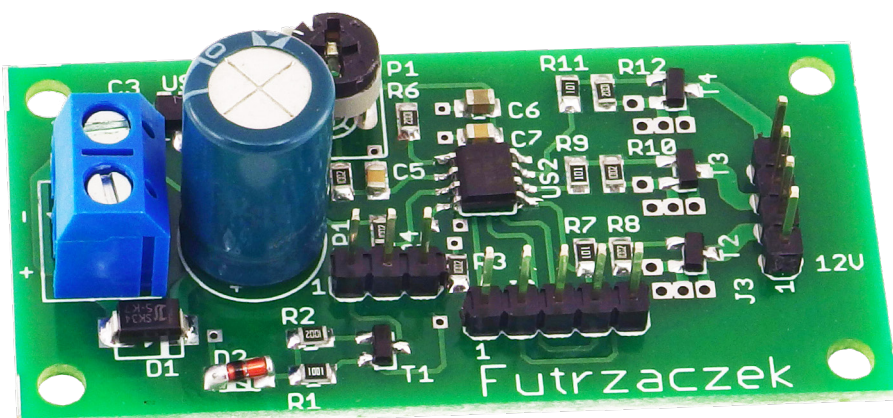


Wolnozmienny sterownik taśmy RGB

Diodowe taśmy RGB dają wiele możliwości, jeżeli chodzi o kreowanie efektów świetlnych. W niektórych przypadkach barwa świecenia powinna zmieniać się płynnie, ale bardzo, bardzo wolno, aby różnica była dostrzegalna po długim czasie. Może to mieć zastosowanie np. w salonie czy ogródku restauracyjnym do tworzenia subtelnego podświetlenia.



Prezentowany układ to regulator jasności trzech kolorów zawartych w taśmie RGB. Kolory zmieniają się samoczynnie i niemalże losowo. Jednak, w przeciwieństwie do innych tego typu układów, tempo zmian może być bardzo wolne, niemal niezauważalne. Dzięki temu obserwatorzy będą mieli wrażenie, że „coś tu się zmieniło”.

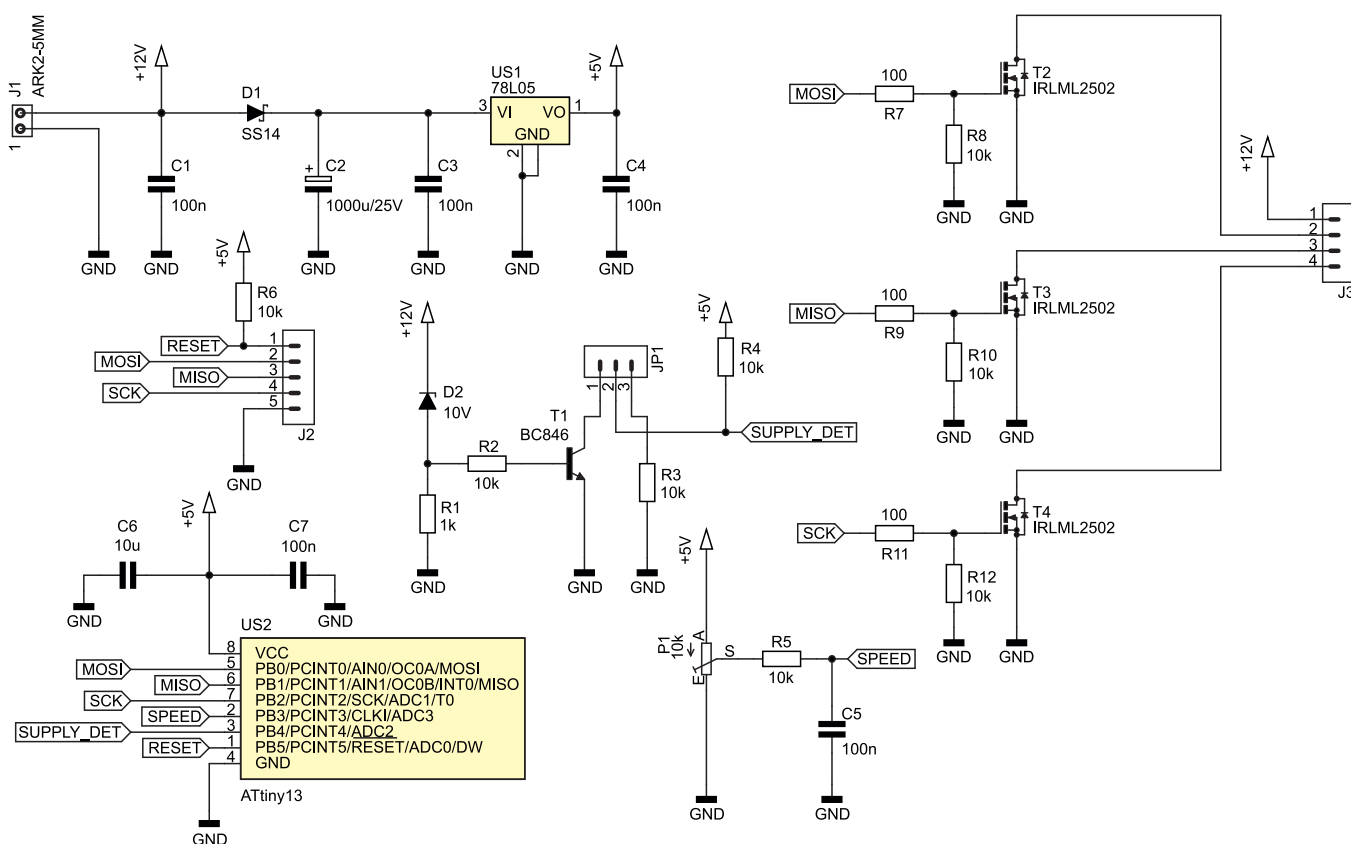
Budowa i działanie

Schemat ideowy został pokazany na **rysunku 1**. Zasilanie układu należy podłączyć do zacisków złącza J1. Stabilizator US1 jest zasilany za pośrednictwem diody D1, aby prąd pobierany przez taśmę nie rozładowywał kondensatora C2, będącego filtrem

zasilania dla stabilizatora. W ten sposób, po wykryciu zaniku napięcia zasilającego, mikrokontroler będzie działał jeszcze przez chwilę, by móc zapisać aktualne wartości składowych kolorów do pamięci nieulotnej EEPROM. Sama detekcja zaniku napięcia została zrealizowana na tranzystorze T1. Kiedy zasilanie jest prawidłowe, dioda Zenera D2 przewodzi, a część jej prądu przepływa przez bazę tranzystora T1. Rezystor R1 ogranicza prąd tej diody do ok. 2 mA. Jeżeli napięcie zasilające układ spadnie poniżej napięcia przewodzenia diody D2, wówczas tranzystor ulega zatkanium, a rezystor R4 podciąga jego kolektor do linii zasilania 5 V. W ten sposób układ uzyskuje dwustanową informację

o obecności zasilania na wejściu J1 – pod warunkiem, że JP1 ma zwarte wyprowadzenia 1 i 2. Jeżeli nie chcemy, aby układ zapamiętywał barwę w momencie wyłączenia go, wystarczy przełożyć zwórkę na wyprowadzenia 2 i 3. Rezystory R3 i R4 stworzą dzielnik napięcia zasilającego w stosunku 1:1, więc na wejście mikrokontrolera zostanie podane napięcie około 2,5 V. Przetwornik analogowo-cyfrowy wbudowany w mikrokontroler zmierzy je i układ podejmie stosowną decyzję.

Szybkość zmian barw jest regulowana potencjometrem P1. Napięcie z jego ślizgacza również jest mierzone przez przetwornik A/C i przeliczane na opóźnienie. Obwód



Rysunek 1. Schemat ideowy układu

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

W ofercie AVT* AVT-5784

Podstawowe parametry:

- sterowanie taśmą LED RGB,
- samoczynna zmiana jasności świecenia trzech barw składowych,
- płynna regulacja tempa zmian,
- możliwość zapamiętania aktualnego koloru w chwili wyłączenia zasilania,
- zasilanie napięciem 12 V.

Wykaz elementów:

- R2..R6, R8, R10, R12: 10 kΩ SMD0805
- R1: 1 kΩ SMD0805
- R7, R9, R11: 100 Ω SMD0805
- P1: 10 kΩ montażowy leżący
- C1, C3..C5, C7: 100 nF SMD0805
- C2: 1000 µF 25 V raster 5 mm
- C6: 10 µF SMD0805
- D1: SS14
- D2: dioda Zenera 10 V MiniMELF
- T1: BC846
- T2..T4: IRLML2502
- US1: 78L05 SOT89
- US2: ATtiny13-20SSU S08
- J1: ARK2/500
- J2: goldpin 5 pin męski 2,54 mm THT
- J3: goldpin 4 pin męski 2,54 mm THT
- JP1: goldpin 3 pin męski 2,54 mm THT + zworka

Projekty pokrewne na www.media.avt.pl:

- AVT-5571 Sterownik taśm LED RGBW zgodny z HomeKit (EP 7/2020)
- AVT-5768 Zasilacz diod power LED 3,5 W (EP 5/2020)
- AVT-5733 Programowany sterownik LED dużej mocy (EP 12/2019)

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

Podstawową wersją 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 wlutować 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] – zmontowany, uruchomiony i przetestowany zestaw
- [B] (elementy wlutowane w płytkę PCB)
- wersja [A] – płytka drukowana bez elementów i dokumentacji
- wersja [A*] – płytka drukowana [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.

R5, C5 tworzy bardzo prosty filtr dolnoprzepustowy, który eliminuje część szumu oraz zakłóceń z uzyskanego napięcia.

Programowanie mikrokontrolera ATtiny13 jest możliwe poprzez złącze J2. Zostały na nie wyprowadzone wszystkie linie niezbędne do wykonania tzw. programowania w systemie (ISP).

Katody diod w taśmie RGB są załączane przez tranzystory MOSFET. Pracują bistabilnie, ponieważ regulacja jasności została zrealizowana przez PWM. Rezystory włączone między bramkę a źródło rozładują istniejącą tam pojemność, chroniąc tranzystor przed uszkodzeniem wywołanym przez nagromadzone ładunki elektrostatyczne. Rezystory szeregowo zwiększają czas przełączania tranzystorów, ograniczając w ten sposób ilość generowanych przez układ zakłóceń. Same tranzystory nie zostały wybrane przypadkowo. Poza tym, że muszą przewodzić możliwie największy prąd i wprowadzać niewiele strat podczas przewodzenia (mała rezystancja otwartego kanału), to jeszcze powinny cechować się jak najniższym napięciem progowym. Dzięki temu jest możliwe ich sterowanie przez mikrokontroler, bez dodatkowych układów pośredniczących.

Listing 1. Fragmenty kodu programu

```
//zmienna inkrementowana przez Timer0, do realizacji programowego PWM-u
volatile uint8_t pwm = 0;
//trzy zmienne do zadawania jasności każdego z kanałów
volatile uint8_t out[3];
//licznik pętli głównej
volatile uint8_t cnt = 0;
//0 - składowa narasta, 1 - opada
volatile uint8_t kierunek[3];

//=====
// Realizacja programowego sterowania PWM na trzech kanałach
ISR(TIM0_OVF_vect){
    TCNT0 = 200;
    pwm++; //każde przerwanie inkrementuje zmienna
    //używaną do porównania z pozostałymi kanałami
    if(pwm >= out[0]){PORTB &= 0b11111110;} else {PORTB |= 0b00000001;}
    if(pwm >= out[1]){PORTB &= 0b11111101;} else {PORTB |= 0b00000010;}
    if(pwm >= out[2]){PORTB &= 0b11111011;} else {PORTB |= 0b00000100;}
}

//=====
//Pętla główna programu
int main(void){
    //PB3 jako wejście potencjometru,
    //PB4 jako wykrycie wyłączenia zasilania,
    //PB0, PB1 i PB2 wyjścia
    DDRB = 0b1100111;
    //wyłączenie wszystkich wyjść
    PORTB = 0b1111000;
    //preskaler przez 1 przy 9,6MHz
    TCCR0B = 0b00000001;
    //da to częstotliwość PWMu równą:
    //9,6MHz / (256*5) = 682Hz
    TCNT0 = 200;
    //przerwanie od przepełnienia
    TIMSK0 = 0b00000010;

    while(1) {
        //instrukcje opóźniające
        cnt++;
        //dzielenie modulo przez różne wartości daje różne szybkości zmian
        if(cnt % 5 == 0){
            //realizacja narastania i opadania wypełnienia
            if(kierunek[0] == 0){out[0]++;}
            if(kierunek[0] == 1){out[0]--;}
            //jeżeli wypełnienie doszło do minimum, przełącz na narastanie
            if(out[0] == 0){kierunek[0] = 0;}
            //jeżeli wypełnienie doszło do maksimum, przełącz na opadanie
            if(out[0] == 255){kierunek[0] = 1;}
        }

        if(cnt % 7 == 0){
            if(kierunek[1] == 0){out[1]++;}
            if(kierunek[1] == 1){out[1]--;}
            if(out[1] == 0){kierunek[1] = 0;}
            if(out[1] == 255){kierunek[1] = 1;}
        }

        if(cnt % 9 == 0){ //ta składowa zmienia się najwolniej
            if(kierunek[2] == 0){out[2]++;}
            if(kierunek[2] == 1){out[2]--;}
            if(out[2] == 0){kierunek[2] = 0;}
            if(out[2] == 255){kierunek[2] = 1;}
        }
    }
}
```

Montaż i uruchomienie

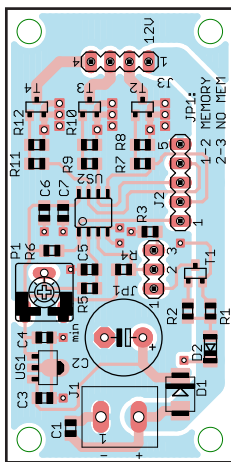
Układ został zmontowany na dwustronnej płytce drukowanej o wymiarach 60×30 mm, a jej schemat pokazuje rysunek 2. W odległości 3 mm od krawędzi płytki znalazły się otwory montażowe.

Znacząca część elementów jest w obudowach do montażu powierzchniowego i to od nich należy rozpocząć montaż. Pozostałe elementy należy wlutować według wysokości ich obudów. Zmontowany i gotowy do działania układ pokazano na fotografii tytułowej. Prawidłowo zmontowany wymaga jedynie zaprogramowania mikrokontrolera, co może odbyć się poprzez złącze J2. W pierwszej kolejności należy zmienić bity zabezpieczające na następujące: High Fuse = 0xFD, Low Fuse = 0x7A. Po tej zmianie można zaprogramować pamięć Flash gotowym wsadem.

Jeżeli programowanie odbyło się bez zakłóceń, to należy wyłączyć zasilanie oraz ustawić zworkę JP1. Zwarte pola 1–2 powodują, że układ zapisuje i odczytuje poziomy

jasności barw składowych. Zwarte pola 2–3 oznaczają, że każde włączenie układu powoduje rozpoczęcie sekwencji barw od nowa. Jeżeli zworki nie będzie, mikrokontroler wstrzyma działanie. Dopiero wyłączenie zasilania, prawidłowe ustawienie zworki i ponowne włączenie układu przywróci jego funkcjonowanie. W przypadku wyboru pierwszej opcji należy zadbać o możliwie szybki zanik napięcia w momencie wyłączenia układu. Może to realizować np. przełącznik lub przekaźnik włączony w szereg z przewodami prowadzącymi do złącza J1. Układ jak najszybciej powinien dostać informację o tym, że napięcie zanika, co jest możliwe tylko poprzez szybkie zatkanie tranzystora T1. Zasilanie dla mikrokontrolera będzie, przez tę krótką chwilę, kiedy kondensator C2 utrzymuje jeszcze napięcie, pochodzące właśnie z niego.

Taśmę LED RGB należy podłączyć do złącza J3. Na płytce zostało zaznaczone wyprowadzenie, na którym jest stałe napięcie



Rysunek 2. Schemat montażowy i wzór ścieżek płytki

12 V. Podłączenie poszczególnych barw składowych nie ma znaczenia, ponieważ i tak

mieszają się one między sobą niemal chaotycznie. Maksymalny prąd, jaki mogą przewodzić tranzystory T2...T4, to teoretycznie 4,2 A. Jednak pracują one impulsowo: częstotliwość sygnału PWM to około 680 Hz. Dlatego, uwzględniając straty wynikające z przełączania, proponowałbym nie obciążać każdego z kanałów prądem większym niż na przykład 2,5 A. Przekłada się to na maksymalny całkowity prąd pobierany przez taśmę równy 7,5 A. Możliwy do podłączenia odcinek taśmy jest zatem bardzo długi. Zakładając pobór mocy 7,2 W/m, istnieje możliwośćysterowania ponad 12 m taśmy. W razie konieczności polecam zastosować wzmacniacz do taśm RGB.

Pobór prądu przez sam sterownik jest niewielki, bo wynosi około 10 mA. Na tle prądu pobieranego przez taśmę można go pominąć podczas doboru odpowiedniego zasilacza.

Dla dociekliwych

Mikrokontroler ATtiny13 zawiera tylko jeden sprzętowy licznik Timer0, który z kolei ma dwa wyjścia PWM. Do sterowania taśmą RGB potrzebne są trzy kanały, więc trzeba było wykonać to programowo. Szczegóły są na **liście 1**, który zawiera najważniejsze fragmenty kodu programu napisanego w języku C. Sygnał PWM jest generowany poprzez cyklicznie wywoływane przerwanie od przepełnienia licznika Timer0, co następuje z częstotliwością ok. 174,5 kHz. Podczas obsługi pojedynczego przerwania inkrementowana jest ośmiobitowa zmienna, z którą porównywane są trzy zmienne ustalające wypełnienie. Zamknięcie cyklu nastąpi po 256 przerwaniach, dlatego wypadkowy sygnał PWM ma częstotliwość ok. 682 Hz. To znacznie więcej niż granica percepcji ludzkiego oka, która wynosi około 20 Hz.

Michał Kurzela, EP