

# ELEKTRONIK ELEKTOR

MIESIĘCZNIK DLA ELEKTRONIKÓW

co miesiąc w  
Elektronice Praktycznej

## Wideo-woltomierz cyfrowy

*Woltomierze cyfrowe są popularnymi narzędziami dla każdego, kto chce zmierzyć napięcia. Mamy nadzieję, że opisany tu układ, unikalny ze względu na swoje właściwości, zyska uznanie naszych Czytelników. Ten cyfrowy woltomierz ma dwa wyjścia: jedno szeregowe i drugie do bezpośredniego połączenia z odbiornikiem lub monitorem TV.*

Wymień kilka najczęściej stosowanych aplikacji mikrokontrolera, a prawdopodobnie będą to pomiary, sterowanie, wyświetlanie itp. Dodaj do tego najświeższą technologię RISC z bardzo krótkim czasem cyklu wykonania instrukcji (co umożliwia nawet generowanie sygnału wizji w czasie rzeczywistym) oraz pojemność pamięci „flash“ dostępnej dla programowania bez kosztownego wyposażenia, i proszę, oto masz cyfrowy przyrząd wideo, tani, zabawny, o wyświetlaczu gigantycznych rozmiarów - może to być woltomierz z wbudowanym interfejsem szeregowym!

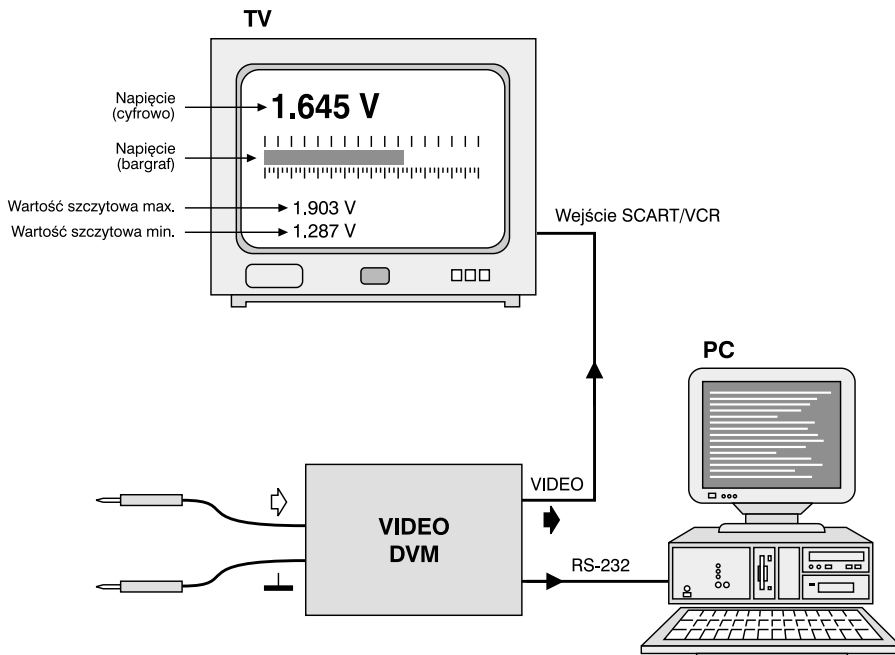
Wideo-woltomierz cyfrowy wyświetla napięcia na ekranie twojego telewizora, zarówno w postaci gigantycznych cyfr jak i paska analogowego. Zapisuje wartości szczytowe: maksymalną i minimalną, jak również przesyła dane pomiarowe poprzez interfejs szeregowy do komputera osobistego.

Schemat blokowy woltomierza przedstawiono na **rys. 1**. A oto jego podstawowe właściwości:

- Wejście analogowe z możliwością pomiaru napięcia w przedziale od 0 do 4,096V.
- Wyjście wizji wyświetlacza, umożliwiające wysterowanie dowolnego telewizora za pośrednictwem jego wejścia SCART (VCR).
- Szeregowe wyjście danych do rejestracji danych przez komputer osobisty.
- Wymaga jedynie pojedynczego napięcia zasilania, typowo 5V/15mA.

Ze swoim ogromnym wyświetlaczem ekranowym, wideo-woltomierz doskonale nadaje się do prezentowania wielkiej widowni wszystkiego, co można przemienić w napięcie: od temperatury w poważnych eksperymentach szkolnych do miłości na imprezach towarzyskich.

Ale wideo-woltomierz to o wiele więcej niż tylko śmieszna zabawka. Jest naprawdę tani i łatwy do modyfikacji dostosowujących go do twoich potrzeb, umożliwiając każdemu eksperymenty z sygnałami wizji i wyświetlaniem da-



Rys. 1. Schemat blokowy woltomierza cyfrowego z wyjściami: szeregowym i wizji.

nych. Samo programowanie napotyka poważne ograniczenia bezpośredniej syntezy sygnału wizji w czasie rzeczywistym, przetwarzania cyfrowego, taktowania danych szeregowych i skróconej długości kodu. Przetwornik analogowo-cyfrowy można łatwo łączyć z różnymi czujnikami lub zastąpić go innym kodem do wyświetlania czasu albo liczby impulsów. Wreszcie, dzięki swojemu szeregowemu łączu wideo-woltomierz może również posłużyć jako prosty rejestrator danych.

Cała ciężka praca przygotowania sterowanego przerywaniami generatora obrazu dla telewizora/monitora, procedur transmisji szeregowej, sterowania przetwornika analogowo-cyfrowego itd. jest wstępnie wykonana programowo, umożliwiając modyfikacje kodu małymi krokami, dla dopasowania do wszystkich specyficznych potrzeb lub błyskotliwych pomysłów.

Innymi słowy, budowa wideo-woltomierza może być bardzo pouczająca, co dziś jest szczególnie cenne, w wieku hiperspecjalizowanych układów scalonych, nie pozostawiających niemal żadnego miejsca dla fantazji i własnych poszukiwań.

### Część sprzętowa

Jak możesz się przekonać oglądając schemat elektryczny na rys. 2, układ jest zbudowany

wokół nowego mikrokontrolera AVR90S1200 firmy Atmel i przetwornika analogowo-cyfrowego (ADC).

Pomimo doskonałych właściwości: zegar 16MHz, pamięć programu „flash“ i wewnętrzna EEPROM, układ AVR jest względnie niedrogi i tym samym idealnie nadaje się dla hobbystów. Co więcej, całe niezbędne oprogramowanie (z potężnym symulatorem włącznie) jest dostępne bezpłatnie na stronie Atmela [www.atmel.com](http://www.atmel.com).

Układ ten może być przeprogramowywany co najmniej 1000 razy za pośrednictwem jednego z wielu już dostępnych programów. Szczegółowy protokół programowania pamięci „flash“ również jest dostępny na wspomnianej stronie WWW.

Procesor ma wbudowane 32 bajty pamięci RAM, 64 bajty pamięci EEPROM, 512 słów pamięci programu i 8-bitowy timer. Zestaw instrukcji jest bardzo zwięzły i bardzo dobrze wyważony, a dzięki architekturze Harvard RISC nawet tak złożone zadania, jak to opisane tutaj, mogą być wykonywane za pośrednictwem nie więcej niż około 400 instrukcji. Cykl instrukcyjny jest bardzo krótki, co umożliwia programowe budowanie sygnałów wizji.

Prosty, dwubitowy asymetryczny przetwornik cyfrowo-analogowy (DAC), złożony z trzech rezys-

torów, dostarcza zespolonego sygnału wizji, na standardowym poziomie  $1V_{pp}$ , do wejścia telewizora ze złączem SCART lub wejścia AV (magnetowidu). Szeregowy strumień danych (znaki ASCII, przepływność 1200 bodów, bez bitu parzystości, 8 bitów danych, 1 bit stopu) również jest tworzony programowo. Poziomy są zgodne ze standardem TTL: większość komputerów osobistych pracuje przy poziomach TTL równie dobrze, jak przy poziomach standardu RS232, przy założeniu, że kabel łączący nie jest zbyt długi.

### WYKAZ ELEMENTÓW

#### Rezystory

- R1: 470Ω
- R2, R5: 10kΩ
- R3: 5,6kΩ
- R4: 1,2kΩ
- R6: 12kΩ
- R7: 8,2kΩ

#### Kondensatory

- C1: 10nF, ceramiczny
- C2, C4..C7, C9: 100nF, ceramiczny
- C3: 4,7μF/16V, stojący
- C8: 100μF/25V, stojący
- C10, C11: 22pF
- C12, C13: 10μF/16V, stojące

#### Półprzewodniki

- D1: 1N4148
- D2: LED, wysokosprawna
- D3: 1N4001
- D4: 5,1V/400mW, dioda Zenera
- T1: BC550C
- IC1: MAX192ACPP
- IC2: AVR90S1200-16PC
- IC3: 7805

#### Różne

- S1: wyłącznik przyciskowy, 1 komplet styków
- S2: 3- lub 4-drożny przetłącznik DIP
- X1: kwarc 16MHz
- K1: 2-drożny blok końcówek śrubowych do druku, rozstaw 5mm
- K2: 9-stykowe gniazdo sub-D (żeńskie) z wyprowadzeniami kątowymi, do druku
- Płytką drukowaną (nr katalogowy 992024-1 - dostępna w EE)
- Płyta CD-ROM „μP/μC hard & software 97/98” (ISBN 90 5381-086-2) - dostępna w EE, zawiera kompletny wydruk {listing} kodu źródłowego dla tego projektu. Plik mieści się w katalogu : \Gb\03.

Dlatego pomyślałem, że przetwornik napięć, w rodzaju MAX232, nie jest konieczny. Jeśli chcesz, możesz go dodać na zewnątrz: w takim przypadku musisz odwrócić polaryzację, zamieniając instrukcje „SET BIT“ i „CLEAR BIT“ w kodzie programu (ich miejsce jest wyraźnie zaznaczone na listingu programu - patrz wykaz elementów).

Wszystkie wyprowadzenia wejściowe są wewnętrznie podciągnięte do zasilania i mają dużą wydajność prądu wpływającego, tak że wyłącznik „max - min clear“ i trzy przełączniki DIP albo zwory wyboru położenia kropki dziesiętnej mogą być włączone bezpośrednio pomiędzy te wyprowadzenia i masę.

Dioda LED, migająca za każdym razem, gdy pomiar się zakończy, jest dołączona do wyprowadzenia wyjściowego.

### Pomysł ADC

Napięcie jest odczytywane za pośrednictwem 10-bitowego szeregowego przetwornika analogowo-cyfrowego typu MAX192ACPP, z własnym napięciem odniesienia ustawionym na poziomie 4,096V.

Do przesyłania danych niezbędne są cztery linie wejścia/wyjścia (I/O): *Data input*, *Data output*, *Chip enable*, *Serial clock*. Pomimo że jest wyspecyfikowany jako 10-bitowy, przetwornik dostarcza dwa lub więcej bitów „sub LSB“ („mniej niż najmniej znaczące“): oznacza to, że z przetwornika w rzeczywistości odczytywanych jest 12 bitów, chociaż tylko 10 z nich ma gwarantowaną dokładność. Pomimo tego stwierdziłem, że te dwa dodatkowe bity są bardzo dokładne. Na wyświetlaczu pojawiają się wszystkie bity, co pokrywa zakres od 0,000 do 4,095V, z bezpośrednim odczytem miliwoltów. Układ wykorzystuje tyl-

ko jedno z ośmiu dostępnych wejść MAX192, ale sterownik programowy jest przystosowany do odczytu dowolnego wejścia, jakie zechcesz wybrać. Poprzez niewielkie zmiany możesz przestawić wideo-woltomierz cyfrowy na pomiar czterech napięć: trzech z wyświetlaniem cyfrowym i jednego jako paska analogowego.

### Bezpośrednia synteza wizji

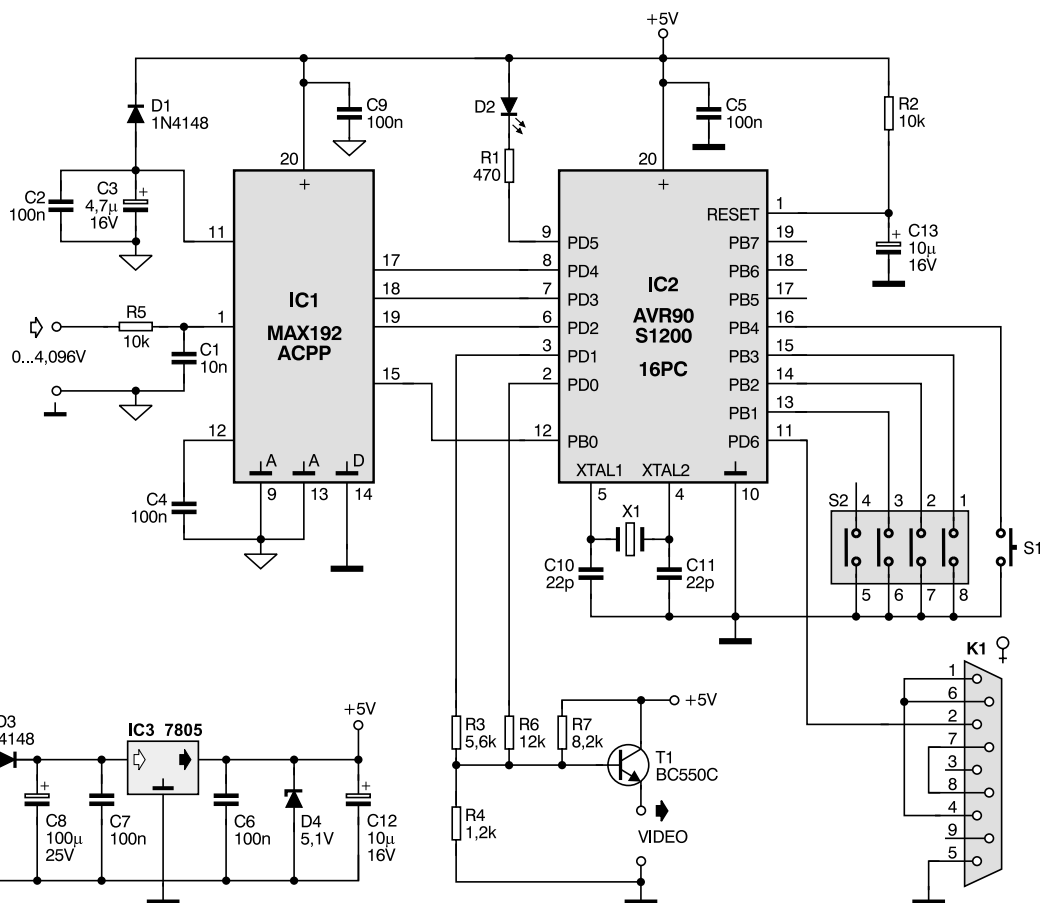
Ze sprzętowego punktu widzenia, część sprzętowa związana z syntezą sygnału wizji jest wyjątkowo prosta: dwa wyprowadzenia wyjściowe i trzy rezystory umożliwiające generowanie czterech poziomów sygnału (synchro, czerń, biel, jasna szarość). Z programowego punktu widzenia, synteza wizji wymaga bardzo krótkiego cyklu instrukcyjnego (tu: 62,5ns) i starannego, wyważonego czasowo, zrównoważonego pod względem instrukcji, dopieszczonego i ręcznie optymalizowanego kodu.

Do uzyskania prawidłowej generacji zespolonego sygnału wizji, solidny system taktowania jest

Tabela 1. Ustawienia przełącznika DIP (S2).			
1	2	3	wyświetlacz
OFF	OFF	OFF	xxx.x
OFF	OFF	ON	xx.xx
OFF	ON	OFF	x.xxx
OFF	ON	ON	.xxxx
ON	OFF	OFF	xx.xx
ON	OFF	ON	x.xxx
ON	ON	OFF	.xxxx
ON	ON	ON	xxxxx

absolutnie niezbędny. Nawet opóźnienie o jeden cykl zegara 16MHz (62,5ns) jest wyraźnie widoczne na ekranie, a więc musisz pomyśleć dwa razy, zanim wstawisz nawet pojedynczą instrukcję.

Przed wszystkim musimy wybrać odpowiednią podstawę czasu: wybranie czasu trwania linii rastra (64µs) jako podstawy czasu pozwoli nam zbudować całą ramkę linia po linii (ramka bez przeplatania składa się z 312 linii), jak również łatwo umieszczać impulsy synchronizacji poziomej (kilka mikrosekund na po-



Rys. 2. Od teorii do praktyki: mikrokontroler i przetwornik analogowo-cyfrowy.

czątku każdej linii) i synchronizacji pionowej (kilka linii na początku każdej ramki).

Jedyny timer dostępny w mikrokontrolerze jest w stanie generować powtarzalne przerwania co każde 16µs bez potrzeby przeładowywania, czyli cztery przerwania na każdą linię rastra.

Co każde cztery przerwania tworzymy nową linię wizji: w tym celu, za każdym wykonaniem podprogramu obsługi przerwania licznik zwiększa swój stan o 1 i co każde cztery przerwania jesteśmy (niemal) na początku linii. „Niemal“, ponieważ po zgłoszeniu przerwania musimy odliczyć czas niezbędny na wykonanie podprogramu obsługi przerwania. Czas ten nie jest stały i zależy od instrukcji wykonywanej właśnie w czasie generacji przerwania. Niektóre instrukcje mają czas wykonywania dłuższy niż inne, a więc czas obsługi przerwania może się zmieniać w sposób nieprzewidywalny, zakłócając wyświetlanie.

Najlepszym rozwiązaniem tego problemu, jakie znalazłem, jest wprowadzanie mikrokontrolera w stan uśpienia tuż przed pojawieniem się czwartego, krytycznego przerwania. Następnie kolejne przerwanie wybudza mikrokontroler ze stałym, znanym przebiegiem czasowym.

Dwa wyprowadzenia wyjściowe są oznaczone jako *CsynBit* (composite sync) i *VideoBit*. Podciąganie zarówno *CsynBit*, jak i *VideoBit* do logicznego poziomu niskiego powoduje, że poziom sygnału wizji ma wartość 0 woltów (poziom synchronizacji). Przy tylko *CsynBit* na poziomie wysokim, otrzymamy poziom czerni, a przy *CsynBit* i *VideoBit* na poziomie wysokim uzyskujemy wyświetlanie bieli.

Nie powinno zaskakiwać, że cała generacja wizji opiera się na podprogramie obsługi przerwania timera. Co każde cztery przerwania nowa linia rozpoczyna się impulsem synchronizacji; następnie są wykonywane powtarzalne podprogramy taktowania (liczenia linii, komunikacja szeregowej). Wszystkie te czynności następują tak szybko, że musimy dodać pętlę opóźniającą, by poczekać na początek widzialnej części linii wizji.

W tym momencie jest wykonywana sekwencja wielu skoków dla określenia, w jakiego rodzaju linii się znajdujemy. Skoki te opierają się na numerze linii (położenie pionowe), tak że w dalszej kolejności mikrokontroler może zdecydować, czy bieżąca linia wyświetli aktualne napięcie (w którym to przypadku następuje wywołanie podprogramu wyświetlania znaku), czy pasek analogowy (podprogram paska lub podprogramy linijek paska), czy też małe liczby przedstawiające wartości szczytowe (podprogramy wyświetlenia znaków ze zmniejszonym parametrem rozmiaru).

Oczywiście, jest również podprogram wyświetlania pustych linii. Dla pustych linii podprogram przerwania po prostu się kończy, zwracając sterowanie do programu głównego. Przy wyświetlaniu linii sterowanie nie jest zwracane, zanim cała linia nie zostanie wyrysowana odpowiednią grafiką. Przy pionowym powrocie linii wyświetlanie jest wygaszane i następuje odwrócenie polaryzacji synchronizacji.

Przy końcu wyświetlania linii albo krótko po trzecim przerwaniu, mikrokontroler jest „zamrażany“ za pośrednictwem instrukcji uśpienia, oczekując na czwarte, w sensie czasu krytyczne, przerwanie.

W trakcie odmierzania czasu na początku każdej linii wizji następuje wysłanie danych szeregowych. Wobec braku dostępu do innych timerów sprzętowych, port szeregowy został utworzony programowo. Co każde 13 linii poziomych jest przesyłany następny bit, uzyskiwana w ten sposób szybkość transmisji 1200 bodów doskonale odpowiada małej liczbie danych, jaką dysponujemy.

Dane są przesyłane jako łańcuch ASCII zakończony znakiem <CR> (ASCII 13). Do odbioru tych danych możesz skorzystać z programu QBasic lub jakiegokolwiek innego programu terminala (jak HyperTerminal dostarczany wraz z Windows 95 lub Terminal z Windows 3.1).

Wszystkie podprogramy generacji sygnału wizji są starannie ręcznie zoptymalizowane. Nie usuwaj z kodu instrukcji NOP, służą one do zrównoważenia cza-

sów wykonywania! Przy wykonywaniu skoków warunkowych należy starannie rozważyć czasy realizowania wszystkich możliwych ścieżek programu. Na szczęście dla Ciebie, wszystkie kody niezbędne do obsługi sygnału wizji są już gotowe i działają samodzielnie pod kontrolą przerwania.

Nie czuj się skrępowany przy modyfikacji lub dodawaniu kolejnych linijek kodów do programu głównego w celu dostosowania go do swoich wymagań. Nawet sekwencja wielokrotnych skoków, wybierająca wyświetlaną zawartość, nie jest zbyt trudna do zrozumienia, tak że możesz łatwo budować setki różnych wyświetlaczy, ze znakami o dowolnych rozmiarach i wieloma paskami analogowymi.

Wszystkie podprogramy i zmienne mają długie nazwy, które powinny same się tłumaczyć. Kody są obszernie skomentowane. Należy pamiętać, że w danym momencie jest dostępne tylko jedno wywołanie podprogramu.

## Własne znaki

Możesz przededefiniować znaki do dowolnie wybranego kształtu: jest to szczególnie przydatne w przypadku jednostki mierzony wielkości, zamiast symbolu wolta (V) możesz wyświetlić dowolną inną literę lub kształt mieszczący się w macierzy 8x5 pikseli. Kształty znaków są przechowywane w pamięci EEPROM zgodnie z plikiem definicji FONTS.INC. Plik ten jest automatycznie włączany w trakcie asemblacji, a wyniki są kompilowane do VIDEODVM.EEP. *Jeśli programowałeś część FLASH, nie zapomnij oddzielnie zaprogramować EEPROM.*

Znaki są przechowywane w pięciu kolejnych bajtach, po obrocie o 90 stopni.

## Montaż układu

Wszystkie zastosowane podzespoły są tanie i łatwe do kupienia. Układ MAX192 ma odpowiedniki o tym samym protokole danych i o różnych rozdzielczościach.

Połączenie z komputerem odbywa się poprzez standardowe 9-stykowe złącze sub-D.

Najprawdopodobniej zechcesz do wejść dołączyć czujniki. Zadbaj, aby ich napięcie wyjściowe w żadnym wypadku nie przekroczyło 4,096V lub nie spadło poniżej 0V, umieszczając rezystor szeregowy (około  $1k\Omega$ ) i diody obcinające, jeśli zakres ten mógłby być przekroczony.

Po włączeniu zasilania migotanie diody LED poinformuje Cię, że mikrokontroler pracuje jak należy. Dioda miga jeden raz po zakończeniu każdego pomiaru.

Nadszedł czas dołączenia wideo-woltomierza do telewizora: w telewizorach wyposażonych w złącze SCART wejściem wizji są wyprowadzenia 20 (sygnał) i 17 (masa). W momencie wtykania złącza SCART telewizor musi być wyłączony z sieci.

Pilotem zdalnego sterowania przełącz telewizor w tryb wejścia AV: starsze telewizory mogą nie mieć tej możliwości, w takim

przypadku wybór wejścia AV można wymusić podciągając w górę wyprowadzenie 8 złącza SCART: po prostu zainstaluj szeregowy rezystor  $1k\Omega$  do wyprowadzenia +5..12V.

Możesz teraz wstawić zwory, aby wybrać położenie kropki dziesiętnej (Tabela 1), pobawić się przykładając różne napięcia do przetwornika analogowo-cyfrowego, sprawdzić działanie przycisku kasowania wskazań min-max lub po prostu podziwiać przez chwilę swoją nową elektroniczną zabawkę.

Ostatnim krokiem jest dołączenie komputera osobistego. Uruchom program Terminal w teczce Accesories (Windows 3.1) lub program narzędziowy HyperTerminal (Windows 95). Ustaw łącze na 1200 bodów, bez parzystości, 8 bitów danych, jeden bit stopu (1200,n,8,1). Standardowy zakres pomiarowy roz-

ciąga się od 0 do 4,095V. Jeśli jest wymagany szerszy zakres pomiarowy, należy dodać tłumik. Jednym z rozwiązań, jakie przychodzą na myśl, jest dodanie rezystora równoległe z C1. Jednak gdy dochodzi do pomiaru wyższych napięć, pojawia się realne niebezpieczeństwo uszkodzenia wejścia MAX192 w przypadku błędnego połączenia. W takim przypadku środkiem zabezpieczającym jest włączenie diody Zenera 5,1V pomiędzy C1 i rezystor. Zakres pomiarowy - 10V na przykład - otrzymuje się poprzez dołączenie rezystancji  $6,9k\Omega$  (rezystora stałego  $5,6k\Omega$  i potencjometru montażowego  $2,2k\Omega$ ) równoległe z C1.

**Projektował A. Ricci Bitti**  
**[902024-1]**

*Artykuł publikujemy na podstawie umowy z redakcją miesięcznika "Elektor Electronics".*