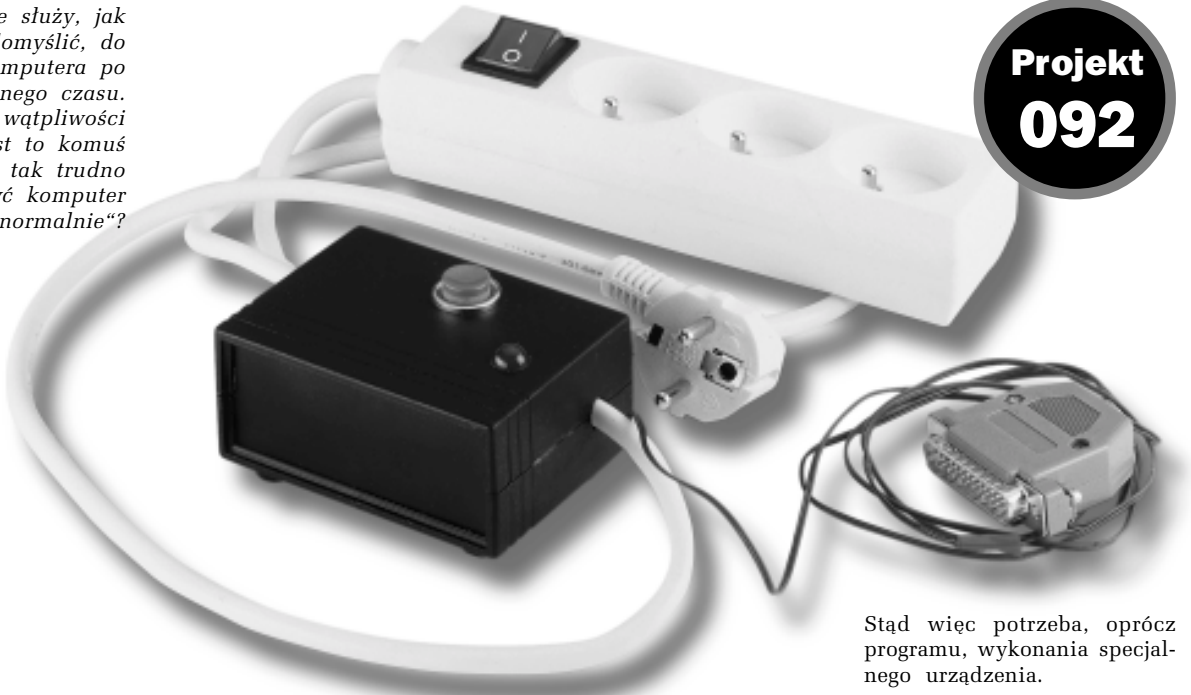


Dział „Projekty Czytelników” zawiera opisy projektów nadesłanych do redakcji EP przez Czytelników. Redakcja nie bierze odpowiedzialności za prawidłowe działanie opisywanych układów, gdyż nie testujemy ich laboratoryjnie, chociaż sprawdzamy poprawność konstrukcji.

Prosimy o nadsyłanie własnych projektów z modelami (do zwrotu). Do artykułu należy dołączyć podpisane **oświadczenie, że artykuł jest własnym opracowaniem autora i nie był dotychczas nigdzie publikowany.** Honorarium za publikację w tym dziale wynosi 250,- zł (brutto) za 1 stronę w EP. Przesyłanych tekstów nie zwracamy. Redakcja zastrzega sobie prawo do dokonywania skrótów.

Czasowy wyłącznik zasilania komputera

Urządzenie służy, jak się łatwo domyślić, do wyłączania komputera po upływie określonego czasu. Można mieć wątpliwości czy jest to komuś potrzebne. Czy tak trudno jest wyłączyć komputer „normalnie“?



Projekt 092

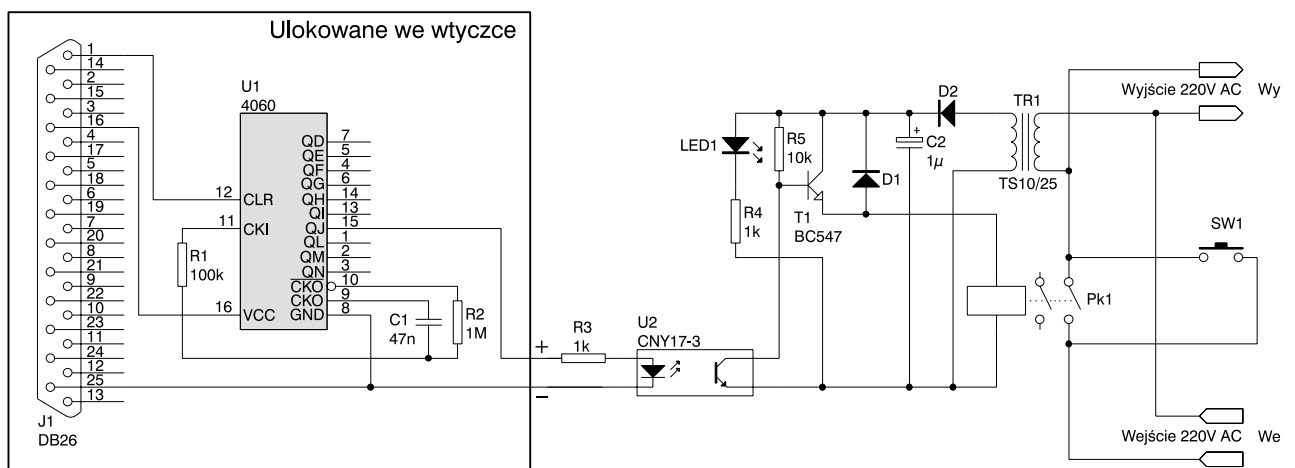
Stąd więc potrzeba, oprócz programu, wykonania specjalnego urządzenia.

Są zadania, które na moim komputerze wykonują się wolno (jest ich coraz więcej), a które nie wymagają interakcji z użytkownikiem, np.: kompresja plików dźwiękowych do formatu MP3, kompilacja dużych programów, symulacje, defragmentacja itp. Często zdarza się nam słuchać muzyki z komputera przed snem, bo lubimy przy muzyce zasypiać. Wówczas

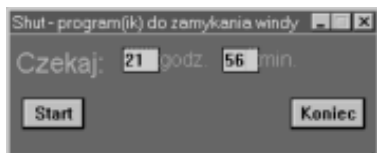
pojawia się problem zamknięcia systemu operacyjnego i odłączenie zasilania. W przypadku współczesnych obudów typu ATX, to drugie jest wykonywane automatycznie, wystarczy napisać odpowiedni program. Jednak mój sprzęt jest dość stary i nie zanosi się na jego wymianę w najbliższym czasie. Dobrze się sprawuje jako szafa grająca, ale sam sobie nie potrafi odciąć zasilania.

Opis działania

Schemat elektryczny urządzenia pokazano na rys. 1. Całość wykonana jest w postaci przedłużacza z zamontowanym na kablu małym pudełkiem (w którym znajduje się elektronika) i wtyczki do portu równoległego, przez który program steruje urządzeniem. Na zewnątrz pudełka wystają: dioda LED (sygnalizująca obec-



Rys. 1.



Rys. 2.

ność napięcia sieci w gniazdkach przedłużacza) oraz przycisk SW1, który służy do włączania. Przewody biegnące od wtyczki sieciowej podłączone są do zacisków „We”. Zaciski „Wy” służą do przyłączenia komputera z ewentualnym osprzętem.

Po naciśnięciu przycisku SW1 przez uzwojenie pierwotne transformatora Tr1 popłynie prąd. Spowoduje to pojawienie się na uzwojeniu wtórnym napięcia, które jest prostowane przez diodę D2 i wyglądane przez kondensator C2. Wyglądzone napięcie ma wartość około 12V. Za pośrednictwem rezystora R5 polaryzowana jest baza tranzystora T1, co wywołuje przepływ prądu przez cewkę przekaźnika Pk1. Następuje połączenie styków przekaźnika i rozwarcie SW1 nie spowoduje już przerwania obwodu. Świecenie diody LED1 świadczy o obecności na zaciskach „Wy” napięcia sieci.

Jeśli teraz do zacisków „-“ i „+” przyłoży się napięcie 5V, przez rezystor ograniczający R3 i diodę LED w strukturze transoptora U2 popłynie prąd o wartości ok. 5mA. Powoduje to zaświecenie diody i nasycenie fototranzystora w U2 (z zachowaniem właściwej polaryzacji). Zwiera on wówczas bazę tranzystora T1 z masą pozorną, co prowadzi do jego zatkania. Prąd przestaje płynąć przez cewkę, styki przekaźnika rozwierają się, LED1 gaśnie i komputer zostaje wyłączony. Aby ponownie uruchomić komputer trzeba nacisnąć SW1.

Dioda D1 zabezpiecza T1 przed zniszczeniem przez impuls napięcia powstający na cewce przy wyłączaniu przekaźnika.

Aby opisywany układ tak działał, to wystarczy dołączyć zaciski „+” i „-“ do portu równoległego i ustawić na jego liniach wyjściowych odpowiednią kombinację stanów logicznych, by wyłączyć zasilanie. Jednak dzisiejszych komputerów nie można tak po prostu wyłączyć bez zamknięcia systemu operacyjnego, a nie da się nic przesłać do portu po zamknięciu systemu. Konieczne jest zatem dodanie układu opóźniającego i uruchamianie go przed rozpoczęciem zamykania systemu. Eksperymentalnie wyznaczyłem czas opóźnienia na jedną minutę ze względu na niestabilność systemu Windows. Standardowo system ten zamyka się znacznie szybciej. Zazwyczaj 15 sekund wystarczy, ale jeśli zostawić pracującego np. Winampa na pięć godzin, to taki czas okazuje się za krótki, sprawdziłem. W przypadku Linuksa problem nie istnieje.

Układ czasowy jest wykonany na układzie 4060. Jest to generator z 14-bitowym licznikiem. Elementy R1, R2, C1 określają częstotliwość generowanego przebiegu. Wykorzystane jest dziesiąte wyjście licznika (nie wszystkie są wyprowadzone), na którym dla wartości elementów R1, R2, C1 podanych na schemacie, pojawia się przebieg o okresie 2 minuty. Przez minutę trwa poziom niski, po czym pojawia się narastające zbocze i następuje przełączenie transoptora.

Zasilanie układu U1 połączone jest ze stykiem nr 16 portu LPT (sygnał INIT - bit nr 2 rejestru sterującego). Aby ustrzec się stanów nie-

ustalonych, wykorzystano wejście zerujące U1 - połączone jest z wyprowadzeniem nr 1 portu (sygnał STROBE - bit nr 0 rejestru sterującego). Sygnał wyjściowy STROBE jest zanegowany, więc po resecie komputera przyjmuje poziom wysoki, wyjście INIT - niski. Jeśli ustawi się bit INIT, generator otrzymuje zasilanie, ale nie startuje, bo jest zerowany przez sygnał STROBE. Dopiero po ustawieniu bitu nr 0 rejestru sterującego, STROBE przyjmuje poziom niski i generator rozpoczyna pracę.

Oprogramowanie

Jak wynika z powyższych uwag, aby rozpocząć „odliczanie” należy w rejestrze sterującym ustawić bit 2, a następnie bit 0. Adres tego rejestru jest przesunięty o 2 względem adresu bazowego, czyli w przypadku LPT1 jest to 37Ah.

Program sterujący został napisany dla dwóch systemów operacyjnych. Pierwszym chronologicznie był *shut.exe* używany pod Windows 95 (powinien też pracować z nowszymi, ale nie sprawdzałem). Korzysta z funkcji *API ExitWindows-Ex()*, która zamyka system (i wyłącza zasilanie, jeśli zasilacz i płyta główna komputera jest wykonana w standardzie ATX).

Zrzut ekranu działającego programu *shut.exe* pokazano na rys. 2. Napisany jest w języku C++ z wykorzystaniem biblioteki OWL i wstawek assemblerowych. Zapewnia tylko to, co jest niezbędne do pracy. Odlicza czas z dokładnością minuty, adres portu drukarkowego domyślnie przyjmuje jako 378h.

O wiele bardziej funkcjonalny jest program dla systemu Linux. Nazywa się *zamek* i jest napisany w C z wykorzystaniem biblioteki *Ncurses* i programu *shutdown*. Działa na konsoli oraz pod X-window w oknie terminala. Może zamykać system po upływie określonego czasu lub o określonej godzinie. Zrzut ekranu pokazano na rys. 3. W programie można zmieniać adres portu i zapisać ustawienia w pliku konfiguracyjnym. Na etapie instalacji trzeba zdecydować czy z programu może korzystać tylko użytkownik uprzywilejowany, czy każdy. Dodatkowo można go

WYKAZ ELEMENTÓW

- Rezystory**
- R1: 100kΩ
- R2: 1MΩ
- R3, R4: 1kΩ
- R5: 10kΩ
- Kondensatory**
- C1: 47nF
- C2: 1μF
- Półprzewodniki**
- D1: dowolna przełączająca
- D2: dowolna prostownicza
- LED1: dowolna świecąca
- T1: BC547 lub podobny
- U1: 4060
- U2: CNY 17-3
- Różne**
- Pk1: przekaźnik - cewka: 12V/-30mA
- SW1: przełącznik monostabilny (przycisk) na nap. 220V
- J1: DB25M
- Tr1: TS 10/25
- Przedłużacz
- Zaciski śrubowe do druku

uruchamiać z opcją „-n”, która powoduje natychmiastowe zamknięcie systemu. Jest to szczególnie przydatne przy pisaniu skryptów, które po wykonaniu jakiegoś zadania mają wyłączyć komputer. Wykorzystuję „zamek” do kompilacji jądra (trwa ponad godzinę) podczas mojej nieobecności. Oto zawartość przykładowego skryptu:

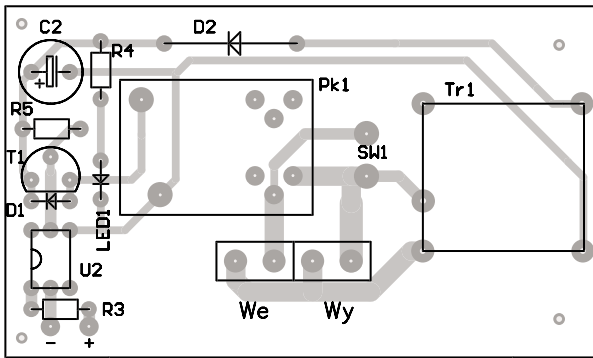
```
#!/bin/sh
cd /usr/src/linux
make dep clean bzImage
modules modules_install
zamek -n
```

Podczas korzystania z programu windowsowego istnieje ryzyko, że nie zadziała, tzn. system nie zostanie poprawnie zamknięty. Nie jest to jednak wina tego programu tylko nie najlepszej konstrukcji systemu Windows. Każdy użytkownik z pewnością spotkał się z sytuacją, kiedy musiał „począkać na zamknięcie systemu” i owego zamknięcia się nie doczekał. Dzieje się tak, bo system Windows podczas zamykania czeka aż wszystkie pracujące programy zakończą działanie i wystarczy, by tylko jeden nie oddał kontroli, a system będzie „wisiał” nieskończenie długo.

Natomiast w przypadku Linuksa (i jemu podobnych systemów operacyjnych) problemu nie ma, bo system zamyka wszystkie programy, odbiera im zasoby, a niepokorni są „zabijani”.



Rys. 3.



Rys. 4.

Montaż

Poskładanie całości nie wymaga specjalnych zabiegów i nie powinno nikomu sprawić trudności. **Należy jednak pamiętać, że urządzenie pracować będzie pod napięciem sieci energetycznej.**

Najwygodniej jest przereźbić gotowy przedłużacz. Elementy U1, R1, R2, C2, J1 zostały zalutowane „na pająka” wewnątrz obudowy wtyczki DB25M i zalane klejem termoplastycznym. Resztę zmonto-

wano na płytce drukowanej pasującej do obudowy Z6. Może być jednak kłopot z zamontowaniem wysokiego przełącznika. Płytkę została przyklejona do obudowy, a więc unika się wiercenia otworów i zminimalizowane jest niebezpieczeństwo porażenia prądem, ponieważ nie wystają na zewnątrz żadne metalowe elementy. Kable przedłużacza należy zabezpieczyć przed wyrwaniem z obudowy. Na obudowie należy

umocować diodę LED1, i przycisk SW1. Początkowo zamiast Tr1 miał być użyty rezystor, ale takie rozwiązanie marnotrawiłoby około 10 watów mocy. Transformator jest niewiele droższy, ale to rozwiązanie jest oszczędniejsze energetycznie i nie będzie się on nagrzewał. Zastosowany TS10/25 jest trochę „nadmiarowym”, gdyż jego wydajność prądowa to ok. 150mA, a powinno w zupełności wystarczyć 50mA (~30mA na przełącznik, 15mA dla LED). Można zastosować dowolny inny o napięciu wtórnym ok. 10V. Przełącznik zastosowany w modelu jest dość nietypowy (z demobilu), więc płytkę drukowaną raczej trzeba będzie przereźbić.

Wadą opisanego urządzenia wydaje się być zajmowanie portu LPT, szczególnie jeśli posiada się drukarkę. Teoretycznie jednak, bo gdyby podłączyć obydwa urządzenia do LPT, to nie powinny się one „gryźć”. Aby wy-

łączyć zasilanie, należy utrzymać odpowiednie stany na wyjściach portu przez całą minutę (celowo nie wstawiłem kondensatorów). Jakakolwiek chwilowa zmiana stanu linii INIT lub STROBE, powoduje przerwanie odliczania. Sygnał STROBE jest, jak nazwa wskazuje, używany do strobowania danych i zmienia się podczas drukowania wiele razy w ciągu sekundy, więc transfer danych do drukarki nie może spowodować wyłączenia zasilania. Natomiast jeśli drukarka jest wyłączona, to obojętne jest dla niej, na których liniach stany się zmieniają. Są to jednak rozważania tylko teoretyczne, bo nie posiadam drukarki i nie mogłem tego sprawdzić.

Opisane urządzenie jest intensywnie użytkowane od prawie pół roku i pracuje bez zarzutu, choć początkowo trudno mi było przyzwyczaić się do włączania komputera nogą.

Sławomir Jeliński
dugi@maluch.bondi.pl