

# Stroik gitarowy

Przełóżając strony internetowe, natknąłem się na bardzo ciekawy, nietuzinkowy projekt stroika gitarowego, który w odróżnieniu od pozostałych urządzeń tego typu wykorzystuje inne zjawiska fizyczne niż fala akustyczna w celu umożliwienia nastrojenia instrumentu. Typowe rozwiązania dostępne w handlu poddają analizie dźwięk lub drgania emitowane przez strunę gitary i wyświetlają procent odstrojenia instrumentu od wartości prawidłowej. Opisywane urządzenie nie wykonuje analizy dźwięku, a używa efektu stroboskopowego – emituje wiązkę światła LED o określonej częstotliwości migotania, oświetlając nią strojoną strunę.

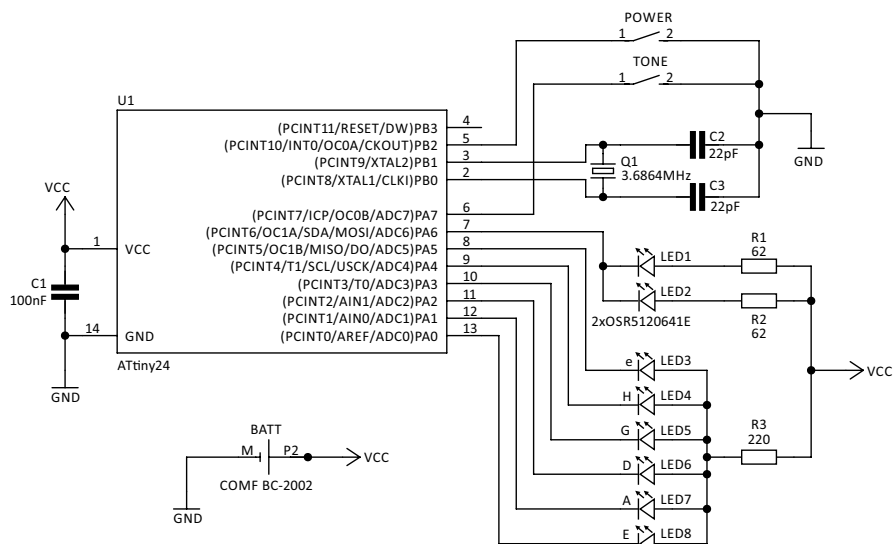
**Rekomendacje:** stroik może przydać się nie tylko muzykom.

Jak wspomniano we wstępie, układ stroika oświetla strojoną strunę za pomocą światła o określonej częstotliwości migotania. Jeśli obserwowana struna drga z częstotliwością inną niż emitowana przez LED, będzie wydawało się, że porusza się w górę i w dół wolniej lub szybciej, zależnie od stopnia odstrojenia. W wypadku nastrojenia struny do częstotliwości migania diod LED struna ta będzie wydawała się nieruchoma. W ten sposób możemy zrealizować układ oryginalnego stroika gitarowego, który z powodzeniem zastąpi rozwiązanie tradycyjne.

Zanim jednak przejdziemy do opisu urządzenia konieczne wydaje się przyswojenie podstawowej wiedzy na temat częstotliwości dźwięków muzycznych. W powszechnie używanym tzw. systemie równomiernie temperowanym stosunek częstotliwości dwóch kolejnych dźwięków jest stały i wynosi  $\sqrt[12]{2}$ , gdyż ten system zakłada podział oktawy na 12 równych części (półtonów). I tak, dla przykładowej skali C kolejne dźwięki mają następujące nazwy: C, Cis, D, Dis,

E, F, Fis, G, Gis, A, Ais (inaczej B), H, po czym po dźwięku H występuje dźwięk o oktawę wyższy w stosunku do C, czyli dźwięk o 2-krotnie większej częstotliwości

(nazwijmy go C2). Wspomniana temperacja polega właśnie na tym, że dźwięki w oktawie różnią się od siebie o stały czynnik, w tym przypadku o  $\sqrt[12]{2}$ . Podążając tym tokiem rozumowania, otrzymujemy co następuje: Cis różni się od C o  $\sqrt[12]{2}$ , D różni się od C o  $\sqrt[12]{2} * \sqrt[12]{2}$ , Dis różni się od C o  $\sqrt[12]{2} * \sqrt[12]{2} * \sqrt[12]{2}$  ... i dochodząc aż do C2 zobaczymy, że C2, różni się od C o  $\sqrt[12]{2} * \sqrt[12]{2} * \sqrt[12]{2} * \sqrt[12]{2} * \sqrt[12]{2} * \sqrt[12]{2} * \sqrt[12]{2} * \sqrt[12]{2} * \sqrt[12]{2} * \sqrt[12]{2} * \sqrt[12]{2} * \sqrt[12]{2}$ , czyli  $(\sqrt[12]{2})^{12}$ , czyli dwukrotnie! Przyjęto ponadto, że podstawową częstotliwością, do której stroi się wszystkie instrumenty, jest częstotliwość dźwięku A1 równa 440 Hz. Dla takich założeń częstotliwości prawidłowo nastrojonych strun gitary klasycznej wynoszą kolejno: E2 → 82,41 Hz, A2 → 110 Hz, D3 → 146,83 Hz, G3 → 196 Hz, B3 → 246,94 Hz, E4 → 329,63 Hz. Na tym etapie posiadamy już



Rysunek 1. Schemat ideowy stroika gitarowego

**Wykaz elementów:****Rezystory:** (SMD 0805)

R1, R2: 62 Ω

R3: 220 Ω

**Kondensatory:** (SMD 0805)

C1: 100 nF

C2, C3: 22 pF

**Półprzewodniki:**

U1: ATtiny24 (SOIC14)

LED1, LED2: LED typu OSR5120641E (SMD 1206)

LED3...LED8: LED czerwona (SMD 1206)

**Inne:**

Q1: rezonator kwarcowy SMD 3,6864 MHz (HC49SM)

POWER, TONE: przycisk SMD typu

TACTM-34N

BATT: gniazdo baterii CR2032 typu COMF BC-2002

**Ustawienia fuse-bitów:**

CKSEL3...0: 1101

SUT1...0: 11

CKDIV8: 1

CKOUT: 1

niezbędną z punktu widzenia projektu wiedzę z zakresu muzyki, w związku z czym pora przejść do rozwiązania sprzętowego.

Schemat ideowy stroika pokazano na **rysunku 1**. Jest to nieskomplikowany system procesorowy, którego sercem jest mikrokontroler ATtiny24 realizujący całą, zakładaną funkcjonalność urządzenia. Mikrokontroler steruje pracą dwóch diod LED1 i LED2 będących oświetlaczem stroboskopowym oraz realizuje interfejs użytkownika zbudowany z 6 diod LED wskazujących rodzaj strojonej struny oraz dwóch przycisków: TONE odpowiedzialnego za wybór rodzaju strojonej struny oraz POWER służącego do włączania/wyłączania zasilania urządzenia. Warto podkreślić, że całe urządzenie jest zasilane z niewielkiej baterii CR2032 o napięciu 3 V, więc w tym projekcie istotne było odpowiednie gospodarowanie pobieraną energią. Do realizacji tego wymagania wykorzystano możliwość wprowadzania mikrokontrolera w tryb uśpienia (w naszym wypadku tryb Power Down), w którym prąd pobierany przez urządzenie spada do wartości poniżej 0,1 μA! W porównaniu do prądu obciążenia włączonego urządzenia (pobieranego głównie przez diody LED) rzędu 30 mA osiągnięto gigantyczną wręcz oszczędność

energii, zapewniając długą pracę urządzenia na jednej baterii zasilającej. Mikrokontroler jest usypiany każdorazowo po naciśnięciu przycisku Power (dokładnie, co drugie naciśnięcie) lub po 2 minutach bezczynności urządzenia.

Przejdźmy zatem do szczegółów implementacyjnych. W celu realizacji efektu stroboskopowego wykorzystano układ czasowo-licznikowy Timer1 pracujący w trybie CTC (porównania wartości licznika TCNT1 z zawartością rejestru OCR1A) taktowany przebiegiem zegarowym o częstotliwości 3,6864 MHz (Prescaler=1). Dzięki odpowiedniej konfiguracji Timer1 przejmie kontrolę nad wyprowadzeniem OC1A mikrokontrolera, pozwalając na generowanie przebiegu prostokątnego o częstotliwości zależnej od wartości wpisanej do rejestru OCR1A a obliczanej według wzoru:

$$f_{OC1A} = \frac{f_{clk}}{2 \cdot N \cdot (1 + OCR1A)}, \text{ gdzie:}$$

- $f_{clk}$  – częstotliwość taktowania mikrokontrolera równa 3,6864 MHz,
- $N$  – wartość preskalera ustawiana w rejestrze TCCR1B (w naszym wypadku 1),
- OCR1A – wartość wpisana do rejestru OCR1A.

Na **listingu 1** pokazano sposób konfiguracji Timera1 oraz niezbędne stałe wyznaczone dla potrzeb generowania przebiegów o określonej częstotliwości (dla kolejnych strun gitary). Zmiany częstotliwości na wyjściu OC1A dokonujemy poprzez wczytanie do rejestru OCR1A stosownej stałej z tablicy **Tones[]**, używając w tym celu makra z pliku *pgmspace.h* dostarczanego wraz z pakietem AVR-GCC w następujący sposób: `OCR1A=pgm_read_word(&Tones[Index]);`

Do omówienia pozostaje poruszona na wstępie kwestia energooszczędności urządzenia i trybu uśpienia mikrokontrolera. W celu zminimalizowania energii pobieranej przez układ zastosowano następujące, charakterystyczne dla wszystkich mikrokontrolerów AVR zabiegi programowo-sprzętowe:

- Wszystkie nieużywane porty I/O ustawiono jako wyjściowe (opcjonalnie można było pozostawić je jako wejściowe z włączonym podciąganiem do napięcia zasilania).

**Listing 1. Sposób konfigurowania Timera1 dla potrzeb generowania przebiegów o określonej częstotliwości**

```
//Tablica przechowująca wartości OCR1A odpowiadające częstotliwościom poszczególnych strun
const uint16_t Tones[] PROGMEM = {22365, 16755, 12552, 9403, 7463, 5590};
//Konfigurowanie Timera1 dla generowania przebiegu na wyjściu OC1A
TCCR1A = (1<<COM1A0); //Zmiana stanu OC1A przy porównaniu
TCCR1B = (1<<WGM12)|(1<<CS10); //Tryb CTC, Prescaler = 1
OCR1A = 22365; //Na starcie ustawiamy częstotliwość struny E
DDRA |= (1<<PA6); //Port OC1A, jako port wyjściowy
```

**Listing 2. Kod odpowiedzialny za minimalizowanie zużycia energii urządzenia**

```
//Redukcja poboru mocy poprzez wyłączenie nieużywanych peryferiów
PRR = (1<<PRTIM0)|(1<<PRUSI1)|(1<<PRADC); //Timer0, Universal Serial Interface, ADC
ACSR = (1<<ACD); //Wyłączenie komparatora analogowego
//Nieużywany port, jako wyjściowy ze stanem 0
DDRB |= (1<<PB3);
```

**DODATKOWE MATERIAŁY NA FTP:**<ftp://ep.com.pl>**USER: 95777, PASS: 53wtjyf6****W ofercie AVT\*****AVT-5614****Podstawowe informacje:**

- Obsługa za pomocą dwóch przycisków.
- Strojenie gitary z użyciem efektu stroboskopowego.
- Wielkość i kształt płytki drukowanej zbliżony do kostki gitarowej.
- Mikrokontroler ATtiny24 taktowany rezonatorem 3,6864 MHz.
- Zasilanie z baterii CR2032.

**Projekty pokrewne na FTP:**

(wymienione artykuły są w całości dostępne na FTP)

AVT-5576	Moduł „delay/reverb” (EP 4/2017)
---	Pogłos analogowy (EP 3/2017)
AVT-5569	Mikser Dry/Wet (EP 2/2017)
AVT-5544	Stereofoniczna, cyfrowa linia opóźniająca (EP 7/2016)
---	DSPfactory – profesjonalny efekt dźwiękowy dla muzyków (EP 3-5/2016)
---	Efekt „Reverb” do gitary lub instrumentu klawiszowego (EP 3/2015)
Projekt 220	Zasilacz do efektów gitarowych (EP 2/2015)
AVT-5484	Delay – efekt do instrumentu muzycznego (EP 1/2015)
AVT-1768	Efekt gitarowo-basowy Fuzz (EP 08/2013)
AVT-1767	Efekt gitarowo-basowy Distortion (EP 08/2013)
AVT-1766	Efekt gitarowo-basowy Overdrive (EP 08/2013)
AVT-1765	Efekt gitarowo-basowy Crunch Drive (EP 08/2013)
AVT-3031	Distortion Plus (EdW 5/2012)
AVT-5344	Efekt gitarowy Fazer (EP 5/2012)
AVT-5215	Cyfrowy efekt gitarowy (EP 12/2009)
AVT-435	Prosty wzmacniacz do ćwiczeń gry na gitarze (EP 7/2005)

**\* Uwaga! Elektroniczne zestawy do samodzielnego montażu.****Wymagana umiejętności lutownia!**

Podstawową wersją zestawu jest wersja [B] nazywana potocznie KiTem (z ang. zestaw). Zestaw w wersji [B] zawiera elementy elektroniczne (w tym [UK] – jeśli występuje w projekcie), które należy samodzielnie wzlutować w dolną 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 wzlutowane w płytce PCB)
- wersja [A] płytka drukowana bez elementów i dokumentacja Kity w których występuje układ scalony wymagający zaprogramowania, posiadają następujące dodatkowe wersje:

- wersja [A\*] płytka drukowana [A] + zaprogramowany układ [UK]

**! 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>

REKLAMA

Projekty na...  
**STM32**

[www.stm32.eu](http://www.stm32.eu)**KAMAMI**

life.augmented

```
Listing 3. Kod odpowiedzialny za uśpienie mikrokontrolera
set_sleep_mode(SLEEP_MODE_PWR_DOWN); //Ustawienie rodzaju trybu uśpienia
sleep_enable(); //Aktywacja możliwości uśpienia mikrokontrolera
sei(); //Zezwolenie na obsługę przerw (w celu późniejszego wybudzenia mikrokontrolera)
sleep_cpu(); //Uśpienie mikrokontrolera
sleep_disable(); //Dezaktywacja możliwości uśpienia mikrokontrolera
```

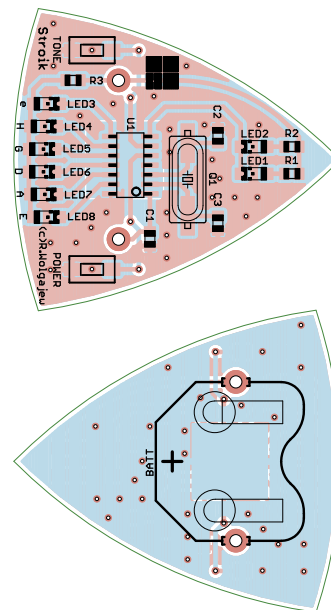
- Wyłączono wszystkie nieużywane peryferia mikrokontrolera.
- Nieużywane urządzenie wprowadzono do stanu uśpienia Power Down.

Za realizację powyższej funkcjonalności odpowiada fragment programu przedstawiony na **listingu 2**.

Do rozwiązania pozostaje jeszcze kwestia uśpienia i wybudzania mikrokontrolera. Do wybudzania mikrokontrolera ze stanu uśpienia wykorzystano wejście przerwania zewnętrznego INT0 skonfigurowane jako wyzwalane poziomem niskim, ponieważ dla tego wejścia to jedyna dostępna możliwość wybudzania. Samą procedurę obsługi przerwania INT0 pozostawiono pustą, gdyż przerwanie służy wyłącznie do wyprowadzenia mikrokontrolera ze stanu uśpienia, w który został wcześniej wprowadzony za pomocą przycisku Power, a które inicjuje wykonanie fragmentu programu odpowiedzialnego za uśpienie mikrokontrolera z **listingu 3**. Zaprezentowany kod z list. 3 korzysta z wygodnych makr

zawartych w pliku *sleep.h* dostarczonym z pakietem AVR-GCC.

To tyle, jeśli chodzi o opis zagadnień implementacyjnych. Pora na pokazanie schematu montażowego urządzenia, który zamieszczono na **rysunku 2**. Jak widać, zaprojektowano niewielki i zwarty obwód drukowany ze zdecydowaną przewagą elementów SMD, który kształtem i wymiarami przypomina kostkę gitarową. Montaż rozpoczynamy od przylutowania elementów umieszczonych na warstwie TOP, przy czym w pierwszej kolejności lutujemy mikrokontroler, następnie wszystkie LED-y zachowując odpowiednią biegunowość, kolejno elementy bierne, a na samym końcu przyciski POWER i TONE oraz rezonator kwarcowy. Następnie przechodzimy na stronę BOTTOM, gdzie wlotujemy gniazdo baterii CR2032 będące jedynym elementem przeznaczonym do montażu przewlekane. Warto także poczynować duże pole lutownicze umieszczone pod gniazdem baterii będące jednocześnie



**Rysunek 2. Schemat montażowy stroika gitarowego**

stykiem ujemnego bieguna baterii. Zaprogramowane urządzenie nie wymaga żadnych procedur konfiguracyjnych i powinno działać tuż po włączeniu zasilania.

Robert Wołgajew, EP

REKLAMA

# Klub Aplikantów Próbek



to inicjatywa redakcji Elektroniki Praktycznej. W kontaktach z firmami redakcja często otrzymuje do przetestowania próbki podzespołów, modułów, a nawet całych urządzeń elektronicznych. Są to zwykle najnowsze typy/modele produktów na rynku. Z chęci podzielenia się z Czytelnikami tymi próbkami zrodziła się inicjatywa pod nazwą Klub Aplikantów Próbek. Członkiem KAP staje się każdy, kto zgłosi chęć przetestowania próbki. Wykaz i krótki opis próbek, którymi dysponuje redakcja EP, można znaleźć poniżej ([www.ep.com.pl/KAP](http://www.ep.com.pl/KAP)). Wystarczy wybrać rodzaj próbek i zwrócić się majłem (na adres: Szef Pracowni Konstrukcyjnej [grzegorz.becker@ep.com.pl](mailto:grzegorz.becker@ep.com.pl)) z prośbą o przesłanie bezpłatnych próbek, podając ich nazwę i adres wysyłki. Warto dopisać jaki jest plan zastosowania tych próbek. Nie jest to konieczne, ale może mieć znaczenie przy podziale próbek w przypadku większej liczby zgłoszeń. Mile widziane, choć nieobowiązkowe, jest też przysłanie do redakcji EP opisu wykonanej aplikacji próbek, oczywiście po jej wykonaniu z zastosowaniem otrzymanej próbki. Autorom przysłanych opisów przyznamy punkty, które będą im dawały pierwszeństwo przy ubieganiu się o kolejne próbki. Najciekawsze opisy aplikacji opublikujemy na forum [ep.com.pl](http://ep.com.pl) lub na łamach Elektroniki Praktycznej. Dla pełnej jasności jeszcze raz podkreślamy, że próbki przekazujemy bezpłatnie i nie trzeba ich zwracać do redakcji.



# www.ep.com.pl/kap