

Kolorofon z Wi-Fi

Trudno dzisiaj wyobrazić sobie prywatkę bez dobrej muzyki i iluminofonii. Dawniej te urządzenia były zarówno trudno dostępne, jak i kosztowne. Wykonywano je z wykorzystaniem aktywnych filtrów LC lub RC sterujących obwodami wykonawczymi na bazie tyrystora lub triaka. Współczesna elektronika w dużej mierze opiera się na zastosowaniu układów cyfrowych – taki też jest opisywany kolorofon. Ma przy tym niespotykaną cechę: łączność pomiędzy elementami zestawu odbywa się za pomocą Wi-Fi.

Rekomendacje: kolorofon nie tylko uatrakcyjni prywatkę i rozświetli wnętrze w rytm muzyki, ale może być też bazą do wykonania własnego systemu oświetlenia sceny lub wnętrza klubu.

Obecnie nie ma problemu z zakupem tanich, efektowych, świecących „błyskotek” – od obrotowych, kolorowych żarówek LED, świecących kul, węży świetlnych, aż po efekty laserowe. Mają one jednak jedną cechę wspólną – zazwyczaj elementy emitujące światło są umieszczone w jednej obudowie, blisko siebie i aby stworzyć scenę świetlną, trzeba zastosować kilka takich urządzeń.

Opisywany kolorofon jest pozbawiony tej wady, ponieważ rozdzielono funkcję analizy spektralnej dźwięku od modułów wykonawczych w formie reflektorów LED. Są to osobne urządzenia, natomiast medium transmisyjnym jest sieć Wi-Fi. Pozwala to na niemal dowolne rozmieszczenie reflektorów z diodami LED w obrębie zasięgu sieci Wi-Fi, a więc na całkiem sporym obszarze.

Jak wspomniano we wstępie, starsze kolorofony wykorzystywały filtry analogowe i zwykle używano w nich jedynie 3 kolorowych żarówek umieszczonych w jednej obudowie. Obecnie wydajność mikrokontrolerów pozwala na zaimplementowanie filtra w formie oprogramowania i ograniczenie toru analogowego wzmacniacza mikrofonowego.

Zwykle do realizacji filtra cyfrowego wykorzystuje się szybką transformację Fouriera. Na rysunku 1 zilustrowano sygnał

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

W ofercie AVT* AVT-5662

Podstawowe parametry:

- Nie ma konieczności galwanicznego połączenia ze źródłem dźwięku. Sygnał wejściowy audio doprowadzany za pomocą mikrofonu.
- Wzmacniacz wejściowy wyposażony w ARW.
- Filtr cyfrowy, zrealizowany na procesorze nadajnika.
- Nadajnik na bazie modułu WROOM-ESP32.
- Odbiornik na bazie modułu WROOM-02.
- Połączenie za pomocą USB.
- Zasilanie przez gniazdo micro USB, co umożliwia użycie power banku lub zasilacza wtyczkowego.

Projekty pokrewne na www.media.avt.pl:

- AVT-5621 Sterownik różnicowy kolektora słonecznego (EP 3/2018)
- AVT-5618 Sterownik bojlera do instalacji PV (EP 2/2018)
- AVT-5620 Wielozadaniowy termostat (EP 1/2018)
- AVT-5589 4-kanałowy termostat z alarmem (EP 6/2017)
- AVT-5354 Zaawansowany, funkcjonalny termostat (EP 11/2016)
- AVT-1908 Termostat 4-kanałowy (EP 5/2016)
- AVT-1878 Prosty termostat cyfrowy (EP 8/2015)
- AVT-3131 Uniwersalny termostat (Edw 6/2015)

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 wlutować w dołączoną płytkę drukowaną (PCB). Wykaz elementów znajduje się w dokumentacji, która jest podlinkowana w opisie kitu.

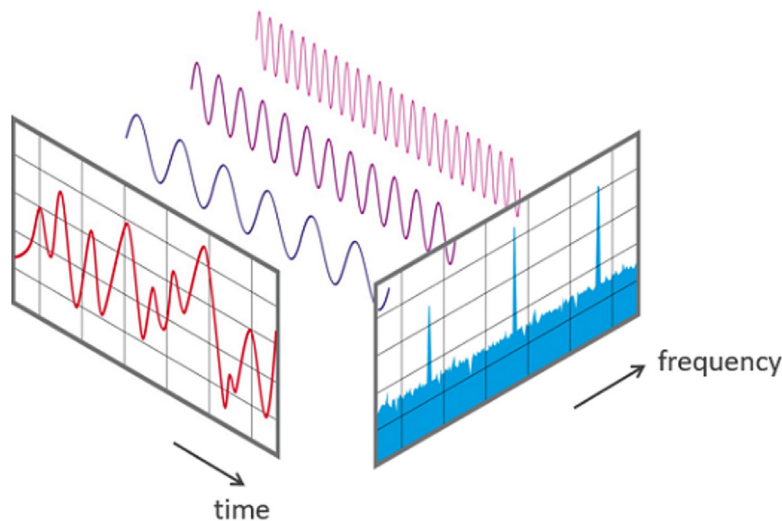
Mając na uwadze różne potrzeby naszych klientów, oferujemy dodatkowe wersje:

- wersja [C] – zmontowany, uruchomiony i przetestowany zestaw [B] (elementy wlutowane w płytkę PCB)
- wersja [A] – płytkę drukowaną bez elementów i dokumentacji
- wersja [UK] – 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.

Redakcja „Elektroniki Praktycznej” dziękuje firmie Soyter Components za dostarczenie zestawów ewaluacyjnych oraz modułów ESP32 i ESP8266 będących bazą urządzeń opisywanych w tym artykule.

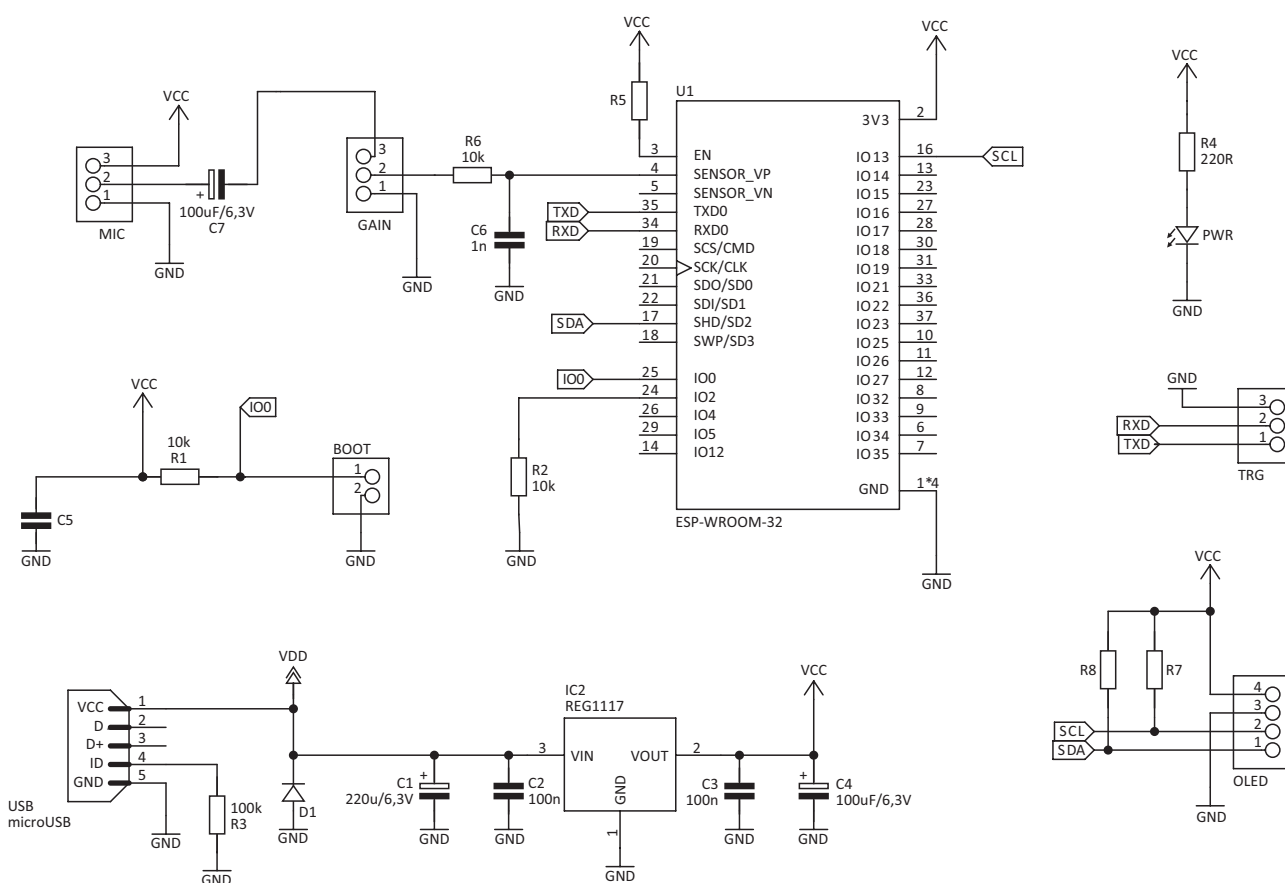




Rysunek 1. Widok sygnału w dziedzinie czasu i częstotliwości (źródło: <http://bit.ly/2CsuML0>)

w nim układ ESP32 wyposażony w bardzo wydajny, dwurdzeniowy, 32-bitowy mikrokontroler. Oba rdzenie mikrokontrolera mogą pracować z częstotliwością taktowania od 80 MHz do 240 MHz. Układ wyposażono w 520 kB pamięci SRAM i 16 MB pamięci Flash. Oprócz obsługi Wi-Fi, ESP32 obsługuje także protokół Bluetooth 4.2, ma interfejsy SPI, I²C oraz szereg bloków funkcjonalnych typu PWM, czujnik temperatury, interfejs kart pamięci i inne. Szczegóły można odnaleźć w dokumentacji układu.

Co być może najważniejsze, aby skorzystać z możliwości zawartych w tym zaawansowanym układzie, nie trzeba być bardzo doświadczonym programistą. Popularne środowisko Arduino od jakiegoś czasu wspiera programowanie ESP32, dzięki czemu nawet



Rysunek 2. Schemat ideowy nadajnika kolorofonu z interfejsem Wi-Fi

w dziedzinie czasu, który jest widoczny na ekranie oscyloskopu jako pojedynczy przebieg. Jak widać, nie jesteśmy w stanie jednoznacznie stwierdzić, z ilu sygnałów i o jakiej częstotliwości składowych się składa ani jaką mają fazę i amplitudę. Jednak po przejściu do dziedzinie częstotliwości zostaną wyodrębnione różne sygnały składowe (na tym rysunku pokazano 3 przykładowe) o różnicowanej amplitudzie. Tak więc można powiedzieć, że w rezultacie wykonania dyskretniej transformacji Fouriera na próbkach sygnału wejściowego otrzymujemy ciąg harmonicznych, z których jest złożony ten sygnał. Lub innymi słowami, transformacja Fouriera

rozkłada funkcję okresową na szereg funkcji okresowych tak, że uzyskana transformata podaje, w jaki sposób poszczególne częstotliwości składają się na funkcję pierwotną. I właśnie tej funkcjonalności użyto w budowie opisywanego kolorofonu.

W opisywanym urządzeniu transformację Fouriera wykonuje moduł nadajnika, natomiast funkcjonalność modułów odbiorczych jest ograniczona do źródła światła sterowanego za pomocą Wi-Fi.

Budowa nadajnika

Schemat ideowy modułu nadajnika kolorofonu pokazano na **rysunku 2**. Zastosowano

REKLAMA

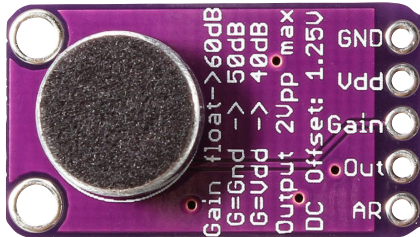
Specjalistyczne szkolenia dla elektroników i automatyków



TECHDAYS

techdays@techdays.pl
TECHDAYS.PL

CERTYFIKOWANY PARTNER SZKOLENIOWY
life.augmented



Fotografia 3. Moduł z gotowym układem wzmacniacza mikrofonowego oparty na układzie MAX9814

początkujący adepci mogą łatwo zaprogramować wspomniany układ przy użyciu sepek darmowych, łatwo dostępnych bibliotek i przykładów programowania.

Opisywany moduł nadajnika kolorofonu zawiera elementy niezbędne do zasilania układu ESP32 oraz jego zaprogramowania. Zasilacz wykonano w oparciu na stabilizatorze napięcia typu LM1117. Ponieważ założeniem projektu była możliwość przenoszenia urządzenia w różne miejsca, całość jest zasilana za pomocą gniazda micro USB. Dzięki takiemu rozwiązaniu źródłem zasilania kolorofonu może być ładowarka do telefonu lub power bank. Ten ostatni dzięki dużej pojemności i niewielkiemu poborowi prądu przez urządzenie zapewni wielogodzinne działanie urządzenia w miejscach, gdzie nie dysponujemy zasilaniem sieciowym.

Uważny czytelnik zauważy brak wzmacniacza mikrofonowego. Ze względu na chęć

Wykaz elementów:

Nadajnik

Rezystory: (SMD 0805)

R1, R2, R3, R5: 10 kΩ
R4: 220 Ω
R6, R7*: 4,7 kΩ

Kondensatory:

C1: 220 μF/6,3 V (SMD)
C2, C3, C5: 100 nF (SMD 0805)
C4, C7: 100 μF/6,3 V (SMD)
C6: 1 nF (SMD 0805)

Półprzewodniki:

U1: WROOM32
U2: LM1117 3,3 V
D1: SS12 SMA (D0-214AC) lub podobna
D2: LED SMD 0805 np. zielona

Inne:

Goldpin
Gniazdo micro USB 10118193-0001LF/C
Obudowa KM-32B
Mikrofon ze wzmacniaczem

Odbiornik

Rezystory: (SMD 0805)

R1, R2, R5: 10 kΩ
R4: 220 Ω
R6, R9, R11, R13, R15, R17, R19, R21: 22 Ω

R7, R10, R12, R14, R16, R18, R20: 100 kΩ
R8, R23, R29: 4,7...5 Ω/1 W

Kondensatory:

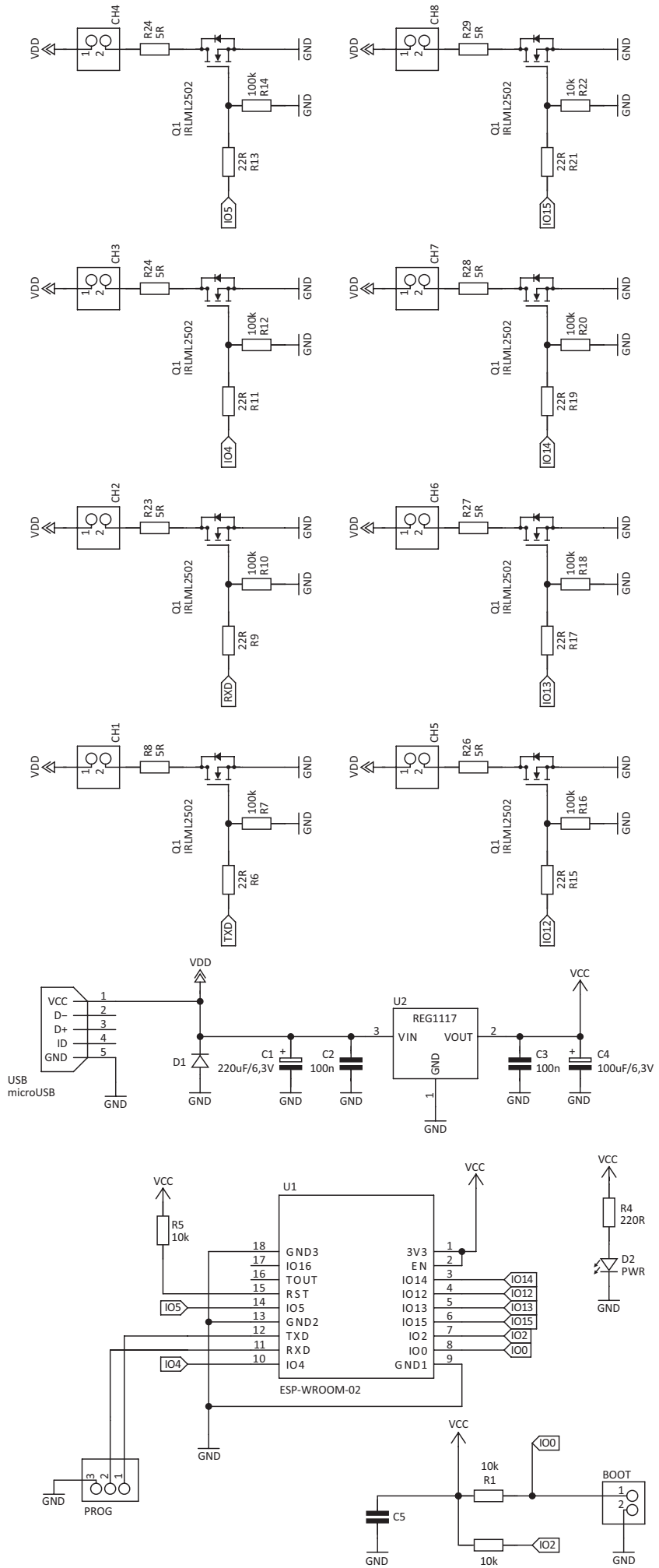
C1: 220 μF/6,3 V (SMD)
C2, C3, C5: 100 nF (SMD 0805)
C4: 100 μF/6,3 V (SMD)

Półprzewodniki:

U1: WROOM-02
U2: LM1117-3,3 V
D1: SS12 SMA (D0-214AC) lub podobna
D2: LED SMD 0805 np. zielona
Diody LED o mocy 1 W

Inne:

Gniazdo micro USB 10118193-0001LF/C
Goldpin
Obudowa KM-32B
Kolimator dla diod LED

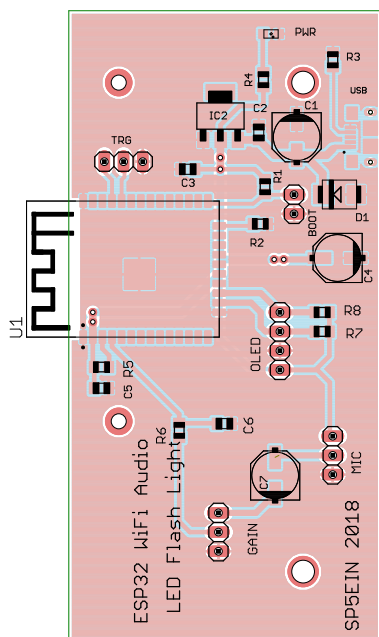


Rysunek 4. Schemat ideowy odbiornika kolorofonu

ograniczenia kosztów budowy kolorofonu zdecydowano się na zastosowanie gotowego modułu wzmacniacza mikrofonowego. Okazuje się bowiem, że koszt elementów do budowy wzmacniacza zasilanego z napięcia 3,3 V jest wyższy niż zakup gotowego modułu... W prototypie zastosowano moduł z mikrofonem i układem wzmacniacza MAX9814 pokazany na **fotografii 3**. Moduł może być zasilany napięciem z zakresu 2,7...5,5 V, ma mały poziom szumów własnych, niewielkie zniekształcenia nieliniowe, wbudowany układ automatycznej regulacji wzmocnienia oraz wybierane wzmocnienie 40, 50 lub 60 dB.

W prototypie ustalono wzmocnienie wynoszące 40 dB, co okazało się optymalne w tym zastosowaniu. Układ reaguje zarówno na cichą mowę, jak i głośną muzykę w zakresie działania ARW. W tym celu połączono doprowadzenie GAIN na płytce wzmacniacza z dodatnim biegunem zasilania – zgodnie z opisem na płytce drukowanej modułu. Oczywiście czytelnik może zastosować dowolny układ wzmacniacza mikrofonowego, co daje spore możliwości eksperymentowania. Opcjonalnie można również zamontować potencjometr w roli tłumika sygnału ze wzmacniacza mikrofonowego. Należy go dołączyć do pinów GAIN. Decyzja o jego montażu zależy od wielkości sygnału wyjściowego ze wzmacniacza i należy do czytelnika. W wypadku, gdy nie montujemy potencjometru tłumika, należy połączyć zworą piny 2 i 3 złącza GAIN.

Jako filtr antyaliasingowy zastosowano filtr dolnoprzepustowy RC o górnej częstotliwości około 16 kHz. Na płytce zamontowano również złącze o nazwie OLED. Umożliwia ono wyposażenie nadajnika w wyświetlacz OLED 1,3", na którym można pokazywać



Rysunek 5. Schemat montażowy nadajnika kolorofonu

widmo sygnału. W opisywanym prototypie nie jest ono używane, ale daje możliwość indywidualnej rozbudowy przez czytelnika.

Budowa odbiornika

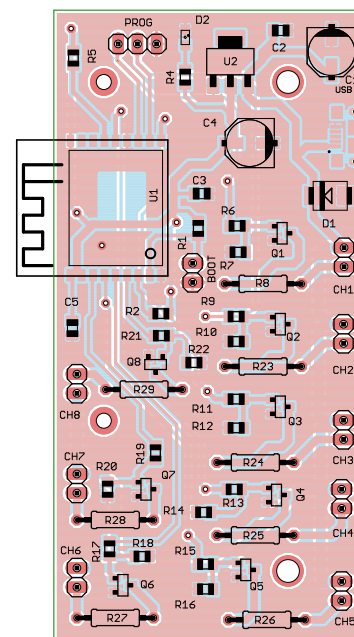
Schemat ideowy modułu odbiornika zamieszczono na **rysunku 4**. Sercem modułu jest układ ESP8266 w wersji WROOM-02. O ile w nadajniku był potrzebny wydajny, szybki mikrokontroler do wykonywania akwizycji dźwięku i transformacji FFT, o tyle w odbiorniku jego funkcja sprowadza się do odbioru danych w postaci pakietów UDP i wygenerowania na ośmiu wyjściach przebiegów PWM służących doysterowania LED.

Układ ESP8266 jest zasilany w taki sam sposób, jak nadajnik. Oprócz zasilacza mamy złącza do wprowadzenia układu w tryb programowania oraz samego programowania. Głównie miejsce zajmuje osiem kanałów PWM do sterowania diodami LED mocy. Jako elementy wykonawcze zastosowano tranzystory MOSFET IRLML2502 o małym napięciu otwarcia drenu i dużej obciążalności prądowej. Siedem kanałów jest identycznych pod względem wartości elementów – ponumerowano je od 1 do 7. W kanale 8 zmniejszono rezystancję opornika R22, co jest związane z funkcjami alternatywnymi wyjścia IO15 układu ESP.

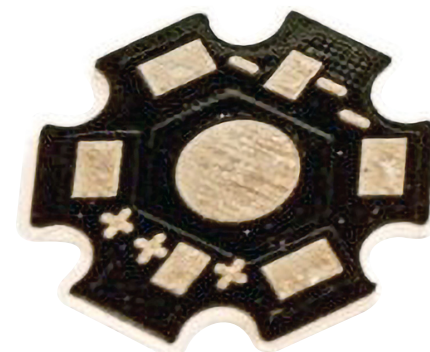
Montaż

Schemat montażowy nadajnika pokazano na **rysunku 5**. Montaż należy rozpocząć od przylutowania elementów zasilacza. Nieco trudności może przysporzyć przylutowanie gniazda micro USB ze względu na mały raster wyprowadzeń. Używać należy cienkiego grotu, a lutowanie przeprowadzić pod szkłem powiększającym. Jeśli wszystko przebiegło prawidłowo, to po dołączeniu zasilacza zaświeci się dioda LED PWR (D2), a napięcie zasilające zmierzone na odpowiednich wyprowadzeniach mikrokontrolera ESP32 będzie wynosiło 3,3 V. W kolejnym kroku należy przylutować układ ESP32. Do jego zaprogramowania potrzebujemy jedynie przejściówki USB-UART w standardzie 3,3 V. Uruchomienie modułu z zapiętą zworą na złączu BOOT wprowadza układ w tryb programowania, o czym w dalszej części artykułu.

Schemat montażowy odbiornika pokazano na **rysunku 6**. Płytkę drukowaną pasuje do obudowy KM-32B, w której na płycie czołowej mieszczą się dwie soczewki z diodami. Oczywiście, nie trzeba montować wszystkich 8 kanałów. W miarę potrzeb można zamontować jeden lub kilka kanałów – w zależności od tego, ile planujemy diod w danej obudowie. Nie należy również zapomnieć o odpowiednim chłodzeniu diod mocy. W tym celu trzeba zaopatrzyć je w radiatory, na przykład takie, jak pokazano na **fotografii 7**.



Rysunek 6. Schemat montażowy odbiornika kolorofonu



Fotografia 7. Radiator służący do chłodzenia LED

Przy montażu i uruchomieniu obowiązują te same zasady, co w wypadku nadajnika. W prototypie zastosowano diody LED o mocy 1 W oraz kolimatory (soczewki) o kącie rozżarcia wiązki 90°. Oczywiście, czytelnik może wykorzystać diody i soczewki wg własnej potrzeby, należy jednak pamiętać, że zazwyczaj maksymalna wydajność prądowa zasilaczy/ladowarek telefonicznych jest rzędu 2 A. Ogranicza to maksymalną

REKLAMA

Specjalistyczne szkolenia dla elektroników i automatyków

STM32

TECHDAYS

techdays@techdays.pl
TECHDAYS.PL

CERTYFIKOWANY PARTNER SZKOLENIOWY

Listing 1. Główna pętla programu nadajnika

```

void loop()
{
  for (int i = 0; i < SAMPLES; i++)
  {
    newTime = micros();
    // Using Arduino ADC nomenclature. A conversion takes about 1uS on an ESP32
    vReal[i] = analogRead(A0);
    vImag[i] = 0;
    while ((micros() - newTime) < sampling_period_us);
  }
  FFT.Windowing(vReal, SAMPLES, FFT_WIN_TYP_HAMMING, FFT_FORWARD);
  FFT.Compute(vReal, vImag, SAMPLES, FFT_FORWARD);
  FFT.ComplexToMagnitude(vReal, vImag, SAMPLES);
  for (int i = 2; i < 512; i++)
  {
    if (vReal[i] > 2000) // regulacja poziomu wyzwalania diod LED
    {
      if (i > 2 && i <= 5) bufor[0] = (int)vReal[i]; // 180Hz (120-200)
      if (i > 7 && i <= 8) bufor[1] = (int)vReal[i]; // 3000Hz (240-360)
      if (i > 9 && i <= 12) bufor[2] = (int)vReal[i]; // 500Hz (400-720)
      if (i > 20 && i <= 30) bufor[3] = (int)vReal[i]; // 1000Hz (800-1200)
      if (i > 45 && i <= 55) bufor[4] = (int)vReal[i]; // 2000Hz (1800-2200)
      if (i > 95 && i <= 110) bufor[5] = (int)vReal[i]; // 4000Hz (3800-4400)
      if (i > 180 && i <= 210) bufor[6] = (int)vReal[i]; // 8000Hz (7200-8400)
      if (i > 250 && i <= 325) bufor[7] = (int)vReal[i]; // 12000Hz (10k-13k)
    }
  }
  udp.beginPacket(udpAddress, udpPort);
  udp.print(String(bufor[0]>>6)+"."+String(bufor[1]>>6)+"."+String(bufor[2]>>6)+"."+
    String(bufor[3]>>6)+"."+String(bufor[4]>>6)+"."+String(bufor[5]>>6)+"."+
    String(bufor[6]>>6)+"."+String(bufor[7]>>6));
  udp.endPacket();
  for(int x=0;x<8;x++) bufor[x]=0; // czyszczenie bufora nadawczego
}

```

Listing 2. Główna pętla programu odbiornika

```

void loop()
{
  int packetSize = Udp.parsePacket();
  if (packetSize)
  {
    // zapisz odebrane dane z pakietu UDP do bufora
    int len = Udp.read(packetBuffer, 255);
    if (len > 0) packetBuffer[len] = 0;
    uint16_t val[8];
    // odseparuj dane dla poszczególnych prążków
    val[0] = atoi(strtok(packetBuffer, " "));
    val[1] = atoi(strtok(NULL, " "));
    val[2] = atoi(strtok(NULL, " "));
    val[3] = atoi(strtok(NULL, " "));
    val[4] = atoi(strtok(NULL, " "));
    val[5] = atoi(strtok(NULL, " "));
    val[6] = atoi(strtok(NULL, " "));
    val[7] = atoi(strtok(NULL, " "));
    // wysterowanie diod LED odpowiednim poziomem PWM
    analogWrite(LED_1, val[0]);
    analogWrite(LED_2, val[1]);
    analogWrite(LED_3, val[2]);
    analogWrite(LED_4, val[3]);
    analogWrite(LED_5, val[4]);
    analogWrite(LED_6, val[5]);
    analogWrite(LED_7, val[6]);
    analogWrite(LED_8, val[7]);
  }
}

```

wydajność zastosowanych zestawów LED-ów do około 10 W.

Oprogramowanie

Programy sterujące pracą nadajnika i odbiornika kolorofonu napisano za pomocą środowiska Arduino. Dzięki użyciu zaimplementowanych w nim bibliotek dość skomplikowana realizacja algorytmu obliczania FFT sprowadza się do określenia liczby próbek, które chcemy poddać transformacji oraz częstotliwości próbkowania. Wynikiem działania biblioteki są dwie tablice z parami danych: częścią rzeczywistą i urojoną obliczonej liczby zespolonej. Otrzymane liczby zespolone opisują parametry badanego sygnału: moduł liczby informuje o amplitudzie prążka. Po szczegóły odnośnie do działania biblioteki FFT odsyłam do dokumentacji Arduino.

Na listingu 1 zamieszczono pętlę główną programu nadajnika, odpowiedzialną za pobranie próbek sygnału, przetworzenie i wysłanie po UDP w ramce rozgłoszeniowej, tak

aby informacja dotarła do wszystkich odbiorników zalogowanych do punktu AP nadajnika. Sygnał wejściowy jest próbkowany z częstotliwością 40 kHz, więc górna granica częstotliwości badanego sygnału wynosi 20 kHz. W tablicy *vReal* mamy 512 punktów określających sygnał w dziedzinie częstotliwości (element zerowy tablicy zawiera informację o składowej stałej), więc szerokość pasma każdej próbki to około 40 Hz. Mając amplitudę każdego prążka, w zależności od interesujących nas zakresów częstotliwości, możemy je „wizualizować”. Służy do tego pętla `for (int x = 2; x < 512; x++) {...}`. Chcąc zmienić zakres częstotliwości, wystarczy edytować warunek wewnątrz pętli. Po wykonaniu pętli w tablicy *bufor[0-7]* są dostępne informacje o amplitudach interesujących nas składowych sygnału.

Zmieniając wartość porównania `if (vReal[i] > 2000)`, zmieniamy próg czułości na dźwięki. Teraz wystarczy przekształcić wartość na zmienną typu *string* i wysłać w pakiecie UDP. Służy do tego instrukcja

```

udp.print(String(bufor[0]>>6)+"."+
String(bufor[1]>>6)+"."+String(
bufor[2]>>6)+"."+String(bufor[3]
>>6)+"."+String(bufor[4]>>6)+"."+
String(bufor[5]>>6)+"."+String(
bufor[6]>>6)+"."+String(bufor[7]>>
6));

```

Pętlę główną programu odbiornika pokazano w listingu 2. Ponieważ po stronie odbiorczej do sterowania diodami LED wykorzystujemy 10-bitowy generator PWM, jednocześnie konwertujemy „w locie” zakres zmiennej *bufor[x]* z 16 bitów do 10, dodając znak dwukropka pomiędzy zmiennymi w celu ich odseparowania dla łatwej analizy.

Po stronie odbiorczej moduł pracuje w trybie nasłuchu pakietów UDP. Gdy pojawi się pakiet, zostaje poddany analizie, po której otrzymujemy stabilizowane wartości poszczególnych prążków. Dane są przekazywane bezpośrednio na wyjścia GPIO, a diody LED są sterowane za pomocą PWM. Wykorzystujemy wszystkie wyjścia do sterowania PWM, więc trzeba zmienić funkcję pinów RX i TX układu ESP8266 na wyjściowe. Można to zrobić instrukcją `pinMode(nr_pinu, FUNCTION_3)`. Nadal jednak można programować układ, ponieważ wejście w tryb bootloadera powoduje zmianę trybu pracy GPIO1 i GPIO3 na TX/RX.

Programowanie układów ESP32 i WROOM-02 (ESP8266) odbywa się z poziomu środowiska Arduino, które trzeba wcześniej odpowiednio skonfigurować. W Internecie można znaleźć wiele poradników, jak to zrobić, więc nie będę tego opisywał. W materiałach dodatkowych dostępne są również źródła obu wersji programów do pobrania.

Aby wejść w tryb programowania, należy założyć zworeczkę na piny BOOT i włączyć zasilanie modułu. W ten sposób uruchamiamy wewnętrzny bootloader, dzięki któremu programujemy układ. Po zaprogramowaniu układy są gotowe do pracy. W pierwszej kolejności włączamy nadajnik i czekamy kilka sekund, aby moduł uruchomił punkt AP dla odbiorników. Po chwili możemy włączać odbiorniki – same łączą się do sieci Wi-Fi utworzonej przez nadajnik. Dla ciekawskich – można połączyć się telefonem z siecią nadajnika i za pomocą programu do analizowania pakietów UDP „podслушать” dane analizowanego dźwięku.

Grzegorz Burzyński
sp5ein@gmail.com

Bibliografia:

1. <http://bit.ly/2ROBV2i>
2. <http://bit.ly/2RyOYFs>
3. <http://bit.ly/2ARGeuT>
4. <http://bit.ly/2Ctixhm>