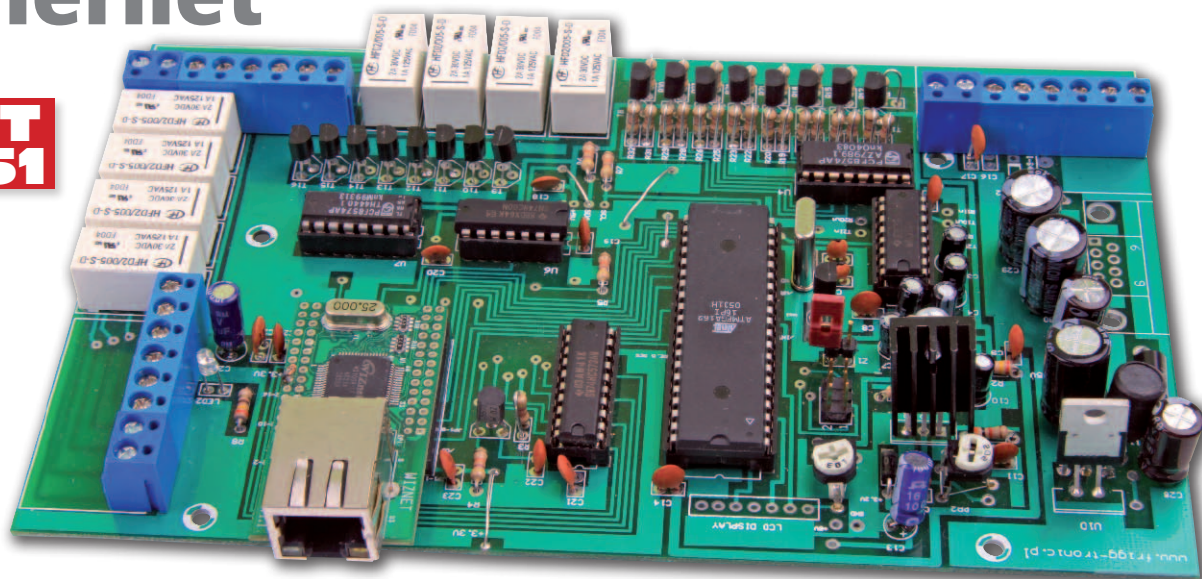


Sterownik urządzeń domowych

Sterowanie poprzez sieć Ethernet


**AVT
5251**


Sterownie urządzeniami poprzez sieć Intranet czy Internet jest ostatnio jednym z tematów coraz częściej poruszanych na łamach Elektroniki Praktycznej. Zaprezentowane rozwiązanie wyróżnia się interfejsem graficznym, który pozwoli na łatwe posługiwanie się sterownikiem nawet pięcioletkowi.

Rekomendacje: osoby zajmujące się automatyzacją budynków, majsterkowicze zainteresowani poprawą funkcjonalności swoich domów i mieszkań.

Układy z wykorzystaniem modułów firmy Wiznet (7000A, 7010) prezentowane były już przynajmniej trzykrotnie. Oczywiście każdy z tych projektów różnił się zarówno konstrukcją jak i „technologią” połączenia (serwery Telnet, Web itp.) Chciałbym w tym temacie dołożyć i swoją, kolejną już cegiełkę i zaprezentować serwer telnetowy zbudowany właśnie z wykorzystaniem modułu 7010A firmy Wiznet.

Żeby na samym początku nie oklejać projektu etykietą „To już było”, postanowiłem pójść troszkę w innym kierunku, mam nadzieję, znacznie podnosząc walory użytkowe projektu. Wykorzystamy do współpracy z urządzeniem serwis WWW zawierający wykonaną profesjonalnie wizualizację obiektu domu. Nietrudno chyba wyobrazić sobie przewagę nad innymi rozwiązaniami jaką niesie połączenie możliwości podglądu pomieszczeń domu z możliwością przemieszczania się po nich, bez ograniczania się

tylko do standardowych komponentów typu przyciski, pola wyboru itp.

Budowa sterownika

Schemat sterownika pokazano na **rysunku 1**. Jego elementem jest 8-bitowy mikrokontroler ATmega162, taktowany sygnałem generatora kwarcowego o częstotliwości 4 MHz. Sygnał zerowania jest generowany przez DS1813-10.

Przeglądając aplikacje z wykorzystaniem modułów Wiznet często można spotkać sytuację, w której doprowadzenia zerowania modułu ethernetowego i mikrokontrolera są sterowane z tego samego źródła. Jest to rozwiązanie poprawne, ale mało funkcjonalne. W momencie restartu mikrokontrolera np. na skutek zadziałania Watchdog'a czy funkcji *Brown Out Detection*, pojawiają się często problemy z ponownym nawiązaniem połączenia z modułem IIM7010A. Dlatego

AVT-5251 w ofercie AVT:
AVT-5251A – płytka drukowana

Podstawowe informacje:

- Zasilanie 12 VDC, mikrokontroler ATmega162.
- Płytko drukowana o wymiarach 183×113 mm
- 8 wyjść przełącznikowych, 8 wyjść cyfrowych.
- Interfejs Ethernet (moduł IIM7010A firmy Wiznet).
- Sterowanie z użyciem protokołu Telnet.
- Wyposażony w oprogramowanie sterujące LinkService Lite umożliwiające wpisanie nastaw, kontrolę sterownika oraz integrację z serwisem WWW.

Dodatkowe materiały na CD i FTP:

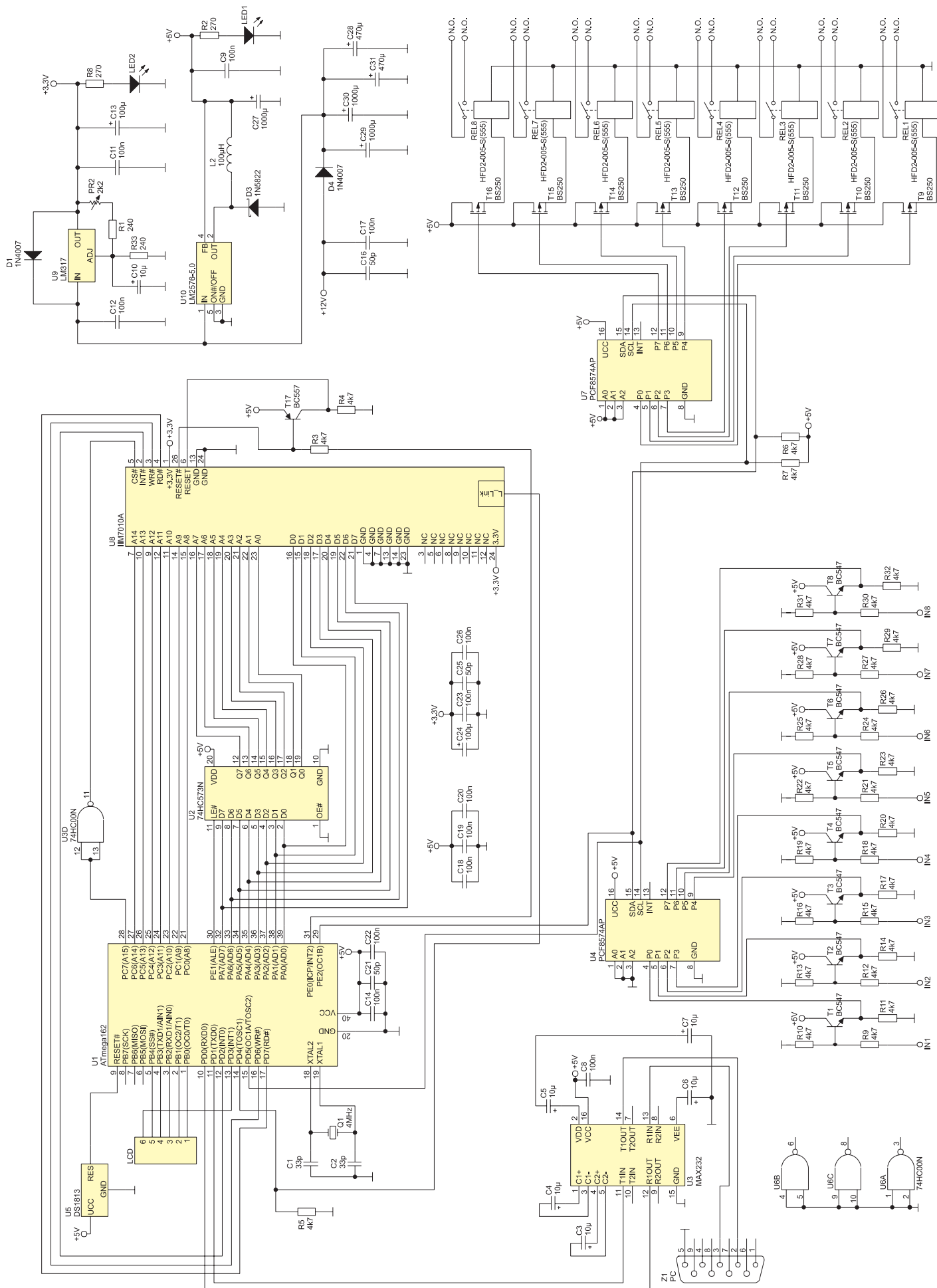
- <ftp://ep.com.pl>, user: 10765, pass: 4t4q4glg
- wzory płytek PCB
 - karty katalogowe i noty aplikacyjne elementów oznaczonych w **wykazie elementów** kolorem czerwonym

Projekty pokrewne na CD i FTP:

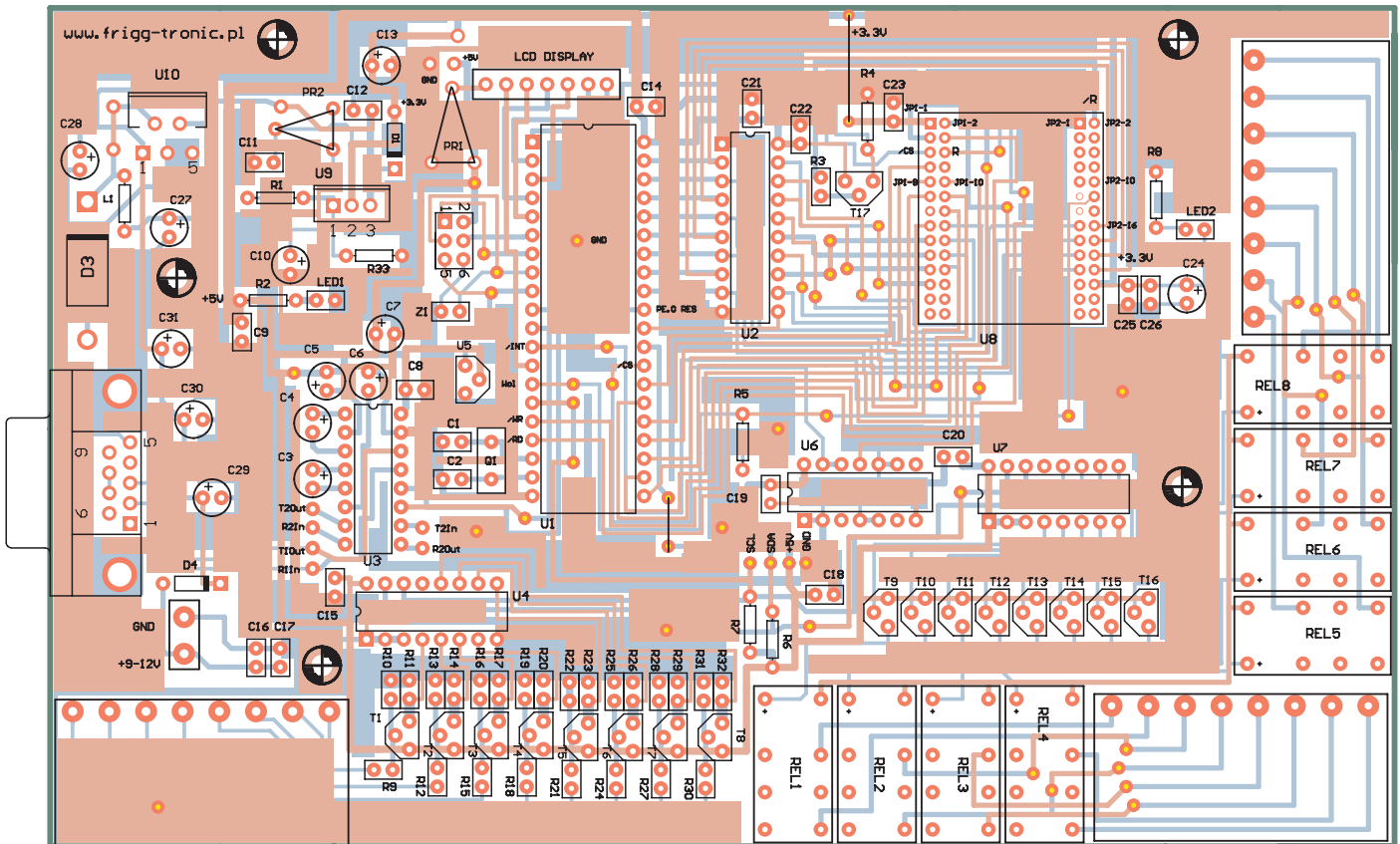
- (wymienione artykuły są w całości dostępne na CD)
- AVT-5200 Uniwersalny sterownik ethernetowy (EP 9/2009)
 - AVT-5166 Serwer HTTP (EP 1/2009)
 - AVT-5157 Przełącznik internetowy (EP 11/2008)
 - AVT-2859 Internetowy sterownik urządzeń (EdW 3/2008)
 - AVT-974 Sterownik z interfejsem TCP/IP (EP 3/2007)
 - AVT-966 Karta przełączników sterowana przez internet (EP 2/2007)

w sterowniku zastosowałem zupełnie inne rozwiązanie. Nad prawidłowym startem modułu ethernetowego czuwa sam mikrokontroler zerując go sygnałem z jednego z wyjść.

Moduł IIM7010A może pracować z wykorzystaniem dwóch interfejsów komuni-



Rysunek 1. Schemat ideowy sterownika domu



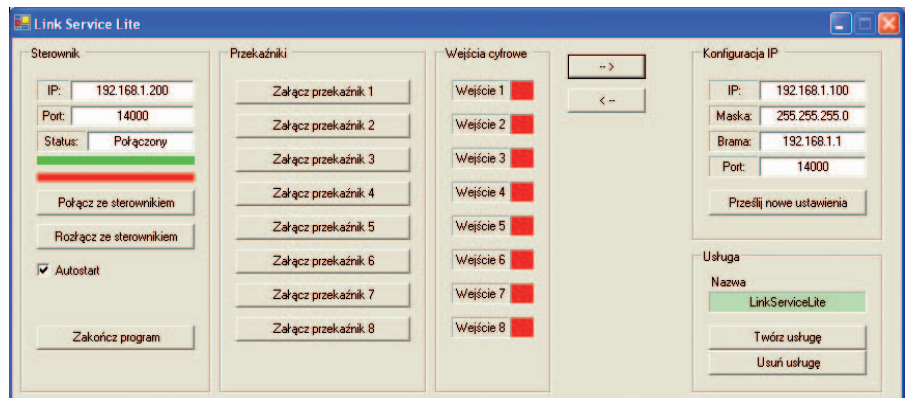
Rysunek 2. Schemat montażowy sterownika domu

kacyjnych. W jednym z przypadków jest widziany przez mikrokontroler jako pamięć zewnętrzna, natomiast w drugim do komunikacji jest używany interfejs TWI. Każda z tych metod ma pewne wady i zalety. Pierwsza wymaga użycia dużej liczby linii połączeniowych. Z drugiej strony wymiana danych przebiega znacznie szybciej, niż przez interfejs TWI. Zaletą rozwiązania z TWI jest niewielka liczba wyprowadzeń, natomiast wadą stosunkowo mała prędkość transmisji. W urządzeniu wykorzystałem rozwiązanie pierwsze.

Układy elektroniczne są zasilane napięciem +5 V, które zapewnia stabilizator impulsowy działający w oparciu o popularny LM2576-5. W urządzeniu pobór prądu nie jest duży, więc układ nie wymaga zastosowania radiatora. Moduł Ethernetowy potrzebuje napięcia +3,3 V. Jest ono uzyskiwane za pomocą stabilizatora liniowego – LM317.

Tabela 1. Konfiguracja fusebit'ów mikrokontrolera ATmega162

M161C	0
BODLEVEL	Brown-out detection disabled
OCDEN	0
JTAG	0
WDTON	0
EESAVE	0
BOOTRST	0
CKDIV8	0
CKOUT	0
SUT_CKSEL	Ext. Crystal Osc. 8.0 MHz; Start-up time: 16K CK + 4,1 ms



Rysunek 3. Wygląd okna aplikacji LinkService Lite

Inaczej niż stabilizator impulsowy, należy go wyposażyć w niewielki radiator. Wejścia modułu 7010A są przystosowane do pracy z napięciami +5 V, dlatego zdecydowałem się na użycie dwóch różnych napięć zasilania.

Sterowanie przekaźników zrealizowano z pośrednictwem tranzystorów BS250 z kanałem typu P. Zastosowanie tego typu tranzystorów, a nie tańszych odpowiedników z kanałem typu N, wiąże się bezpośrednio z układem PCF8574AP, które posłużył jako ekspander we/wy ze względu na ograniczoną liczbę portów mikrokontrolera. W momencie zasilenia układu, PCF8574AP na moment ustawia wszystkie porty na poziomie wysokim, co może spowodować zadziałanie przekaźników. W momencie zasilenia, mikrokontroler ustawia wszystkie porty w stan 1, po czym ustawia flagę startu i zapamiętuje w pamięci ulotnej stan przekaźników.

W chwili utraty połączenia czy kontrolowanego restartu mikrokontrolera, przekaźniki nie opadają tylko pozostają w poprzednim stanie, aż do momentu wyłączenia zasilania.

Montaż i uruchomienie

Schemat montażowy sterownika zamieszczono na **rysunku 2**. Montaż jest łatwy i nie powinien sprawić problemu. Przed włożeniem modułu w podstawkę, należy za pomocą potencjometru PR2 ustawić napięcie 3,3 V wymagane do jego zasilania.

Do zaprogramowania mikrokontrolera wykorzystano AVR-Studio 4.18. Na płytce znajduje się 6-pinowe, dwurzędowe gniazdo programatora ISP, dzięki któremu mikrokontroler można zaprogramować bez wyjmowania go z podstawki lub wylutowywania z płytki. Konfigurację fusebitów zamieszczono w **tabeli 1**.

Poprawnie zmontowany i zaprogramowany sterownik powinien uruchomić się od razu po włączeniu zasilania. Przy pierwszym starcie oprogramowanie nadaje mu domyślne parametry, dzięki którym bez problemu można go odnaleźć w sieci lub – w razie problemów – uruchomić go z domyślnymi nastawami. Nastawy domyślne to:

- adres IPv4: 192.168.1.200,
- port: 14000,
- maska podsieci: 255.255.255.0.

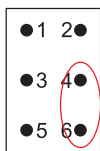
Obsługa sterownika

Sterownik pracuje jako serwer Telnet o znanym adresie, łączący się poprzez ustalony port. Do wprowadzenia nowych nastaw można użyć np. dołączonej do niego testowej aplikacji *LinkService Lite* (**rysunek 3**), umożliwiającej zarówno zmianę ustawień sieciowych sterownika, jak również sterowanie przekaźnikami, podgląd ich stanów oraz ośmiu wejść cyfrowych.

Po przesłaniu nowych nastaw sieciowych należy odczekać 5 sekund, po czym wyłączyć zasilanie, wpiąć zworkę w łączę programatora w miejscu pokazanym na **rysunku 4** i ponownie uruchomić sterownik. W celu przywrócenia ustawień domyślnych, należy odłączyć zasilanie, ściągnąć zworkę i ponownie zasilić urządzenie.

Sterownik w krótkich odstępach czasu zwraca informację z aktualnymi statusami wejść oraz wyjść, w zależności od których zmienia się nazewnictwo przycisków sterujących. Jeśli przekaźnik 1 będzie załączony, wówczas przycisk przewidziany dla przekaźnika przyjmie nazwę – „Wyłącz przekaźnik 1”, jednocześnie umożliwiając jego wyłączenie, po czym nazwa zostanie zmieniona na „Załącz przekaźnik 1”

Aplikacja *LinkService Lite* ma jeszcze jedną ważną funkcję. Umożliwia ona połączenie programu z dowolnym serwisem WWW czy aplikacją okienkową za pomocą plików tekstowych, wymienionych w ustalonym katalogu. W tej wersji można zrobić z programu usługę, dzięki przyciskom „Twórz usługę” oraz „Usuń usługę”, która startuje samoczynnie wraz ze startem komputera. Usługa nawiązuje i podtrzymuje połączenie ze sterownikiem. Pliki tekstowe generowane z dowolnego innego programu są przez nią przetwarzane i jeśli zostaną zrozumiane, to zostaną przesłane do sterownika, a następnie usunięte. Jest to wygodne rozwiązanie dla mniej wprawionych w programowaniu



Rysunek 4. Sposób wpięcia zworki w łączę programatora

użytkowników. Sytuacja szczególnie dogodna przy wykorzystaniu komputera jako typowego serwera WWW.

Bardziej zaawansowani programiści mogą napisać własne oprogramowanie sterujące. Dla nich jest przeznaczona duża część tekstu.

Protokół komunikacyjny

Jak wspomniano wcześniej, sterownik można kontrolować z użyciem protokołu Telnet. Niżej umieszczono wymagane formaty poleceń.

Sterowanie przekaźnikami

Rozkaz	Suffix
O\$OUTXY	Chr(3)

X: przyjmuje wartość S lub R (od słów Set – włącz oraz Reset – wyłącz)

Y: ma wartość 1...8 i określa numer wyjścia

Informacja zwrotna o stanie we/wy

Rozkaz	Suffix
O807060504030201817161514131211	Chr(3)

O8...O1: aktualne statusy wyjść. Zmienia na przyjmuje wartość 0 lub 1, przy czym statusy podawane są w inwersji ze względu na zastosowane tranzystory z kanałem typu P. „0” oznacza, że przekaźnik jest załączony.

I8...I1: aktualne statusy wejść. Przyjmuje wartość 0 lub 1, gdzie 0 oznacza poziom niski na danym wejściu

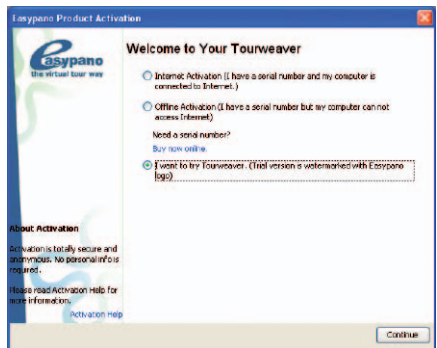
Podtrzymywanie połączenia

Rozkaz	Suffix
A\$ALIVE	Chr(3)

Informacja wysyłana cyklicznie co 8 sekund w celu podtrzymania połączenia. Wysłanie komendy sterującej przekaźnikami jest równoznaczne z komendą podtrzymywania połączenia.

Tworzenie graficznego interfejsu użytkownika

Do tworzenia oprogramowania wizualizacji budynku wykorzystamy narzędzie



Rysunek 5. Okno powitalne programu Easypano – Tourweaver 4.0 Professional

firmy *Easypano – Tourweaver 4.0 Professional*. Aktualnie jest też dostępna wersja 5.0. W wersji demonstracyjnej można go pobrać ze strony producenta http://www.easypano.com/Virtual-tour-software-44_351.html. Wersja demo jest w pełni funkcjonalna, jedynie pozostawia na projekcie znak wodny oraz informację o konieczności rejestracji. Nie przeszkodzi nam to jednak w nauce i poznaniu tego rewelacyjnego narzędzia.

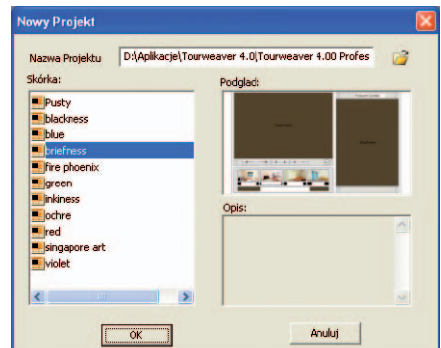
Aby nie utrudniać sobie nauki, początkowo zastосуemy gotowe elementy zawarte w programie, a dostępne po jego instalacji. Docelowo, po nabraniu wprawy każdy będzie mógł sam utworzyć własną wizualizację i użyć jej w przyszłych projektach. Tematyka artykułu dotyczy sterowania urządzeniami domowymi, co oczywiście nie ogranicza użycia tego narzędzia w przypadku innych zastosowań.

Domyślnie program instaluje się w angielskiej wersji językowej, jednak podczas instalacji można wybrać język polski. Przed uruchomieniem zainstalowanego programu proponuję założyć (np. na pulpicie) katalog o nazwie „Projekt”, w którym będziemy przechowywać nasze opracowania i skopiować do niego z materiały udostępnione na CD EP8/2010 w katalogach Images, Map oraz Hotspots.

Jeśli korzystamy z wersji demonstracyjnej, to program należy uruchomić korzystając z trzeciej opcji umieszczonej w oknie powitalnym, wyświetlanym po uruchomieniu programu (**rysunek 5**).

Po dokonaniu wyboru przechodzimy dalej. W kolejnym oknie poprzedzającym rozpoczęcie pracy wybieramy opcję „New Project” umożliwiającą rozpoczęcie pracy nad nowym projektem. W kolejnym etapie pozostanie nam jeszcze wybór szablonu. Na razie zdecydujemy się na gotowy szablon o nazwie „briefness” (**rysunek 6**), wchodzący w skład pakietu. Po nabraniu wprawy w pracy z programem, będziemy mogli stworzyć własne i wtedy wybierzemy opcję „Pusty”. Otwiera się okno główne, które producent podzielił na kilka części.

Tak naprawdę trudno w ramach tego artykułu napisać instrukcję obsługi programu. Wszystkie opcje można przetestować



Rysunek 6. Wybór szablonu o nazwie briefness

samemu, a ja w tym opisie skupię się tylko na tych funkcjach, które umożliwiają nam utworzenie przykładowego interfejsu przeznaczonego do współpracy z opisywanym sterownikiem.

Środkowa część ekranu będzie zawierać tworzoną przez nas wirtualną wizualizację domu. W części dolnej znajdziemy opcje dotyczące ustawień oraz akcji, które zostaną przyporządkowane umieszczanym

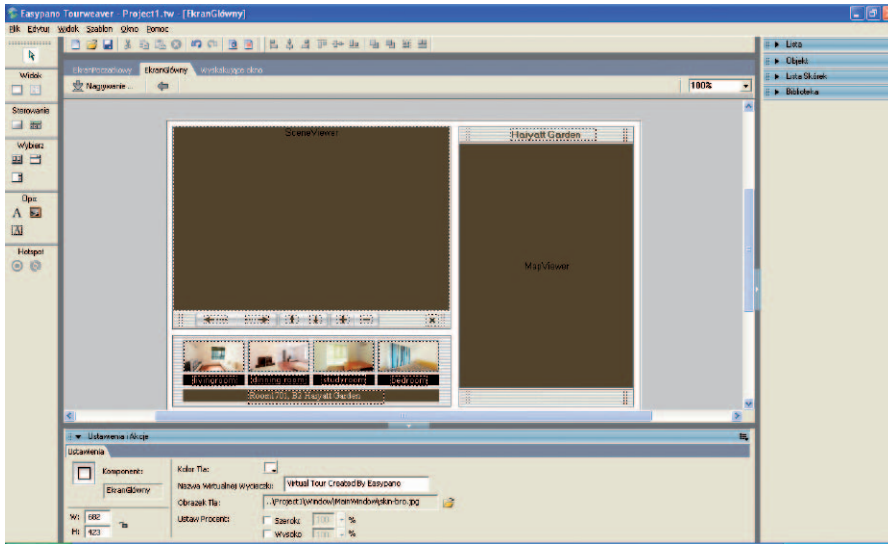
przez nas obiektom, takim jak „Hotspots” czy przyciski, umożliwiając przemieszczanie się między pomieszczeniami wirtualnego budynku (w przypadku większej ich liczby), oraz dzięki którym w danym pomieszczeniu będziemy mogli generować wymagane przez nas zdarzenia. W części lewej i prawej znajdziemy narzędzia umożliwiające podgląd i w pewnym sensie edycję użytych obiektów (rysunek 7).

Zanim przejdziemy dalej, zapiszmy nasz projekt w utworzonym katalogu i nadajmy mu nazwę „Projekt”. Opcja zapisu odbywa się w standardowy sposób, dobrze znany z systemu Windows.

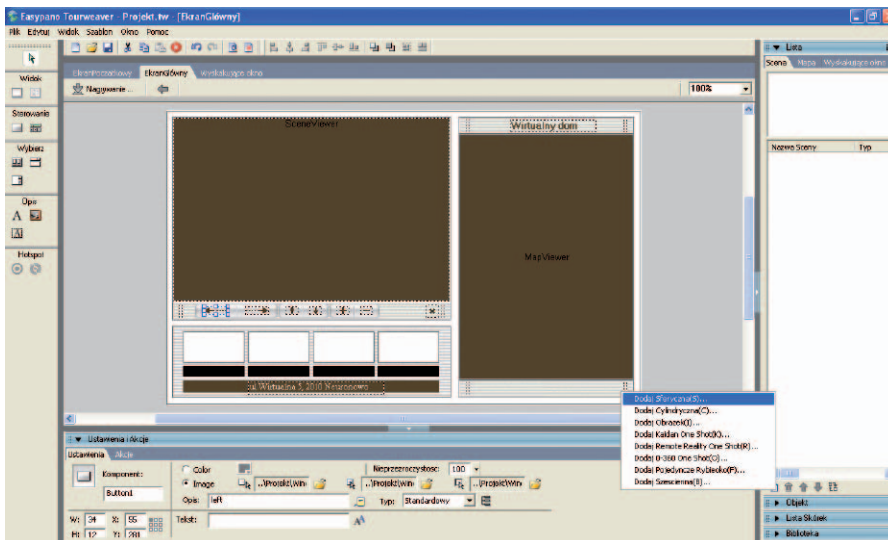
Tworzymy projekt. Aby ułatwić nieco pracę, przeniosłem zdjęcia do katalogu, który jest dołączony do materiałów projektu. Znajdują się one we wspomnianych wcześniej folderach Hotspots, Images oraz Map.

Na samym początku zaczniemy od wykasowania niepotrzebnych elementów w istniejącym szablonie. Nad polem MapViewer zmienimy napis „Haiyatt Garden” na „Wirtualny dom EP” (rysunek 8). Pod polem SceneViewer wyrzucamy cztery niepotrzebne zdjęcia oraz napisy pod nimi, a tekst „ul. Wirtualna 5, 2010 Neuronowa”. Oczywiście jeśli ktoś woli, napis może również wykasować. Stosujemy szablon, ponieważ zaoszczędzi nam to sporo wysiłku podczas nauki obsługi programu. Jeśli rozwiniemy zakładkę „Obiekt” zauważymy, że cały projekt składa się z takich podstawowych obiektów jak OknoSceny oraz OknoMapy. Kolejne, to elementy wspomagające, takie jak przyciski, umożliwiające generowanie zdarzeń czy etykiety tekstowe pozwalające na zamieszczenie dodatkowych opisów. Miejsca w których jeszcze przed chwilą znajdowały się obrazeczki nie są niczym innym jak graficznym szablonem będącym po prostu obrazkiem tła. Po zaznaczeniu dowolnego z komponentów, w zakładce „Akcje” w dolnej części możemy podglądać zdarzenie, które zostało przypisane temu komponentowi. Zdarzenie można wybrać z dostępnych 5 kategorii, takich jak: Scena, Dźwięk, Różne, Okno, Mapa.

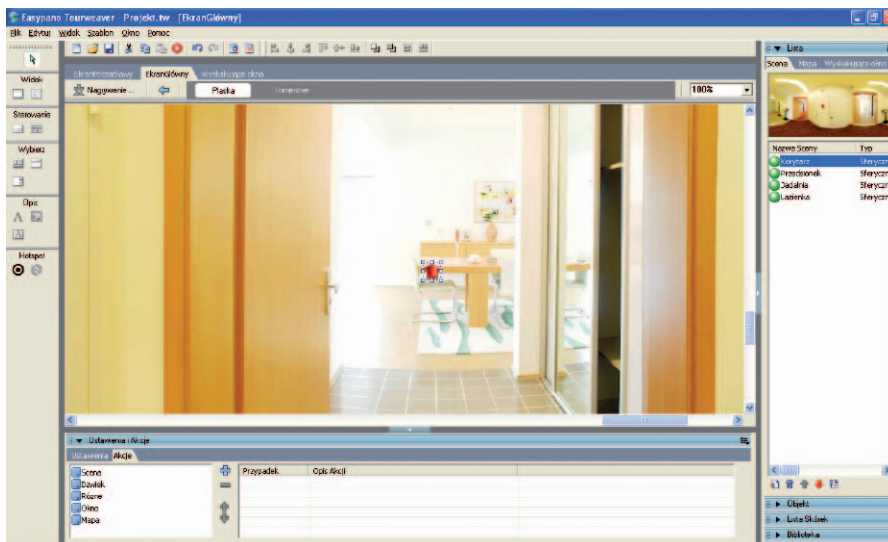
Wreszcie czas rozpocząć budowę naszego wirtualnego domu. Rozwijamy zakładkę „Lista” i wybieramy wchodzącą w jej skład kartę „Scena”. Następnie przyciskiem „Dodaj” (ikonka dokumentu z czerwonym krzyżykiem) wybieramy opcję „Dodaj sferyczna(S)” i dodajemy kolejno przygotowane zdjęcia: Korytarz.jpeg, Przedsionek.jpeg, Jadalnia.jpeg i Lazienka.jpeg. Następnie przechodzimy do karty „Mapa” i w analogiczny sposób dodajemy zdjęcie mapy z katalogu „Map”. Teraz wracamy do karty scena i dwukrotnie klikamy na scenie „Korytarz”. Powinno otworzyć się okno zawierające cały obraz sceny. Za pomocą scrollbarów umiesz-



Rysunek 7. Wygląd okna szablonu projektu briefness



Rysunek 8. Wygląd okna projektu po proponowanych zmianach



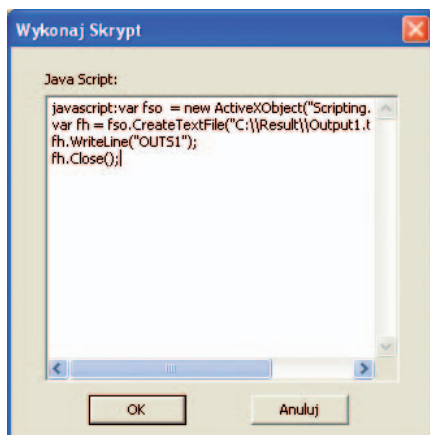
Rysunek 9. Umieszczenie obiektu hotspot na fotografii wnętrza



Rysunek 10. Wybór sceny przedśionek

przemierzamy się po obrazie i wyszukujemy drzwi, u spodu których widoczne są kafelki. Jednocześnie zwróćmy uwagę na to, że w lewym pasku narzędziowym aktywne stało się narzędzie „Hotspot”. Klikamy je i zamieszczamy wybranego hotspota na odnalezionych przez nas drzwiach. Przyjmuje on domyślny wygląd w kolorze zielonym. Teraz zaznaczamy opcję „Image”. Następnie z pierwszej pozycji klikamy na ikonę otwarcia folderu i wyszukujemy w katalogu „Hotspots” pliku o nazwie 1-7.gif i wybieramy go. W tym momencie nasz punkt powinien zmienić się w strzałkę na wprost (rysunek 9).

Teraz zmieniamy obszar naszych działań na dolną część paska narzędziowego. Otwieramy kartę „Akcja”, wybieramy opcję „Scena”, a następnie dwukrotnie klikamy na „Link do sceny”. Z wirtualnego korytarza przechodzimy do przedśionka (wybieramy scenę przedśionek, jak na rysunku 10).



Rysunek 11. Efekt działania skryptu w języku JavaScript

Dodatkowo możemy ustawić jeszcze efekt przejścia oraz czas przejścia.

Następnie klikamy dwukrotnie na scenę przedśionek i wykonujemy tę samą czynność, co poprzednio dla korytarza, dodając dwa hotspoty z przejściem do łazienki i jadalni oraz powrót do korytarza. Po przejściu w kartę „Mapa” i kliknięciu na obrazku mapy, dokładnie w ten sam sposób możemy dodać hotspot i od razu przechodzić do żądanych pomieszczeń bez przemieszczania się przez kolejne. Na mapie dla przykładu naniosłem tylko jeden punkt, za pomocą którego możemy przejść od razu do łazienki.

Po zakończeniu projektu możemy obejrzeć nasz wirtualny dom, korzystając z funkcji „Podgląd”, którą znajdziemy w postaci ikonki w górnym pasku narzędziowym lub w rozwijanym menu „Plik”.

Integracja z aplikacją LinkService Lite oraz sterownikiem

Pozostała nam jeszcze do rozwiązania kwestia integracji projektu z aplikacją LinkService oraz sterownikiem. Wykorzystamy do tego celu język JavaScript, który pozwoli nam tworzyć pliki tekstowe.

Przykłady obiektów do sterowania w naszym domu można mnożyć. Począwszy od oświetlenia, telewizora, radia, rolet okiennych, a skończywszy na piecu C.O. Zrobimy to również korzystając z obiektów typu hotspot, które możemy umieścić w wymaganym przez nas miejscu – np. w pobliżu włączników światła.

Przyjąć możemy dwie opcje. W pierwszej wykorzystamy dwa osobne obiekty obok siebie – jeden umożliwi włączenie, drugi wyłączenie danego odbiornika. Jest to opcja najbardziej praktyczna. W drugiej zamieścimy w pierwszej kolejności hotspota umożliwiającego włączenie odbiornika. Użycie go, oprócz wygenerowaniu skryptu tworzącego plik, spowoduje przejście do identycznej sceny, tylko z hotspotem generującym funkcję wyłączenia danego odbiornika. Scena może być ciemniejsza, symulując jednocześnie np. zgaszenie światła. Aby wygenerować plik tekstowy skorzystamy z następującego skryptu:

```
var fso = new
ActiveXObject („Scripting.File-
SystemObject”);
var fh = fso.
CreateTextFile („C:\\Result\\Out-
put1.txt „, true);
fh.WriteLine („OUTS1”);
fh.Close ();
```

Oczywiście ścieżka dostępu powinna odnosić się do miejsca zadeklarowanego w aplikacji LinkService Lite. Zamieścimy zatem hotspot w dowolnym miejscu i w dowolnej scenie. Korzystając z pola „Tekst” na karcie „Ustawienia” możemy dowolnie go opisać, np. REL1 ON. Wejźmy teraz do za-

Wykaz elementów

- Rezystory:**
R1, R33: 240 Ω
R2, R8: 270 Ω
R3...R7, R10...R32: 4,7 Ω
PR2: 2,2 kΩ
- Kondensatory:**
C1, C2: 33pF
C8, C9, C11, C12, C14, C17...C20, C22, C23, C26: 100 nF
C16, C21, C25: 50 pF
C3...C7, C10: 10 μF/16 V
C13, C24: 100 μF/16 V
C27, C29, C30: 1000 μF/16 V
C28, C31: 470 μF/16 V
- Półprzewodniki:**
U1: ATmega162
U2: 74HC573N
U3: 74HC00N
U4, U7: PCF8574AP
U5: DS1813-10
U6: 74HC00N
U8: IIM7010A
U9: LM317
U10: LM2576-5.0
D1, D4: 1N4007
D3: 1N5822
LED1, LED2: dioda LED 3 mm
T1...T8: BC547
T9...T16: BS250
T17: BC557B lub C
- Inne:**
Q1: 4 MHz
L2: 100 μH
REL1...REL8: HFD2-005-S
Złącza dwurzędowe, raster 2 mm
DSUB9 do druku
ARK2

kładki umożliwiającej zadeklarowanie akcji. Tym razem nie skorzystamy z opcji „Link do sceny” tylko w pozycji „Różne” wykorzystamy funkcję „Wykonaj skrypt” (rysunek 11). Po dwukrotnym kliknięciu otworzy się nam okno umożliwiająca wpisanie skryptu. Przekopiujemy zatem nasz skrypt, zatwierdzimy go i dla pewności zapiszmy projekt.

Uruchommy teraz opcję „Podgląd”, aby sprawdzić rezultat naszej pracy. Zanim wykorzystamy nasze dzieło, utworzymy na dysku C katalog o nazwie „Result”. W przeciwnym razie skrypt wygeneruje błąd, nie mogąc znaleźć zadeklarowanej ścieżki. Gdy już to zrobimy, możemy kliknąć na utworzonym punkcie, a później odszukać nowo powstały plik tekstowy. Jeśli plik zostałby stworzony w katalogu zadeklarowanym dla programu LinkService Lite, spowodowałoby to załączenie przekaźnika nr 1.

Kiedy uznamy, że zakończyliśmy projektować nasz wirtualny dom, możemy przystąpić do ostatniej czynności, którą jest publikacja. Opcję publikacji znajdziemy w górnym pasku narzędziowym w menu „Plik”. Otrzymamy stronę WWW, praktycznie gotową do umieszczenia na serwerze. Przykład znaleźć można pod adresem: <http://wirtualnydomep.eu.interia.pl>.

Rafał Chromik
rchromik@frigga-tronic.pl
www.frigga-tronic.pl

