

# MidiCtrl

## – prosty sterownik MIDI

*W swojej praktyce inżynierskiej kilkakrotnie podejmowałem wyzwanie skonstruowania różnorodnych urządzeń audio wyposażonych w prosty interfejs MIDI. Dotyczyło to zwłaszcza tych urządzeń, które zdradzały konotacje estradowe. Wynika to z faktu, że mając do dyspozycji wspomniany interfejs komunikacyjny, możemy zautomatyzować proces produkcji dźwięku, czyniąc go w pełni powtarzalnym. Nie bez znaczenia jest też fakt, że MIDI, mimo że choć wymyślone w latach 80., jest na tyle uniwersalne, iż z powodzeniem wykorzystują je najnowsze urządzenia estradowe. W praktyce prawie każde urządzenie estradowe korzysta z dobrodziejstw tego medium transmisyjnego. Jednym z powodów użycia technologii MIDI jest szeroka dostępność i uniwersalność takich sterowników. Jako że czasami zdarza mi się stosować elektroniczne urządzenia muzyczne (np. syntezatory), wygodne byłoby zaopatrzenie się w jakiś prosty, sterownik MIDI, który ułatwi zapanowanie nad dziesiątkami parametrów edycyjnych, wszak nie zawsze mają one odzwierciedlenie w postaci rzeczywistych elementów regulacyjnych na panelu urządzenia.*

Zanim jednak przejdziemy do opisu prezentowanego urządzenia, należy, choćby pokrótce, przypomnieć sobie podstawowe informacje na temat tego dość leciwego, jednak świetnie pomyślanego standardu.

### Podstawowe informacje o MIDI

Interfejs MIDI jest szeregowym interfejsem komunikacyjnym pracującym z szybkością

31250 ( $\pm 1\%$ ) bits/s, w którym dane przesyłane są w paczkach po 8 bitów, z jednym bitem startu i jednym bitem stopu, bez bitów kontroli parzystości. Układ wejściowy interfejsu MIDI wykonuje się zwykle przy użyciu szybkiego transoptora (6N138), który zamienia prąd w linii (ok. 5 mA) na przebiegi napięciowe, zaś interfejs wyjściowy z wykorzystaniem pary rezystorów ograniczających

Dodatkowe materiały do pobrania ze strony [www.media.avt.pl](http://www.media.avt.pl)

**W ofercie AVT\* AVT-5692**

#### Podstawowe parametry:

- Liczba obsługiwanych kontrolerów MIDI: 128
- Liczba obsługiwanych komunikatów MMC: 5
- Liczba obsługiwanych identyfikatorów urządzeń: 128
- Liczba obsługiwanych kanałów MIDI: 16
- Zasilanie: 7,5...9 VDC, 100 mA

#### Projekty pokrewne na [www.media.avt.pl](http://www.media.avt.pl):

- AVT-5594 Konwerter MIDI-SPI (EP 7/2017)
- AVT-960 Płyta ewaluacyjna i klawiatura muzyczna MIDI (EP 12/2006)
- AVT-842 Asynchroniczny konwerter RS232-MIDI (EP 12/1999)

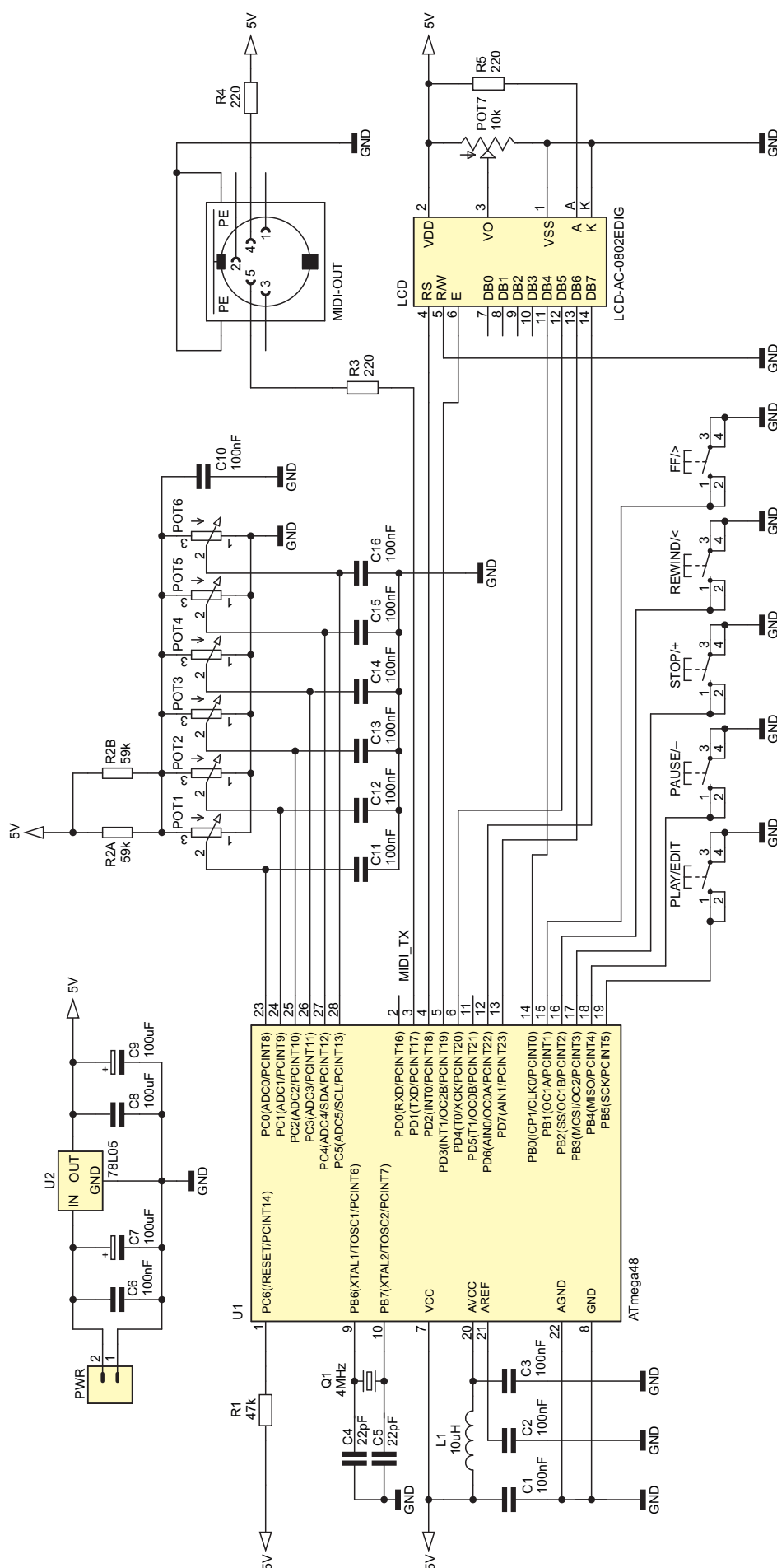
**Uwaga!** Elektroniczne zestawy do samodzielnego montażu. Wymagana umiejętność lutowania!

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] – płytkę drukowaną bez elementów i dokumentacji
- 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](mailto:kity@avt.pl).



### Rysunek 1. Schemat ideowy sterownika MIDI

**Wykaz elementów:**

**Rezystory:** (obudowy miniaturowe, 1/8 W)  
R1: 47 kΩ  
R2A, R2B: 59 kΩ 1%  
R3...R5: 220 Ω  
POT1..POT6: potencjometr suwakowy 50 kΩ  
typu POT50K/B-SUWD (wym. 88×12,5×11 mm,  
posuw suwaka 60 mm)  
POT7: potencjometr montażowy leżący  
10 kΩ

**Kondensatory:** (raster 0,1")  
C1...C3, C6...C16: ceramiczny 100 nF  
C4, C5: ceramiczny 22 pF

### Półprzewodniki:

U1: ATmega48 (DIL28)  
U2: 78L05 (TO-92)  
LCD: wyświetlacz alfanumeryczny 2x8  
znaków typu LCD-AC-0802E-DIG G/KK

Inne:

Q1: rezonator kwarcowy 4 MHz (raster 0,2")  
L1: dławik osiowy 10  $\mu$ H (raster 0,2")  
PWR: złącze śrubowe AK500/2 (raster 0,1")  
FF/>, PAUSE/-, PLAY/EDIT, REWIND/<,  
STOP/-: mikroszwitch TACT 9 mm  
MDI-OUT: gniazdo DIN-5 do montażu THT

prąd diody LED w urządzeniu odbiorczym. W standardzie MIDI dane przesyłane są grupowo w formie tzw. komunikatów (*Messages*), przy czym wprowadzono bardzo prosty sposób na odróżnienie bajtów poleceń sterujących (*Status Byte*) od bajtów danych (*Data Byte*): bajty poleceń mają ustawiony najstarszy bit (0xFF...0x80) a bajty danych najstarszy bit mają wyzerowany (0x7F...0x00). Zwykle informacje przesyłane są w kolejności: bajt polecenia i po nim jeden lub dwa bajty danych (w zależności od rodzaju polecenia). Polecenia wysyłane są tylko przy zmianie danego elementu sterującego. Bajt polecenia określa jedną ze standardowo zdefiniowanych funkcji, którą instrument ma wykonać (4 najstarsze bity), np. *Note On/Off* (włącz/wyłącz nutę), *Control Change* (zmień parametr urządzenia), *Program Change* (zmień rodzaj brzmienia) oraz numer kanału MIDI, na którym informacja ma być odebrana (pozostałe 4 bity określające jeden z 16 kanałów MIDI). Dla porządku należy wspomnieć o możliwości ograniczenia transferu danych poprzez usunięcie redundancji, z której korzysta metoda *Running Status*. Polega ona na wysłaniu jednego bajta polecenia i wielu bajtów danych (bez każdorazowego ponawiania bajta polecenia) w przypadku przesyłania tego samego rodzaju sygnałów sterujących jeden za drugim, np. sygnały wywołane zmianą jednego i tego samego regulatora.

## Budowa urządzenia

Prezentowany układ to prosty sterownik MIDI złożony z 6 elementów regulacyjnych (potencjometrów suwakowych), którym możemy przypisać dowolny kontroler MIDI (*Control Change*) i 5 przycisków funkcyjnych przeznaczonych do sterowania dowolnym rejestratorem MIDI, wspierającym rozszerzenie MIDI zwane MMC (*Midi Machine Control*). Schemat ideowy urządzenia pokazano na **rysunku 1**. Zaprojektowano bardzo prosty system mikroprocesorowy,



**Ustawienia Fuse-bitów (ważniejszych):**

```
CKSEL3...0: 1101
SUT1...0: 11
CKDIV8: 1
EESAVE: 0
```

którego sercem jest niewielki mikrokontroler firmy Microchip (dawniej Atmel) o oznaczeniu ATmega48 taktowany zewnętrznym rezonatorem kwarcowym o częstotliwości 4 MHz. Zastosowanie rezonatora kwarcowego jako źródła taktowania mikrokontrolera wynikało z potrzeby zapewnienia dużej dokładności prędkości transmisji MIDI. Mikrokontroler nasz odpowiedzialny jest za obsługę 6 potencjometrów suwakowych połączonych w konfiguracji dzielnika napięcia, które mierzone jest każdorazowo przez kolejne kanały wbudowanego w strukturę mikroprocesora przetwornika ADC, obsługę 5 przycisków do zdalnej kontroli dowolnego rejestratora MIDI (za pomocą komunikatów MMC), obsługę niewielkiego wyświetlacza alfanumerycznego o organizacji 2×8 znaków stanowiącego element interfejsu użytkownika oraz obsługę wyjściowego interfejsu MIDI zbudowanego zaledwie z 2 rezystorów i gniazda DIN 5.

## Działanie i obsługa

Zasada działania programu obsługi aplikacji jest banalna. Mikrokontroler cały czas mierzy napięcia na zaciskach kolejnych

### Listing 1. Plik nagłówkowy modułu obsługującego interfejs MIDI

```
//Definicje prędkości interfejsu MIDI
#define MIDI_BAUD 31250
//Obliczamy UBRR dla U2X=0
#define __UBRR_MIDI F_CPU/16/MIDI_BAUD-1

//Definicje wybranych komunikatów MIDI
#define MIDI_CTRL_CHANGE 0xB0
#define MIDI_PRGM_CHANGE 0xC0

//Definicje dla MIDI SYSEX
#define MIDI_SYSEX_START 0xF0
#define MIDI_SYSEX_END 0xF7
#define MIDI_UNIVERSAL_ID 0xF7

//Definicje dla MIDI Machine Control
#define MIDI_MMC_COMMAND 0x06
#define MIDI_MMC_STOP 0x01
#define MIDI_MMC_PLAY 0x02
#define MIDI_MMC_FF 0x04
#define MIDI_MMC_REWIND 0x05
#define MIDI_MMC_PAUSE 0x09
```

### Listing 2. Funkcja inicjująca interfejs USART mikrokontrolera do pracy jako sprzęg MIDI

```
void initMIDI(void){
    //Ustawienie prędkości dla
    MIDI
    UBRR0H = (__UBRR_MIDI)>>8;
    UBRR0L = __UBRR_MIDI;
    //Załączenie nadajnika
    UCSRB |= (1<<TXEN0);
    //Ustawienie formatu ramki:
    //8bitów danych, 1 bit stopu
    UCSRC = (3<<UCSZ00);
}
```

potencjometrów POT1...POT6 i na skutek zmiany ich wartości wysyła stosowny komunikat typu Control Change (0xB0) ze zdefiniowanym wcześniej numerem kontrolera MIDI i wartością zależną od położenia suwaka każdego z potencjometrów (zakres wartości 0...127). Podobnie mikrokontroler monitoruje cały czas stan przycisków sterujących i w przypadku ich naciśnięcia wysyła stosowny komunikat MIDI. Komunikat ten należy do grupy komunikatów typu System Exclusive, czyli dedykowanych konkretnemu urządzeniu odbiorczemu, którego identyfikator ustawiamy w ramach konfiguracji niniejszego sterownika. Konstrukcja komunikatu System Exclusive takiego rodzaju wygląda następująco:

**0xF0, 0xF7, Device-ID,  
Sub-ID1, Sub-ID2, 0xF7**

gdzie:

- Device-ID – określa identyfikator docelowego urządzenia odbiorczego (0...127),
- Sub-ID1 – określa rodzaj przesyłanego komunikatu, w naszym przypadku wartość ta równa 0x06 oznacza rozkaz standardu MMC,
- Sub-ID2 – określa typ rozkazu MMC, dla przykładu 0x01 → polecenie Stop, 0x02 → polecenie Play, 0x04 → polecenie Fast forward itd.

Dla dociekliwych – wartości 0xF0 oraz 0xF7 to znaczniki początku i końca komunikatu System Exclusive, zaś 0xF7 to wartość uniwersalnego urządzenia SysEx, dla którego przeznaczony jest bieżący komunikat.

Warto również podkreślić, że przyciski funkcyjne nie służą tylko do generowania komunikatów MIDI (typu MMC), lecz za ich

### Listing 3. Funkcja wysyłająca jeden bajt za pomocą interfejsu MIDI

```
inline void MIDISendByte(uint8_t Byte){
    //Czekamy na ewentualne opróżnienie bufora nadawczego
    while (!(UCSR0A & (1<<UDRE0)));
    UD0 = Byte;
}
```

### Listing 4. Funkcja wysyłająca komunikat typu Control Change

```
inline void MIDISendCC(uint8_t ctrlNumber, uint8_t ctrlValue, uint8_t Channel){
    MIDISendByte(MIDI_CTRL_CHANGE|Channel);
    MIDISendByte(ctrlNumber);
    MIDISendByte(ctrlValue);
}
```

### Listing 5. Funkcja wysyłająca komunikat typu MMC

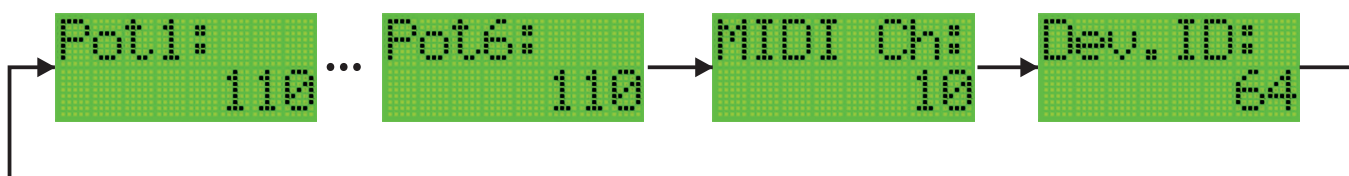
```
inline void MIDISendMMC(uint8_t commandType, uint8_t deviceID){
    MIDISendByte(MIDI_SYSEX_START);
    MIDISendByte(MIDI_UNIVERSAL_ID);
    MIDISendByte(deviceID);
    MIDISendByte(MIDI_MMC_COMMAND);
    MIDISendByte(commandType);
    MIDISendByte(MIDI_SYSEX_END);
}
```



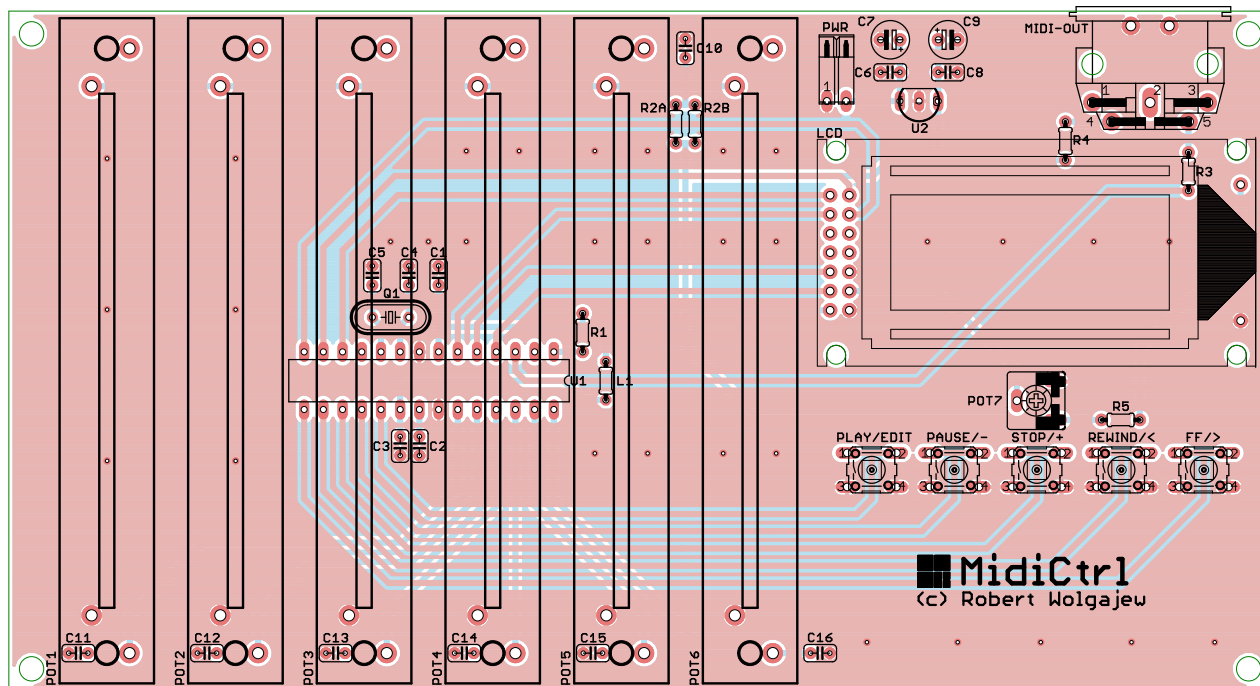
Rysunek 2. Wygląd ekranu graficznego interfejsu użytkownika w trybie normalnym

pomocą dokonujemy również konfiguracji sterownika MidiCtrl, korzystając ze specjalnego trybu edycyjnego uruchamianego poprzez długie naciśnięcie przycisku PLAY/EDIT. W trybie edycji klawisze oznaczone REWIND/< i FF/> służą do przełączania kolejnych pozycji menu edycyjnego, zaś PAUSE/–, STOP+ służą do regulacji parametru wybranego menu edycyjnego. Wyjście z menu edycyjnego możliwe jest poprzez ponowne, długie przyciśnięcie klawisza PLAY/EDIT, czemu towarzyszy zapisanie wszystkich ustawień konfiguracyjnych w nieulotnej pamięci EEPROM mikrokontrolera. Ustawienia te to: numery kontrolerów MIDI przyporządkowane do kolejnych potencjometrów regulacyjnych, numer aktywnego kanału MIDI, na którym odbywa się transmisja danych oraz identyfikator urządzenia odbiorczego, dla którego przeznaczone są komunikaty MMC.

Wygląd ekranu graficznego interfejsu użytkownika w trybie normalnym pokazano na **rysunku 2**, na **rysunku 3** pokazano wygląd poszczególnych ekranów menu edycyjnego (konfiguracyjnego). Warto zauważyć, że w trybie normalnym na ekranie urządzenia MidiCtrl pokazywane są dodatkowo w sposób graficzny wartości poszczególnych kontrolerów MIDI, którym odpowiadają położenia suwaków kolejnych potencjometrów suwakowych (za pomocą bargrafów).



Rysunek 3. Wygląd ekranu graficznego interfejsu użytkownika w trybie edycji (konfiguracji)



Rysunek 4. Schemat montażowy sterownika MIDI

### Opis programu sterującego

Na początek przedstawiam zawartość pliku nagłówkowego, który pokazano na **listingu 1**. Dalej, na **listingu 2**, przedstawiono funkcję inicjującą interfejs USART mikrokontrolera do pracy jako sprzęg MIDI (w tym wypadku wyłącznie jako nadajnik komunikatów MIDI). Z kolei na **listingu 3** zilustrowano banalną wręcz funkcję wysyłającą jeden bajt za pomocą interfejsu MIDI (USART). I na sam koniec dwie proste funkcje narzędziowe pozwalające wysłać komunikat MIDI typu Control Change (**listing 4**) i MMC (**listing 5**). Tyle w kwestiach

implementacyjnych, które, jak widzicie, okazały się bajecznie proste.

### Montaż i uruchomienie

Schemat montażowy urządzenia MidiCtrl pokazano na **rysunku 4**. Jak widać, zaprojektowano bardzo zwarty obwód drukowany przeznaczony do montażu elementów przewlekanych. Montaż urządzenia rozpoczynamy od wlutowania mikrokontrolera i rezonatora kwarcowego. Dalej lutujemy stabilizator napięcia, rezystory i kondensatory a na samym końcu potencjometry suwakowe, gniazda połączeniowe i przyciski sterujące.

Ostatnim etapem montażu urządzenia jest przylutowanie alfanumerycznego wyświetlacza LCD czego dokonujemy korzystając z typowych listew typu goldpin. Poprawnie zmontowany układ wymaga wyłączenie regulacji kontrastu wyświetlacza LCD (potencjometr montażowy POT7) oraz konfiguracji parametrów MIDI (menu edycyjne). Dla zmniejszenia ryzyka ewentualnych zakłóceń w pracy urządzenia (generowanie „niepotrzebnych” komunikatów Control Change) metalowe obudowy potencjometrów suwakowych najlepiej połączyć z masą sterownika.

**Robert Wołgajew, EP**

REKLAMA

## ELEKTRONIKA PRAKTYCZNA MEDIA

Aby skorzystać z materiałów dodatkowych dołączonych do numeru, należy:

1. Wejść na stronę **www.media.avt.pl**
2. Zarejestrować się lub zalogować
3. Wybrać wydanie „Elektroniki Praktycznej”, które ma trafić do biblioteki osobistej
4. Odpowiedzieć na proste pytanie dotyczące bieżącego numeru
5. Pobrać pliki

