

RPU - Regulator Poziomu Umystu

Niestety, urządzenie to nie podnosi współczynnika IQ ani też nie dodaje mądrości (choć posiada walory edukacyjne ujawniające się podczas jego budowy). Jego zadaniem jest stymulacja mózgu do pracy na danej częstotliwości, czyli wywołania pożądanych fal mózgowych. Ponieważ mózg człowieka pracuje zawsze na jakiejś częstotliwości, można powiedzieć, że umysł jest na jakimś poziomie. Z takim określeniem spotkałem się zarówno w Internecie, jak i w literaturze. Wyróżnia się następujące stany-poziomy pracy umysłu: ALPHA, BETA, THETA, DELTA i GAMMA. Fale odpowiadające za określony poziom można zaobserwować za pomocą aparatu EEG. Wykres przedstawiający zarejestrowane za jego pomocą przebiegi poszczególnych stanów umysłu pokazany jest na **rysunku 1**.

Stan ALPHA, któremu towarzyszą fale o częstotliwości od 7 do 13 Hz, występuje podczas odpoczynku i charakteryzują go spo-

kój, pozytywne myślenie, lepsze przyswajanie informacji oraz lepsza koncentracja. Jest to chyba najbardziej pożądany stan, ponieważ szybciej postępuje nauka (np. języków obcych) i umysł staje się bardziej kreatywny.

Pozostałe stany charakteryzują się następująco:

DELTA (0,5–4 Hz) to stan nieświadomości występujący podczas głębokiego snu;

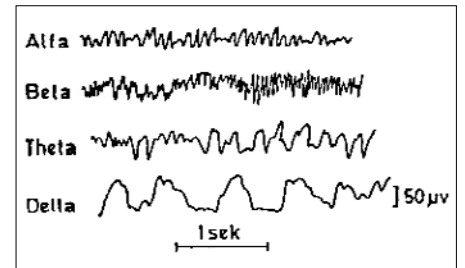
THETA (4–7 Hz) to stan np. podczas płytkiego snu;

BETA (14–30 Hz) stan występujący przy normalnym funkcjonowaniu z pełną świadomością;

GAMMA (30 Hz i wyższ) to stan towarzyszący silnie przeżywanym emocjom.

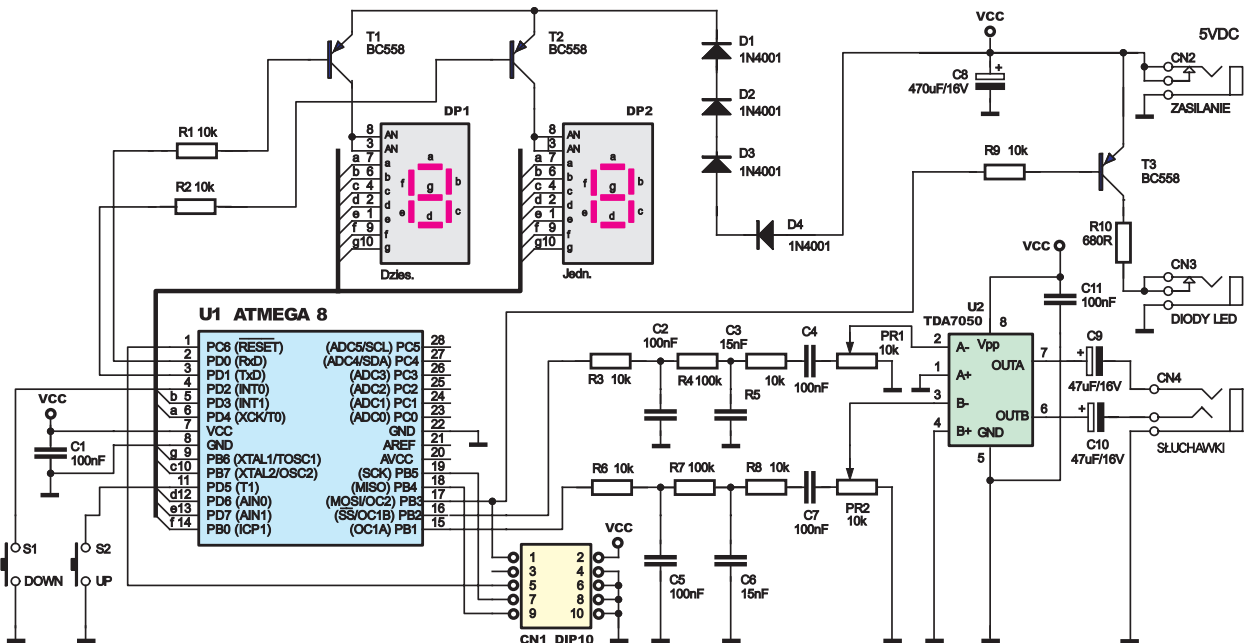
Tego typu urządzenia obecne są w sklepach (głównie na Zachodzie), a ludzie kupują je z nadzieją, że osiągną dzięki nim relaks i odpoczynek. Istnieją również programy komputerowe, za pomocą których można „nastroić

umysł”, takie jak: BrainWave Generator albo CoolEdit, który ma opcję modulowania dźwięków falami o częstotliwościach ALPHA, BETA i pozostałych. Niektórzy są zachwyceni ich działaniem i opisują swoje niesamowite przeżycia podczas użytkowania, a inni mówią, że po ich użyciu kręci im się w głowie i nic więcej. Skuteczność urządzenia



Rys. 1

Rys. 2 Schemat ideowy



proponuję sprawdzić samemu i zachęcam do budowy opisanego dalej układu elektronicznego.

Opis układu

Na wstępie zaznaczam, że stosowanie urządzenia nie wymaga od użytkownika przyklejania sobie elektrod do czoła albo zakładania hełmu na głowę (np. jak miało to miejsce w filmie o przygodach podopiecznego dr. Frankensteina) - wystarczy słuchawki stereofoniczne i dodatkowo okulary wyposażone w diody LED mrugające w kierunku oczu.

Do zrealizowania układu elektronicznego potrzebne są dwa generatory wytwarzające przebieg o częstotliwości akustycznej, najlepiej sinusoidalny. Częstotliwość przebiegu jednego i drugiego generatora powinna się różnić o kilka-kilkanaście Hz, a różnica ta powinna być stała w czasie. Przebieg z pierwszego generatora podajemy na lewą słuchawkę, natomiast przebieg z drugiego na prawą. Słuchawki zakładamy na uszy i co słyszymy? Oczywiście pojedynczy dźwięk w jednym uchu i podobnie w drugim. Ale oba tony zdunniają się w mózgu i dodatkowo słyszymy ton o częstotliwości równej absolutnej różnicy częstotliwości tonu z obydwu słuchawek. I właśnie tak można wywołać określony poziom umysłu. Efekt potęgują migające z częstotliwością różnicową diody LED przystawione blisko zamkniętych oczu.

Żeby jeszcze bardziej uzmysłowić zasadę działania, można zaznaczyć, że stosowanie głośników zamiast słuchawek jest bezcelowe, ponieważ obydwie fale akustyczne pochodzące od membran głośników zdunniają się już w powietrzu i pomimo że słyszymy ton różnicowy, efekt z tego żaden.

Praktyczna realizacja koncepcji widoczna jest na **rysunku 2**. W roli głównej występuje procesor AVR ATMEGA8. Pełni on rolę generatorów przebiegów sinusoidalnych podawanych na słuchawki, generatora przebiegu dla diod LED oraz sterownika wyświetlaczy, na których widoczna jest częstotliwość zdunnień podana w Hz.

Zastosowanie w układzie wydajnego procesora AVR rozwiązało szereg problemów, z których najbardziej istotnym jest trudność w wytworzeniu dwóch przebiegów sinusoidalnych o niewielkiej różnicy częstotliwości i podtrzymanie tej różnicy. Problem utrzymania stabilności różnicy obu częstotliwości odpadł automatycznie, natomiast samo wytworzenie przebiegu bardzo podobnego do sinusoidy jest trochę skomplikowane.

Procesor, który jest napędzany wewnętrznym oscylatorem o częstotliwości 8 MHz, generuje przebieg bardzo zbliżony do sinusoidalnego za pomocą dwóch sprzętowych generatorów PWM (OCR1A i OCR1B). Pracują one w trybie fast-PWM bez preskalera i częstotliwość

przebiegu PWM wynosi 31250Hz. A więc, co bardzo istotne, jest to częstotliwość niesłyszalna dla ucha ludzkiego.

W ten sposób (za pomocą modulacji PWM) teoretycznie można uzyskać w słuchawce czysty ton sinusoidalny, bez stosowania

Listing 1. Obsługa przerwania od przepelnienia T1.

```

/*****
SIGNAL (SIG_OVERFLOW1)
{
    unsigned char x1,x2,x3;//omega t dla fn=1 i dla fn=2
    if (fn==2) { if (LED_i==1) SIN_i++; }
    else SIN_i++;
    //::::::::::::sinus AUDIO //przebieg odniesienia f0
    x1=(1310*SIN_i)>>8; //1310=ftable[0]
    OCR1A=SIN[x1]; //przebieg o czestotliwosci rownej f0+f
    x2=(ftable[f]*SIN_i)>>8;
    OCR1B=SIN[x2];
    //::::::::::::LED
    x3=x2-x1;
    OCR2=x3;
    //::::::::::::wyswietlacz
    PORTD|=0xfb;
    PORTB|=0xc1;
    if (LED_i==1) //jednosci f
    {
        PORTB&=LED_jedn_PB[f_LED];
        PORTD&=LED_jedn_PD[f_LED];
    }
    else //dziesiatki f
    {
        PORTB&=LED_dzies_PB[f_LED];
        PORTD&=LED_dzies_PD[f_LED];
        LED_i=0;
    }
    LED_i++;
};
*****/

```

Listing 2. Kod pętli głównej programu.

```

/*****
for(;;)
{
    //czy wcisnieto 2 przyciski i czestotliwosc==0?
    //jesli tak to nastepuje przełaczenie czestotliwosci nośnej
    if (up()==1 && down()==1 && f==0)
    {
        if (fn==1) fn=2;
        else fn=1;
        do { }
        while (up()==1 || down()==1); //czekaj az zostana puszczone przyciski
        loop(1000); //gaszenie drgan przyciskow
    };
    //zwiększ lub zmniejsz czestotliwosc
    if (up()==1 || down()==1)
    {
        SIN_i=0;
        if (up()==1 && f/fn<max_f) f+=fn; //zwiększanie czestotliwosci
        if (down()==1 && f/fn>min_f) f-=fn; //zmniejszanie f
        if (faza<3) licznik=2500; //opoznienie dla pierwszych 3 przeskocow
        else licznik=500; //opoznienie dla kolejnych przeskocow
        do
        {
            licznik--;
            warunek=up()+down();
        }
        while (!(licznik==0 || warunek==0)); //petelka opozniajaca przeskakiwanie
        //cyferek na wyswietlaczu przy wcisnietym przycisku
        faza++;
        if (faza>4) faza=4;
        loop(100); //gaszenie drgan przyciskow
    }
    else faza=0;
    f_LED=f/fn; //zeby wyswietlacz pokazywal wlasciwa czestotliwosc
};
*****/

```

żadnych filtrów. Niestety tylko w przypadku z góry ustalonej częstotliwości sinusa i to nie dowolnej. Wtedy z każdym wystąpieniem przerwania timera1 wystarczy pobrać ze specjalnie przygotowanej tablicy odpowiednią wartość i wprowadzić ją do rejestru OCR1n.

Konieczne stało się opracowanie algorytmu, który szybko wyznaczy wartość $\sin(x)$ wprowadzaną do OCR1n dla 50 różnych częstotliwości (wartości naturalne od 0 do 49Hz).

Algorytm obliczania wartości współczynnika PWM bazuje na wzorze:

$$y = \sin(\omega t) \quad (1)$$

Ponieważ rozdzielczość przebiegu PWM w trybie fast-PWM wynosi 8 bitów, fala sinusoidalna jest tworzona za pomocą 256 wartości współczynnika wypełnienia. A więc wartość y musi przyjmować wartości całkowite od 0 do 255. Przedstawia to wzór (2) będący modyfikacją wzoru (1):

$$y = \text{round}(128\sin(\omega t)) + 128 \quad (2)$$

Obliczanie funkcji $\sin(x)$ nie wchodzi w grę, ponieważ mamy do dyspozycji dużo mniej niż 256 taktów zegara między kolejnymi wywołaniami przerwań od przepelnienia timera T1, podczas gdy obliczenie funkcji $\sin(x)$ w programie skompilowanym czy to w BASCOMIE, czy za pomocą GCC zajmuje procesorowi ponad 1000 cykli zegara. Oprócz tego konieczna jest eliminacja wszelkich obliczeń, nawet kosztem zajęcia większego obszaru pamięci RAM. W zastosowanym mikrokontrolerze pamięci RAM jest pod dostatkiem (1024B). Można sobie więc pozwolić na stabilizowanie wszystkiego co się da, eliminując takie „cyklochłonne” operacje jak właśnie funkcja $\sin(x)$ albo mnożenie i dzielenie, w szczególności liczb zmiennoprzecinkowych.

We wzorze (2) ω można przedstawić jako iloczyn: nf , gdzie n to liczba kwantów składających się na jeden okres sinusoidy, zaś f to częstotliwość przebiegu. Wartości każdego z n kwantów zostały zapisane w tablicy 256-elementowej, czyli dla $n=256$. Pozostają dwie zmienne: f i t . Wartość $\omega(f)$ jest umieszczona w tablicy 50-elementowej, żeby pozbyc się mnożenia nf . Ostatecznie pozostaje tylko jedno mnożenie, którego już nie da się wyeliminować za pomocą tablicy (musiałaby mieć ponad 30000 elementów) i wzór, którym obliczana jest wartość wpisywana do rejestrów OCRn, ma postać:

$$y = \text{SIN}[F[f] t] \quad (3)$$

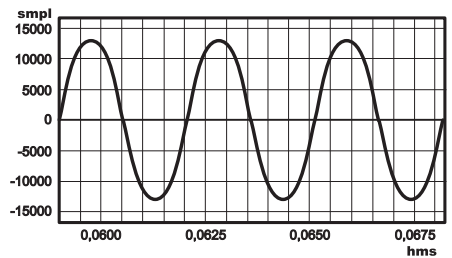
SIN[] – tablica z rozłożonym okresem sinusoidy; F[] – tablica wartości ω dla poszczególnych f .

Wartość y jest więc obliczana dla jedynego argumentu: t . Jest to licznik modulo 256, a więc obliczenia z jego udziałem są wykonywane szybko.

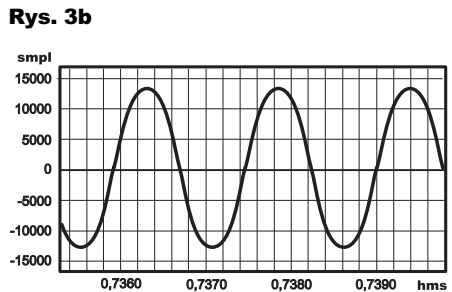
Program skompilowany został za pomocą kompilatora *gcc-3.4.3-2.15-1.2.3*, a napisany w edytorze *AVRSide* przy wspomaganii *AVRStudio 3.56*. Listing części programu realizującej opisany sposób przedstawia listing 1.

Podczas wystąpienia przerwania od przepelnienia timera1 wykonywana jest procedura SIGNAL. Wykonywane są w niej przypisania do rejestrów OCRn, czyli generowanie sygnału dla słuchawek i przy okazji sygnału dla diod LED przystawionych do oczu, oraz sterowanie wyświetlaczami LED. Po wyjściu z procedury przerwania program znajduje się w pętli i oczekuje na naciśnięcie przycisków. Każde naciśnięcie przycisku jest filtrowane w programie zakłóceń poprzez wnoszenie niewielkiego opóźnienia w postaci pętli (funkcja LOOP). Wciśnięcie przycisku powoduje natychmiastowe zwiększenie lub zmniejszenie częstotliwości o 1Hz. Dłuższe przytrzymanie przycisku spowoduje automatyczne „przeskakiwanie” częstotliwości – najpierw wolne, później szybkie. Naciśnięcie dwóch przycisków jednocześnie, gdy na wyświetlaczu jest 00Hz, spowoduje przełączenie częstotliwości nośnej na drugą. Warty zwrócenia uwagi jest sposób zmiany częstotliwości nośnej: dla $f=625\text{Hz}$ obliczenia są wykonywane za każdym wywołaniem przerwania od przepelnienia timera1, natomiast dla częstotliwości 312.5Hz obliczenia wykonywane są w co drugie wywołanie tego przerwania. Kod pętli głównej widoczny jest na listingu 2.

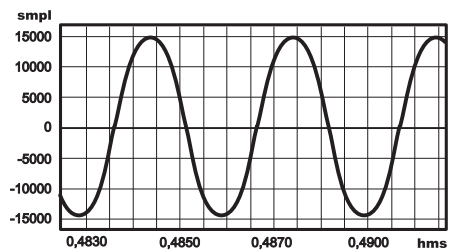
To tyle na temat programu, dalej przeczytasz, co się dzieje z sygnałami opuszczającymi mikrokontroler. Po opuszczeniu wyjść Ocn mikrokontrolera obydwu fal PWM, wędrują one do prostych filtrów dolnoprzepustowych (R3-R5,C2,C3; R6-R8,C5,C6). Mają one za zadanie pozbawić sinusoidę zakłóceń i wszelkiego rodzaju niewielkich, aczkolwiek słyszalnych „poszarpan”, wynikających z niedoskonałości obliczeń i kumulujących się błędów zaokrąglenia. Sygnał na wyjściu i na wyjściu przedstawia rysunek 3. Rysunki 3a i 3b przedstawiają sygnał pobrany prosto z wyjść procesora OC dla obydwu częstotliwości (a – 312,5Hz, b – 625Hz), rys. 3c pokazuje sygnał na wyjściu wzmacniacza. Po opuszczeniu filtrów przebieg jest podawany na wzmacniacz audio U2 TDA7050. Poprzedzają go jeszcze potencjometry regulacji głośności PR1 i PR2. Wzmacniacz ten wybrałem z uwagi na minimalną wymaganą liczbę elementów zewnętrznych oraz niskie napięcie zasilania.



Rys. 3a



Rys. 3b



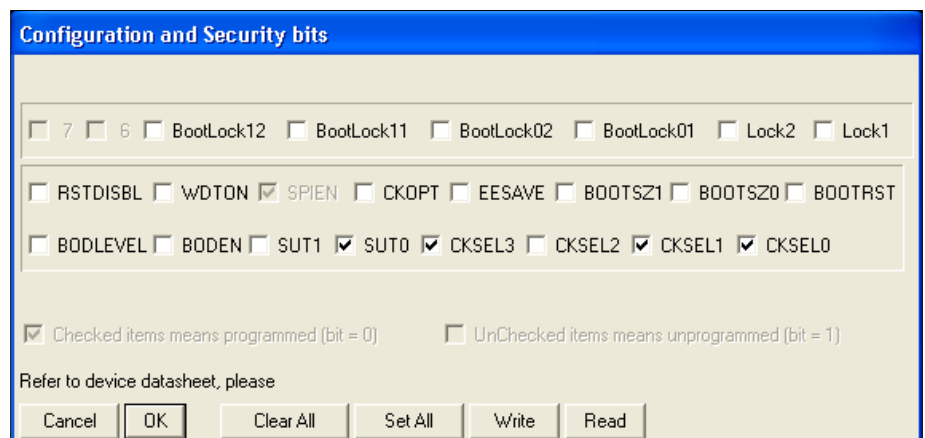
Rys. 3c

Diody stymulujące wzrok zasilane są poprzez tranzystor T3 z wyjścia OC2. Z wyjścia OC2 wychodzi piłokształtny przebieg (w postaci fali PWM) o częstotliwości różnicowej przebiegów w kanałach audio.

Wyświetlacze LED są standardowo sterowane multipleksowo i zasilane poprzez diody D1..D4 obniżające napięcie na wyświetlaczach.

Mikrokontroler zaprogramowałem za pomocą programu *PonyProg2000* programatorem *Avr ISP*. Na rysunku 4 widoczny jest zrzut przedstawiający ustawienie bitów konfiguracyjnych procesora U1.

Rys. 4



Montaż i uruchomienie

Układ można zamontować na płycie drukowanej pokazanej na **rysunku 5**. Jest na niej miejsce na standardowe złącze programatora ISP (oznaczone jako CN1), a więc można szybko zmodyfikować program mikrokontrolera i ewentualnie dostosować do własnych wymagań. Program można ściągnąć z Portalu EdW. Jako regulator głośności w układzie modelowym zastosowałem dwa potencjometry montażowe. Może się wydawać, że jest to rozwiązanie mało wygodne dla użytkownika, ponieważ trzeba ustawiać każdy potencjometr z osobna i to na dodatek mało ergonomicznie, bo z użyciem śrubokręta. Jednak z przeznaczenia urządzenia wynika, że będzie ono używane w spokoju i ciszy na zewnątrz, a więc wystarczy jeden raz ustawić głośność obu kanałów, odpowiednio dostosowaną do wrażliwości słuchu. Kto jednak chce, może bez problemu zastosować potencjometr stereo i zaopatrzyć go w pokrętło. W układzie nie trzeba nic ustawiać i powinien zadziałać po zmontowaniu i podłączeniu zasilania. Napięcie zasilania nie powinno przekraczać 5,5V, natomiast minimalne napięcie zasilania zależy od rodzaju mikrokontrolera U1. Dla ATMEGA8L wystarczy, że będzie ono równe 2,7V. Układ można wtedy zasilać z 2 paluszków 1,5V (ale dla tak niskiego napięcia zasilania należy dobrać odpowiednią liczbę diod połączonych w szereg z anodami wyświetlaczy). Standardowo urządzenie powinno być zasilane ze stabilizowanego zasilacza o napięciu 5V podłączonego do gniazda zasilania CN2.

Słuchawki natomiast podłączamy do gniazda jack stereo 3,5mm oznaczonego na schemacie jako CN4, a diody LED do gniazda CN3 (jack mono 3,5mm).

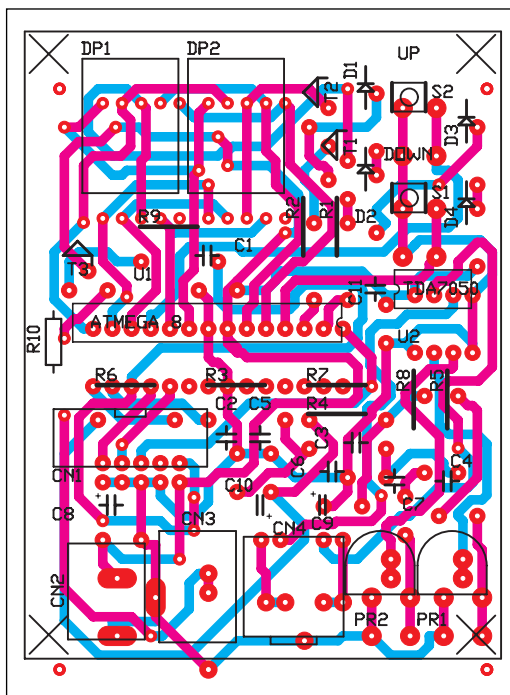
Kwestię obudowy oraz wykonania „okularów”, w których umieszczone będą mrugające LED-y pozostawiam do własnego opracowania Czytelnikom.

Urządzeniem sterujemy za pomocą przycisków S1 i S2. Ponieważ funkcja realizowana przez układ jest prosta, sterowanie nim musi być łatwe – ogranicza się ono do ustawienia częstotliwości dudnień wybranej z przedziału 0-49Hz oraz do przełączania jednej z dwóch częstotliwości tonu słyszalnego w słuchawkach, która może mieć wartość 625Hz i 312,5Hz.

Piotr Wójtowicz

piotr.wojtowicz@edw.com.pl

Rys. 5 Schemat montażowy



Od Redakcji: Układy podobne do opisanego swoją pracą wywołują rozmaite reakcje, poszczególnych osób. W przypadku osób wrażliwych oraz chorych (np. na epilepsję) mogą spowodować niekontrolowane reakcje (np. atak padaczki). Dlatego prezentowany układ należy potraktować jako przykład rozwiązania technicznego problemu generowania dwóch częstotliwości różniących się o stałą wartość, a nie jako przyrząd do ćwiczeń psychotechnicznych.

Wykaz elementów

Rezystory

R1-R3,R5,R6,R8,R9	10kΩ
R4,R7	100kΩ
R10	680Ω
PR1,PR2	10kΩ

Kondensatory

C1,C2,C4,C5,C7,C11	100nF
C3,C6	15nF
C8	470μF/16V
C9,C10	47μF/16V

Półprzewodniki

D1..D4	1N4001
DP1,DP2	wyświetlacz LED wspólna ANODA
T1-T3	BC558
U1	ATMEGA8(L) zaprogramowany
U2	TDA7050

Inne

S1,S2	mikroswitch
CN2	gniazdo zasilania
CN3	jack mono 3,5 druk
CN4	jack stereo 3,5 druk podstawki 28-DIP wąska i 8-DIP

Komplet podzespołów z płytka jest dostępny w sieci handlowej AVT jako kit szkolny AVT-2766