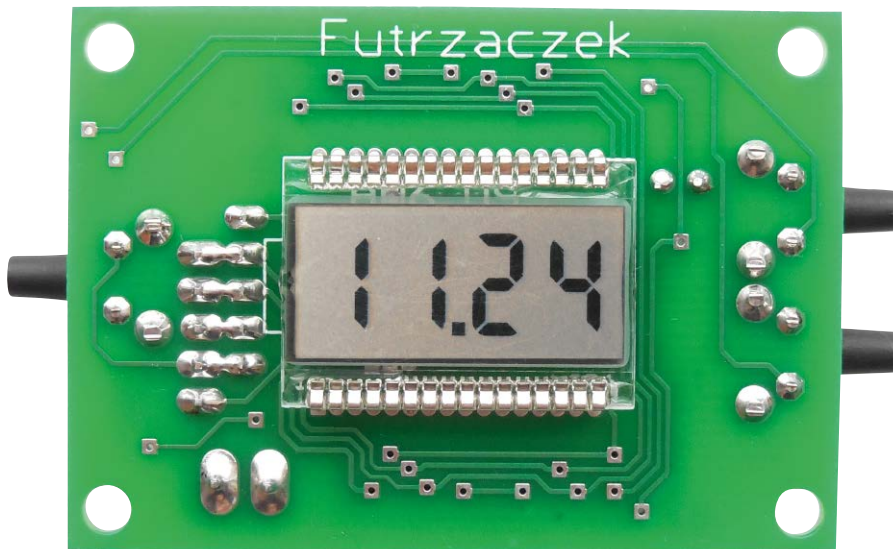


# Energooszczędny zegar z wyjściem sterującym

Przedstawiony projekt zegara włącz/wyłącz wyróżnia się spośród innych bardzo małym poborem prądu. Pozwala to na zasilanie go z baterii przez wiele miesięcy. Ma jedno wyjście, którego godzina załączenia oraz wyłączenia może być dowolnie ustawiana przez użytkownika. Obsługę zegara ułatwia przejrzyste, intuicyjne menu.

**Rekomendacje:** projekt przydatny szczególnie tym, którzy poszukują łatwego w obsłudze zegara do sterowania cyklicznie załączanym urządzeniem.

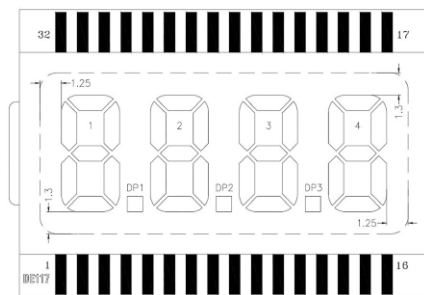
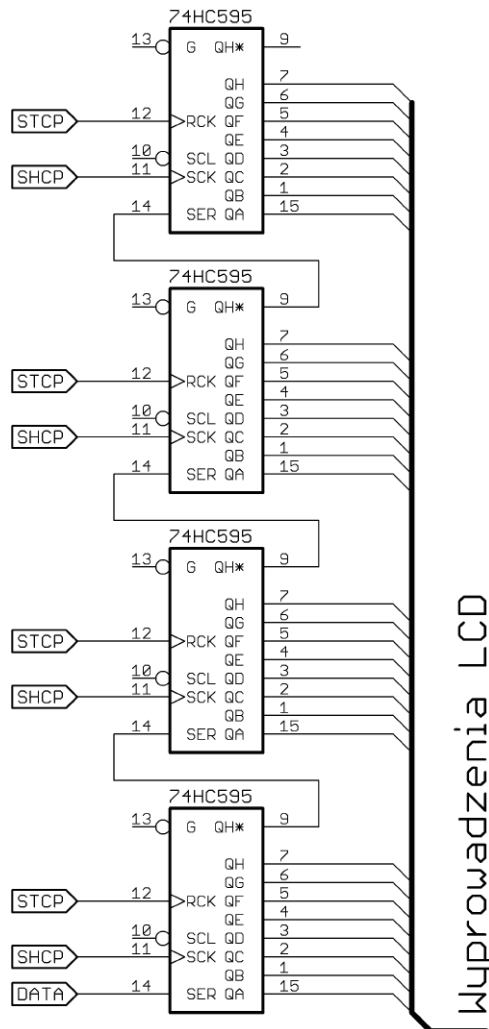


Największym problemem, jaki został napotkany podczas realizacji tego projektu, było znalezienie niewielkiego, taniego wyświetlacza, który pobierałby znikomy prąd. Oczywiście wyborem są tu wyświetlacze LCD, jednak te najbardziej popularne są w zasadzie modułami z wbudowanym sterownikiem. To znacząco upraszcza komunikację, ale powoduje też znaczący wzrost poboru prądu, więc taki wyświetlacz raczej nie nadaje się do zasilania z baterii – pobór prądu rzędu kilku miliamperów szybko rozładuje nawet największe ogniwa. Z tego powodu konieczne było wykonanie własnego układu sterującego wszystkimi elektrodami wyświetlacza.

Zdecydowałem się na użycie wyświetlacza LCD typu DE 117. Jest to 4-cyfrowy,

7-segmentowy wyświetlacz LCD bez podświetlenia tła, przystosowany do zasilania napięciem 3 V. Na jedną nóżkę wyprowadzono elektrodę wspólną, a pozostałe 31 prowadzi bezpośrednio do segmentów (rysunek 1). Łatwo policzyć, że ten wyświetlacz dołączony wprost do mikrokontrolera zajmuje aż 32 wyprowadzenia. Można jednak obejść ten problem, np. stosując rejestr przesuwny z wyjściem równoległym. W tej roli użyto czterech 8-bitowych układów 74HC595. Układy z serii HC mogą być zasilane napięciem z zakresu 2...6 V, dzięki czemu idealnie nadają się do tej aplikacji. Dodatkowym atutem jest niewielka cena detaliczna – około 1 zł.

Wszystkie rejestry połączone w łańcuch, jak pokazano na rysunku 2. Mikrokontroler wpisuje 32-bitowy ciąg na linię DATA w takt przebiegu wysyłanego na linii SHCP. Po wpisaniu całości, zboczem narastającym na linii STCP, aktualizuje wyjścia rejestrów. Dzięki temu do sterowania całym



Rysunek 1. Wyświetlacz DE 117 firmy Display Elektronik GmbH

Rysunek 2. Układ sterowania wyświetlaczem DE 117

### Wykaz elementów:

#### Rezystory:

R1, R2: 4,7 kΩ (SMD 0805)  
RN1: 4×100 kΩ (SIL5)

#### Kondensatory:

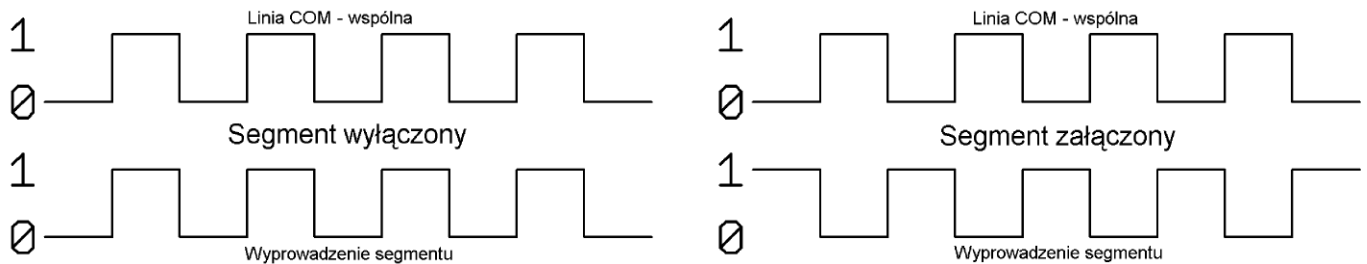
C1, C5, C6: 100 nF (SMD 0805)  
C2, C3: 10 nF (SMD 0805)  
C4: 10 μF (SMD 0805)

#### Półprzewodniki:

T1, T2: IRLML2502  
US1: M41T00 (SO8)  
US2: ATtiny44A-SSU  
US3...US6: 74HC595 (SO16)

#### Inne:

J1: goldpin kątowny, męski 5-pin/2,54 mm  
J2: złącze ARK2/3,5 mm  
LCD1: wyświetlacz DE 117-RS-20/7,5/V  
Q1: rezonator kwarcowy 32,768 kHz  
S1...S3: przycisk 6 mm kątowny  
Koszyk baterii 2×AAA (opis w tekście)



Rysunek 3. Przebiegi czasowe napięć na wyprowadzeniach wyświetlacza LCD

**DODATKOWE MATERIAŁY NA FTP:**

ftp://ep.com.pl

USER: 97325, PASS: 6yfwxr8q

**W ofercie AVT\*****AVT-5572****Podstawowe informacje:**

- Dwustronna płytka drukowana o wymiarach 38 mm×54 mm.
- Ciągłe wyświetlanie aktualnego czasu (minuty i godziny).
- Załączanie i wyłączanie jednego wyjścia o dowolnie wybranej porze (raz w ciągu doby).
- Zapamiętywanie nastaw godziny włączenia i wyłączenia w nieulotnej pamięci EEPROM.
- Zasilanie napięciem 3 V lub o zbliżonej wartości.
- Pobór prądu na tyle niski, aby komplet baterii zapewnił wiele miesięcy bezobrotowego działania.

**Projekty pokrewne na FTP:**

(wymienione artykuły są w całości dostępne na FTP)

AVT-5560	Programowalny układ czasowy „Tajmerek” (EP 9/2016)
AVT-1879	Przełącznik czasowy start-stop (EP 8/2015)
AVT-5467	Programowany Timer (EP 9/2014)
AVT-1821	Czasówka ON/OFF (EP 8/2014)
AVT-1820	Programowany przełącznik czasowy (EP 8/2014)
AVT-5410	Time-ek – sterownik czasowy (EP 10/2013)
AVT-1710	Regulowany wyłącznik opóźniający (EP 10/2012)
AVT-1689	Przełącznikowy wyłącznik czasowy (EP 8/2012)
AVT-1684	Automatyczny wyłącznik czasowy (EP 8/2012)
AVT-1535	Przełącznik czasowy (EP 8/2009)
AVT-1459	Uniwersalny układ czasowy (EP 12/2007)
AVT-724	Uniwersalny układ czasowy (inteligentny wyłącznik schodowy) (EdW 7/2004)
AVT-2622	Uniwersalny przełącznik czasowy (EdW 11/2003)
AVT-2442	Automatyczny wyłącznik czasowy (automat schodowy) (EdW 11/2000)

\* Uwaga:  
 Zestaw AVT mogą występować w następujących wersjach:  
 AVT xxxx UK to zaprogramowany układ, tylko i wyłącznie. Bez elementów dodatkowych.  
 AVT xxxx A płytka drukowana PCB (lub płytki drukowane, jeśli w opisie wyrażenie zaznaczono), bez elementów dodatkowych.  
 AVT xxxx AP płytka drukowana i zaprogramowany układ (czyli połączenie wersji A i wersji UK) bez elementów dodatkowych.  
 AVT xxxx B płytka drukowana (lub płytki) oraz komplet elementów wymienionych w załączniku pdf.  
 AVT xxxx G to nie inego jak zmontowany zestaw B, czyli elementy wlotowane w PCB. Należy mieć na uwadze, że o ile nie zaznaczono wyraźnie w opisie, zestaw ten nie ma obudowy ani elementów dodatkowych, które nie zostały wymienione w załączniku pdf.  
 AVT xxxx CD oprogramowanie (nieczęsto spotykana wersja, lecz jeśli występuje, to niezbędne oprogramowanie można ściągnąć, klikając w link umieszczony w opisie kitu).  
 Nie każdy zestaw AVT występuje we wszystkich wersjach! Każda wersja ma załączony ten sam plik pdf. Podczas składania zamówienia upewnij się, którą wersję zamawiasz! (UK, A, A+, B lub C). <http://sklep.avt.pl>

wyświetlaczem angażowane są tylko 3 wyprowadzenia mikrokontrolera.

Wyświetlacz LCD powinien być sterowany w taki sposób, aby pomiędzy segmentami a elektrodą wspólną nie występowała składowa stała, ponieważ jej długotrwała obecność może uszkodzić wyświetlacz. Jednocześnie należy zapewnić odświeżanie z częstotliwością kilkudziesięciu herców. Dlatego przyjęto, że napięcie na wyprowadzeniu wygaszonego segmentu zmienia się w fazie z potencjałem elektrody wspólnej, a na załączonym – w przeciwfazie. Ilustruje to **rysunek 3**. W obu sytuacjach średnia różnica potencjałów między segmentem a elektrodą wspólną jest zerowa.

**Budowa zegara**

Schemat ideowy zegara pokazano na **rysunku 4**. W sterowniku wyświetlacza LCD, złożonym z układów US3...US6, wszystkie wejścia SCL (poziom niski czysty zawartość rejestru) zostały podciągnięte do dodatniego bieguna zasilania. Z kolei wejścia OE (poziom niski uaktywnia wyjścia) zostały połączone z masą. W ten sposób, wprowadzając tranzystor T1 w stan zatkania, można wyłączyć zasilanie rejestrów. Należy jednocześnie ustawić linie DATA, STCP i SHCP, aby prąd nie przedostawał się przez diody wygaszające. Zasilanie rejestrów jest załączane tylko na czas odświeżenia ich zawartości, przez pozostały czas pozostaje wyłączone. Kondensatory C5 i C6 redukują tętnienia na liniach zasilających rejestry podczas wpiśnięcia do nich nowej wartości.

Wyjście, które jest załączane i wyłączane przez zegar, znajduje się na zaciskach złącza J2. W stanie załączenia zwiera je tranzystor T2. Jest to tranzystor MOSFET typu N, więc napięcie przyłożone do tych zacisków musi mieć odpowiednią polaryzację: wyższy potencjał jest przyłożony do drenu (HI), a niższy do źródła (LO). Użyty w tym miejscu tranzystor unipolarny nie pobiera z baterii żadnego prądu, ponieważ prąd płynący przez jego bramkę jest pomijalnie mały. Tranzystory typu IRLML2502 dobrze nadają się do aplikacji z tak niskim napięciem zasilającym. Potwierdza to nota katalogowa, według której napięcie progowe wynosi od 0,6 V do 1,2 V (przy  $ID=250 \mu A$ ). Z kolei, przy  $UGS=3 V$ , prąd drenu może sięgać nawet 3 A ze spadkiem napięcia dren-źródło nieprzekraczającym 100 mV.

Za odmierzenie czasu jest odpowiedzialny układ US2 typu M41T00. W swojej strukturze ma kompletny zegar czasu rzeczywistego (RTC) z generatorem dla rezonatora kwarcowego 32768 Hz. Z mikrokontrolerem komunikuje się za pomocą interfejsu I<sup>2</sup>C.

Wejście VBAT, które służy do podtrzymania odliczania, na stałe dołączono do baterii. Z kolei wejście VCC, z którego jest zasilana część odpowiedzialna za komunikację szeregową, jest sterowane wprost z wyprowadzenia mikrokontrolera. Do tego samego wyprowadzenia dołączono rezystory podciągające magistrali I<sup>2</sup>C. Gotowość do zaktualizowania informacji o czasie jest potrzebna rzadko, dlatego przez większość czasu potencjał wejścia VCC jest bliski zeru. Dopiero wtedy, kiedy zachodzi konieczność nawiązania łączności z układem M41T00, wyprowadzenie VCC zostaje podciągnięte do wysokiego poziomu logicznego, co jest tożsamy z podaniem nań zasilania. Kondensatory C1 i C2 zmniejszają impedancję widzianą przez wejścia zasilania.

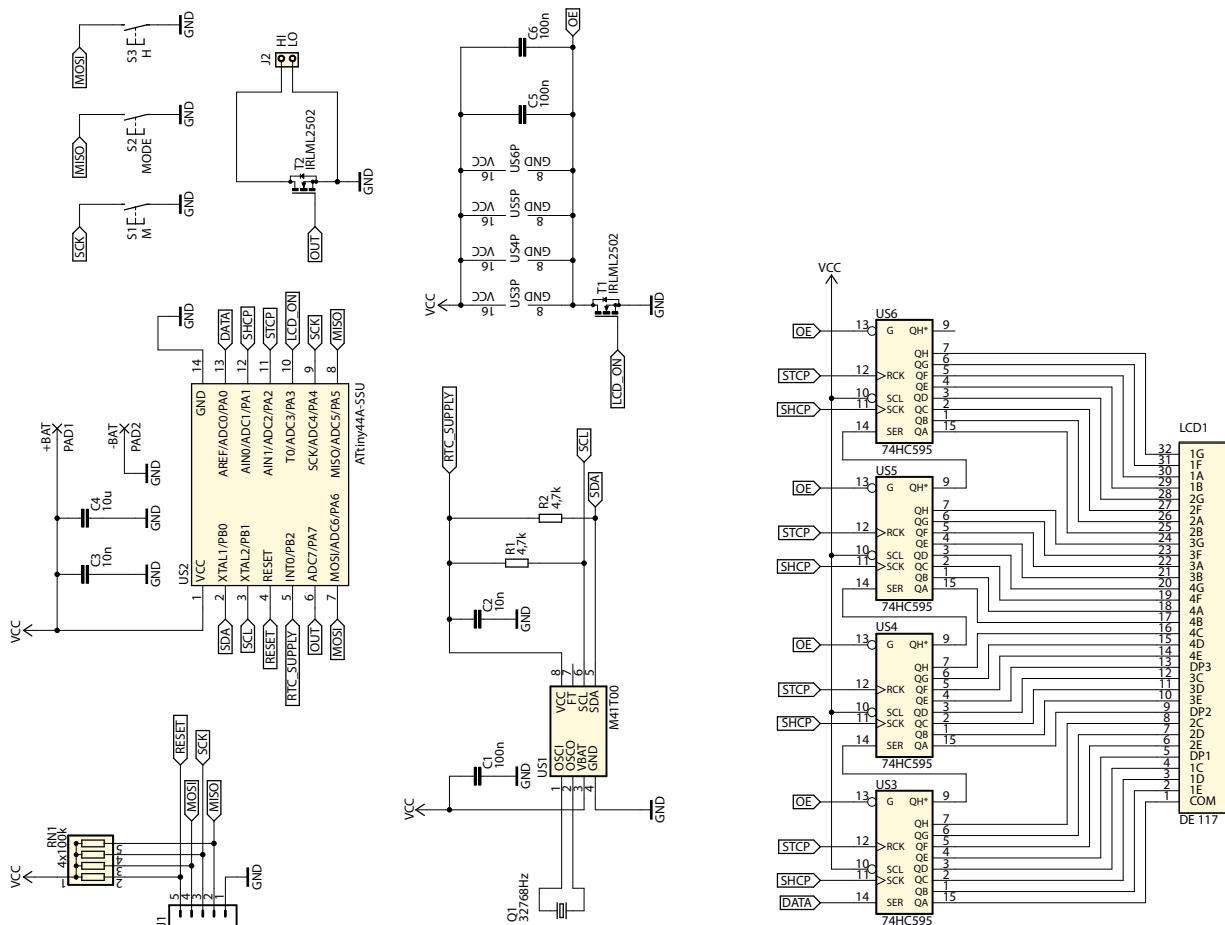
Całością zarządza mikrokontroler ATtiny44A w obudowie SMD. Ma dostateczną liczbę wyprowadzeń do realizacji tego zadania. Ta wersja może być zasilana napięciem już od 1,8 V i dlatego dobrze nadaje się do zasilania bateriami. Taktowany jest wewnętrznym oscylatorem RC. Dla zredukowania poboru mocy przez większość czasu pozostaje w uśpieniu. Kondensatory C3 i C4, dołączone blisko wyprowadzeń zasilania, zmniejszają tętnienia tegoż – zwłaszcza gdy rezystancja wewnętrzna baterii, wskutek rozładowania, stanie się znacząca.

Obsługa odbywa się za pomocą 3 przycisków monostabilnych, które dołączono do tych wyprowadzeń mikrokontrolera, które odpowiadają jednocześnie za programowanie ISP. Wewnętrzne rezystory podciągające w mikrokontrolerze zostały wyłączone, aby zmniejszyć pobór prądu, dlatego trzeba użyć zewnętrznych z drabinki RN1.

W całym układzie zastosowano kondensatory ceramiczne ze względu na małe wymiary oraz – co ważniejsze – znacznie mniejszy prąd upływu w porównaniu z kondensatorami elektrolitycznymi.

**Montaż i uruchomienie**

Układ zegara został zmontowany na dwustronnej płytce drukowanej o wymiarach



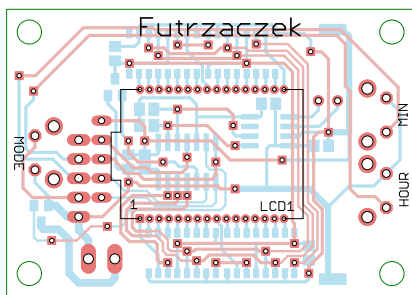
Rysunek 4. Schemat ideowy zegara włącz/wyłącz

38 mm×54 mm, której schemat montażowy pokazano na **rysunku 5**. Na płytce są elementy powierzchniowe i przewlekane, więc proponuję następującą kolejność montażu: wszystkie podzespoły SMD, rezonator kwarcowy Q1, drabinka RN1, złącze goldpin J1, złącze ARK2 J2, przyciski S1...S3, wyświetlacz LCD1. Należy pamiętać, że wszystkie elementy, poza wyświetlaczem, znajdują się po spodniej stronie płytki drukowanej. Podczas wkładania wyprowadzeń wyświetlacza w otworki należy zachować ostrożność, ponieważ silniejsze wygięcie którejkolwiek z nich może połamać kruche szkło wyświetlacza. Widok płytki obsadzonej częściami prezentuje **fotografia 6**. Na samym końcu można doprowadzić zasilanie do pól lutowniczych +BAT i -BAT. Powinno ono wynosić ok. 3 V i może pochodzić, na przykład, z dwóch połączonych szeregowo baterii

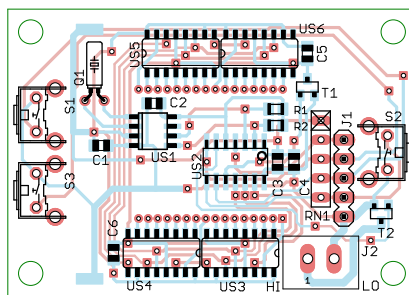
AAA. Warto zauważyć przy tym, że nawet niewielka bateria typu CR2032 pozwoli na pracę tego układu przez długi czas. Po włączeniu zasilania należy przejść do zaprogramowania mikrokontrolera. Ustawienie bitów konfiguracyjnych jest następujące:  
 Low Fuse: 0xE2.  
 High Fuse: 0xDF.  
 Extended Fuse: 0xFF.  
 Na **rysunku 7** pokazano zrzut ekranu z programu BitBurner. Względem konfiguracji domyślnej (fabrycznej), zmianie uległ jedynie bit CKDIV8 – rdzeń jest taktowany przebiegiem o częstotliwości 8 MHz. Mogłoby się wydawać, że podwyższenie częstotliwości taktowania przełoży się na zwiększenie pobieranej mocy, lecz w rzeczywistości, jeśli oprogramowanie mikrokontrolera zostanie odpowiednio przemyślane i napisane, jest inaczej.

### Eksplotacja

Po włączeniu zasilania na wyświetlaczu zostaną wyświetlone 4 zera, ponieważ zegar działa w trybie 24-godzinnym. Kropka pośrodku powinna regularnie migać – oznacza to poprawną komunikację z układem RTC. Po wciśnięciu przycisku S2 zegar przejdzie w tryb nastaw aktualnego czasu, o czym informuje, pokazując przez chwilę litery cL (od „clock”) – **fotografia 8**. Wcisnąc przycisk S1, zwiększymy aktualną liczbę minut, a przyciskiem S3 liczbę godzin. Ponownie wciskając S2, możemy zmienić czas załączenia wyjścia (**fotografia 9**). Przyciski S1 i S3 działają w taki sam sposób, jak poprzednio. Po ponownym wciśnięciu S2 zegar przejdzie do podmenu zmiany czasu wyłączenia wyjścia – komunikat pokazano na **fotografii 10**. Przez załączenie wyjścia jest rozumiane jego zwarcie do masy.



Rysunek 5. Schemat montażowy płytki drukowanej zegara



Fotografia 6. Widok zmontowanej płytki od spodu

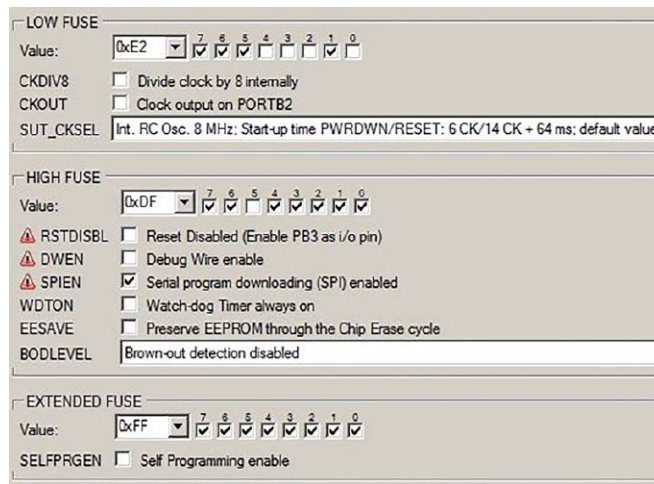
Układ zapamięta nastawy i powróci do pracy po upływie około 5 sekund od ostatniej operacji wykonanej przez użytkownika lub po czwartym z kolei wciśnięciu S2. W czasie pracy, jak i podczas zmiany nastaw, ekran wygląda tak, jak na **fotografii 11**. Jedyną różnicą jest kropka: w czasie pracy kropka miga, a w pozostałych przypadkach pozostaje stale zaczerniona.

Pobór prądu ze źródła o napięciu 3 V wynosi ok. 25  $\mu$ A. Zakładając pojemność baterii CR2032 na poziomie 220 mAh, powinna ona wystarczyć na rok pracy zegara. Na dwóch bateriach AAA (pojemność ok. 900 mAh) może on działać nawet przez 4 lata!

Jeżeli napięcie spadnie do 2 V, zegar nadal działa i pobiera wtedy prąd ok. 15  $\mu$ A. Mogą jednak pojawić się problemy z prawidłowym otwarciem tranzystora T2 sterującego wyjściem. Z kolei zwiększanie napięcia powyżej 3 V jest niewskazane ze względu na wyświetlacz, który jest przystosowany do tego właśnie napięcia. Maksymalne napięcie, jakie można przyłożyć do zacisków złącza J2, wynosi 20 V, ponieważ tyle wynosi maksymalne napięcie dren-źródło tranzystora IRLML2502. Musi być to napięcie stałe, aby dioda zawarta w tranzystorze MOSFET była spolaryzowana zaporowo. Można przyjąć, że maksymalny prąd przewodzony przez ten tranzystor wynosi do 2 A. To z powodzeniem wystarczy do załączenia przekaźnika, małego silnika lub innego odbiornika. W przypadku obciążeń indukcyjnych odpowiednią diodę zabezpieczającą należy umieścić na zaciskach tego obciążenia.

## Dla ciekawskich

W bardzo łatwy sposób osiągnięto odpowiednie odwracanie poziomów logicznych względem nóżki COM – wystarczy



**Rysunek 7. Zrzut okna konfiguracji bitów zabezpieczających programu BitBurner**

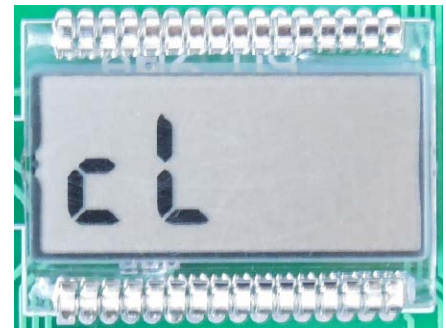
zanegować całą 32-bitową zmienną, bit po bicie. Segment wyłączony pozostanie w tym stanie, ponieważ wyprowadzenia COM i wyłączony segment nadal będą miały ten sam poziom logiczny. Ta sama reguła dotyczy segmentów załączonych (zaczernionych). Podczas jednego wywołania zawartość tej zmiennej jest aktualizowana (według tego, co ma się znaleźć na wyświetlaczu), a podczas drugiego tylko negowana.

Odpowiednie interwały czasowe, niezbędne do sterowania wyświetlaczem z określoną częstotliwością, są realizowane poprzez cykliczne usypianie i budzenie mikrokontrolera. Za budzenie jest odpowiedzialny układ watchdog, który generuje przerwania co ok. 32 ms. Ma on własny generator zegara o częstotliwości 128 kHz, więc działa niezależnie od rdzenia. Po obudzeniu, mikrokontroler wykonuje zadane czynności, po czym ponownie przechodzi w stan uśpienia.

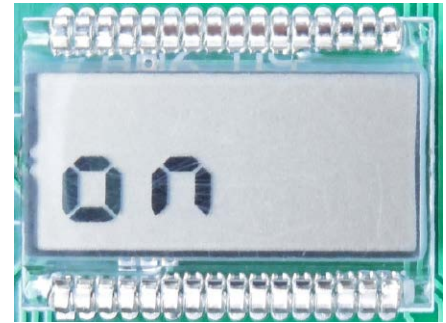
Paradoksalnie, wyższa częstotliwość zegara głównego przyczyniła się do zmniejszenia średniego natężenia pobieranego prądu z 35  $\mu$ A przy 1 MHz do 25  $\mu$ A przy 8 MHz (dla napięcia 3 V), czyli o blisko 30%. Powodem jest krótszy czas załączenia zasilania dla rejestrów 74HC595 – sprzyja temu szybsze wpisywanie do nich nowej zawartości. Fragment kodu, odpowiedzialny za usypianie, budzenie oraz konfigurację, pokazano na **listingu 1**.

Michał Kurzela, EP

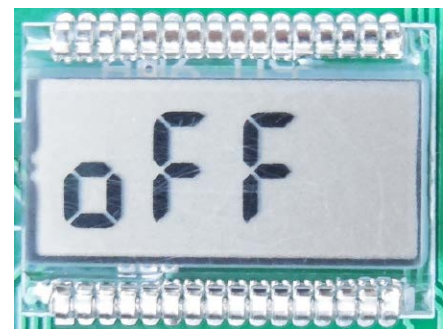
```
Listing 1. Kod odpowiedzialny za ograniczenie pobieranej mocy
ISR(WDT_vect){ } //pusta obsługa przerwania, aby nie pozostało nieobsłużone
int main(void)
{
    sei(); //uruchomienie systemu przerw
    WDTCSR = (1<<WDIE) | (1<<WDP0); //uruchomienie przerw od watchdoga + timeout 32ms
    while(1){
        set_sleep_mode(SLEEP_MODE_PWR_DOWN);
        PORTA &= 0b11110111; //wyłączenie tranzystora w ,595
        PORTA |= 0b00000111; //ustawienie linii danych ,595 na 1, aby przez diody clampujące nie płynął prąd
        sleep_enable();
        sleep_cpu();
        sleep_disable();
        WDTCSR = 0; //wyłączenie watchdoga
        PORTA |= 0b00001000; //załączenie zasilania w ,595
        PORTA &= 0b11111000; //wyzerowanie linii sterujących
    }
}
```



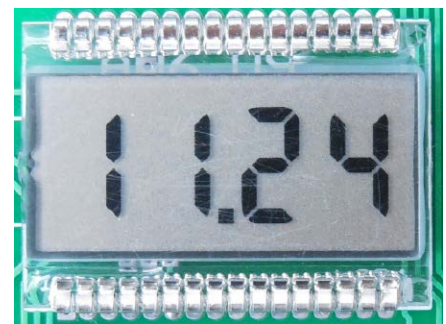
**Fotografia 8. Informacja o przejściu w tryb nastawy czasu**



**Fotografia 9. Komunikat o przejściu do czasu załączenia wyjścia**



**Fotografia 10. Komunikat o przejściu do czasu wyłączenia wyjścia**



**Fotografia 11. Obraz na wyświetlaczu pracującego zegara**