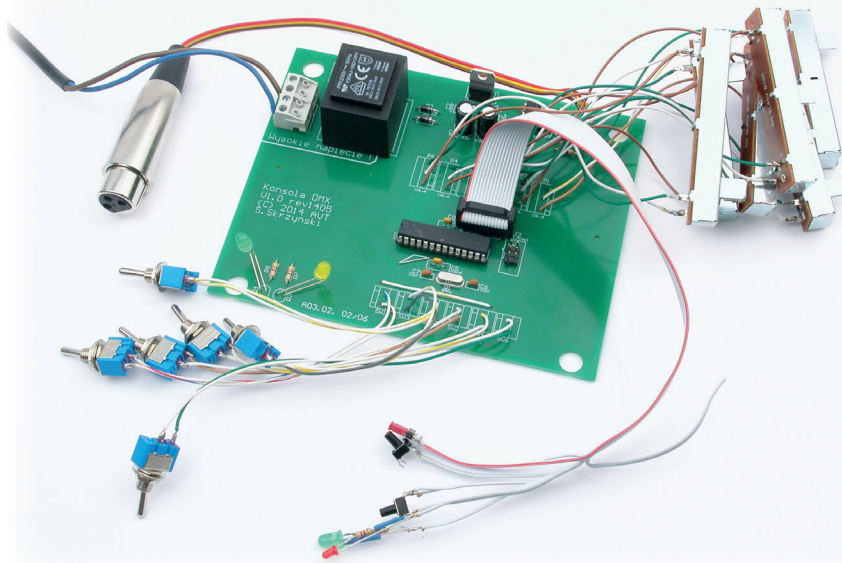


# Miniaturowa konsola z interfejsem DMX

**AVT  
5456**

Na łamach EP już opisano kilkanaście projektów urządzeń z interfejsem DMX. Przeważnie były to odbiorniki, takie jak Dimmery, przekaźniki, sterownik listew RGB. Przyszła pora na nadajnik – miniaturową konsolę. Potrafi ona sterować 12 urządzeniami. Umożliwia także zapis i późniejsze odtworzenie sekwencji sterującej.

**Rekomendacje:** urządzenie przyda się do obsługi małych imprez, gdzie nie jest konieczne używanie dużej konsoli lub komputera.



Schemat ideowy miniaturowej konsoli z interfejsem DMX pokazano na **rysunku 1**. Urządzenie jest zasilane z sieci 230 V AC. Transformator TR1 obniża napięcie do 9 V. Stabilizator U1 zapewnia +5 V DC na potrzeby mikrokontrolera i nadajnika RS485 stosowanego powszechnie w urządzeniach DMX. Układ U3 konwertuje sygnał standardu TTL na RS485. Rezystory włączone na wyjściu konwertera do terminatory linii. Rozwiązanie z trzema rezystorami jest stosowane w sieciach RS485 w sytuacji, gdy magistrala jest dwukierunkowa i czasem nie jest do niej dołączony żaden nadajnik. Wtedy to rezystory dołączone do masy i zasilania zapewniają polaryzację magistrali. W systemie DMX nadajnik jest cały czas dołączony do magistrali, więc można pominąć rezystory polaryzujące, a rezystor 240 Ω zamienić na 120 Ω (impedancja

kabla). Mikrokontroler, za pośrednictwem wbudowanego przetwornika A/C, odczytuje napięcie z potencjometrów i umieszcza je w ramce nadawczej DMX w kanałach od 1 do 6. Dane z przetwornika A/C są uśredniane. W konsoli wykorzystano uśrednianie wykładnicze (**listing 1**). Zaletą takiego uśredniania jest mała zajętość pamięci RAM (trzeba pamiętać tylko poprzedni wynik przetwarzania).

Sygnaly z przełączników są odczytywane przez porty cyfrowe i zależnie od ich stanu, w ramce DMX w kanałach od 7 do 12, zapisywana jest wartość 0 lub 255. Tak skompletowana ramka, po uzupełnieniu sygnałami BREAK, MAB i SC, jest wysyłana na magistralę (**listing 2**).

Wysyłane są 24 kanały DMX, ponieważ takie wymagania stawia norma. W kanałach od 14 do 24 na stałe jest zapisana wartość 0.

Budowę ramki wysyłanej przez konsolę umieszczono w **tabeli 1**.

Naciśnięcie przycisku REC spowoduje zaświecenie diody REC oraz rozpocznie proces rejestracji nastaw potencjometrów i przełączników do pamięci EEPROM. Zapis kończy się po ponownym naciśnięciu przy-

#### W ofercie AVT\*

**AVT-5456 A**      **AVT-5456 B**

#### Podstawowe informacje:

- Zasilanie 230 V AC/2 VA.
- 12-kanałowy interfejs DMX.
- Rejestrowanie i odtwarzanie sekwencji sterujących.
- Mikrokontroler Atmega88, Atmega168 lub Atmega328.
- Obudowa KM-60.

#### Dodatkowe materiały na FTP:

<ftp://ep.com.pl>, user: **28637**, pass: **752sjb64**

#### • wzory płytek PCB

#### Projekty pokrewne na FTP:

(wymienione artykuły są w całości dostępne na FTP)

- AVT-5435 Sterownik DMX-RGB (EP 2/2014)
- AVT-5429 Transmisja DMX512 przez sieć Ethernet (EP 1/2014)
- AVT-5400 DMX Dimmer & Relay (EP 6/2013)
- AVT-5181 Sześciokanałowy dimmer z DMX512 (EP 4/2009)
- AVT-5129 Cyfrowy sterownik DMX512 (EP 4/2008)
- AVT-930 Konwerter USB-DMX512 (EP 5-6/2006)

#### \* Uwaga:

Zestawy AVT mogą występować w następujących wersjach:  
 AVT xxxx UK to zaprogramowany układ. Tylko i wyłącznie. Bez elementów dodatkowych.  
 AVT xxxx A płytka drukowana PCB (lub płytki drukowane, jeśli w opisie wyraźnie zaznaczono), bez elementów dodatkowych.  
 AVT xxxx A+ płytka drukowana i zaprogramowany układ (czyli połączenie wersji A i wersji UK) bez elementów dodatkowych.  
 AVT xxxx B płytka drukowana (lub płytki) oraz komplet elementów wymienionych w załączniku pdf  
 AVT xxxx C to nie innego jak zmontowany zestaw B, czyli elementy wmontowane w PCB. Należy mieć na uwadze, że o ile nie zaznaczono wyraźnie w opisie, zestaw ten nie ma obudowy ani elementów dodatkowych, które nie zostały wymienione w załączniku pdf  
 AVT xxxx CD oprogramowanie (nieczęsto spotykana wersja, lecz jeśli występuje, to niezbędne oprogramowanie można ściągnąć, klikając w link umieszczony w opisie kitu)

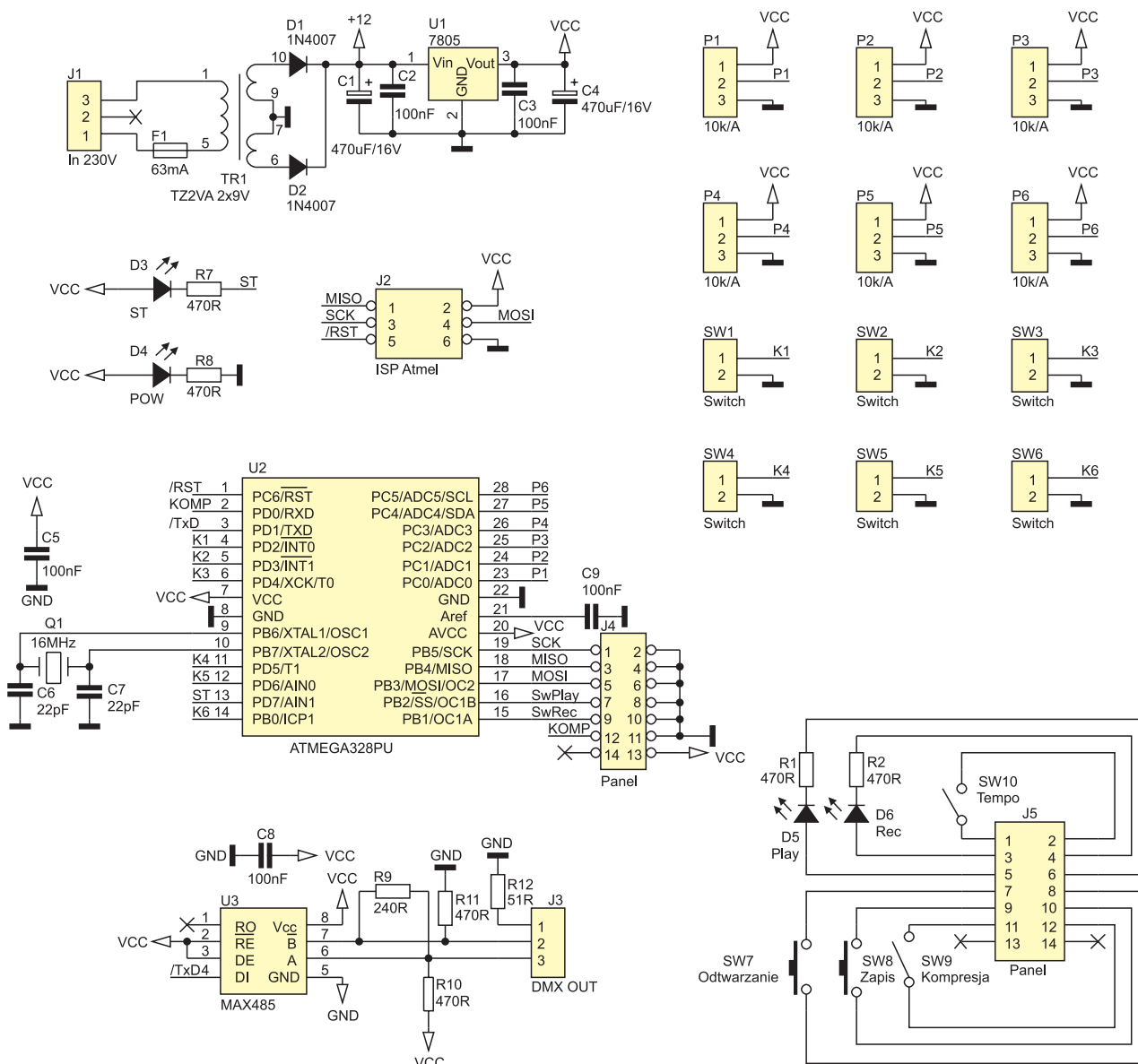
Nie każdy zestaw AVT występuje we wszystkich wersjach! Każda wersja ma załączony ten sam plik pdf! Podczas składania zamówienia upewnij się, którą wersję zamawiasz! (UK, A, A+, B lub C). <http://sklep.avt.pl>

#### Listing 1. Uśrednianie wykładnicze

```
#define WAGA 0.9
srednia[ AdcCnt ] = srednia[ AdcCnt ] * WAGA + (1-WAGA) * ADCH;
// „+1” bo na 0 jest sygnał „SC”
NadawczyDmx[ AdcCnt+1 ] = srednia[ AdcCnt ];
```

#### Listing 2. Procedura generująca sygnały BREAK i MAB:

```
void SendDmx()
{
  // Program główny generuje BREAK i MAB po czym wyśle dane na przerwaniach
  (wpis do UDR0)
  DDRD |= (1<<PD1); // Port wyjściem
  Usart0_Off(); // wyłącz UARTA
  PORTD ^= ~(1<<PD1); _delay_us(200); // wygeneruj BREAK
  PORTD |= (1<<PD1); _delay_us(88); // wygeneruj MAB
  Usart0_Init(); // włącz UARTA
  UDR0 = NadawczyDmx[ PtrTxRs=0 ]; // Wpisz do nadajnika pierwszy znak (start transmisji)
}
```



Rysunek 1. Schemat ideowy miniaturowej konsoli DMX

Tabela 1. Budowa ramki Mini konsoli			
Nr bajtu	Nazwa	Wartość/funkcja	Uwagi
	Break	Wartość 0 przez 200us	
	MAB	Wartość 1 przez 88us	
	SC	0	
1	Kanał 1	0...255	Ustawienie potencjometru P1
2	Kanał 2	0...255	Ustawienie potencjometru P2
3	Kanał 3	0...255	Ustawienie potencjometru P3
4	Kanał 4	0...255	Ustawienie potencjometru P4
5	Kanał 5	0...255	Ustawienie potencjometru P5
6	Kanał 6	0...255	Ustawienie potencjometru P6
7	Kanał 7	0 lub 255	Ustawienie przełącznika SW1
8	Kanał 8	0 lub 255	Ustawienie przełącznika SW2
9	Kanał 9	0 lub 255	Ustawienie przełącznika SW3
10	Kanał 10	0 lub 255	Ustawienie przełącznika SW4
11	Kanał 11	0 lub 255	Ustawienie przełącznika SW5
12	Kanał 12	0 lub 255	Ustawienie przełącznika SW6
13	Kanał 13	0	Dopełnienie do 24 kanałów
14	Kanał 14	0	Dopełnienie do 24 kanałów
...	...		Dopełnienie do 24 kanałów
24	Kanał 24	0	Dopełnienie do 24 kanałów

cisku REC lub wypełnieniu pamięci. Budowę rekordu zapisywanego w pamięci EEPROM opisuje tabela 2.

Ze względu na niewielką pojemność pamięci procesora przy próbkowaniu co 100 ms (10 razy na sekundę) w procesorze ATmega328 można zapisać tylko 14 sekund, co wynika z wyrażenia. W mikrokontrolerze

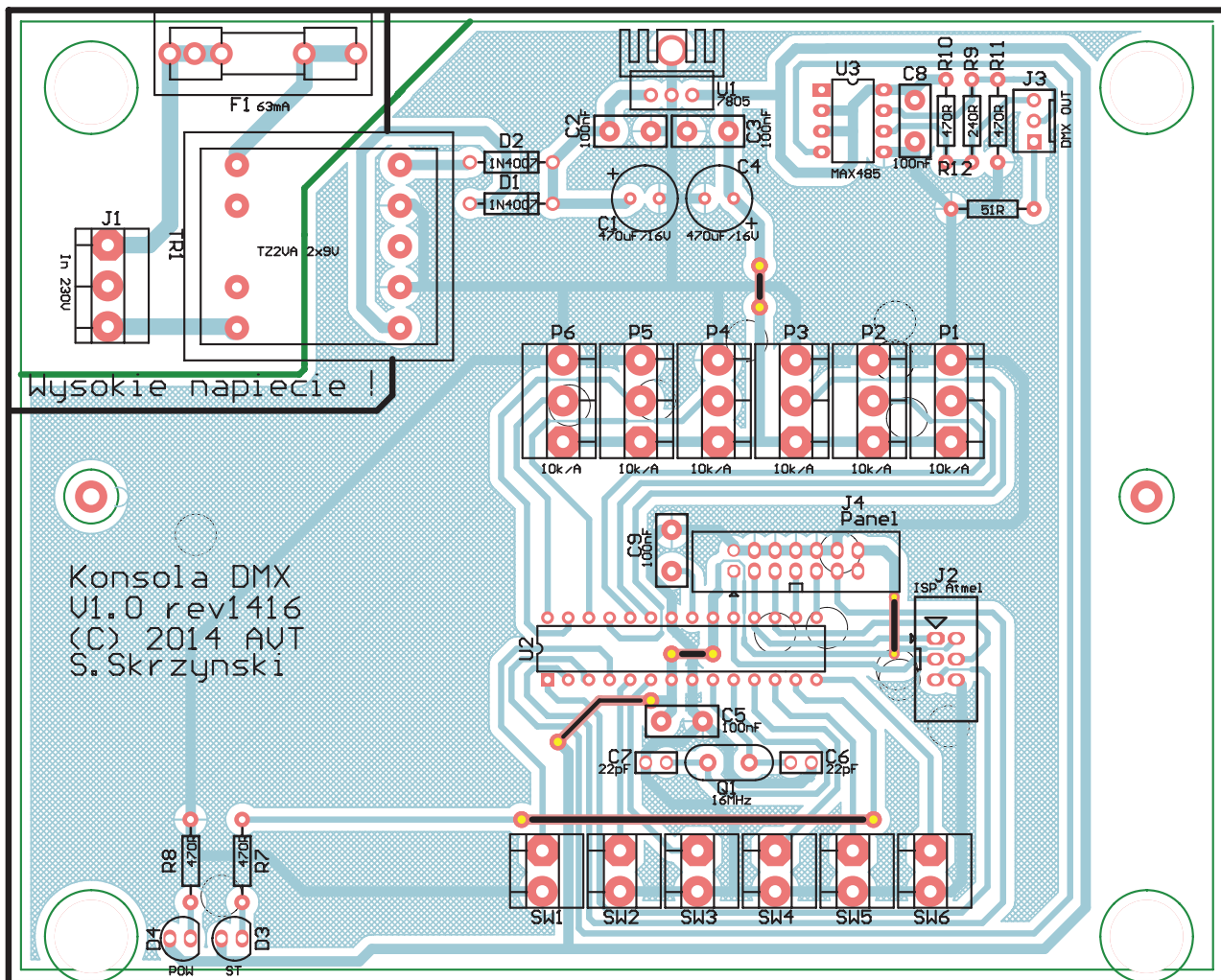
REKLAMA

Projekty na...  
**STM32**

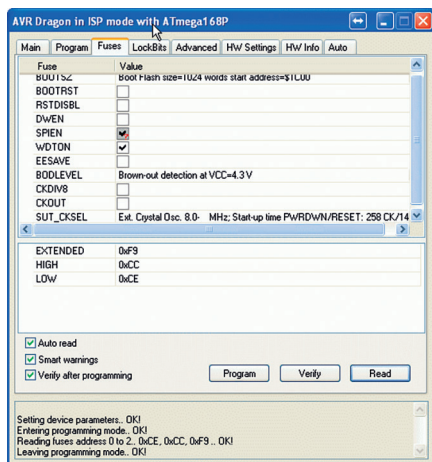
[www.stm32.eu](http://www.stm32.eu)

life.augmented

**KAMAMI**



Rysunek 2. Schemat montażowy miniaturowej konsoli DMX



Rysunek 3. Ustawienia ważniejszych fusebitów

racz ATmega168 lub ATmega88 jest to już tylko 7 sekund, ponieważ pamięć EEPROM ma pojemność 500 bajtów. Aby wydłużyć czas zapisu zastosowano małą „sztuczkę”. Dane zapisywane są do EEPROM co 80, 160, 320 lub 640 ms, zależnie od ustawienie przełączników *KOMPRESJA* i *TEMPO*. Dane o po-

łożeniu potencjometrów są wysyłane zawsze co 80 ms, ale pobierane są z EEPROM co 80 lub 160 ms, a dane pomiędzy próbkami są uśredniane (**listing 3**).

Zmienna *FLdziel* jest zmieniana po każdym odczycie z EEPROM, więc jeden odczyt zapisuje do bufora DMX wartość

Tabela 2. Budowa rekordu w EEPROM

Nr bajtu	Nazwa	Funkcja
1	POT1	Ustawienie potencjometru P1
2	POT2	Ustawienie potencjometru P2
3	POT3	Ustawienie potencjometru P3
4	POT4	Ustawienie potencjometru P4
5	POT5	Ustawienie potencjometru P5
6	POT6	Ustawienie potencjometru P6
7	SW	Ustawienie przełączników SW1...SW6 (bity 0...5)

Tabela 3. Ustawienie przełączników *KOMPRESJA* i *TEMPO*

Ustawienie przełącznika <i>KOMPRESJA</i>	Ustawienie przełącznika <i>TEMPO</i>	Procesor Atmel	Próbkowanie/Czas zapisu	
Off	Off	Mega88/168	Kompresja 4:1	Próbkowanie 640 ms (160*4) Czas zapisu 44 sek.
On	Off	Mega88/168	Kompresja 4:1	Próbkowanie 320 ms (80*4) Czas zapisu 22 sek.
Off	On	Mega88/168	Kompresja 2:1	Próbkowanie 160 ms (80*2) Czas zapisu 11 sek.
On	On	Mega88/168	Kompresja 1:1	Próbkowanie 80 ms (80*1) Czas zapisu 6 sek.
Off	Off	Mega328	Kompresja 4:1	Próbkowanie 640 ms (160*4) Czas zapisu 88 sek.
On	Off	Mega328	Kompresja 4:1	Próbkowanie 320 ms (80*4) Czas zapisu 44 sek.
Off	On	Mega328	Kompresja 2:1	Próbkowanie 160 ms (80*2) Czas zapisu 22 sek.
On	On	Mega328	Kompresja 1:1	Próbkowanie 80 ms (80*1) Czas zapisu 12 sek.

```

Listing 3. Uśrednianie próbek przy odczycie z kompresją 2:1
for (x=0; x<MEMREC; x++)
{
    dana = eeprom_read_byte( &EEmemory[ rekord ][ x ] );
    danaN = eeprom_read_byte( &EEmemory[ rekord+1 ][ x ] );
    if ( (FLdziel & 1) == 1 )
    {
        dana = (dana + danaN) >> 1; // uśrednianie
    }
    NadawczyDmx[ x+1 ] = dana;
}
if ( (FLdziel & 1) == 1 )
{
    rekord++; if ( rekord >= MEMLEN-1 ) PlayStop(); // następny rekord
}
++FLdziel;
    
```

```

Listing 4. Uśrednianie próbek przy odczycie z kompresją 4:1
for (x=0; x<MEMREC; x++)
{
    d0 = eeprom_read_byte( &EEmemory[ rekord ][ x ] );
    /* SW czytamy z wyprzedzeniem, więc nie będzie problemu z interpolacją, ale
    odczyt skończy się o jeden rekord wcześniej */
    danaN = eeprom_read_byte( &EEmemory[ rekord+1 ][ x ] );
    d2 = (d0 + danaN) >> 1;
    switch( FLdziel & 3 )
    {
        case 0:
            dana = d0;
            break;
        case 1:
            dana = (d0+d2) >> 1;
            break;
        case 2:
            dana = d2;
            break;
        case 3:
            dana = (danaN+d2) >> 1;
            break;
    }
    NadawczyDmx[ x+1 ] = dana;
}
if ( (FLdziel & 3) == 3 )
{
    rekord++; if ( rekord >= MEMLEN-1 ) PlayStop(); // następny rekord
}
++FLdziel;
    
```

Tabela 4. Funkcje Diod LED

Nazwa	Funkcja
D4 „POW”	Świeci gdy włączone zasilanie
D5 „Play”	Świeci podczas odtwarzania zapisu z EEPROM
D6 „Rec”	Świeci podczas zapisu do EEPROM
D3 „ST”	<ul style="list-style-type: none"> <li>Miga z częstotliwością około 1 Hz w stanie spoczynku (około 1000 razy mniej niż od częstotliwości wysyłania ramek DMX)</li> <li>Miga z połową częstotliwości próbkowania podczas zapisu do EEPROM</li> <li>Miga z połową częstotliwości odtwarzania próbek podczas odczytu</li> </ul>

z EEPROM, drugi uśrednioną wartość wyliczoną z aktualnej i kolejnej próbki (**listing 4**).

Możliwe ustawienia przełączników *KOMPRESJA* i *TEMPO* pokazano w **tabeli 3**. Oczywiście można użyć innych algorytmów pakowania. Zakładając, że równocześnie zmieniają się nastawy kilku, a nie wszystkich potencjometrów można by użyć algorytmu ByteRun. Długość zapisu można też zwiększyć przez zastosowanie mikrokontrolera z większą ilością pamięci ram lub użyć zewnętrznej pamięci RAM/EEPROM komunikującej się przez SPI lub IIC. Zaletą zewnętrznej pamięci byłaby możliwość zwiększenia częstotliwości próbkowania grubo poniżej 1 ms, co zwiększyłoby płynność odtwarzania. Konsola miała być prosta, więc zrezygnowałem z tej opcji, bo prościej będzie zastosować komputer z przejściówką USB-DMX.

Jeśli w EEPROM znajduje się jakiś zapis, naciśnięcie przycisku PLAY spowoduje za-

świecenie diody PLAY oraz rozpocznie się proces odtwarzania nastaw potencjometrów i przełączników z pamięci EEPROM. Odczyt kończy się po ponownym naciśnięciu przycisku PLAY lub odczytaniu całego zapamiętanego zapisu. Stan pracy konsoli jest sygnalizowany za pomocą diod LED (**tabela 4**).

### Montaż i uruchomienie:

Schemat montażowy konsoli pokazano na **rysunku 2**. Montaż jest typowy i nie wymaga omawiania. Pod układy warto zastosować podstawki. Uruchomienie rozpoczyna się od zasilacza. Jeśli napięcie jest poprawne można umieścić układy w podstawkach. Jeśli procesor nie jest zaprogramowany, możemy to zrobić przy wykorzystaniu złącza J2. Ustawienie bitów konfiguracyjnych przedstawiono na **rysunku 3**. Podczas programowania trzeba pamiętać, że ustawienie przełącznika SW10 „TEMPO” w pozycję On (zwarcie do masy), zablokuje możliwość programowania (zablokowana linia SCK).

### Wykaz elementów

**Rezystory:** (1/8 W):

R12: 51 Ω

R9: 240 Ω

R1, R2, R7, R8, R10, R11: 470 Ω

P1...P6: 10 kΩ/A (potencjometr suwakowy)

**Kondensatory:**

C6, C7: 22 pF/50 V (ceram.)

C2, C3, C5, C8, C9: 100 nF/50 V (ceram.)

C1, C4: 470 μF/16 V/16 V (elektrolit.)

**Półprzewodniki:**

D1, D2: 1N4007

U1: 7805 (TO-220)

U2: ATmega328PU (można użyć ATmega88 lub 168)

U3: MAX485

D4, D5: dioda LED, zielona

D6: dioda LED, czerwona

D3: dioda LED, żółta

**Inne:**

Q1: 16 MHz (kwarc w obudowie HC49 lub HC495)

F1: 63 mA (gniazdo+bezpiecznik 5×20 mm)

J3: NS25-W3, gniazdo NS25 3 pin

XLR-3G-C, gniazdo XLR-3 do obudowy

NS25-T, 3 szt. terminali do wtyku NS25

J2: ZL231-6PG gniazdo ZL231-6PG (6 pin proste) lub listwa kołkowa ZL202-6G goldpin 2×3

J1: TB-5.0-P-PP-3, TB-5.0-PIN złącze TB z listwą kołkową

J4: ZL231-10PG, gniazdo ZL231-10PG (10 pin, proste)

FT14, wtyk zaciskany na taśmie + taśma FLAT 14-żyłowa

SW1...SW6, SW9, SW10: MTS-101

(przełącznik pojedynczy SPST 2P)

SW7, SW10: przycisk chwilowy, zielony

SW8, SW10: przycisk chwilowy, czerwony

TR1: T22VA 2×9 V AC

DIP28S: podstawka precyzyjna 28 pin wąska (300 mils)

PDIP8: podstawka precyzyjna 8 pin

Po włączeniu zasilania dioda D3 powinna migać z częstotliwością około 1 Hz. Elementy umieszczone w ramce znajdują się poza płytką i są z nią połączone taśmą. Urządzenie można zamknąć w obudowie KM-60. W materiałach dodatkowych, na FTP, znajduje się kod źródłowy i wynikowy dla Atmega88, Atmega168 i Atmega328.

**Sławomir Skrzyński, EP**

REKLAMA

Projekty na  
**STM32**

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

**ST** **KAMAMI**  
life.augmented