewski
dzej89@o2.pl

R E K L A M A

KONKURS WIĘCEJ ŚWIATŁA

Zwróć uwagę na projekt
Oświetlenie w kabinie
samochodu – strona 58

Diody superluminescencyjne mają tak intensywne światło i są tak tanie (w ofercie www.sklep.avt.pl), że każdemu elektronikowi serce się rwie żeby coś fajnego z nimi zrobić, np. światła dyskotekowe, podświetlenia, dekoracje, reklamy świetlne, światła ostrzegawcze itd. I Ty powinienes coś zrobić. Włącz się. Weź udział w nieustającym konkursie WIĘCEJ ŚWIATŁA. Zrób zdjęcia uzyskanych efektów świetlnych i zamieść je na specjalnej stronie konkursowej (wiecejswiatla.ep.com.pl). Opisy i schematy układów prześlij pod adresem redakcji redakcja@ep.com.pl.

NAGRODY: za materiał zdjęciowy umieszczony na wiecejswiatla.ep.com.pl dostaniesz darmową prenumeratę 2 numerów EP (prenumeratom przedłużamy ich prenumeratę o 2 gratisowe numery). Opisy i schematy układów mają szansę publikacji na łamach EP – honorarium 250 zł/stronę.