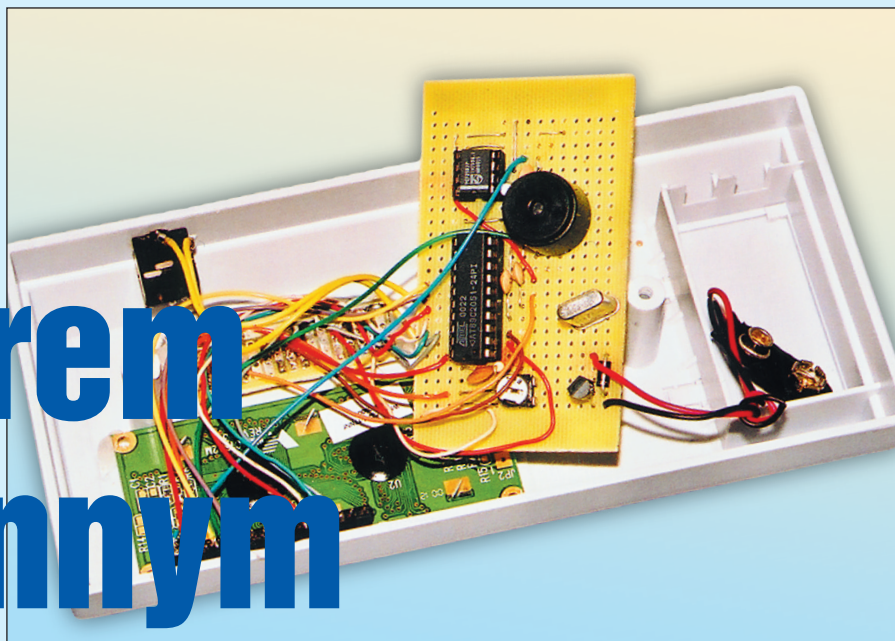




Zegar z timerem kuchennym



Do czego to służy?

Prezentowane urządzenie, zegar z timerem kuchennym, służy do urozmaicenia wyposażenia kuchni o element pomocny przy wykonywaniu różnych czasowych czynności. Dzięki niemu pomocą już nigdy nie zapomnisz wyjąć ciasta z piekarnika, dokładnie przeprowadzisz proces gotowania jajek i wiele innych. Dodatkową funkcją jest wyświetlanie aktualnego czasu i bieżącej daty w chwilach wolnych od pracy z timerem.

Jak to działa?

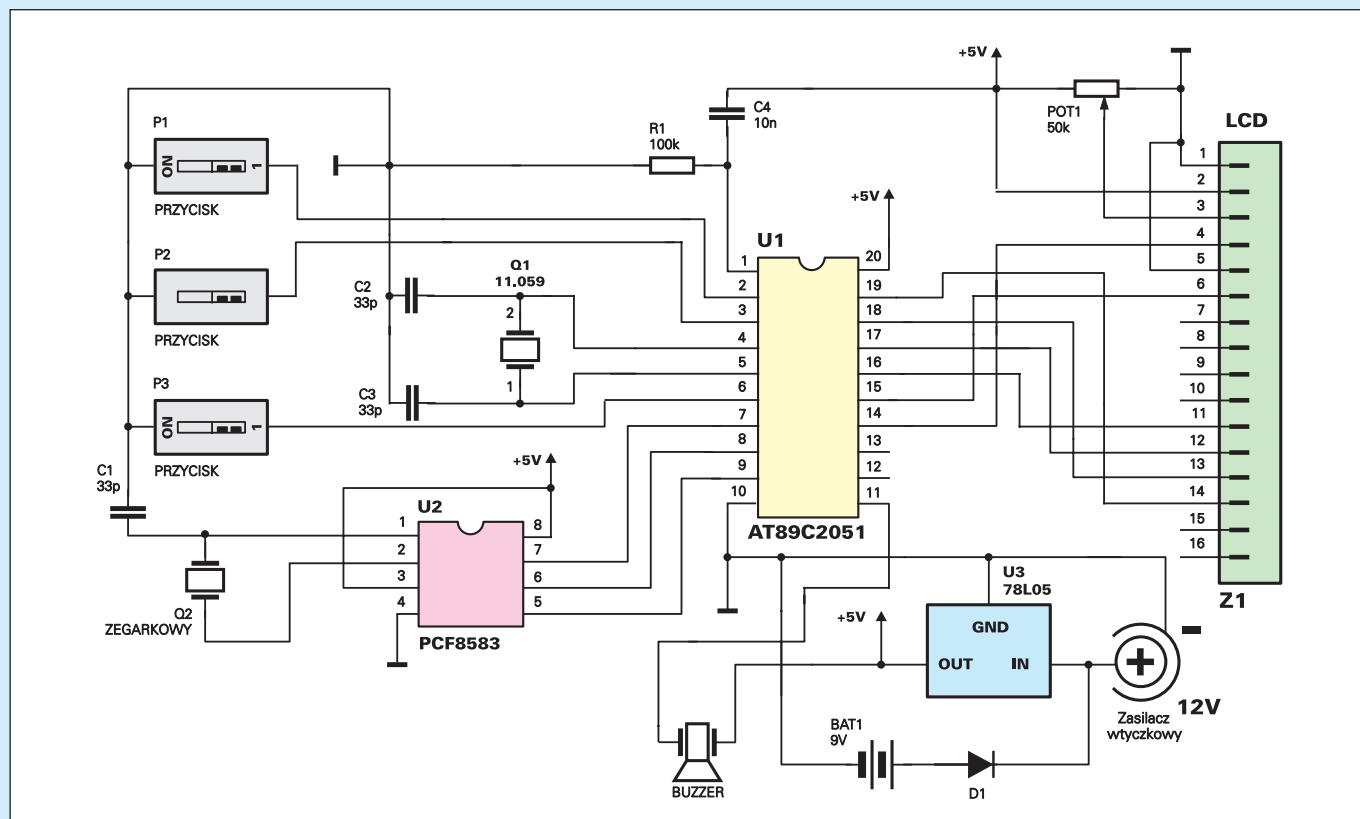
Zasadniczo układ można podzielić na cztery bloki funkcjonalne:

- 1 - blok zasilacza,
- 2 - blok klawiatury,
- 3 - blok procesorowy ze sterownikiem LCD i sygnalizatorem,
- 4 - blok zegara RTC.

Schemat elektryczny zegara przedstawiony jest na **rysunku 1**. W zasadzie praktycznie żaden z elementów nie wymaga (chyba

omawiania, poza układem, U2, który w swoim wnętrzu zawiera zegar czasu rzeczywistego PCF8583 pracujący w trybach 12/24h, 4-letni kalendarz, licznik i kilka rejestrów dodatkowych jak np. rejestry alarmów. Z układem tym procesor porozumiewa się wykorzystując magistralę I²C opracowaną przez firmę Philips. Układ klawiatury tworzy zespół trzech przycisków zwiernych chwilowych,

Rys. 1 Schemat ideowy



których zadaniem jest zwarcie do masy w momencie naciśnięcia pinu, do którego są podłączone. Lwia część portu P1 obsługuje wymianę danych wyświetlaczem LCD, wykorzystując przy tym magistralę danych 4-bitową. Procesor dodatkowo wysterowuje buzzer, którego zadaniem jest sygnalizowanie zakończenia cyklu odliczania timera.

Aby dokładniej uzmysłowić sobie zasadę działania, prześledźmy najważniejsze części programu zawartego w listingu. Na początku zdefiniowane są rozkłady pinów dla wyświetlacza LCD, jak i dla magistrali I²C po to, by uniezależnić się od ustawień kompilatora. Później zadeklarowane są zmienne, aliasy i podprogramy. Jeszcze przed rozpoczęciem pracy we właściwej pętli dokonane są zabiegi kosmetyczne zapewniające: poprawny start programu po włączeniu zasilania, poprawną interpretację sygnałów przzerwania INT1 oraz wyczyszczenie wyświetlacza i wyłączenie kursora. Pętla główna zawarta pomiędzy pierwszymi z rozkazów do...loop jest swoistą osią systemu i to ona zarządza pracą całego układu. Jej zadaniem jest reagowanie na przyciskanie przycisków, odpowiedzialna jest za aktualne wyświetlanie i odświeżanie odpowiednich informacji na wyświetlaczu i wywołuje wszystkie funkcje zegara. Na początku pracy układ zbiera informacje o aktualnym czasie i nastawionej dacie i prezentuje te dane na wyświetlaczu. Użytkownik może wybrać jedną z trzech dostępnych funkcji:

1 - nastawianie zegara,

2 - nastawianie daty,

3 - ustawianie czasu do odliczenia

przez timer.

Przy nastawianiu czasu lub daty automatycznie sprawdzane jest czy wartość zadana nie przekracza dopuszczalnych wartości. Sposób wyświetlania nastaw czasu i daty jest taki, że czas pojawia się zawsze w pierwszej linii, a data w drugiej. Taki sposób prezentacji został niejako wymuszony przez ograniczenia co do wielkości kodu programu procesora AT89C2051. Chcąc zaoszczędzić na cennej pamięci podjąłem się znalezienia takich części programu, w których na pewnych etapach wykonuje się dokładnie te same operacje, połączenia ich w podprogramy i swobodne odwoływanie się do nich w miejscach, w których należałoby zastąpić je równoważnym kodem. Sposób ten pozwolił na wyłonienie kilku

takich części programu i znakomite „odchudzenie” kodu. Przykładem niech będzie wyświetlanie czasu i daty oraz ich osobne nastawianie. W obu przypadkach istnieje ten sam sposób prezentacji na LCD i ten sam sposób dopisywania zer przed liczbami jednocyfrowymi. Po co więc za każdym razem „klepać” kawałki programu, który zajmuje pamięć, jeśli można się odwołać krótkim poleceniem do procedury, która taką funkcję wykonuje? Dalej widzimy procedury odpowiedzialne za odczyt i zapis danych do i z RTC. Różnica między odczytem a zapisem polega na tym, że w fazie odczytu, przypadającej w czasie wyświetlania czasu i daty, jednocześnie dane są „ściągane” hurtowo i trafiają, po konwersji, do odpowiednich rejestrów skąd są posyłane na wyświetlacz. Instrukcji zapisu mamy natomiast dwie, ponieważ przy nastawianiu daty lub czasu, gdy ustawiany jest tylko jeden pasek wyświetlacza, nie ma sensu tracić czasu na zapisywanie danych, których wartości nie uległy zmianie. Użycie takiego sposobu również przyczyniło się do odchudzenia programu. Kolejne podprogramy traktują o nastawach timera, dopisywania zer przed liczbami jednocyfrowymi i wyświetlaniu poszczególnych składników. Ostatnia z procedur to obsługa przzerwania z INT1, które zgłaszane jest przy opadającym zboczu na tym wejściu. Służy ono do zliczania sekund jakby „za darmo”, nie obciążając procesora przy pracy z timerem. Przeglądając dokumentację układu PCF8583, natknąłem się na informację o tym, że po uruchomieniu układu, bez modyfikacji rejestrów kontrolnych, na jednym z jego wyjść otrzymuje się sygnał 1Hz. Właściwość ta wykorzystana jest w owej procedurze, której zadaniem jest odliczanie w dół od zadanego czasu z krokiem 1s i sygnalizacja gdy osiągnięte zostanie zero (czas minie). Proste – prawda? Jeszcze jedna informacja odnośnie tajemniczego opóźnienia 100ms w pętli głównej. Otóż

zwlóka ta jest nałożona po to, by:

1 - zbyt często nie aktualizować wyświetlacza (możliwy efekt mrugania lub pływania tekstu),

2 - zbyt często nie zapisywać do rejestrów kontrolnych RTC informacji o chęci pobrania z niego danych (żywność EEPROM).

System zasilania został opracowany z myślą o wykorzystaniu zasilacza wtyczkowego 9-12V. Gdy napięcie sieci zaniknie, układ automatycznie przełączy się na zasilanie baterijne (D1). Praca z timerem powinna raczej odbywać się z zasilaczem, gdyż buzzer pobiera sporo prądu i możliwe jest „wywieszenie się” programu przy słabszej baterii.

I to byłoby na tyle, jeśli chodzi o opis działania. Zachęcam do przeanalizowania listingu, w którym ująłem dużo komentarza pomocnego przy „rozgrzaniu” programu.

Możliwości zmian...

Wykonując ten projekt, w założeniach miałem zastosować również czujnik AF50 do detekcji ulatniającego się gazu. Ponieważ chwilowo nie zostało to zrealizowane, pozostały dwa wolne wyprowadzenia procesora do, właściwie, dowolnego wykorzystania. Są to P1.0 i P1.1. Najlepiej wykorzystałby zalety układu i wykonał jakiś układ komparacyjny, jak by to miało miejsce w przypadku czujnika gazu. Możliwe są także inne rozwiązania w zależności od wyobraźni. Największą jednak przeszkodą jest bardzo mało wolnej pamięci, którą można jeszcze wykorzystać. Jeśli użyć układu AT89C4051, sytuacja radykalnie zmieniłaby się. Dlatego też, gdy tylko uzyskam dostęp do takiego mikrokontrolera, wykonam nowszą wersję oprogramowania obsługującego albo wspomniany czujnik gazu, albo cyfrowy termometr DS1820, a może oba naraz.

Grzegorz Kaczmarek

Uwaga! Pliki z programem można ściągnąć ze strony internetowej EdW
www.edw.com.pl/library/pliki/zegarGK.zip

Wykaz elementów

Rezystory

R1100kΩ

Kondensatory

C1-C333pF

C410nF

Półprzewodniki

U1AT89C2051

U2PCF8583

U378L03

D11N4148

Inne

P1-P3mikroswitch

POT1potencjometr 50kA

Q1kwarc 11.059

Q2zegarkowy

Buzzer

Wyświetlacz LCD