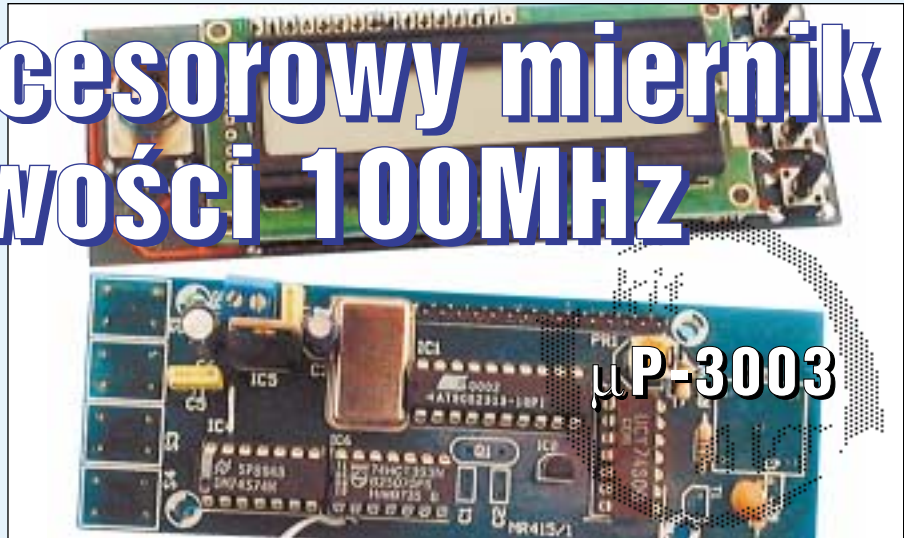


Mikroprocesorowy miernik częstotliwości 100MHz



Chciałbym zaproponować Czytelnikom Elektroniki dla Wszystkich budowę jednego z najbardziej użytecznych przyrządów laboratoryjnych: miernika częstotliwości. Nie ma chyba wśród Was nikogo, kto chciałby podać w wątpliwość celowość konstruowania takiego urządzenia. Przez określenie "miernik częstotliwości" rozumiemy przecież nie tylko przyrząd badający częstotliwość jakiegoś przebiegu, ale także urządzenie mierzące inne wartości, takie jak liczby zarejestrowanych impulsów czy też czas trwania pojedynczego impulsu. Schemat elektryczny miernika częstotliwości o zakresie pomiarowym 1Hz ... 100MHz został pokazany poniżej na **rysunkach 1 i 2**. Jak łatwo zauważyć, układ jest stosunkowo prosty i zawiera nie więcej niż ... 36 układów scalonych. Jest to konstrukcja z zamierzonej epoki: sprzed ponad ośmiu lat. Wspomniany miernik wykonany był w setkach egzemplarzy, był także tematem niejednej pracy dyplomowej. Po co zatem wspominam tego dinozaura? Po to, aby w dobitny sposób ukazać Wam, jakie możliwości kryje w sobie technika mikroprocesorowa. Na **rysunku 3** został pokazany schemat miernika, z którego budową zapoznamy się za chwilę. Został on zbudowany z wykorzystaniem w zasadzie tylko jednego układu scalonego, o którym warto wspominać: mikroprocesora RISC najnowszej generacji typu AT90S2313. Miernik wykonany z wykorzystaniem tylko tego układu dysponowałby zakresem pomiarowym do ... 5MHz! W większości przypad-

ków hobbyści uznaliby taki zakres za zupełnie wystarczający, ale postanowiłem jednak skonstruować urządzenie w niczym nie ustępujące swojemu poprzednikowi i dodałem do układu banalnie prosty preskaler, zrealizowany na dwóch tanich układach TTL. Preskaler ten rozszerza zakres pomiarowy miernika właśnie do "magicznych" 100MHz.

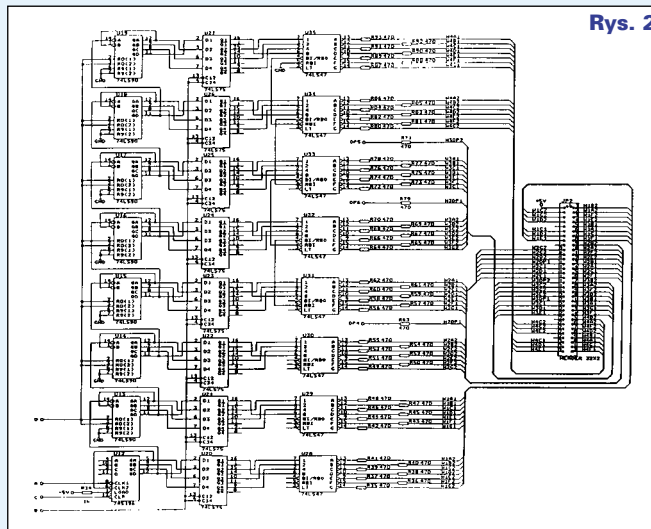
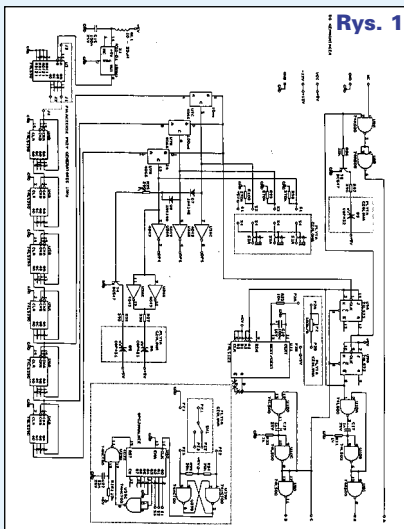
A zatem widzimy już pierwszą i chyba najważniejszą korzyść z zastosowania do budowy miernika nowoczesnego mikroprocesora. Zmieniliśmy 36 układów scalonych na trzy, no powiedzmy na pięć, jeżeli doliczymy jeszcze stabilizator napięcia (zbędny, jeżeli do zasilania miernika wykorzystamy zewnętrzny zasilacz stabilizowany +5VDC) i trzynóżkowy układzik resetu. Nie jest to jednak jedyna korzyść, jaką wyciągnęliśmy z zamiany sterty archaicznych TTL-i na nowoczesny procesor. Nie będę Wam pokazywał rysunków płytek obwodów drukowanych do protoplasty naszego miernika mikroprocesorowego, a to z prostego powodu: ich rysunki zajęłyby ze dwie strony naszego pisma! Popatrzcie teraz na rysunek przedstawiający płytkę obwodu drukowanego miernika mikroprocesorowego. Płytkę swobodnie mieści się na dłoni, wyko-

nana jest na tanim laminacie jednostronnym. Nie zniżajmy się nad pokonaniem i tym razem także nie porównujemy kosztów wykonania PCB do obydwóch urządzeń!

Napisałem, że zakres pomiarowy obydwu przyrządów jest jednakowy. Jest to prawda, ale nie cała prawda. W pamięci procesora naszego nowego miernika zawarty jest program, realizujący także inne funkcje, nie tylko prosty pomiar częstotliwości. W miernikach starszego typu zakresy pomiarowe trzeba było przełączać najczęściej ręcznie. W nowym układzie procesor sam ocenia, na jakim z dwóch zakresów ma pracować, a wybór sygnalizuje odpowiednim napisem na wyświetlaczu LCD. Nowy przyrząd może służyć także jako licznik zdarzeń o imponującym zakresie: do 2 147 483 646 impulsów!

W "dziadku" naszego miernika do obrazowania wyników pomiaru zastosowane były wyświetlacze siedmiosegmentowe LED. W nowej konstrukcji zdecydowałem się na użycie taniego wyświetlacza alfanumerycznego LCD. Korzyści płynące z tej decyzji także są ogromne. Nie będę nawet rozwodził się nad nieporównywalnie mniejszym zużyciu energii i o tym, że nasz przyrząd może być zasilany nawet z baterii. Najistotniejszą zaletą wyświetlacza LCD zastosowanego w każdym przyrządzie pomiarowym jest to, że można na nim ukazywać nie tylko cyfry, ale i dowolne inne znaki. Na szesnastopozycyjnym polu odczytowym zmieści się nie tylko wynik pomiaru, ale i stosowny komentarz, informujący np. czy odczyt dokonywany jest w hercach czy kilohercach.

Zanim przejdziemy do szczegółowego opisu proponowanego układu,



chciałbym wspomnieć jeszcze o jednej, bardzo ważnej sprawie. Układ miernika częstotliwości jest z jednej strony gotowym, wymagającym jedynie nieskomplikowanego montażu urządzeniem. Można go zbudować w ciągu krótkiego czasu, włożyć w podstawkę zakupiony procesor z zaszytym w nim programem i spokojnie przystąpić do eksploatacji własnoręcznie wykonanego miernika. Możemy jednak spojrzeć na proponowaną konstrukcję z innej strony i potraktować ją jako tworzywo, glinę, z której dopiero ulepimy potrzebne nam urządzenie. Wiemy już sporo o programowaniu procesorów '51, a jak możecie się przekonać, programowanie układów AVR z poziomu pakietu BASCOM AVR praktycznie nie różni się od programowania "pięćdziesiątek jedynek". Jeszcze kilka wskazówek dla tych Czytelników, którzy zechcieliby wykonać opisany w tym artykule przyrząd. Macie następujące możliwości:

1. Zakupić płytkę obwodu drukowanego i zaprogramowany procesor. Jest to metoda najprostszą, ale także najkosztowniejszą.

2. Zakupić tylko płytkę i zaprogramować procesor programem, którego kod źródłowy zostanie opublikowany w Internecie.

3. Zaprogramować procesor samodzielnie napisanym programem lub przerobionym programem "fabrycznym".

W pierwszym przypadku nie napotkamy na najmniejsze trudności podczas pracy i po najdalej kilku godzinach staniemy się posiadaczami niezłej klasy miernika częstotliwości. Natomiast podczas samodzielnego programowania procesora napotkamy na pewien, dość trudny do przezwyciężenia problem. Wersja DEMO pakietu BASCOM AVR posiada ograniczenie kodu wynikowego do 1kB i program napisany, co całkowicie uniemożliwia zaprogramowanie procesora napisanym przez mnie programem. Prowadzę wprawdzie w tej chwili rozmowy z firmą MCS Electronics w sprawie udostępnienia Czytelnikom EP i EdW rozszerzonej wersji Demo pakietu BASCOM AVR, której możliwości będą podobne do możliwości BASCOM-a SE-EP. Nie mogę jednak w tej chwili obiecać Wam niczego konkretnego i zanim nie ukaże się nowa wersja BASCOM-a AVR, macie do dyspozycji następujące sposoby rozwiązania problemu długiego kodu wynikowego:

1. Rozwiązanie optymalne: zakupie-

nie pełnej wersji BASCOM-a AVR, która nie posiada żadnych ograniczeń i umożliwi programowanie nawet procesorów AT MEGA z 128kB pamięci programu.

Rozwiązanie "rozpacziwe": spróbować przerobić program tak, aby zajął nie więcej niż 1024B pamięci EEPROM procesora. Można np. spróbować zrezygnować z układu automatyki lub uprościć niektóre procedury.

Kod źródłowy napisanego przeze mnie programu zostanie podany do wiadomości publicznej i pomimo że wcale nie uważam się za bardzo dobrego programistę, może stanowić bazę do stworzenia Waszych własnych programów. Nic za tym nie stoi na przeszkodzie, aby w dowolny sposób modernizować układ miernika, dostosowując go do swoich potrzeb. Macie tu ogromne pole do popisu: dysponując hardware możecie do niego dorabiać praktycznie dowolny software, a granicą takich poczynań może być tylko Wasza wyobraźnia. Reasumując: program obsługujący nasz miernik będziecie mogli dowolnie zmieniać, modyfikować lub napisać własny posługując się wyłącznie darmowym oprogramowaniem i banalnie prostym i tanim sprzętem. Równie dobrze możecie także po prostu włożyć zaprogramowany "fabrycznie" procesor w podstawkę i po prostu korzystać z wykonanego przyrządu.

Ważnym elementem jest tutaj wybór odpowiedniego procesora. W tym celu należy wziąć pod uwagę przede wszystkim pojemność pamięci EEPROM i pamięci programu. Wersja 1024B EEPROM jest dla nas za mała, ponieważ program obsługujący nasz miernik będziecie mogli dowolnie zmieniać, modyfikować lub napisać własny posługując się wyłącznie darmowym oprogramowaniem i banalnie prostym i tanim sprzętem.

Opis układu

Schemat elektryczny miernika częstotliwości, długości impulsu i licznika zdarzeń został pokazany na rysunku 3. Sercem układu jest zaprogramowany procesor typu AT90S2313, nowoczesny chip o architekturze RISC, o bardzo dużej szybkości pracy. Z pewnością już zauważyliście, że układ ten jest "pinowym"

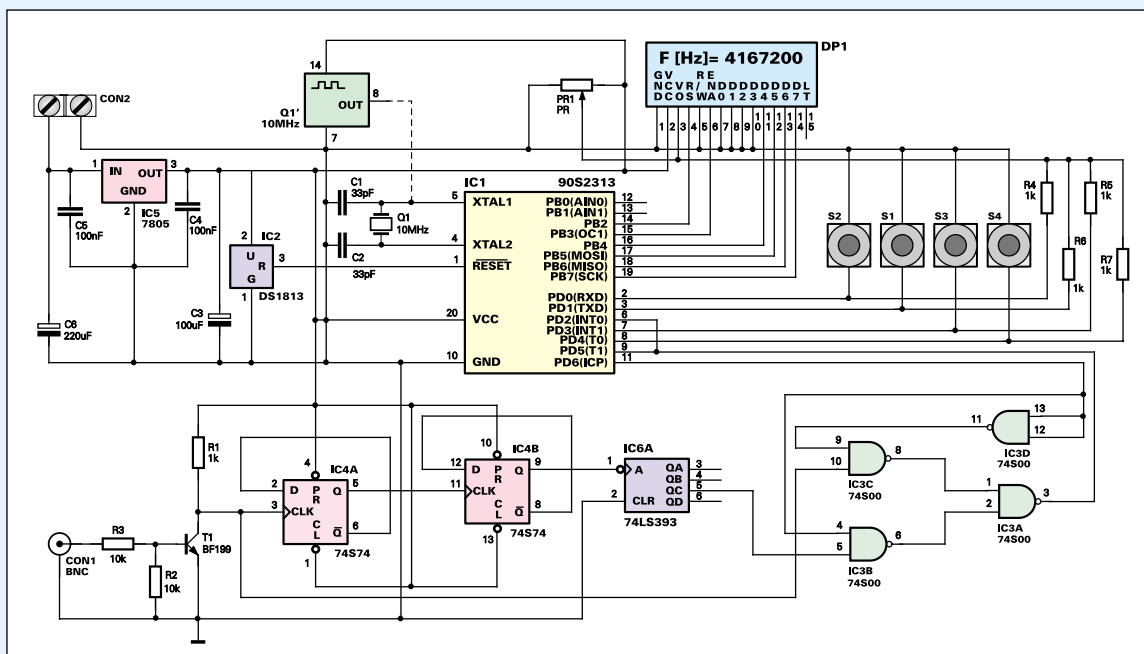
odpowiednikiem dobrze Wam znanego procesora AT89C2051. Co więc spowodowało, że do budowy miernika zastosowałem ten właśnie układ, a nie '2051? Powodem tej decyzji jest znacznie większa szybkość pracy procesorów AVR. Wprawdzie maksymalna częstotliwość rezonatora kwarcowego, jaki możemy dołączyć do AVR-a wynosi tylko 10MHz (czyli trzykrotnie mniej niż do '51), ale w procesorach AVR częstotliwość wewnętrznego oscylatora nie jest, tak jak w '51 dzielona przez 12. Tak więc procesor z kwarcem 10MHz pracuje co najmniej czterokrotnie szybciej niż „pięćdziesiątka jedynka“ z kwarcem 30MHz! Calowo napisałam "co najmniej" ponieważ w rzeczywistości szybkość pracy AVR-a jest znacznie większa, ponieważ większość instrukcji wykonywanych jest w nim w jednym cyklu maszynowym.

Nie będziemy tu wnikać w szczegóły budowy procesora '2313. Podam Wam tylko jego najważniejsze dane techniczne:

Napięcie zasilania (wersja 4)	2,7 ... 6VDC
Napięcie zasilania (wersja 10)	4 ... 6VDC
Częstotliwość zegara systemowego (wersja 4)	0 ... 4MHz
Częstotliwość zegara systemowego (wersja 10)	0 ... 10MHz
Pobór prądu przy 4 MHz	2,8mA
2kB pamięci EEPROM programu	
128B pamięci danych	
128B nieulotnej pamięci danych EEPROM	
Jeden 8 bitowy timer z preskalerem	
Jeden 16 bitowy timer z preskalerem	
Programowany Watchdog z preskalerem	
Wbudowany interfejs SPI do programowania układu w uruchamianym systemie	

Badany przebieg podawany jest na wejście CON1. Ze względu na zastosowanie tranzystora wejściowego T1, amplituda tego przebiegu może mieścić się w standardzie TTL i może być od niego zarówno mniejsza, jak i większa. Impulsy prostokątne podawane na wejście miernika kierowane są do dwóch

Rys. 3



punktów układu: do przełącznika zbudowanego "po bożemu" na szybkich bramkach NAND TTLS - IC3 i na wejście wstępnego dzielnika częstotliwości - preskalera zbudowanego także tradycyjnymi metodami: z dwóch szybkich przerzutników D typu 74. Przełącznik zbudowany na bramkach NAND sterowany jest z wyjścia PD.6 procesora. Jeżeli na tym wyjściu utrzymuje się stan niski, to do wejścia PD5(T1) procesora dociera przebieg pobierany bezpośrednio z wejścia CON1. Przy stanie wysokim na wyjściu sterującym pracą przełącznika częstotliwość wejściowa przed skierowaniem jej na wejście PD5(T0) procesora ulega podziałowi w preskalerze.

Zastanówmy się teraz, jaki sens ma stosowanie przełącznika i alternatywne dzielnie badanej częstotliwości. Timery - liczniki procesora '2313 pracującego z oscylatorem o częstotliwości 10MHz są w stanie zliczać przebiegi zewnętrzne o maksymalnej częstotliwości 5MHz. Wynika to z następującego uwarunkowania:

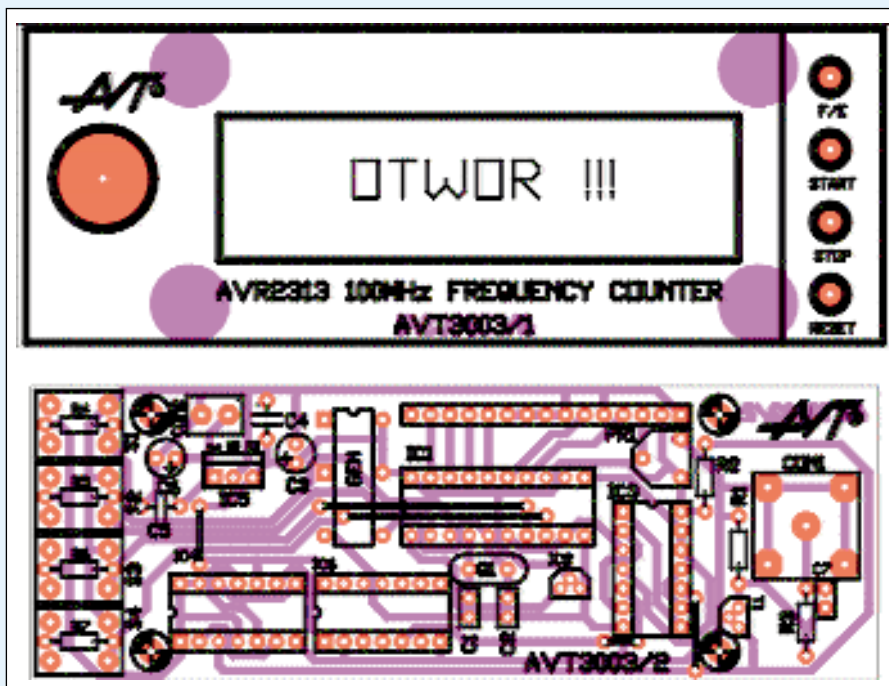
Przy pracy w trybie timera inkrementacja licznika następuje w każdym kolejnym cyklu maszynowym, a zatem maksymalna częstotliwość zliczania jest równa częstotliwości zegarowej mikrokontrolera. W przypadku wykorzystywania układu w trybie licznika, jego zawartość jest zwiększana w odpowiedzi na opadające zbocze sygnału wejściowego. Detekcja zbocza odbywa się jednak synchronicznie z cyklem pracy mikroprocesora - przez testowanie stanu w odpowiedniej linii wejściowej w każdym kolejnym cyklu maszynowym. Jeśli testowanie wykazuje stan wysoki linii w jednym cyklu maszynowym oraz stan niski linii w następnym cyklu maszynowym, zawartość licznika jest zwiększana. Tak więc, aby zagwarantować wykrycie wszystkich impul-

sów, każdy (zarówno niski, jak i wysoki) stan testowanej linii wejściowej musi trwać co najmniej jeden pełny cykl maszynowy. Skutkiem tego maksymalna częstotliwość pracy układu w trybie licznika jest ograniczona do 1/2 częstotliwości zegarowej mikrokontrolera. Warto tu zwrócić uwagę na fakt, że w przypadku procesorów '51 ograniczenie to wynosiłoby 1/24 częstotliwości oscylatora mikroprocesora!

Tak więc sens stosowania preskalera jest już chyba oczywisty: przebiegi o częstotliwości nieco mniejszej od 5MHz będziemy mierzyć podając je bezpośrednio na wejście timera procesora, a przebiegi większe będą ulegać podziałowi przez 16. Dlaczego jednak wybrałem właśnie taki stopień podziału i jak działa preskaler? Założyłem, że maksymalna częstotliwość mierzona przez nasz przyrząd powinna wynosić 100MHz. Ten wybór nie był przypadkowy, ponieważ jest to maksymalna częstotliwość, jaką możemy wprowadzić na tanie i łatwo dostępne układy serii TTL-S, bez stosowania stosunkowo kosztownych i trudniejszych do zdobycia preskalerów wykonanych w technologii.

Częstotliwość podawana na wejście preskalera jest wstępnie dzielona przez 4 za pomocą dwóch połączonych szeregowo przerzutników typu D, pracujących w układzie dwójek liczących. Dalszy podział następuje w liczniku binarnym typu 74LS393 - IC5A. Ponieważ przebieg pobierany jest w z wyjścia QC tego licznika, częstotliwość wejściowa zostanie ostatecznie podzielona przez 32. A więc, zakładając, że na wejście miernika podana zostanie częstotliwość maksymalna - 100MHz, to na wejście timera procesora dotrze jedynie 3 125 000Hz, czyli wartość akceptowana "z zapasem" przez timer.

Rys. 4 Schemat montażowy



Zanim przejdziemy do pobieżnej analizy programu sterującego pracą miernika, wspomnijmy jeszcze o dwóch elementach widocznych na schemacie. Mam tu przede wszystkim na myśli podejrzaną kombinację dołączoną do wejść oscylatora procesora. Widzimy tam dwa elementy: rezonator kwarcowy Q1 i generator oznaczony jako Q1'. Sprawa jest bardzo prosta: generator Q1' jest elementem opcjonalnym, mogącym zastąpić rezonator Q1 w przypadku, kiedy zależeć nam będzie na szczególnie dokładnych wskazaniach naszego przyrządu. Stabilność częstotliwości wytwarzanej przez generator jest zawsze o rząd wielkości lepsza od generowanej przez wewnętrzny oscylator procesora stabilizowany typowym kwarem zewnętrznym. Niestety, cena generatora jest także o rząd wielkości większa od ceny rezonatora i dlatego przewidziałem w układzie dwa rozwiązania, różniące się nie tylko jakością pomiarów, ale i ceną wykonania przyrządu.

Ostatnim elementem wartym wzmianki jest układ IC2 - DS1813. Jest to układ standardowo stosowany w systemach mikroprocesorowych, którego zadaniem jest wykonanie resetu sprzetowego procesora w przypadku spadku napięcia zasilającego poniżej 4,75VDC.

Zajmijmy się teraz fragmentami programu sterującego pracą miernika. Podawanie całego listingu nie miałoby sensu, ponieważ i tak zostanie on opublikowany i w każdej chwili jest do ściągnięcia ze strony www.edw.com.pl. Ponadto, każdy z Was może poprosić o przesłanie mu kodu źródłowego tego programu

Wykaz elementów

Kondensatory	
C1, C233pF
C3100µF/10
C4, C5100nF
C6220µF/16
Rezystory	
PR1	...potencjometr montażowy miniaturowy 1kΩ
R1, R4, R5, R6, R71kΩ
R2, R310kΩ
Półprzewodniki	
IC1zaprogramowany procesor AT90S2313
IC2DS1813
IC374S00
IC474S74
IC57805
IC674LS393
T1BF199
Pozostałe	
DP1wyświetlacz alfanumeryczny LCD 16*1
CON1złącze BNC lutowane w płytce
CON2ARK2 (3,5mm)
Q1rezonator kwarcowy 10MHz
S1 ... S4przycisk microswitch 15 mm
Rząd 16	goldpinów

Komplet podzespołów z płytką jest dostępny w sieci handlowej AVT jako kit szkolny μP-3003

(zbniew.raabe@edw.com.pl). Skupimy się więc zatem na podprogramie realizującym funkcję pomiaru częstotliwości. Studenci BASCOM College i jego sympatycy proszeni są o szczególnie wnikliwe przeanalizowanie fragmentów listingów. Sądzę, że wnioski mogą ich zachęcić do dalszego poznawania BASCOM-a i nauki języka MCS BASIC.

Po wykonaniu czynności zawartych w **listingu 1** program przechodzi do pętli, w której oczekuje na upływie czasu bramkowania, czyli 1 sekundy. Każde przepełnienie timer0 powoduje skok do procedur pokazanych na **listingu 2**.

Sądzę, że ten fragment listingu pozwoli Czytelnikom zapoznać się z grubsza z metodami programowania mierników częstotliwości i zachęci ich do dokonywania przeróbek programu, a także napisania własnego oprogramowania.

Wyjaśnienia wymaga jeszcze rola przycisków S1...S4. Podczas dokonywania pomiarów częstotliwości miernik pracuje całkowicie automatycznie, sam dobierając sobie potrzebny zakres pomiarowy. Jeżeli jednak zechcemy wykorzystać nasz przyrząd jako licznik zdarzeń, to za pomocą przycisku S1 możemy przełączyć układ w ten tryb pracy (ponowne naciśnięcie S1 spowoduje przejście układu ponownie w tryb pracy jako miernika częstotliwości). W trybie pracy licznika zdarzeń wykorzystywane są trzy pozostałe przyciski. Naciśnięcie przycisku S2 powoduje rozpoczęcie zliczania podawanych na wejście układu impulsów, S3 - zatrzymanie zliczania, a S4 wyzerowanie licznika.

Montaż i uruchomienie

Na **rysunku 4** została pokazana mozaika ścieżek płytki obwodu drukowanego układu miernika oraz rozmieszczenie na niej elementów. Na rysunku widoczna jest także druga płytka, na której jednak nie będziemy umieszczać jakichkolwiek elementów elektronicznych. Płytkę ta może służyć jako względnie efektywna i estetyczna płyta czołowa, wykonana z laminatu i w prosty sposób łączona z płytką układu elektronicznego i wyświetlaczem.

Montaż układu przeprowadzamy w typowy sposób, rozpoczynając od elementów o najmniejszych gabarytach: kilku zworek, a następnie od rezystorów R4 ... R7, które muszą być wlutowane w płytkę jako pierwsze, a w każdym razie przed przylutowaniem **przycisków S1 ... S4**, które podobnie jak **wyświetlacz LCD i złącze BNC** muszą być zamocowane **OD STRONY DRUKU**. Musimy także podjąć decyzję, czy zastosujemy w układzie typowy kwarc 10MHz, czy też znacznie bardziej dokładny, ale też znacznie droższy generator. Jeżeli zdecydujemy się na to drugie rozwiązanie, to kondensatory C1 i C2 nie będą potrzebne. W miejscu przeznaczonym na wyświetlacz wlutowujemy w płytkę **OD STRONY DRUKU** rząd gold-

```

Listing 1
sub frequency          'podprogram pomiaru częstotliwości
Config Timer0 = Timer , Prescale = 8      'konfiguracja timer0, odpowiadającego za odmierzenie 'czasu bramkowania.
'Dodatkowy komentarz:
'Timer0 = Counter - Timer0 będzie pracował jako timer
'Prescale = 8 - na wejście timera podawana będzie częstotliwość zegara systemowego podzielona 'przez 8
Config Timer1 = Counter , Edge = Falling  'konfiguracja timer1, zliczającego 'impulsy wejściowe
'Dodatkowy komentarz:
'Config Timer1 = Counter - Timer1 będzie pracował jako licznik zliczający zewnętrzne impulsy
'Edge = Falling - detekcja impulsu wejściowego będzie rozpoczynać się od opadającego zbocza
Count = 0              'wyzerowanie zmiennej pomocniczej COUNT
Ccount = 0             'wyzerowanie zmiennej pomocniczej COUNT
Set portd.6            'przy pierwszym pomiarze włączamy preskaler
Cls                    'czyszczenie ekranu wyświetlacza
Cursor Off            'usunięcie kursora z ekranu
Enable Timer0         'ezwolenie na obsługę przerwania timer0
Enable Timer1         'ezwolenie na obsługę przerwania timer1
Enable Interrupts     'ezwolenie na obsługę przerwań
On Timer0 int_0        'w przypadku wystąpienia przerwania timer0 skok do podprogramu INT_0
On Timer1 int_1        'w przypadku wystąpienia przerwania timer1 skok do podprogramu INT_1
Timer1 = 0            'wyzerowanie timer1
Start Timer0          'uruchomienie timer0
Start Timer1          'uruchomienie timer1
Do                    'początek pętli programu oczekiwania na wynik pomiaru
Loop
    
```

```

Listing 2
int_0:
inc Count              'zwiększ zmienną pomocniczą COUNT o 1
If Count = 4883 Then  'jeżeli zmienna COUNT przyjęła wartość 4833, czyli upłynęła 1 sekunda, to 'wykonaj wszystkie
                                                                'niższe czynności:
Stop Timer1           'zatrzymaj timer0
Stop Timer0           'zatrzymaj timer1
Maincounter = Ccount * 65536      'wynik pomiaru (MAINCOUNTER) równy jest ilości 'przepełnień timer1 *
                                                                'pojemność timer1
Maincounter = Maincounter + Timer1 'wynik pomiaru równy jest poprzednio obliczonej 'wartości + wynik ostatniego
                                                                'zliczania timer1
'Dodatkowy komentarz: 'Podczas pomiaru większych częstotliwości pojemność timer1 najczęściej okaże się 'niewystarczająca i timer
'ten będzie rozpoczął zliczanie od początku, zgłaszając przy każdym 'przepełnieniu przerwanie, powodujące zwiększenie zmiennej
'CCOUNT o 1. A zatem, po zakończeniu 'pomiaru ilość impulsów wejściowych będzie równa ilości przepełnień timer1 pomnożonej o
'jego 'pojemność. Ponieważ timer1 jest licznikiem szesnastobitowym, jego pojemność wynosi 65536. W 'memencie upływności czasu
'bramkowania, w timerze1 może pozostać 'reszta' 'wyniku zliczania 'impul-
'sów wejściowych, którą oczywiście należy dodać do uprzednio obliczonej wartości.
If Maincounter = 0 Then 'jeżeli żadna częstotliwość nie została zmierzona, to:
    set Portd.6         'włącz preskaler
    End If              'koniec warunku
If Portd.6 = 1 Then    'jeżeli na wyjściu PORTD.6 jest stan wysoki (włączony 'preskaler), to:
    Maincounter = Maincounter / 31 'podziel wynik pomiaru przez 31
'Dodatkowy komentarz: 'Przy zastosowaniu preskalera mierzona wartość jest wstępnie dzielona przez 16. Z kolei, podczas 'pomiarów
'wartości częstotliwości większych od kilku MHz wyświetlanie wyniku z dokładnością 'większą niż 1kHz nie ma najmniejszego sensu.
'Dlatego też wynik pomiaru z preskalerem 'jest 'mnożony przez 16, a następnie dzielony przez 1000.
End If                 'koniec warunku
Ccount = 0             'wyzerowanie zmiennej pomocniczej
Timer0 = 0            'wyzerowanie zmiennej pomocniczej
Cls                    'wyczyść ekran wyświetlacza
If Portd.6 = 0 Then    'jeżeli preskaler jest wyłączony, to:
    Lcd "F [Hz] ="      'wyświetl komunikat "F[Hz]=
Else                    'w przeciwnym wypadku:
    Lcd "F[KHz] ="      'wyświetl komunikat "F[KHz]=
End If                 'koniec warunku
Locate 2 , 1           'umieść kursor na dziewiątej pozycji wyświetlacza
'Dodatkowy komentarz: 'W naszym programie wyświetlacz 16*1 jest traktowany jako 8*2. Jest to wynikiem niekompatybilności 'niek-
'tórych wyświetlaczy ze standardem i kłopotów z sterowaniem nimi z poziomu MCS BASIC. W 'najbliższym cza-
'się problemy te zostaną usunięte przez MCS Electronics
Lcd *                  'zastąp spacjami poprzedni wynik pomiaru
Locate 2 , 1           'umieść kursor na dziewiątej pozycji wyświetlacza
Lcd Maincounter ;      'wyświetl wynik pomiaru
Locate 2 , 8           'umieść kursor na szesnastej pozycji wyświetlacza
Lcd "*"                'wyświetl znak *
Waitms 200             'zaczekaj 200 msek
Locate 2 , 8           'umieść kursor na szesnastej pozycji wyświetlacza
Lcd " "                'usuń znak " , sygnalizujący poprawną pracę miernika
Maincounter = 0        'zerowanie zmiennych
Timer1 = 0
Start Timer0           'ponowne uruchomienie timerów
Start Timer1
End If                 'koniec globalnego uwarunkowania
Return
    
```

pinów, do którego po bardzo starannym sprawdzeniu poprawności montażu przylutowujemy wyświetlacz.

Ostatnią czynnością, jaką będziemy musieli wykonać będzie zamocowanie płyty czołowej. Do tego celu będą nam potrzebne cztery śrubki M3 i gąstka nakrętek. Kolejność postępowania jest następująca:

1. Pomiędzy wyświetlacz, a przylutowaną do niego płytkę elektroniki miernika wsuwamy tulejki dystansowe o odpowiedniej długości i skręcamy całość za pomocą śrubek tak, aby ich łebki wystawały ok. 5 mm ponad powierzchnię płytki wyświetlacza. Do każdej śrubki dajemy w związku z tym dwie nakrętki.

2. Tak zmontowaną konstrukcję układamy na płycie czołowej i starannie wyrównujemy. Następnie lutujemy łebki śrubek do dużych punktów lutowniczych wykonanych na spodniej stronie płyty czołowej.

3. Jak zauważyliście, płyta czołowa jest nieco większa od płytki miernika. Pozwoli to na łatwe dobudowanie tylnej części i boków obudowy, które możemy wykonać z kawałków laminatu lub tworzywa sztucznego.

Po zmontowaniu układu i włożeniu układów scalonych w podstawki włączamy zasilanie miernika. Urządzenie może być zasilane napięciem stałym o wartości 9 ... 16VDC, niekoniecznie stabilizowanym. Po włączeniu zasilania i upływie ok. 1 sekundy na wyświetlaczu powinien ukazać się napis "F[kHz]= 0", co świadczy o poprawnym działaniu układu.

Możemy teraz podłączyć do miernika źródło prądu prostokątnego o częstotliwości mieszczącej się w zakresie miernika i dokonać pierwszego pomiaru.

Zbigniew Raabe
e-mail: zbniew.raabe@edw.com.pl