

# Nietypowy zamek szyfrowy

## Do czego służy?

Powszechnie wiadomo, że złodzieje mają dostęp do fabrycznych urządzeń alarmowych i szyfrowych. Dzięki temu łatwiej łamią zabezpieczenia. Dlatego panuje słuszne przekonanie, iż to właśnie nietypowe rozwiązania okazują się najlepsze. Typowy zamek szyfrowy z kombinacją 4 cyfr posiada 10 000 kombinacji, lecz stojąc obok wprowadzającego kod, bez problemu można go podejrzec i później wejść do strzeżonego pomieszczenia. Zaprezentowany w artykule zamek ma tylko 1440 kombinacji. Posiada jednak tę zaletę, że kod zmienia się co minutę i tylko właściciel wie o ile faktycznie kod ten się zmienił. Jeśli kod zostanie wpisany nieprawidłowo trzy razy z rzędu, to układ zostanie zablokowany na kilka minut, co skutecznie powinno zniechęcić złodzieja.

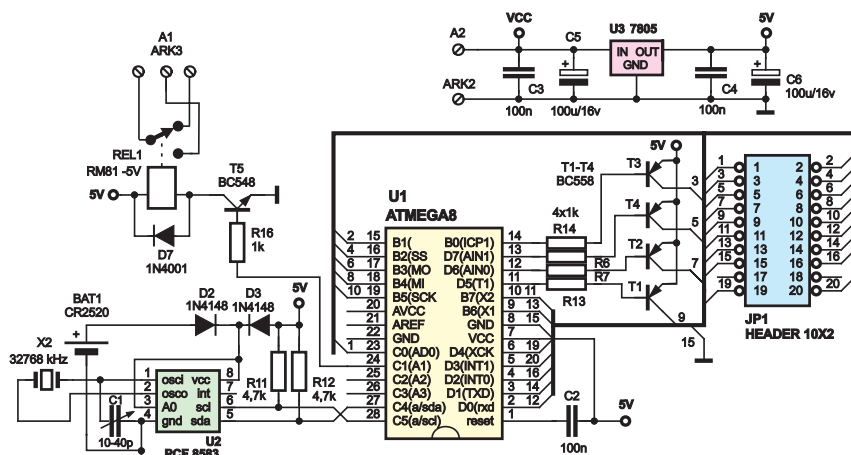
Warto wspomnieć, że kod zamka zależny jest od pokazywanej przez wyświetlacz aktualnej godziny.

## Jak to działa?

Schemat ideowy części sterującej jest przedstawiony na rysunku 1. Głównym elementem jest procesor ATMEGA8, który steruje pracą całego układu. Tranzystory T1-T4 służą do multipleksowania wyświetlacza znajdującego się na drugiej płytce. PCF8583 (U2) to typowa aplikacja zegara czasu rzeczywistego wraz z rezerwowym zasilaniem w postaci baterii CR2520 komunikujący się z procesorem za pomocą magistrali I<sup>2</sup>C. Wyjścia przekaźnika podłączone pod rygiel w drzwiach sterują otwieraniem i zamykaniem drzwi.

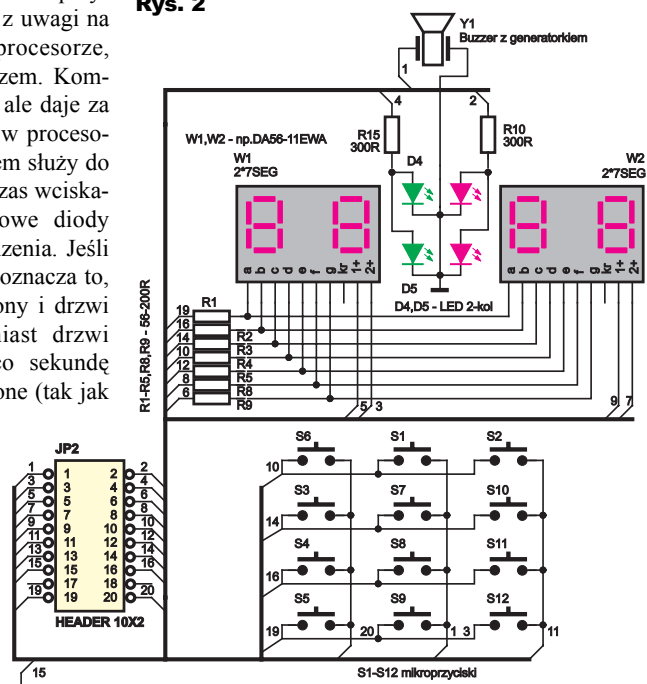
Schemat wyświetlacza i klawiatury jest przedstawiony na rysunku 2. Wszystkie segmenty wyświetlaczy od a do g są ze sobą połączone, a który wyświetlacz aktualnie wyświetla czas, zależy od tego, na której anodzie pojawi się napięcie zasilania, czyli który

z tranzystorów T1-T4 będzie otwarty. Klawiatura sterująca posiada 12 przycisków, z których 4 linie, z uwagi na małą liczbę wyjść w procesorze, współdzielili z wyświetlaczem. Kompilując to nieco program, ale daje za to więcej wolnych wyjść w procesorze. Buzzer z generatorkiem służy do wydawania dźwięku podczas wciskania klawiszy. Dwukolorowe diody LED wskazują stan urządzenia. Jeśli świecą się diody zielone, oznacza to, iż przekaźnik jest załączony i drzwi są otwarte. Jeśli natomiast drzwi zostaną zamknięte, to co sekundę będą migały diody czerwone (tak jak w zwykłym zegarku). Jeśli diody te będą stale świeciły się na czerwono, oznacza to, iż ktoś próbował złamać kod i trzykrotnie źle go wprowadził. W takiej sytuacji należy odczekać około 3 minut, aby diody



Rys. 1

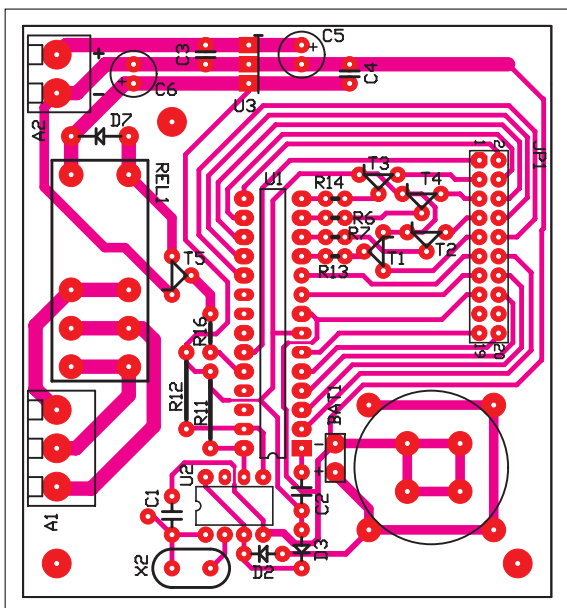
Rys. 2



z powrotem zaczęły migać. Rezystory R1-R5, R8, R9 ustalają jasność świecenia wyświetlaczy (56-200Ω).

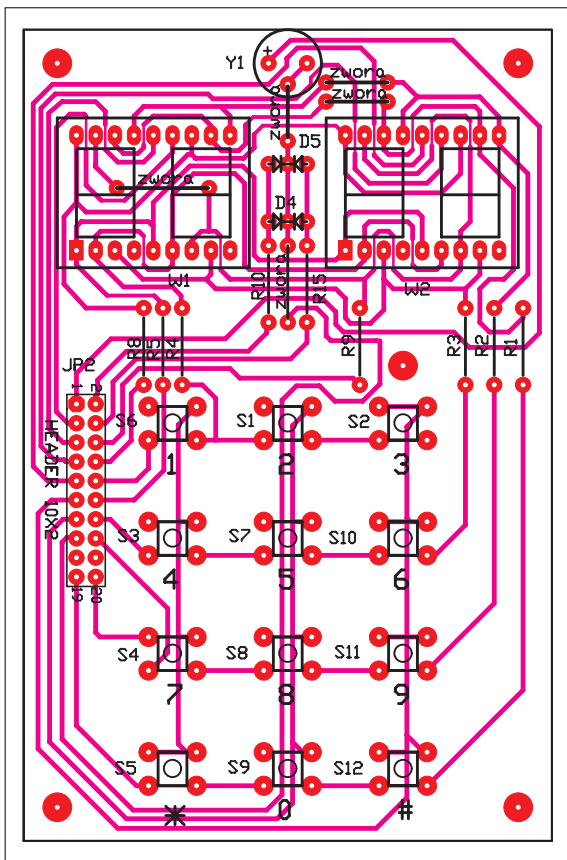
## Montaż i uruchomienie

Układ został zmontowany na dwóch płytkach. Na jednej z nich znajduje się układ sterujący oraz stabilizator napięcia (rysunek 3), a na drugiej wyświetlacze i przyciski (rysunek 4).



Rys. 3

Rys. 4



Montaż wykonujemy w typowy sposób, zaczynając od czterech zworek na płycie wyświetlacza, a kończąc na największych gabarytowo elementach. Złącze JP2 powinno być kątowe. Po wlutowaniu wszystkich elementów oraz zarobieniu i wpięciu tasiemki komputerowej (20-tki), można płytki połączyć ze sobą za pomocą dystansów, ścieżkami do siebie. Aby zwiększyć stopień bezpieczeństwa, płytka sterownika powinna znajdować się w miejscu niedostępnym dla potencjalnego złodzieja, czyli jak najdalej od układu z wyświetlaczami. Złodziej jednak nie powinien się domyślić jak układ jest zbudowany i dlatego proponuję zwartą budowę w postaci dwóch płytek obok siebie. Ewentualnie do istniejącego już systemu alarmowego możnadodać czujnik antysabotażowy, umieszczony tuż pod obudową układu.

Po włączeniu zasilania powinna ukazać się godzina 00:00. Jeśli jednak ktoś wcześniej włożył baterię - godzina ta może się różnić o kilka minut. Aby ustawić aktualny czas, należy wyłączyć zasilanie (baterii nie trzeba wyjmować), a następnie nacisnąć przycisk z gwiazdką \* i przytrzymując go, włączyć zasilanie. Gdy wyświetlacze „zgasną”, należy puścić przycisk \*, na wyświetlaczach pojawi się napis hour. Następnie wpisujemy aktualną godzinę. W razie pomyłki wpisujemy godzinę jeszcze raz. Naciskamy teraz przycisk z krzyżykiem # i diody powinny zacząć migać na czerwono co około 1s.

Aby ustawić kod, należy ponownie wyłączyć zasilanie i nacisnąć przycisk #. Po włączeniu zasilania i puszczeniu przycisku powinien ukazać się napis oPEn, oznaczający możliwość wpisywania wartości, o jaką ma różnić się kod od aktualnej godziny. I tak np. wpis 0015 oznacza, że jeśli jest przykładowo godzina 14:20 - to kod będzie wynosił 1435, czyli będzie większy o 15 od godziny. W przypadku gdy będzie godzina 14:55, to kod wyniesie 1510.

Po ustawieniu kodu naciskamy przycisk # i testujemy układ (na początek można ustawić 0000, wtedy kod będzie odpowiadał aktualnej godzinie). Po ustawieniu kodu 0000 naciskamy po kolei liczby, które są aktualnie na wyświetlaczach. Jeśli ustawiony został inny kod, to należy do wyświetlanej godziny dodać odpowiednią wartość, a następnie wpisać kod, który aktualnie wypada. Po wpisaniu kodu naciskamy krzyżyk # albo gwiazdkę \*. Od tego co naciśniemy zależy działanie przekaźnika. Po naciśnięciu krzyżyka # przekaźnik załączy się na czas 4 sekund, a następnie się wyłączy. Gdy po wpisaniu kodu naciśniemy gwiazdkę \* - wówczas przekaźnik załączy się na stałe. Wyłączenie go nastąpi po naciśnięciu krzyżyka #.

Największą wartość kodu, jaką można wpisać, jest 9999, co jest równoznaczne z wpisaniem 0439 (4\*24h+4h39m=99h,99 min), zostawiłem tę możliwość na wypadek gdyby komuś się przydała. W normalnych warunkach największą wartością, jaką wpisujemy, jest 2359, co będzie odpowiadało kodowi o jedną minutę mniej niż widzimy na wyświetlaczu.

A co się stanie, gdy w momencie wpisywania kodu zmieni się godzina? Taka sytuacja jak najbardziej może się zdarzyć i jest na nią sposób. Po prostu zignorować zmieniającą się godzinę. Program został tak napisany, aby zapamiętywał godzinę w momencie wciśnięcia pierwszej cyfry. Czyli po prostu wpisujemy kod taki, jaki mieliśmy wpisać, nie zważając na zmianę godziny. Stan przekaźnika możemy obserwować poprzez stan świecenia

Wyznaczony jest napis oPEn, oznaczający możliwość wpisywania wartości, o jaką ma różnić się kod od aktualnej godziny. I tak np. wpis 0015 oznacza, że jeśli jest przykładowo godzina 14:20 - to kod będzie wynosił 1435, czyli będzie większy o 15 od godziny. W przypadku gdy będzie godzina 14:55, to kod wyniesie 1510.

Po ustawieniu kodu naciskamy przycisk # i testujemy układ (na początek można ustawić 0000, wtedy kod będzie odpowiadał aktualnej godzinie). Po ustawieniu kodu 0000 naciskamy po kolei liczby, które są aktualnie na wyświetlaczach. Jeśli ustawiony został inny kod, to należy do wyświetlanej godziny dodać odpowiednią wartość, a następnie wpisać kod, który aktualnie wypada. Po wpisaniu kodu naciskamy krzyżyk # albo gwiazdkę \*. Od tego co naciśniemy zależy działanie przekaźnika. Po naciśnięciu krzyżyka # przekaźnik załączy się na czas 4 sekund, a następnie się wyłączy. Gdy po wpisaniu kodu naciśniemy gwiazdkę \* - wówczas przekaźnik załączy się na stałe. Wyłączenie go nastąpi po naciśnięciu krzyżyka #.

Największą wartość kodu, jaką można wpisać, jest 9999, co jest równoznaczne z wpisaniem 0439 (4\*24h+4h39m=99h,99 min), zostawiłem tę możliwość na wypadek gdyby komuś się przydała. W normalnych warunkach największą wartością, jaką wpisujemy, jest 2359, co będzie odpowiadało kodowi o jedną minutę mniej niż widzimy na wyświetlaczu.

A co się stanie, gdy w momencie wpisywania kodu zmieni się godzina? Taka sytuacja jak najbardziej może się zdarzyć i jest na nią sposób. Po prostu zignorować zmieniającą się godzinę. Program został tak napisany, aby zapamiętywał godzinę w momencie wciśnięcia pierwszej cyfry. Czyli po prostu wpisujemy kod taki, jaki mieliśmy wpisać, nie zważając na zmianę godziny. Stan przekaźnika możemy obserwować poprzez stan świecenia

Wyznaczony jest napis oPEn, oznaczający możliwość wpisywania wartości, o jaką ma różnić się kod od aktualnej godziny. I tak np. wpis 0015 oznacza, że jeśli jest przykładowo godzina 14:20 - to kod będzie wynosił 1435, czyli będzie większy o 15 od godziny. W przypadku gdy będzie godzina 14:55, to kod wyniesie 1510.

Po ustawieniu kodu naciskamy przycisk # i testujemy układ (na początek można ustawić 0000, wtedy kod będzie odpowiadał aktualnej godzinie). Po ustawieniu kodu 0000 naciskamy po kolei liczby, które są aktualnie na wyświetlaczach. Jeśli ustawiony został inny kod, to należy do wyświetlanej godziny dodać odpowiednią wartość, a następnie wpisać kod, który aktualnie wypada. Po wpisaniu kodu naciskamy krzyżyk # albo gwiazdkę \*. Od tego co naciśniemy zależy działanie przekaźnika. Po naciśnięciu krzyżyka # przekaźnik załączy się na czas 4 sekund, a następnie się wyłączy. Gdy po wpisaniu kodu naciśniemy gwiazdkę \* - wówczas przekaźnik załączy się na stałe. Wyłączenie go nastąpi po naciśnięciu krzyżyka #.

Największą wartość kodu, jaką można wpisać, jest 9999, co jest równoznaczne z wpisaniem 0439 (4\*24h+4h39m=99h,99 min), zostawiłem tę możliwość na wypadek gdyby komuś się przydała. W normalnych warunkach największą wartością, jaką wpisujemy, jest 2359, co będzie odpowiadało kodowi o jedną minutę mniej niż widzimy na wyświetlaczu.

A co się stanie, gdy w momencie wpisywania kodu zmieni się godzina? Taka sytuacja jak najbardziej może się zdarzyć i jest na nią sposób. Po prostu zignorować zmieniającą się godzinę. Program został tak napisany, aby zapamiętywał godzinę w momencie wciśnięcia pierwszej cyfry. Czyli po prostu wpisujemy kod taki, jaki mieliśmy wpisać, nie zważając na zmianę godziny. Stan przekaźnika możemy obserwować poprzez stan świecenia

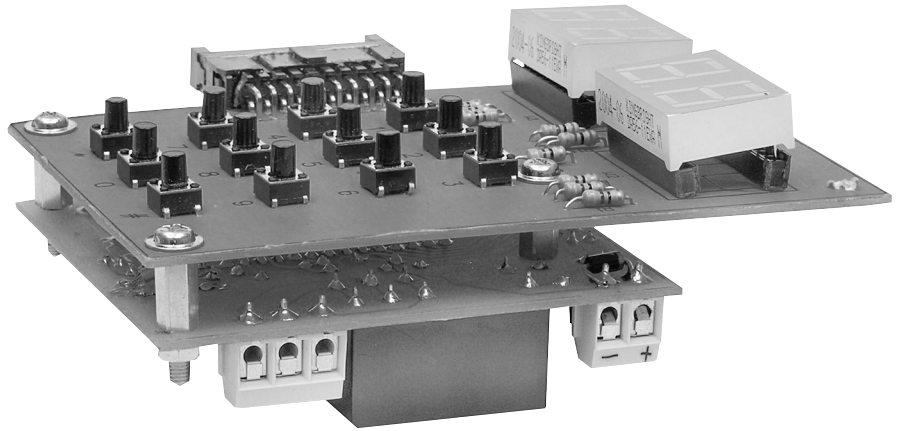
| Wykaz elementów                               |                                    |
|---|------------------------------------|
| <b>Rezystory</b>                              | T5..... BC548                      |
| R1-R5,R8,R9..... 56-200Ω                      | U1..... ATMEGA 8                   |
| R6,R7,R13,R14,R16..... 1kΩ                    | U2..... PCF 8583                   |
| R10,R15..... 300Ω                             | U3..... 7805                       |
| R11,R12..... 4,7kΩ                            | W1,W2..... DA56-11EWA              |
| <b>Kondensatory</b>                           | <b>Pozostałe</b>                   |
| C1... trymer 10-40pF lub kondensator ok. 30pF | A1..... ARK3                       |
| C2-C4..... 100nF                              | A2..... ARK2                       |
| C5,C6..... 100μF/16V                          | X2..... 32768kHz                   |
| D1..... 1N4001                                | Y1..... buzzer z generatorem       |
| D2,D3..... 1N4148                             | S1-S12..... mikroprzyciski         |
| D4,D5..... LED dwukolorowa 3mm                | REL1..... przekaźnik 5V            |
| T1-T4..... BC558                              | JP1,JP2..... złącza 2*10pin kątowe |
|   | Podstawki: 28-PIN i 8-PIN          |
|   | 5cm taśmy komputerowej 20          |

Płytką drukowaną jest dostępna w sieci handlowej AVT jako kit szkolny AVT-2796

diod - Zaświecone zielone świadczą o tym, że przekaźnik jest załączony, inny stan - przekaźnik wyłączony.

Może się zdarzyć, że zegar będzie się spóźniał lub spieszył o kilka sekund na dobę. Można to skorygować trymerem C1. Ponieważ odczyt godziny następuje co 10s, należy trymerem zmienić czas dopiero po upływie ok. 24 godzin.

Zasilanie podłączamy do gniazda A2. Powinno mieć ono wartość 6-12V, z tym że im mniejsza jego wartość, tym mniej będzie się grzał stabilizator U3. Przy napięciu zasilania 10-12V stabilizator grzeje dość mocno, przydałby mu się więc mały radiator. Dzieje się tak, gdyż przekaźnik został podłączony za stabilizatorem i po włączeniu pobiera on ok. 80-100mA prądu. Podłączyłem ten przekaźnik w ten sposób, ponieważ chciałem, aby układ (a raczej przekaźnik) nie determinował wartości zasilania. Ponieważ zwykle centraliki są zasilane z 12V, można było zastosować przekaźnik 12V i stabilizator nie musiałby



mieć malutkiego radiatora. Jednak zdarzają się wersje na 6V, a w przypadku oryginalnych rozwiązań także inne wartości zasilania, więc postanowiłem go podłączyć właśnie w ten sposób. Zawsze przecież można dodać niewielki radiator. Nie zwiększy to znacząco kosztów, a polepszy właściwości chłodzące.

Oczywiście na grzanie się stabilizatora wpływa jasność wyświetlacza, którą ustawiamy rezystorami R1-R4, R8, R9.

Płytki drukowane i program można ściągnąć z Elportalu EdW.

**Rafał Kuchta**  
dj\_rav@interia.pl