

Zasilacz sterowany komputerowo



Do czego to służy?

Zasilacze na łamach EdW było już wiele – i słusznie. Jest to bowiem jeden z najbardziej niezbędnych przyrządów w pracowni elektronika. Zbigniew Raabe powiedział kiedyś przy opisie takowego, że koncepcji budowy zasilacza jest tyle, ilu jest na świecie elektroników. Na różne okazje potrzebne są różne zasilacze. Poza tym wiem z doświadczenia, że dobrych zasilaczy nigdy za wiele. Dziś ja chciałbym zaproponować Czytelnikom budowę niecodziennego zasilacza sieciowego, przydatnego zwłaszcza przy pracy nad układami mającymi współpracować z komputerem czy też wszelkimi systemami mikroprocesorowymi. Gdy pracujemy nad takim układem, cały czas używamy komputera – zwykle budowa części sprzętowej jest prosta lub wręcz banalna, a cały trud polega na napisaniu programu dla mikrokontrolera lub PC-ta. Do zasilania wykorzystujemy jakiś zasilacz, ale rzadko bywa tak, że jest on pod ręką i wszelkie manipulacje przy nim są utrudnione. Zwłaszcza w sytuacji, gdy „pracownia” elektronika hobbysty to skromny stolik mogący pomieścić komputer, opracowywany układ, parę narzędzi i ... niewiele więcej. Zasilacz stoi wtedy gdzieś w rogu lub wetknięty jest w nie zawsze wygodnie położone gniazdko. Pomyślałem, że byłoby czymś niezwykle wygodnym mieć dostęp do zasilacza z poziomu komputera PC. Chcemy zaprogramować procka (bez ISP) lub dokonać zmian w układzie – jedno kliknięcie i zasilacz jest wyłączony, możemy bez obaw wyjąć układ z podstawki. Po dokonaniu zmian klikamy i układ jest znów zasilany. Chcemy zmienić napięcie – klikamy na odpowiedni przycisk i ... no, myślę że wystarczy już tej wyliczanki i Czytelnicy zgodzą się ze mną, jak użyteczny może być taki niezwykły zasilacz. Oprócz możliwości zmiany napięcia oferuje on stałą kontrolę napięcia wyjściowego, które obrazowane jest na „wyświetlaczu” w głównym oknie programu. Wynik aktualizowany

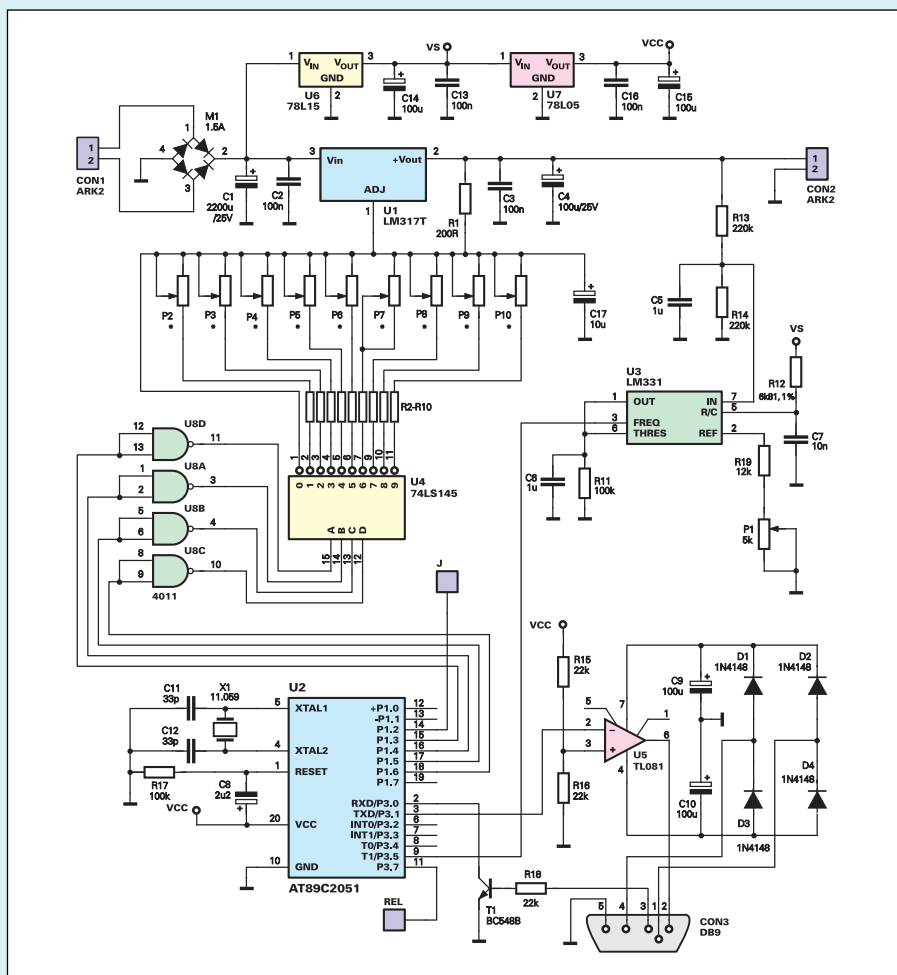
jest ok. 5 razy na sekundę. Model wyposażyłem w niewielki transformator TS6/40 z myślą o zastosowaniu go do zasilania głównie układów mikroprocesorowych i cyfrowych nie czerpiących prawie nigdy więcej niż 300...500mA prądu. Nic nie stoi jednak na przeszkodzie, aby zastosować większy transformator i duży radiator na układzie LM317 – taki zasilacz mogący dostarczyć prądu ponad 1A sprostą chyba wszystkim wymaga-

niom jakie napotka w pracowni niejednego elektronika.

Jak to działa?

Schemat ideowy układu przedstawiony jest na **rysunku 1**. Pomimo dość dużej liczby elementów sposób funkcjonowania zasilacza jest w sumie bardzo prosty. Komunikuje się

Rys. 1 Schemat ideowy



on z komputerem za pomocą portu szeregowego, którym może być dowolny z portów COM1...COM4 (jest wybierany przy uruchomieniu programu w PC-cie). Do konwertowania napięć w jedną stronę wykorzystałem zwykły wzmacniacz operacyjny TL081. Pracuje on tu w roli komparatora, a do jego zasilania użyte są linie RTS i DTR, na które w chwili otwarcia portu podawane są przeciwnie stany. Konwersję w drugą stronę zapewnia prosty bufor z tranzystorem T1. Działanie całości jest następujące: przez cały czas (co ok. 200ms) procesor U2 wysyła do komputera informację o napięciu wyjściowym. Robi to „na ślepo”, nie zastanawiając się, czy została odebrana, czy nie. Informację tę uzyskuje za pośrednictwem, ośmielę się użyć tego słowa, rewelacyjnego układu U3 typu LM331. Pracuje on w typowej aplikacji przetwornika U/F z wyjątkiem włączenia rezystorów R13 i R14 zamiast jednego rezystora na wejściu. Tworzą one dzielnik przez dwa, potrzebny ze względu na fakt, że zakres dopuszczalnych napięć wejściowych tej kostki wynosi w przybliżeniu 0...Uzas-2V, co przy Uzas=15V daje wartości 0...13V. To stanowczo za mało. Dzięki R13 i R14 możliwe jest mierzenie napięć z zakresu ok. 0...25V, co jest wręcz idealne w naszym zastosowaniu. Wartości tych rezystorów zostały zgodnie z prostą zasadą Thevenina dobrane tak, aby układ U3 „widział” rezystor o zalecanej wartości mniej więcej równej 100kΩ. Choć w modelu użyłem zamiennika kostki LM331 o nieco gorszych parametrach – układu XR4151 – uzyskane wyniki i tak przeszły moje oczekiwania, mogę z czystym sumieniem polecić ten układ wszystkim jako alternatywne rozwiązanie przetwornika analogowo-cyfrowego. Po szczegóły odsyłam do EdW12/2000 i 1/2001 oraz na stronę internetową EdW – są tam noty aplikacyjne wymienionych układów.

Współczynnik przetwarzania wynosi 1kHz/V, a więc napięcie wyjściowe zasilacza w voltach jest dwukrotnie (dzielnik) większe od częstotliwości przebiegu na nóżce 3 U3 wyrażonej w kilohercach. Jest to wyjście typu otwarty kolektor, co załatwiło problem z dopasowaniem napięć ze strefy 15V do 5V. Procesor (za pomocą pozostałego Timera0 – Timer1 wytwarza baudrate dla portu) mierzy czas trwania 50 okresów tego przebiegu i wysyła wynik (najpierw starszy a potem młodszy bajt Timera0) przez port szeregowy. Następnie czeka 200ms, znowu mierzy i wysyła - i tak w kółko. Nie interesuje go przy tym, czy dane zostały odebrane – opisana transmisja ma miejsce stale, nawet bez względu na to, czy port jest otwarty, czy nie. Komputer może te liczby odebrać i zinterpretować lub zignorować – wedle „uznania”. Jeśli program Zasilacz.exe jest uruchomiony i port jest otwarty, przychodząca informacja jest przez komputer przetwarzana z uwzględnieniem dzielnika przez dwa, liczby mie-

rzonych okresów oraz długości trwania okresu zegara w procesorze (ok. 1,085μs). Wynik trafia na „wyświetlacz”. Tak wygląda komunikacja w kierunku procesor - komputer.

W drugą stronę jest ciekawiej, bo przesyłanych jest kilka rodzajów informacji. **Rysunek 2** zawiera przykładowy zrzut z ekranu głównego okna programu Zasilacz.exe.

Rys. 2 Główne okno programu Zasilacz.exe



Obecny jest na nim wspomniany wyświetlacz, 9 przycisków służących do wyboru napięć, przycisk on/off, mały checkbox oraz oczywiście help i about. Do czego służy ów checkbox wyjaśnię później, teraz jedynie przeproszam obrońców ojczyzny - polszczyzny, ale taki termin chyba najlepiej oddaje pełnią funkcję („miejsce służące zaznaczeniu pewnej opcji” lub coś podobnego mało mnie pociąga) – dla równowagi w helpie (ha! znów angielskie określenie) użyłem niezbyt trafnego słowa „opcja”. To samo dotyczy użytego wcześniej określenia baudrate. Wciśnięcie jednego z przycisków wyboru napięć lub włącznika/wyłącznika powoduje wysłanie przez komputer jednego bajtu, w którym starsza połówka stanowi rozkaz mówiący procesorowi co ma zrobić, a młodsza to argument tego rozkazu. Procesor U2 odbiera ten bajt w procedurze obsługi przerwania UART-a i - zależnie od jego wartości - odpowiednio reaguje. Jeśli zmieniamy napięcie wyjściowe, rozkaz wynosi zero a argument określa, które napięcie wybieramy i przyjmujemy wartości od 6 (napięcie numer 9 – regulowane P10) do 14 (napięcie numer 1 regulowane P2). Wysłanie liczby 15 powoduje połączenie nóżki ADJ układu U1 z masą, oczywiście przez jeden z tranzystorów układu U4. Powoduje to podanie na wyjście zasilacza najmniejszego możliwego napięcia, z większością kostek 74LS145 równego ok. 1,4V (1,25V + napięcie kolektor-emiter tranzystora). Jak widać, liczby 14...6 są dokładnym zanegowaniem liczb 1...9. Jest tak ze względu na obecność w zasilaczu bramek z układu U8. Wielu zapyta, po co zastosowałem tę kostkę, skoro negacja ta jest tu zupełnie niepotrzebna, a nawet gdyby była potrzebna, to z dziecinną łatwością można ją zrealizować programowo? Otóż po włączeniu urządzenia do sieci układ U2 jest resetowany, co jak wiadomo, powoduje podanie na wszystkie porty stanów wysokich. Nawet

jeśli pierwszą instrukcją programu w procesorze byłoby podanie na linię P1.3...P1.6 stanów niskich, to i tak przez krótki czas byłyby tam stany wysokie, a to spowodowałyby wystąpienie na wyjściu zasilacza przez mniej więcej ten czas maksymalnego możliwego do uzyskania z danym transformatorem napięcia. Układ U8 zapobiega wystąpieniu takiego niekorzystnego zjawiska – tuż po włączeniu zasilacza na wyjściu pojawia się ok. 1,4V.

Po wciśnięciu przycisku ON zasilacz jest włączany. Komputer otwiera port, po czym wysyła liczbę 15 (włącz 1,4V na wyjściu) oraz liczbę 176 odpowiadającą rozkazowi 11 i argumentowi 0. Ponowne wciśnięcie tego przycisku wyłączy zasilacz - zaowocuje mianowicie wysłaniem liczby 15, potem 208 (rozkaz 13) i zamknięciem portu. Skąd takie dziwne liczby? A no z nikąd – ich wartość jest zupełnie nieistotna, ważne żeby były unikalne. Jeśli procesor odbierze liczbę 176, to podaje na nóżkę P3.7 stan niski, jeśli 208 – stan wysoki. A do portu P3.7 ... nic nie jest podłączone! Jedyne napis REL sugeruje, że chodzi o jakiś przekaźnik. Dlaczego jednak nie ma go na schemacie i płytce drukowanej? Otóż na tym etapie projektowania zasilacza powstał mały dylemat. Nasz zasilacz trzeba jakoś włączyć i wyłączać, ściślej – dołączać i odłączać od niego zasilany układ. W najprostszym przypadku można by realizować wyłączenie, podając po prostu na wyjście najmniejsze z możliwych napięć, co zresztą jest czynione. Zgodzicie się jednak wszyscy, że uznanie urządzenia, na którym występuje napięcie sporo ponad 1V, za odłączone od zasilacza jest grubą przesadą. Nasuwa się więc użycie przekaźnika włączanego w obwód wyjścia zasilacza. Popularne przekaźniki przeznaczone do pracy przy napięciu sieci 230V nie są dobrym rozwiązaniem, bo czerpią zwykle 50...150mA. Najlepszy byłby przekaźnik małosygnałowy ze względu na dużą rezystancję cewki, a co za tym idzie – mały pobór prądu. Jednak przekaźniki takie cechują się zadziwiającym wręcz zróżnicowaniem rozkładów wyprowadzeń i ich liczbą. Poza tym z doświadczenia wiem, że są one znacznie trudniej dostępne niż ich duzi koleździ, zwłaszcza w niewielkich sklepach elektronicznych w małych miastach. Gdybym więc zastosował jakiś konkretny typ lub nawet dodał w miarę uniwersalne miejsce na płytce, to i tak istniałoby wiele typów wręcz idealnie się nadających pod względem parametrów, które trzeba by wlotować „na drutach” lub na oddzielnej płytce. Ostatecznie uznałem, że na płytce drukowanej zasilacza nie będzie przekaźnika. Konkretny typ jaki zastosujecie należy zamontować na niewielkim kawałku płytki uniwersalnej wraz z tranzystorem (-ami) sterującym i (koniecznie!) diodą likwidującą przepięcia. Na płytce są stosowne punkty do pobrania napięć 5V i 15V oraz punkt REL. Przykładowo

– w modelu zastosowałem tani (2,50PLN) i prosty przekaźnik JZC pobierający przy nominalnym napięciu 12V prąd 30mA. Nie jest to może element o jakichś rewelacyjnych parametrach, ale nadaje się tu całkiem nieźle. Jest zasilany z wyjścia 15V poprzez szeregową diodę LED, co daje na cewce przekaźnika napięcie troszkę ponad 12V. Diodę można przy okazji wykorzystać w roli kontrolki. Sterowanie odbywa się za pomocą dwóch tranzystorów NPN włączonych jeden za drugim w konwencji wspólnego emitera. W obwodzie kolektora drugiego z nich jest przekaźnik i dioda. Pierwszy pełni rolę bufora między dwiema strefami napięć (5V – 15V) oraz inwertera – dzięki niemu przekaźnik jest włączany, gdy na P3.7 panuje stan niski, a nie wysoki. Dysponując przekaźnikiem na 5V wystarczy tu jeden typowo włączony tranzystor PNP.

No cóż – omówienie programu rozpocząłem niejako od końca. Nie wspominałem o ważnym zjawisku mającym miejsce zaraz po jego uruchomieniu – o skanowaniu wyjścia zasilacza. Skanowanie służy umieszczeniu na przyciskach odpowiadających im wartości napięć. Polega ono na tym, że komputer wybiera kolejne napięcia (1...9) i - po odczekaniu chwili na ich ustalenie się - odczytuje ich wartości, które następnie umieszcza na przyciskach. Użytkownik widzi wtedy małe okienko z paskiem postępu sygnalizującym zaawansowanie skanowania. Proces może być w każdej chwili przerwany, ale nie w połowie kroku. Dzięki skanowaniu przy starcie programu informacja na przyciskach jest zawsze aktualna. Choć trwa ono ok. 5s, nie jest uciążliwe, gdyż występuje jeden jedyny raz po uruchomieniu. Jeśli podczas korzystania z zasilacza okno programu jest chwilowo niepotrzebne, nie należy go zamykać, tylko zminimalizować – dwukrotne kliknięcie na ikonkę w Trayu natychmiast je przywraca. Samo skanowanie nie jest w żaden sposób niebezpieczne dla układu dołączonego do zasilacza o ile zastosujemy przekaźnik na jego wyjściu. Jednak znajdują się pewnie tacy, którzy go nie zastosują, zdając się na własną pamięć, aby podczas skanowania nie pozostawiać żadnych urządzeń podłączonych do zasilacza. Ponieważ pamięć ludzka jest zawodna, przed roz-

poczęciem tego procesu ukazuje się ostrzeżenie, mające na celu ustrzec ich przed skutkami pomyłki. Przy zastosowaniu przekaźnika ostrzeżenie to tylko niepotrzebnie demerwowałoby użytkownika – w takim przypadku należy je wyłączyć, zwierając do masy nóżkę 14 procesora U2 (P1.2). Nie polecam takich praktyk, gdy w układzie przekaźnika nie ma. Obecność przekaźnika jest niemal obowiązkowa.

Na koniec wyjaśnienie funkcji małego checkboksu znajdującego się po lewej stronie panelu z przyciskami. Jego zaznaczenie znacznie ułatwia proces regulacji zasilacza – sprawia, że po wybraniu któregośkolwiek z napięć wartość poprzednio ustawionego napięcia kopiowana jest na odpowiadającym mu przycisku. Dzięki temu bez konieczności ponownego skanowania widać, które napięcia są już wyregulowane, a które jeszcze nie. Podczas normalnej pracy opcja ta powinna być odznaczona, gdyż jedynie przeszkadzałyby, fałszując napisy przy szybkim przełączaniu napięć. Przyczyną jest niezbyt duża szybkość zmian napięcia na wyjściu U1 po skokowej zmianie rezystancji pomiędzy końcówką ADJ a masą.

Montaż i uruchomienie

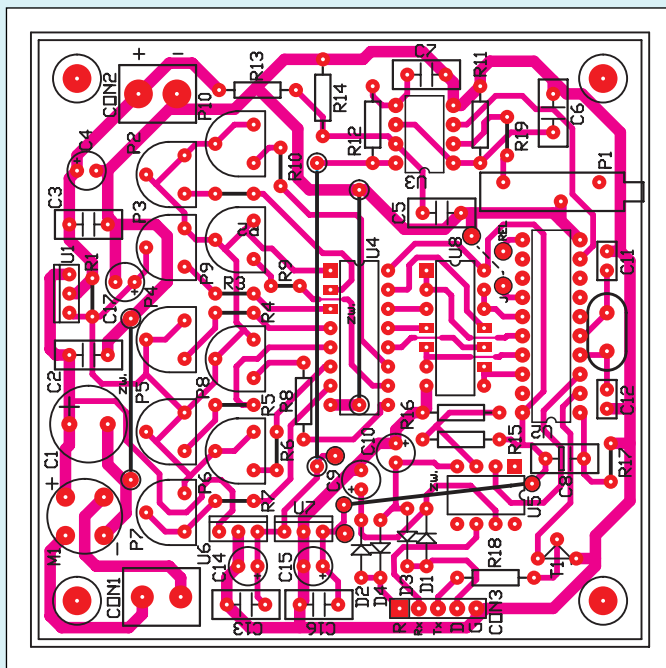
Schemat montażowy znajduje się na **rysunku 3**. Sam montaż jest typowy. Kolejność jest właściwie dowolna, najlepiej rozpocząć go od zwerek i podstawek pod układy scalone. Zależnie od wartości napięć jakie chcemy uzyskać, należy włutować rezystory R2...R10 i potencjometr P2...P10 o odpowiednich wartościach, zgodnie z prostym wzorem: $U_{wy} = 1,25V * (Q / (200R + 1))$; $R_n + P_n > Q > R_n$; $n = 2, \dots, 10$. W miarę możli-

wości należy wybrać potencjometr o małej wartości w porównaniu ze współpracującym rezystorem (zwiększy to precyzję regulacji), aby tylko suma ich rezystancji gwarantowała uzyskanie potrzebnego napięcia. Dla leniwych w wykazie podaję wyliczone wartości dla kilku popularnych napięć ustawionych w modelu.

Uruchomienie sprowadza się do wyregulowania przetwornika U/F za pomocą helitrima P1. W tym celu należy uruchomić program Zasilacz.exe – po skanowaniu na przyciskach pojawią się przypadkowe wartości. Na dowolnym kanale trzeba ustawić napięcie ok. 20V lub inne będące w okolicy największego napięcia jakie chcemy uzyskać z naszego zasilacza (ostatecznie może to być wręcz dowolne stosunkowo spore napięcie - liniowość LM331 okaże się i tak wystarczająco dobra).

ciąg dalszy na stronie 57.

Rys. 3 Schemat montażowy



Wykaz elementów

Rezystory:	
R1200Ω
R2240Ω
R3560Ω
R4680Ω
R51kΩ
R61,5kΩ
R72kΩ
R8,R9,R10dobraczależnieodpotrzeb
R11,R17100kΩ
R126,81kΩ 1%
R13,R14220kΩ
R15,R16,R1822kΩ
R1912kΩ
P1helitrim 5k
P247Ω PR
P3,P4100Ω PR
P5,P6,P7470Ω PR
P8,P9,P10PR (dobraczależnieodpotrzeb)
Kondensatory:	
C12200μF/25V
C2,C3,C13,C16100nF
C4,C9,C10,C14,C15100μF/25V
C5,C61μF MKT
C710nF MKT
C82,2μF/16V
C11,C1233pF
C1710μF/16V
Półprzewodniki:	
D1...D41N4148
M1mostek 1,5A
T1BC548B
U1LM317T
U289C2051
U3LM331 ('4151)
U474LS145
U5TL081
U67815
U77805
U84011
Inne:	
X111,0592MHz
CON1,CON2ARK2
CON3DB9F

Komplet podzespołów z płytką jest dostępny w sieci handlowej AVT jako kit szkolny AVT-2655.