

Adres	ATtiny2313	ATmega8	Opis
0x000	0x1E	0x1E	Producent
0x001	0x91	0x93	Rozmiar pamięci FLASH
0x002	0x0A	0x07	Typ układu

Tab. 1 Sygnatura układów AVR

Tab. 2 Wartości domyślnie fuse-bitów

	ATtiny2313	ATmega8
Fuse low	0b01100100, 0x64	0b11100001, 0xE1
Fuse high	0b11011111, 0xDF	0b11011001, 0xD9

du ATmega8, podobnie procedury zapisu/ odczytu również mogą wyglądać inaczej dla poszczególnych typów mikrokontrolerów. Z racji ograniczonego zastosowania opisywanego programatora, wykorzystywane są tylko procedury zapisu i odczytu fuse-bitów oraz procedury odczytu sygnatury układu. Po szczegóły dotyczące algorytmów stosowanych przy programowaniu równoległym układów AVR odsyłam zainteresowanych do dokumentów PDF dostępnych na stronie Atmela:

ATtiny2313: www.atmel.com/dyn/resources/prod_documents/DOC2543.PDF

ATmega8: www.atmel.com/dyn/resources/prod_documents/doc2486.pdf

Stamtąd też czerpałem informacje niezbędne do zbudowania urządzenia.

Procedura wprowadzenia układu w tryb programowania równoległego wymaga możliwości sterowania napięciem 12V. Ponieważ napięcie pracy mikrokontrolera AVR nie może osiągać takiej wartości, należało zastosować bufor w postaci tranzystora T1. Aby zrekompenzować różnicę napięć pomiędzy napięciem emitera a stanem wysokim, wynoszącym 5V, wprowadzona została dioda Zenera na napięcie 7,5V. Dzięki niej poprawnie zostaje interpretowany stan logiczny na porcie PB1 kostki U1. Zasilanie układów programowanych czerpane jest bezpośrednio z portu PC5 układu U1 (połączenie PVcc).

Na schemacie widoczne są wyprowadzenia układów U3 i U4, które biorą udział w programowaniu. Połączenia oznaczone jako D0...D7 stanowią szynę danych przekazywanych do i z mikrokontrolera. Pozostałe wyprowadzenia przekazują sygnały sterujące procesem programowania, których rolę można poznać, analizując wyżej wspomniane specyfikacje układów AVR.

Program sterujący pracą programatora napisany został w języku C (WinAVR).

Funkcja main programu pokazana jest na **listingu 1**.

W układzie zrezygnowałem z jakichkolwiek „interfejsów użytkownika” w postaci przełączników albo zwerek; układ automatycznie wykrywa typ programowanego układu, a jedyną czynnością, jaką ma wykonać użytkownik przed zaprogramowaniem układu, jest wciśnięcie przycisku S1 – Reset. Jak widać na **listingu**, program na początku sprawdza, z jakim układem ma do czynienia. Dokonuje tego, odczytując sygnaturę układu, na którą składają się trzy bajty. Adresy i wartości bajtów składowych sygnatury dla obydwu układów przedstawia **tabela 1**.

Następnie kontroler dokonuje zapisu konfiguracji fuse-bitów. **Tabela 2** prezentuje wartości domyślne konfiguracji fuse-bitów, które zostają zapisywane do programowanego układu.

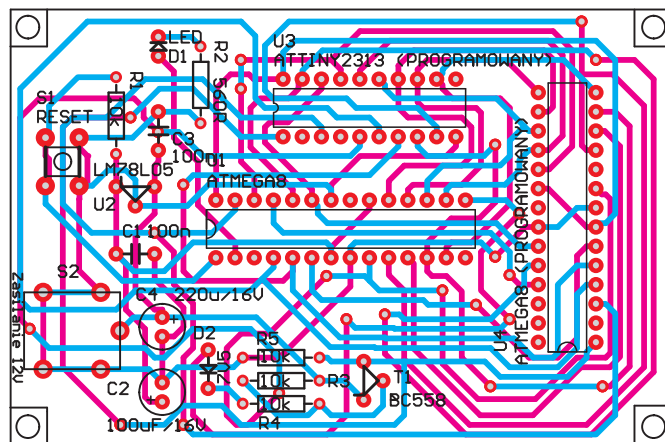
Po zapisie program dokonuje weryfikacji, która polega na odczycie konfiguracji fuse-bitów i porównaniu jej z konfiguracją domyślną dla danego typu układu. Pozytywny wynik weryfikacji oznacza, że układ został prawidłowo zaprogramowany i można go traktować jak świeżo zakupiony mikrokontroler. Prawidłowy proces „naprawy” kostki sygnalizowany jest zaświeceniem zielonej diody LED, natomiast negatywny wynik weryfikacji albo błąd podczas odczytu/zapisu nie jest sygnalizowany w żaden sposób, tzn. dioda LED nie

świeci. Może się też zdarzyć, że brak odzewu ze strony mikrokontrolera podczas programowania szeregowego nie jest spowodowany nieprawidłowym ustawieniem bitów konfiguracyjnych, lecz uszkodzeniem układu. Wtedy nie pozostaje nic innego jak wymienić go na nowy egzemplarz.

Dociekliwi Czytelnicy znajdą w Elportalu dodatkowe informacje na temat funkcji stosowanych w programie w pozostałych plikach projektu w formacie AVRStudio 4 (*m8proc.c*, *t2313proc.c*, *ports.c*).

Montaż i uruchomienie

Układ modelowy został zmontowany na uniwersalnej płytce drukowanej. W celu uniknięcia mnóstwa połączeń użyłem adapterów – dodatkowych płytek stanowiących platformę dla układów ATtiny2313 i ATmega8. Zaprojektowałem również płytkę dwustronną, jednolitą z dwiema podstawkami, zamiast adapterów. Jej widok pokazuje rysunek 2. Montaż w tym przypadku wykonujemy według standardowych reguł, z jedną uwagą – pod układy programowane najlepiej zastosować podstawki precyzyjne.



Rys. 2

Listing 1

```

void main() {
    init(); //initialize
    setLEDG(false);

    if(epm_t2313() == SIGN_2313[2]) {
        plf_t2313(DEF_FUSE_2313[0]); //write default low fuses
        phf_t2313(DEF_FUSE_2313[1]); //write default high fuses
        if(rfb_t2313(0)==DEF_FUSE_2313[0] && rfb_t2313(1)==DEF_FUSE_2313[1]) //verify
            setLEDG(true);
    } else if(epm_m8() == SIGN_M8[2]) {
        plf_m8(DEF_FUSE_M8[0]); //write default low fuses
        phf_m8(DEF_FUSE_M8[1]); //write default high fuses
        if(rfb_m8(0)==DEF_FUSE_M8[0] && rfb_m8(1)==DEF_FUSE_M8[1]) //verify
            setLEDG(true);
    }

    for(;;);
}
    
```

Mikrokontroler U1 został zaprogramowany za pomocą programu *ISP Prog*, a **rysunek 3** przedstawia zrzut z konfiguracją fuse-bitów. Prawidłowo zmontowany układ nie wymaga specjalnych zabiegów uruchomieniowych i po podłączeniu zasilania od razu jest gotów do działania. Podczas użytkowania należy uważać, aby w podstawkach znajdował się tylko jeden układ programowany, nie przetestowano bowiem, jak się zachowa programator w przypadku, jeśli podłączone są ATtiny2313 i ATmega8 jednocześnie do programatora.

Prawdopodobnie dla niektórych ograniczenie obsługi do dwóch typów układów AVR będzie wadą, jednak pójdzie ku rozwojowi urządzenia dla większej liczby układów oznacza zbudowanie kolejnego programatora uniwersalnego dla rodziny AVR. Nic nie stoi na prze-

szkodzie realizacji takiego projektu, a wszelkie sugestie od Czytelników będą mile widziane.

Piotr Wójtowicz
pw@elportal.pl

Wykaz elementów

Rezystory

R1,R3-R5.....10kΩ
R2.....560Ω

Kondensatory

C1,C3.....100nF
C2.....100μF/16V
C4.....220μF/25V

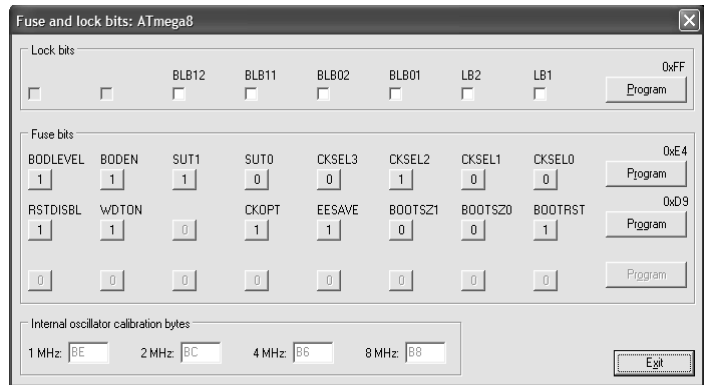
Półprzewodniki

U1..... Atmega8 zaprogramowany
U2..... LM78L05
T1..... BC558
D1..... LED
D2..... dioda Zenera 7V5

Inne

S1..... microswitch
S2..... gniazdo zasilania 12V do druku
Podstawka dip28 precyzyjna wąska x 2 szt.
Podstawka dip20 precyzyjna

Komplet podzespołów z płytką jest dostępny w sieci handlowej AVT jako kit szkolny AVT-2897.



Rys. 3

