

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.

Falownik trójfazowy

Prezentujemy urządzenie wkraczające w obszary mało przez nas (zazwyczaj) lubiane: wysokie napięcia, duże prądy i moce.



Projekt
057

Muszę zacząć od rozszyfrowania tej, dziwnie brzmiącej, nazwy: falownik trójfazowy. Falownik trójfazowy jest urządzeniem, które dostarcza na wyjściu napięcie trójfazowe, tzn. trzy fazy napięcia zmiennego o określonej częstotliwości, przesunięte względem siebie w czasie o kąt 120° .

Idea konstrukcyjna takiego falownika jest przedstawiona na rys. 1. Falownik, to po prostu sześć przełączników, które przy odpowiednim sterowaniu otwierają się i zamykają w określonym cyklu, co sprawia, że z punktu widzenia zacisków obciążenia otrzymujemy przebieg trójfazowy.

Całość zasilana jest ze źródła napięcia stałego, tak więc konstruując ten układ staramy się „oszukać” po-

tencjalne obciążenie, którym w większości przypadków będzie silnik indukcyjny trójfazowy asynchroniczny i zmusić go do wykonania pracy, której normalnie przy zasilaniu prądem stałym nie chciałby wykonać, a mógłby się nam co najwyżej spalić wydzielając przy tym duże ilości nieprzyjemnego zapachu.

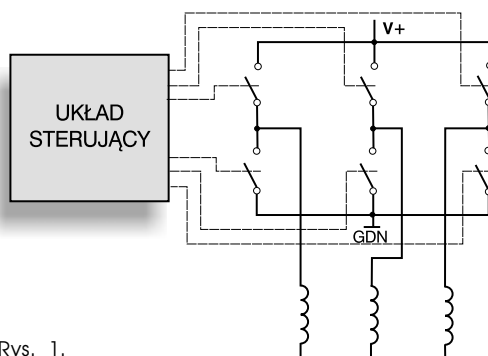
Skąd pomysł takiego układu i jakie jest jego potencjalne zastosowanie? Istnieją dwa główne powody, dla których zdecydowałem się na konstrukcję urządzenia. Po pierwsze, jest ono zasilane z sieci jednofazowej jaka najczęściej występuje w typowym mieszkaniu. Wykluczone jest bezpośrednie podłączenie silnika do sieci jednofazowej. Możemy tego dokonać po dołączeniu kondensatora, jednak jest to kłopotliwy sposób i powoduje znaczny spadek mocy silnika. Drugim, być może ważniejszym powodem przyjęcia takiego rozwiązania jest możliwość ustawienia dowolnej, oczywiście w rozsądnych granicach, prędkości wirowania wirnika. Prędkość ta zależy głównie od częstotliwości. Normalnie jest ona praktycznie niemożliwa do zmiany, gdyż przez zmianę napięcia zasilania uzyskamy bardzo mizerne rezultaty.

Teraz wymienię wady tego urządzenia, które niewątpliwie posiada i nieomówienie ich we wstępie byłoby po prostu nieuczciwe w stosunku do Czytelników.

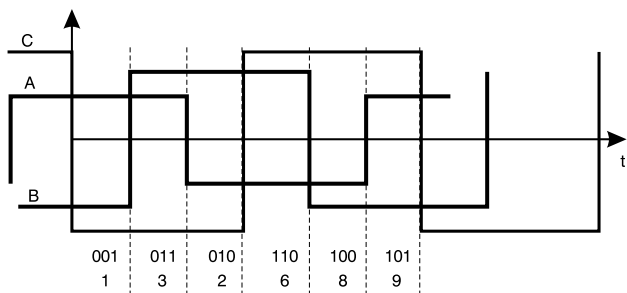
Układ posiada w zasadzie trzy zasadnicze wady (więcej wad niż zalet, mógłby stwierdzić uszczypliwy Czytelnik), jednak czy są to wady, i w jakim stopniu, pozostawiam do oceny Czytelnikom.

Pierwszą z nich jest to, iż kształt napięcia na wyjściu jest prostokątny, a nie sinusoidalny, jak to zazwyczaj bywa (rys. 2). Takie zasilanie powoduje, że nie możemy analizować naszego układu metodą symboliczną, a dane podawane przez producenta silnika na tabliczce znamionowej są dalekie od rzeczywistych.

Dokładną analizę pracy takiego silnika możemy przeprowadzić w oparciu o metodę widmową, po rozwinięciu naszego przebiegu w szereg Fouriera. Drugą wadą tej konstrukcji jest konieczność sterowania komputerem, co powoduje, że potencjalny użytkownik musi nabyć komputer, aby połączyć silnik trójfazowy. To zupełnie tak, jak kupowanie tokarki do otwarcia puszki konserw. Przyznacie więc, że kompletne nieporozumienie.



Rys. 1.



Rys. 2.

Muszę tu jednak bronić troszeczkę mojego pomysłu, bowiem kosztem zaprzęgnięcia do pracy komputera znacznie upraszcza się konstrukcja układu. Trzecią wadą jest cena urządzenia. Aby przybliżyć koszt całej zabawki, należy sobie powiedzieć, co tak naprawdę będzie pracowało w miejscu przełączników. Obecnie na rynku przyrządów półprzewodnikowych mamy w czym wybierać i w zasadzie wszystko się do takiego urządzenia nadaje.

Możemy zastosować np. tyrystory, wtedy jednak należałoby się zastanowić nad sposobem wyłączania załączonych tyrystorów, co jest możliwe, gdyż istnieją takie

sposoby, nie jest to jednak zagadnienie proste.

Tranzystory bipolarne możemy również z powodzeniem zastosować, jednak musimy się liczyć z ich dość kłopotliwym sterowaniem, praktycznie od zatkania aż do nasycenia. Dochodzi również fakt, że tranzystory bardzo niechętnie (długo) wychodzą ze stanu nasycenia. Jeżeli natomiast górny, nasycony tranzystor (z kolektorem na zasilaniu) nie zamknie się odpowiednio wcześniej, zanim otwori się dolny tranzystor (z emiterem na masie), to popłynie prąd zwarcia, który w najgorszym przypadku „usmaży“ tranzystory.

Można się oczywiście zastanowić także nad zastosowaniem przekaźników. Pierwsze komputery były wykonane z tysięcy przekaźników, jednak zbliża się XXI wiek, a to do czegoś zobowiązuje. Dlatego w moim rozwiązaniu postanowiłem wykorzystać jedną z nowszych zabawek półprzewodnikowych, jaką jest tranzystor polowy MOS z kanałem indukowanym typu n. Zastosowanie tego przyrządu jest stosunkowo proste, straty mocy na elemencie są niewielkie, jednak cena jednego tranzystora jest stosunkowo wysoka. Cena użytych tranzystorów typu BUZ90 waha się, w zależności od dostawców, od czterech do czternastu złotych. O cenie całego urządzenia decyduje głównie cena sześciu tranzystorów polowych i sześciu wysokonapięciowych tranzystorów bipolarnych.

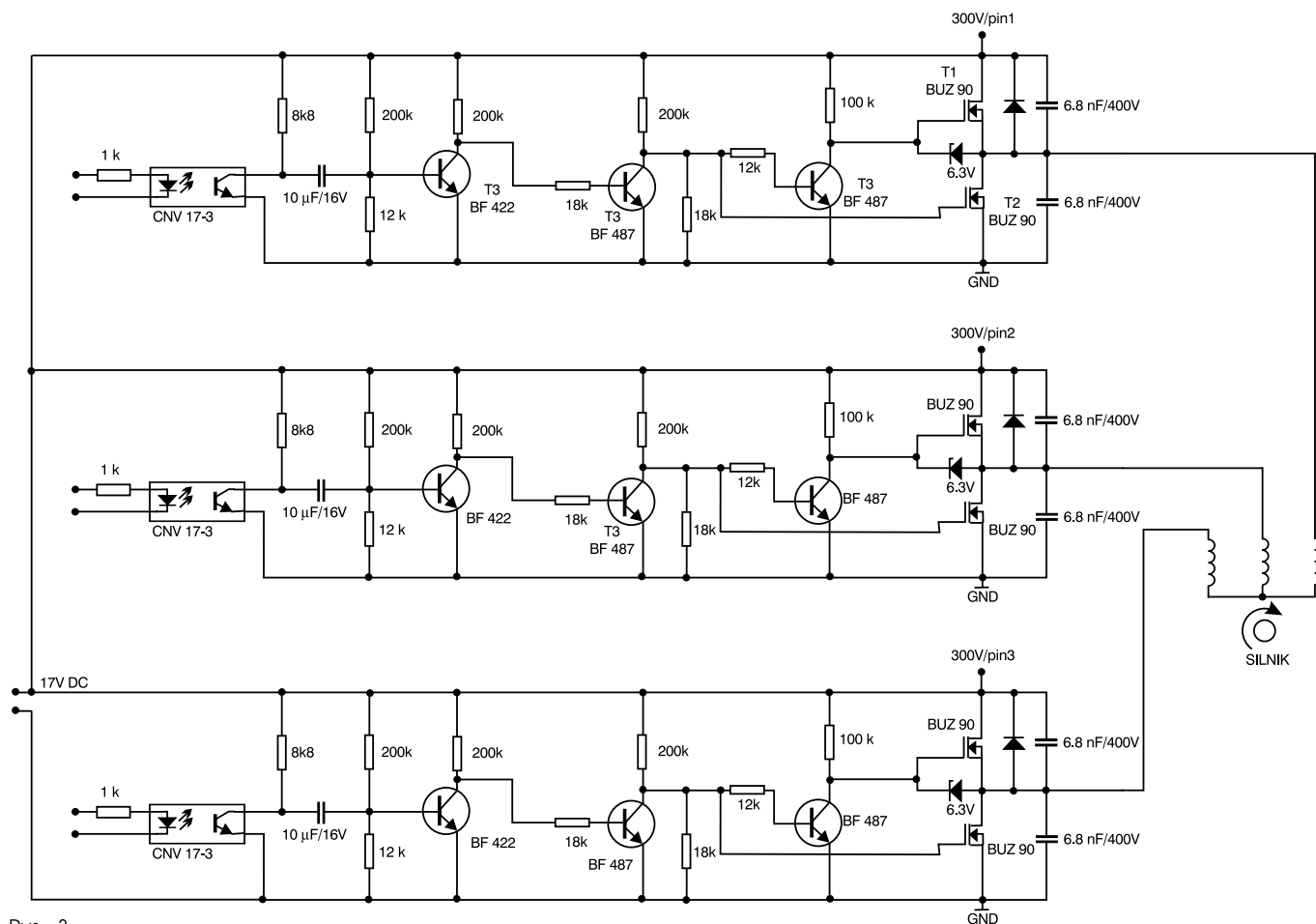
Falownik jest sterowany przez komputer PC z odpowiednim programem (list. 1). Komputer steruje falownikiem przez złącze drukarki. Za pomocą komputera zała-

czamy silnik oraz podajemy częstotliwość, która decyduje o prędkości obrotowej wirnika. Należy jednak pamiętać, iż zwiększając częstotliwość ponad 50Hz nie mamy gwarancji, że konstrukcja mechaniczna silnika jest wystarczająco wytrzymała. Natomiast obniżając częstotliwość pamiętajmy, iż przez uzwojenia popłynie większy prąd niż prąd znamionowy, co może spowodować spalenie uzwojeń.

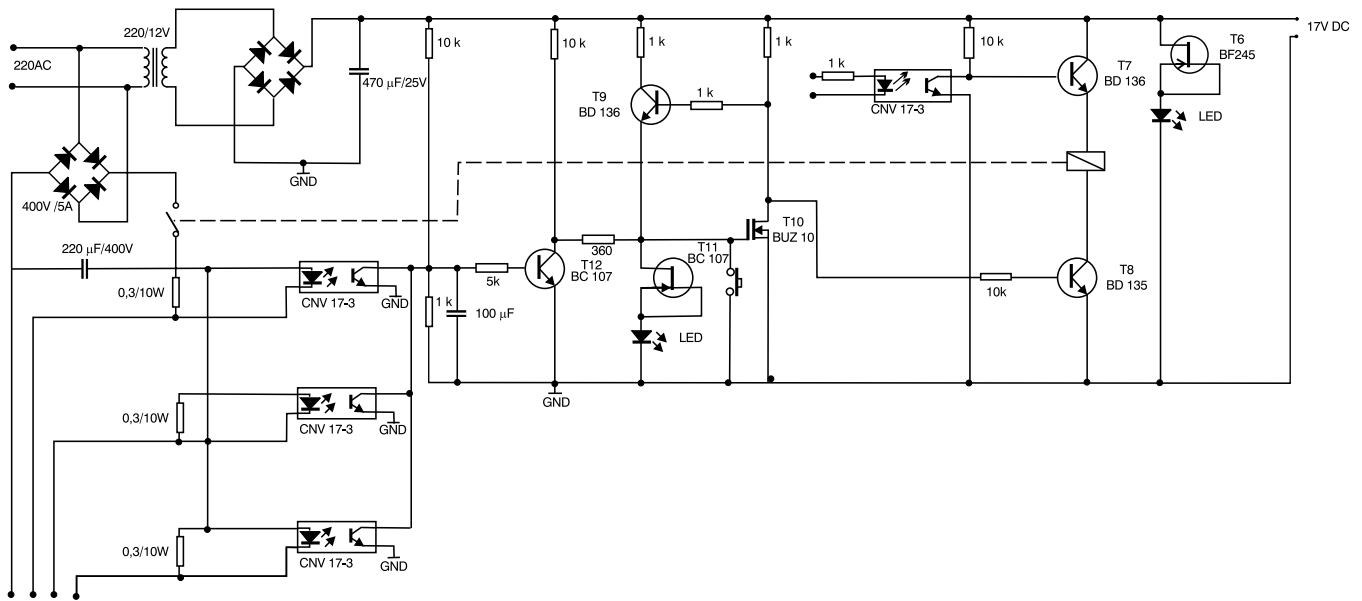
Jak to działa?

Patrząc na schemat głównej części falownika (rys. 3) dostrzegamy, że jest to konstrukcja bardzo prosta i w zasadzie nie bardzo jest co omawiać. Każdy z trzech bloków falownika składa się z przedwzmacniacza oraz dwóch tranzystorów zasilanych z wysokiego napięcia, które sterują bramkami tranzystorów polowych.

Należy jednak zwrócić uwagę na kilka elementów. Pierwszym rzucającym się w oczy rozwiązaniem konstrukcyjnym, które chciałbym wyjaśnić, jest zastosowanie



Rys. 3.



Rys. 4.

transoptorów. Przyjąłem je tylko i wyłącznie ze względu na bezpieczeństwo komputera. Uruchamiany układ jest układem dużej mocy, pracującym przy dużym napięciu, a w takim układzie wszystko się może zdarzyć, szczególnie niedoświadczonym elektronikom. Dlatego chcę tu bardzo silnie podkreślić, aby nie rezygnować z takiego rozwiązania, bo koszty tej decyzji są łatwe do wyliczenia. Należy zauważyć, że komputer jest galwanicznie separowany od urządzenia, co zapewnia maksimum jego bezpieczeństwa.

Drugim problemem, który pragnę omówić, jest zabezpieczenie tranzystorów polowych. Tranzystory te są zabezpieczone ze względu na dwa niebezpieczne zjawiska, które bezpowrotnie niszczą tranzystory. Pierwsze to powstawanie dużych szpilek napięcia przy wyłączaniu tranzystorów obciążonych indukcyjnością. Aby zapobiec niszczącym następstwom tego zjawiska, tranzystory są zabezpieczone diodami i kondensatorami włączonymi między źródłem a drenem. Drugą drogą do szybkiego i skutecznego uszkodzenia tych tranzystorów jest podanie na bramkę napięcia przekraczającego 20V (dla BUZ90). Dlatego bramka została zabezpieczona diodami Zenera. Ponieważ tranzystor łączy się po napięciu ok. 4V, dlatego też zastosowałem diody 6,2V, które wyeliminowały

zjawisko przebijania bramek. To by były wszystkie uwagi dotyczące głównej części falownika. Należałoby teraz powiedzieć kilka słów o zasilaniu układu.

W obwodzie zasilania układu można by zastosować mostek Graetza i kondensator. Postanowiłem jednak trochę rozbudować tę część projektu. Zasilacz (rys. 4) został wyposażony w zabezpieczenie nadprądowe oraz załączanie napięcia na układ sterowany komputerem. Elementem wykonawczym jest krytykowany wcześniej przekaźnik, jednak w tym zastosowaniu wydaje się on być odpowiedni. Cewka przekaźnika jest sterowana dwoma tranzystorami T7 i T8 i tylko przy załączeniu obydwu przekaźnik zostanie załączony i poda napięcie na układ. Tranzystor T7 odpowiada za załączenie przekaźnika przy podaniu sygnału z komputera, natomiast T8 za odłączenie przekaźnika w razie przekroczenia dopuszczalnego poboru prądu. Układ zabezpieczenia nadprądowego powinien być zaprojektowany tak, aby nie został przekroczony maksymalny prąd tranzystorów, gdyż układ ten ma zabezpieczać tranzystory, a nie podłączać obciążenie. Badanie prądu następuje przy wykorzystaniu transoptorów. Jeżeli napięcie na rezystorze 0,3Ω przekroczy 0,6V, to transoptor spowoduje przejście

układu w stan, w którym zostaje zatkany tranzystor sterujący T8. Aby system działał poprawnie, należało skonstruować układ, który po podaniu impulsu z transoptora przechodziłby w stan zatkania tranzystora sterującego i pozostawał w tym stanie. Jest to bardzo ważne, gdyż gdyby nie został zastosowany taki system zatrasku, to układ po prostu włączałby się i wyłączał, gdyż po podaniu impulsu z transoptora i rozłączeniu przekaźnika znikałaby przyczyna rozłączenia i układ ponownie załączałby przekaźnik. Układ działa następująco: normalnie tranzystor T12 jest nasycony, a T10 jest zatkany, gdyż ma niski potencjał na bramce. Przy zatkaniu tranzystora T12 przez transoptor następuje załączenie tranzystora T10, jednak prąd płynący przez jego dren powoduje załączenie tranzystora T9, który podaje wysoki potencjał na bramkę, zatraskując układ stabilnie. Układ można ponownie przywrócić do poprzedniego stanu przez wciśnięcie przycisku *reset*.

Oprogramowanie

Pozostał jeszcze jeden problem do rozwiązania: jak zmusić komputer do sterowania falownikiem? Do tego celu został wykorzystany standardowy port drukarkowy. Jeżeli kogoś interesuje dlaczego właśnie ten port, a nie jakiś inny, to wyjaśnię, że informatyka jest

moją słabą stroną, a obsługa portu drukarkowego wydała mi się najprostszą.

Cały sekret sterowania polega na tym, że w komputerze PC liczba zapisana pod ściśle określonym adresem jednocześnie znajduje się na wyjściu portu drukarki. Aby więc zmienić stan wyjścia drukarki, wpisujemy do tej komórki nową wartość. Program został napisany w języku C, a instrukcja pozwalająca wpisać wartość pod określony adres to *outp* (adres, wartość). Aby więc uzyskać sterowanie trójfazowe, do naszego układu należy przekazać określoną sekwencję liczb, w odpowiednich odstępach czasowych. Aby znaleźć odpowiednią sekwencję, należy odpowiedzieć sobie na pytanie, jaka jest zmienność przebiegu trójfazowego w czasie, podczas jednego okresu. Nazwijmy trzy kolejne fazy odpowiednio A, B, C. Przyjmijmy, iż okres rozpoczyna się, gdy A jest w stanie wysokim, a B i C w niskim. Wpisana sekwencja binarna to 001, a dziesiętna to 1. W następnej fazie A i B są w stanie wysokim, a C w niskim, więc sekwencja binarna to 011, a dziesiętna to 3. Następnie B pozostaje w stanie wysokim, a A i C jest w niskim, więc 010 i 2 itd.

Zasadę tę wyjaśnia rys. 2. Po uwzględnieniu sygnału załączającego układ, jedynka na najmłodszym bicie, otrzymamy sekwencje:

11, 3, 7, 5, 13, 9. Czas trwania odpowiedniego stanu określa częstotliwość przebiegu, więc przez zmianę tej wartości regulujemy częstotliwość. Oto główna część programu:

```

outp (888, 11) ;
delay (f) ;
outp (888, 3) ;
delay (f) ;
outp (888, 7) ;
delay (f) ;
outp (888, 5) ;
delay (f) ;
outp (888, 13) ;
delay (f) ;
outp (888, 9) ;
delay (f) ;
if (a) break ;
outp (888, 11) ;
delay (i) ;
outp (888, 3) ;
delay (i) ;
outp (888, 7) ;
delay (i) ;
outp (888, 5) ;
delay (i) ;
outp (888, 13) ;
delay (i) ;
outp (888, 9) ;
delay (i) ;
a=kbhit () ;
if (a) break
}

```

```

clrscr () ;
printf ("PRACA @d Hz/njesli
chcesz przerwac wcisnij dowolny
klawisz", b) ;
for (;)
{
if(a) break;
outp (888,11);
delay (f) ;
outp (888, 3) ;
delay (f) ;
outp (888, 7) ;
delay (f) ;
outp (888, 5) ;
delay (f) ;
outp (888, 13) ;
delay (f) ;
outp (888, 9) ;
delay (f) ;
}

```

```

delay (f) ;
a=kbhit () ;
if (a) break ;
}
clrscr () ;
outp (888, 0) ;
printf ("STOP\np-powrot w-wy-
jscie") ;
getch () ; a=getch () ;
if (a=='w'@@a=='W') break;
}
}

```

Wyjaśnienia wymagają jeszcze dwa ważne szczegóły programu. Pierwszym jest to, iż parametr instrukcji *delay* nie jest wartością bezwzględną, a zależy od użyciego komputera. Dlatego też w moim programie wartość ta przechowywana jest w pliku i przed pierwszym włączeniem układu należy wpisać taką liczbę, przy której częstotliwość na wyjściu będzie się zgadzać z częstotliwością zadaną z klawiatury. Drugim problemem, o którym pragnę wspomnieć, jest rozruch silnika. Program po uruchomieniu zmienia płynnie częstotliwość od 20Hz do częstotliwości zadanej, co powoduje znacznie szybszy rozruch oraz ogranicza prąd rozruchowy o około 50%. Dalsze ograniczenie prądu rozruchu można próbować osiągnąć przez podział impulsów poszczególnych faz na krótsze, co spowoduje, iż amplituda podstawowej harmonicznej zmaleje. Dla wyższych harmonicznych indukcyjność silnika stanowi duży opór indukcyjny, dlatego tą drogą można próbować przeprowadzać rozruch przy małym prądzie początkowym. Wymaga to jednak znacznej komplikacji programu.

Kończąc ten artykuł pragnę poinformować, iż układ był testowany i działał poprawnie przy obciążeniu silnikiem 0,25kW. Jeszcze jedna praktyczna uwaga: podczas prac nad urządzeniem rozsądnie jest się posługiwać transformatorem z regulowanym napięciem wyjściowym (autotransformatorem), co znacznie ograniczy niebezpieczeństwo porażenia prądem. Zainteresowanym osobom udzielię bardzo chętnie informacji dodatkowych oraz prześlę gotowy skompilowany program.

Sposób obsługi falownika

1. Przed pierwszym włączeniem urządzenia uruchom program, podaj częstotliwość i sprawdź czy zgadza się ona z częstotliwością na wyjściu portu drukarki. Jeśli nie, to zmień wartość w pliku kalibracja (po liczbie musi być wpisany symbol*). Należy powtarzać operację aż do uzyskania poprawnego działania systemu. Zmian można dokonywać bezpośrednio w "Nortonie".

2. Podłącz urządzenie do komputera oraz do silnika.

3. Włącz komputer oraz załaduj program.

4. Włącz zasilanie urządzenia.

5. Możesz uruchomić silnik.

6. Nie zamykaj programu zanim nie wyłączysz urządzenia z sieci energetycznej!!! Tylko taka kolejność czynności gwarantuje poprawność zadziałania urządzenia. Niezastosowanie się do niniejszej instrukcji może spowodować uszkodzenie falownika. Czerwona dioda LED oznacza, iż zadziałało zabezpieczenie nadprądowe odłączając napięcie. Jednak zaraz po włączeniu do sieci układ automatycznie ustawi się w stan odcięcia i należy go zresetować. Jest to zachowanie normalne. Jeżeli jednak układ po wciśnięciu klawisza reset dalej przechodzi w stan zablokowania, to oznacza, iż pobór prądu jest większy od dopuszczalnego.

Sławomir Gruszczyński

Wszelkie uwagi, wnioski i komentarze proszę kierować pod adresem: Wrocław, ul. Wyszyńskiego 106/8, tel. (071) 72-11-31.

List. 1.

```

#include<stdio.h>
#include<conio.h>
#include<dos.h>
void main(void)
{
FILE*plik;
char k[1],kalib[10];
int a,b,i,l,n;
double f;
float h=0;
outp(888,0);

clrscr();
plik=fopen("a:kalibrac","r[t]");
for(i=0;i<10;i++)
{kalib[i]=0;
for(i=0;i<10;i++)
{
k[0]=fgetc(plik);
if(k[0]=='') break;
kalib[i]=k[0];
}

kalib[i]='\0';
i=0;
while(kalib[i]!='\0')
{
h=10*h+kalib[i+1]-'0';
}

for(;;)
{
clrscr();
printf("
STEROWANIE SILNIKOW
TROJFAZOWYCH\n");
printf("PODAJ CZESTOTLIWOSC (70-95
Hz) ");
scanf("%d",&b);
if(b<5||b>70) continue;

f=h/b;

for(i=1;i<11;i++)
{
clrscr();
printf("START ZA %d
SEKUND",11-i);
printf("\n jesli chcesz
przerwac wcisnij dowolny klawisz");
a=kbhit();
if (a) break;
delay(h);
}

clrscr();
printf("ROZRUCH\n jesli chcesz
przerwac wcisnij dowolny klawisz");
n=20;
for(i=h/n;i>h/b;i=(h/(n++)))
{
if(a) break;
outp(888,11);
delay(i);
outp(888,3);
delay(i);
outp(888,7);
delay(i);
outp(888,5);
delay(i);
outp(888,13);
delay(i);
outp(888,9);
delay(i);
a=kbhit();
if(a) break;
}

clrscr();
printf("PRACA %d Hz\njesli chcesz
przerwac wcisnij dowolny
klawisz",b);
for(;;)
{
if(a) break;
outp(888,11);
delay(f);
outp(888,3);
delay(f);
outp(888,7);
delay(f);
outp(888,5);
delay(f);
outp(888,13);
delay(f);
outp(888,9);
delay(f);
a=kbhit();
if (a) break;
}

}
clrscr();
outp(888,0);
printf("STOP\np-powrot w-
wyjscie");
getch();a=getch();
if(a=='w'||a=='W') break;
}
}

```