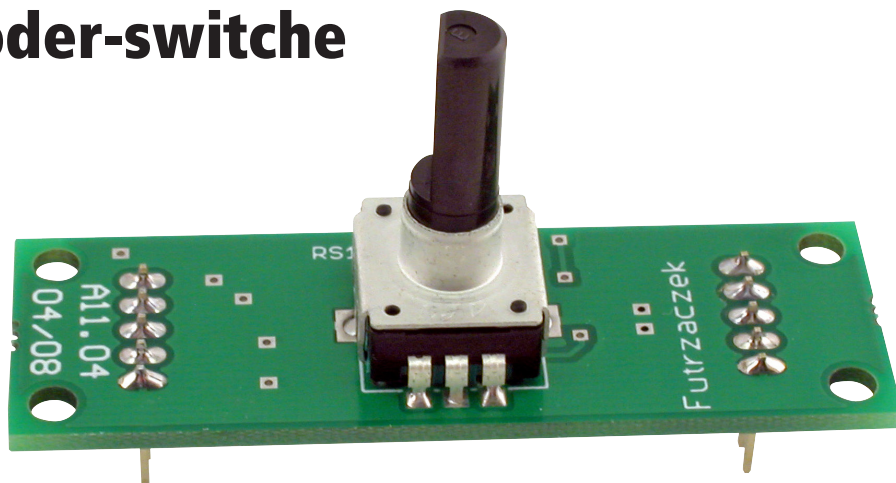


Konwerter enkoder-switcha

W EP 12/13 został opisany układ emulatora enkodera, który generuje sygnał odpowiadający wyjściu kwadraturowemu, po wciśnięciu jednego z przycisków lewo/prawo. Opisane niżej urządzenie działa w drugą stronę, czyli sygnał z mechanicznego enkodera zamienia na ciąg impulsów odpowiadający naciskanym ręcznie przyciskom.



Dodatkowe materiały do pobrania ze strony www.media.avt.pl

W ofercie AVT* AVT-5702

Podstawowe parametry:

- zasilanie napięciem stałym 3..5,5 V
- pobór prądu 1..1,5 mA przy 5 V

Wykaz elementów:

R1, R2, R5..R8, R10: 10 kΩ SMD0805
R3, R4: 330 Ω SMD0805
R9, R11: 1 kΩ SMD0805
C1..C3, C6: 100 nF SMD0805
C4, C5: 10 μF/16 V SMD0805
T1, T2: BC846 lub podobne
U1: Attiny13A-20SSU
J1, J2: goldpin 5 pin męski THT 2,54 mm
RS1: ED1212S-24C24-25F (opis w tekście)

Uwaga! Elektroniczne zestawy do samodzielnego montażu. Wymagana umiejętność lutowania!

Podstawowa wersja zestawu jest wersja [B] nazywana potocznie KIT-em (z ang. zestaw). Zestaw w wersji [B] zawiera elementy elektroniczne (w tym [UK] – jeśli występuje w projekcie), które należy samodzielnie wylutować w dołączoną płytkę drukowaną (PCB). Wykaz elementów znajduje się w dokumentacji, która jest podlinkowana w opisie kitu.

Mając na uwadze różne potrzeby naszych klientów, oferujemy dodatkowe wersje:

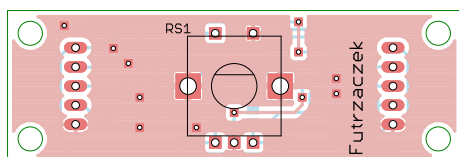
- wersja [C] – zamontowany, uruchomiony i przetestowany zestaw [B] (elementy wylutowane w płytkę PCB)
- wersja [A] – płytkę drukowaną bez elementów i dokumentacji Kity w których występuje układ scalony wymagający zaprogramowania, mają następujące dodatkowe wersje:
- wersja [A*] – płytkę drukowaną [A] + zaprogramowany układ [UK] i dokumentacja
- wersja [UK] – zaprogramowany układ

Nie każdy zestaw AVT występuje we wszystkich wersjach! Każda wersja ma załączony ten sam plik pdf! Podczas składania zamówienia upewnij się, którą wersję zamawiasz! <http://sklep.avt.pl>. W przypadku braku dostępności na <http://sklep.avt.pl>, osoby zainteresowane zakupem płytek drukowanych (PCB) prosimy o kontakt via e-mail: kity@avt.pl.

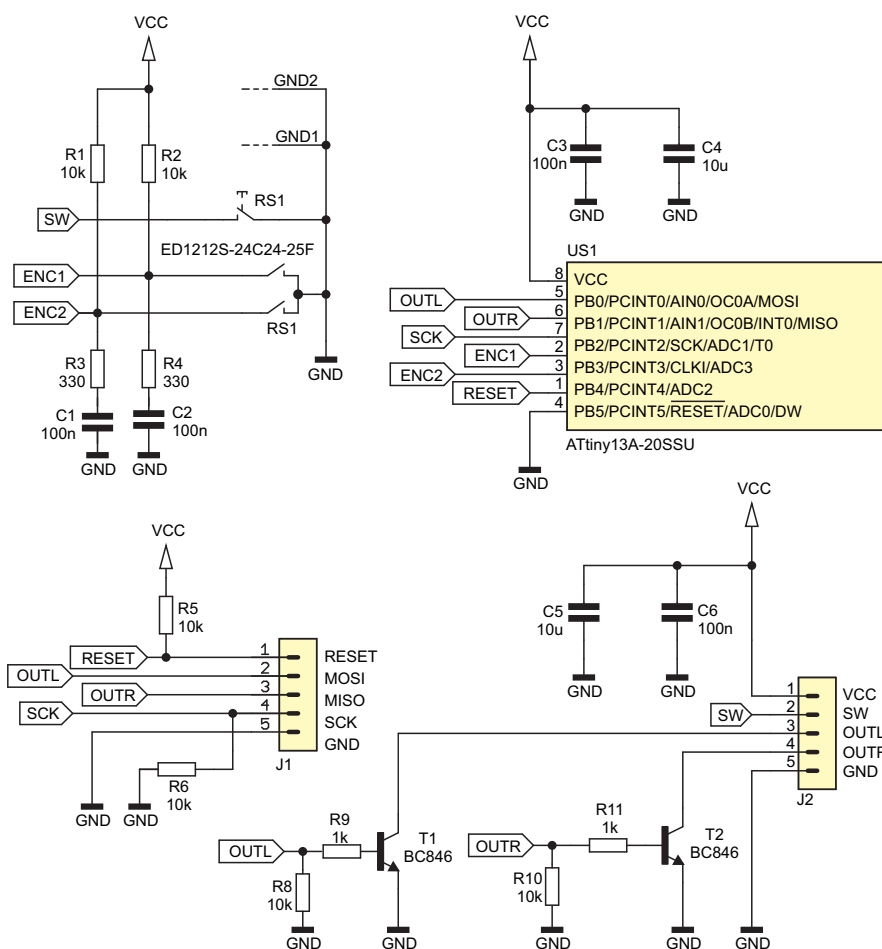
Budowa i działanie

Schemat ideowy układu znajduje się na rysunku 1. Styki enkodera są podciągnięte do dodatniego bieguna zasilania przez rezystory R1 i R2, a styk wspólny został podłączony do masy. Zwarcie wymusi zatem niski stan logiczny. Aby zminimalizować wpływ drgania styków podczas przełączania, układ uzupełniono o kondensatory filtrujące ten sygnał, C1 i C2, wraz z szeregowymi opornikami R3, R4.

Wyjścia symulujące przyciski mają konfigurację OC (open collector) – czyli w stanie aktywnym zwierają do masy. Rezystory R9 i R11 ograniczają prądy baz



Rysunek 2. Wzór ścieżek i schemat montażowy płytki PCB, strona TOP



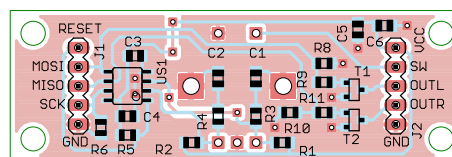
Rysunek 1. Schemat ideowy konwertera

tranzystorów, zaś R8 i R10 utrzymują je w stanie zatkania.

Podzespołem odpowiedzialnym za interpretację sygnałów wystawianych przez enkoder i generowania sygnałów przycisków jest mikrokontroler ATtiny13A. Ten niewielki układ doskonale się do tego celu nadaje.

Do programowania mikrokontrolera przewidziano złącze J1, na które wyprowadzono sygnały ISP. Potencjały nieużywanych wyprowadzeń RESET i SCK zostały ustalone zewnętrznymi rezystorami R5 i R6.

Sygnały wyjściowe konwertera, masa oraz zasilanie znajdują się na wyprowadzeniach



Rysunek 3. Wzór ścieżek i schemat montażowy płytki PCB, strona BOTTOM

Listing 1. Kod źródłowy pętli głównej i procedury obsługi przerwania

```

//-----
// Przerwanie od styków enkodera
//-----
ISR(PCINT0_vect){
    //jeżeli oba impulsy są zakończone
    if(outr == 0 && outl == 0){
        if(bit_is_clear(PINB, PINB3) && bit_is_set(PINB, PINB4)){ enkod[0]++; enkod[1] = 0; }
        if(bit_is_set(PINB, PINB3) && bit_is_set(PINB, PINB4)){ enkod[1]++; enkod[0] = 0; }
        if(bit_is_set(PINB, PINB3) && bit_is_clear(PINB, PINB4)){ enkod[0]++; enkod[1] = 0; }
        if(bit_is_clear(PINB, PINB3) && bit_is_clear(PINB, PINB4)){ enkod[1]++; enkod[0] = 0; }
        if(enkod[0] >= 2) {
            outr = PULSE_LEN;
            enkod[0] = 0;
            TCNT0 = 0;
        }
        if(enkod[1] >= 2) {
            outl = PULSE_LEN;
            enkod[1] = 0;
            TCNT0 = 0;
        }
    }
}
//-----
// GŁÓWNA FUNKCJA PROGRAMU
//-----
int main(void){
    //konfiguracja wyprowadzeń
    DDRB = (1 << DDB1) | (1 << DDB0); //wyjścia symulujące przyciski
    OUTL_0;
    OUTR_0; //wyłączenie tranzystorów wyjściowych
    //uruchomienie przerwań od styków enkodera
    GIMSK = (1 << PCIE); //aktywacja systemu przerwań PCINT
    PCMSK = (1 << PCINT4); //przerwanie od pinu PB4

    TCCR0B |= (1 << CS01) | (1 << CS00); //preskaler przez 64

    sei(); //uruchomienie systemu przerwań

    while(1){
        //jeżeli licznik TCNT0 nie jest pusty
        if(TCNT0){
            if(outr > TCNT0){ //dopóki trwa odliczanie
                OUTR_1; //zwrzyj wyjście do masy
            } else {
                //w przeciwnym razie, zatkaj tranzystor wyjściowy
                OUTR_0;
                outr = 0; //i zakończ odliczanie
            }

            if(outl > TCNT0){
                OUTL_1;
            } else {
                OUTL_0;
                outl = 0;
            }
        }
    }
}
}

```

złącza J2. Doprowadzono tam również styk przycisku enkodera, który można dowolnie zagospodarować.

Montaż i uruchomienie

Układ został zmontowany na dwustronnej płytce drukowanej o wymiarach 20×60 mm, której wzór ścieżek i schemat montażowy przedstawiają **rysunki 2 i 3**. W odległości 3 mm od krawędzi znalazły się otwory montażowe. Enkoder położony jest na środku płytki.

Wszystkie elementy montowane powierzchniowo znalazły się po stronie bottom płytki i to od tych elementów należy rozpocząć montaż – **fotografia 4**. Następnie należy włutować złącza typu goldpin, zaś

enkoder na samym końcu, po przeciwnej stronie płytki. Zmontowany układ należy zaprogramować dostarczonym wsadem. Ustawienia fuse-bitów pozostają niezmienione.

Układ należy zasilac napięciem stałym z przedziału 3...5,5 V. Pobór prądu wynosi ok. 1...1,5 mA przy napięciu 5 V.

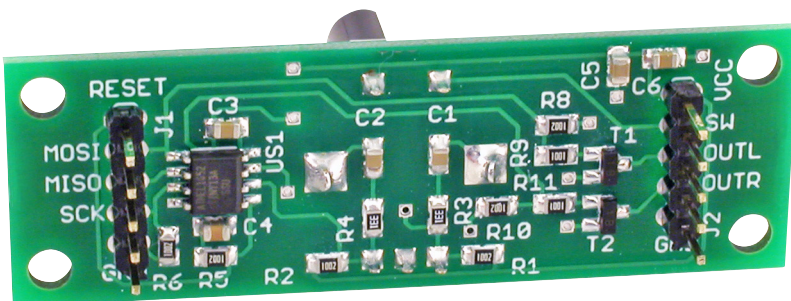
Dla ciekawskich

Kod źródłowy programu został napisany w języku C i jego istotny fragment znajduje się na **listingu 1**. Obsługa enkodera odbywa się w przerwaniu zewnętrznym typu PCINT (Pin Change Interrupt). Aby możliwie zminimalizować liczbę błędnych odczytów stanu enkodera, zaimplementowano prosty podział liczby

odebranych impulsów przez dwa. Jeżeli zostanie uchwycona nagła zmiana kierunku ruchu enkodera, zmienna odliczająca dotychczasową wartość zostaje wyzerowana. Dopiero po prawidłowym odliczeniu dwóch następujących po sobie impulsów układ zezwala na wygenerowanie impulsu wyjściowego.

Do odmierzenia długości impulsów wyjściowych służy licznik Timer0. Jego zawartość jest zerowana w momencie rozpoczęcia się impulsu wyjściowego. Zastosowany preskaler przez 64 oraz odliczanie do uzyskania wartości równej 200 przekłada się na impulsy wyjściowe o długości ok. 10 ms. W sytuacji, kiedy oś enkodera obraca się bardzo szybko, impulsy zlewają się ze sobą, ponieważ kolejny impuls jest generowany tuż po zakończeniu poprzedniego.

Pobór prądu waha się w zależności od tego, w jakim położeniu zatrzymał się enkoder. Jeżeli któryś z jego styków zwiera do masy, to przez rezystor podciągający stale płynie prąd o natężeniu ok. 0,3...0,5 mA (w zależności od napięcia zasilającego). Implementacja trybów oszczędzania energii w mikrokontrolerze nie była zatem celowa, ponieważ jego pobór prądu nie przekracza 1 mA w stanie aktywnym.



Fotografia 4. Widok zmontowanej płytki od strony elementów SMD

Michał Kurzela, EP