

ELEKTRONIKA

dla wszystkich

3/2018 MARZEC • CENA 12zł (w tym 5% VAT) • NAKŁAD: 14 990 egz.

www.elportal.pl

Alexander Amplifier

czyli stałoprądowy wzmacniacz słuchawkowy

Kurs Arduino – Pierwsze kroki

- ▶ Układy PLD – Krótki wstęp
- ▶ Regulator mocy z wyświetlaczem LCD
- ▶ Gasnąca lampka LED
- ▶ Latarka do... zębów?
- ▶ Energooszczędny termometr diodowy
- ▶ Smog
- ▶ DipTrace – Trójwymiarowy podgląd płytki drukowanej
- ▶ Zegar Nixie z „kukułka”
- ▶ Kurs C
- ▶ Szumy stabilizatorów i zasilaczy
- ▶ Taśmy LED
- Wskazówki praktyczne



Drukarki 3D
filamenty, części zapasowe



www.thingiverse.com/thing:486983
sklep avt.pl

Portale branżowe
AutomatykaB2B.pl
ElektronikaB2B.pl

Miejsca dla
specjalistów



FIRMA PIEKARZ
CZĘŚCI ELEKTRONICZNE

przełączniki
półprzewodniki
złącza
przełączniki
radiatory
obudowy
i wiele więcej...

www.piekarz.pl

AVT 5272 AVTduino - pomysł na AVR - płytka bazowa

Podstawowy moduł zgodny z Arduino. Jest to doskonała baza do tworzenia własnych systemów mikroprocesorowych. Niewielkie wymiary, bezproblemowe uruchomienie i duże wsparcie ze strony społeczności internetowych powodują że moduł przydatny będzie zarówno zawodowcom jak i nowicjuszom – uczącym się programowania.



Wybrane parametry:

- mikrokontroler ATmega328
- konwerter USB/RS232 (spełniający rolę programatora)
- elektroniczny włącznik napięcia zasilającego
- sygnalizacja stanu - diody LED
- zasilanie: 8...12 VDC lub 5 VDC z USB
- wymiary płytki: 69x53 mm



AVT 1615 AVTduino LCD - wyświetlacz LCD dla Arduino

Płytkę zawiera elementy peryferyjne, niezbędne w każdym urządzeniu bazującym na mikrokontrolerach. Całość z wyświetlaczem i przyciskami w prosty sposób pomoże zbudować np.: sterownik, miernik lub zegar.



Wybrane parametry:

- uniwersalny moduł z peryferiami do platformy Arduino
- wyświetlacz LCD 2x16 znaków z podświetleniem
- dioda LED - sygnalizacja zasilania
- 4 dodatkowe diody LED
- 4 przyciski oraz Reset
- czujnik temperatury LM35
- przetwornik piezo z wbudowanym generatorem
- zasilanie: 5 VDC (z płytki bazowej)
- wymiary płytki: 53x92mm



AVT 1616 AVTduino LED - wyświetlacz LED dla Arduino

Moduł jest nakładką na płytkę bazową AVTduino (Arduino). Oprócz czterocyfrowego wyświetlacza LED, przycisku Reset i złącza programowania ISP, płytka została wyposażona w układ zegara RTC z interfejsem I2C oraz termometr DS18B20.



Wybrane parametry:

- 4 - cyfrowy wyświetlacz LED
- zegar RTC
- 2 dodatkowe przyciski oraz Reset
- czujnik temperatury - DS18B20
- przetwornik piezo z wbudowanym generatorem
- fotorezystor
- dioda LED - sygnalizacja zasilania
- zasilanie: 5 VDC (z płytki bazowej)
- wymiary płytki: 54 71 mm



AVT 1618 AVTduino JOY - manipulator dla Arduino

Moduł został wyposażony w niewielki joystick, dwa przyciski oraz złącza umożliwiające dołączenie trzech serwo mechanizmów. Dzięki takiej konfiguracji możliwe jest zbudowanie prostego manipulatora sterowanego z platformy Arduino.



Wybrane parametry:

- moduł w standardzie Arduino
- joystick
- 3 złącza do serwo mechanizmów
- dioda LED - sygnalizacja zasilania
- 2 dodatkowe diody LED
- 2 dodatkowe przyciski oraz Reset
- zasilanie: 5 VDC (z płytki bazowej)
- wymiary płytki: 52x72 mm



AVT 1619 AVTduino Motor - driver silników

Moduł odpowiedzialny za sterowanie silnikami elektrycznymi. Układ opracowano wykorzystując dwa drivery L293. Każdy z nich zawiera w swej strukturze po dwa mostki H. W efekcie moduł może sterować maksymalnie czterema silnikami prądu stałego lub dwoma silnikami krokowymi.



Wybrane parametry:

- drivery silników elektrycznych - L293
- możliwość sterowania 4 silnikami prądu stałego lub dwoma krokowymi
- 4 dodatkowe przyciski
- dioda LED - sygnalizacja zasilania
- 2 dodatkowe diody LED
- zasilanie: 5 VDC (z płytki bazowej)
- wymiary płytki: 72x52 mm



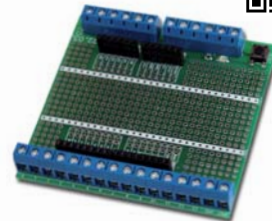
AVT 1633 Uniwersalny moduł rozszerzeń dla Arduino

Uniwersalny moduł rozszerzeń do płytki bazowej AVTduino (Arduino). Oprócz pól stykowych pozwalających zlutować dowolne, współpracujące obwody na krawędziach znajdują się złącza śrubowe typu ARK.



Wybrane parametry:

- moduł ze złączami w standardzie Arduino
- uniwersalne pola lutownicze
- złącza śrubowe typu ARK
- przycisk Reset
- dioda LED - sygnalizacja zasilania
- zasilanie: 5 VDC (z płytki bazowej)
- wymiary płytki: 77x80 mm



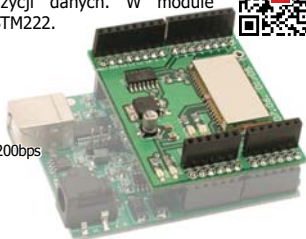
AVT 1646 Moduł Bluetooth kompatybilny z Arduino

Moduł komunikacyjny umożliwia współpracę z AVTduino (Arduino) poprzez interfejs Bluetooth. Pozwoli to zaprojektować system zdalnego sterowania lub bezprzewodowej akwizycji danych. W module wykorzystano popularny moduł transmisji BTM222.



Wybrane parametry:

- popularny moduł Bluetooth BTM222
- domyślna transmisja szeregową z prędkością 19200bps
- wbudowany stabilizator napięcia 3,3V
- sygnalizacja stanu – diody LED
- zasilanie: 5 VDC (z płytki bazowej)
- wymiary płytki: 56x45 mm



AVT 1649 Moduł karty pamięci kompatybilny z Arduino

Moduł rozszerza możliwości płytki bazowej AVTduino (Arduino) o funkcję zapisu danych na karcie pamięci SD. Dzięki temu AVTduino może być używane np.: do akwizycji danych w systemach pomiarowych.



Wybrane parametry:

- moduł ze złączami w standardzie Arduino
- złącze karty SD
- bufor sygnałów karty
- dioda LED - sygnalizacja zasilania
- wbudowany stabilizator 3,3 V
- zasilanie: 5 VDC (z płytki bazowej)
- wymiary płytki: 56x45 mm



AVT 1666 AVTduino RELAY - moduł przekaźników

Moduł przekaźnikowy do serii modułów AVTduino. Na płytce umieszczone zostały cztery przekaźniki ze stykami przełącznymi pozwalające sterować dowolnymi odbiornikami zewnętrznymi.



Wybrane parametry:

- moduł w standardzie płytek AVTduino
- 4 przekaźniki, każdy o obciążalności styków 10 A
- diody LED - sygnalizujące stan każdego z przekaźników
- dioda LED - sygnalizacja zasilania
- złącza śrubowe typu ARK
- złącza pozwalające dołączać kolejne moduły
- zasilanie: 5 VDC (z płytki bazowej)
- wymiary płytki: 66x56 mm



AVT 1722 Miniaturowy panel operatora dla Arduino

Moduł jest nakładką na płytkę bazową Arduino. Oprócz niewielkiego wyświetlacza LCD i kilku przycisków, moduł wyposażono w dodatkowe peryferia, dzięki którym każdy może bez większych problemów zbudować i przetestować praktycznie dowolne urządzenie prototypowe.



Wybrane parametry:

- wyświetlacz LCD 2x8 znaków z podświetleniem
- dioda LED RGB
- dioda LED - sygnalizacja zasilania
- 4 przyciski oraz przycisk Reset
- impulsator obrotowy z przyciskiem
- złącze do magistrali 1-Wire (DS18B20)
- miniaturowy przekaźnik 1 A
- zasilanie: 5 VDC (z płytki bazowej)
- wymiary płytki: 71x53 mm





OPTYMALNE WYNIKI – RÓWNIŻ W TRUDNYCH WARUNKACH: NOWY MULTIMETR CYFROWY

Solidny multimetr cyfrowy z lampą LED

Doskonale przydatny podczas codziennej pracy w terenie w sektorze serwisu, budowy i przemysłu oraz dla ambitnego majsterkowicza.

- Automatyczne podświetlenie przycisków funkcyjnych i przełącznika obrotowego
- Mocna lampa LED do oświetlenia miejsca pomiaru
- 3 5/6-cyfrowy wyświetlacz wartości pomiarowych (6000 Counts)
- Wyświetlacz LCD z pomiarem wartości względnych i 42-segmentowym wykresem słupkowym
- Test ciągłości obwodu z brzęczykiem i testem diody

PeakTech®



EN 61010-1
CAT IV
600 V

EN 61010-1
CAT III
1000 V

Nr zam.: PEAKTECH 3443

NOWOŚĆ **Zł 278,98**



MOŻLIWOŚĆ ZASTOSOWANIA RÓWNIŻ W CIEMNOŚCI!



Podświetlenie przycisków + zintegrowane oświetlenie robocze:

PEAKTECH 3443 i
PEAKTECH 3445

Solidny multimetr cyfrowy z Bluetooth

Podobny do PeakTech 3443 z tymi dodatkowymi funkcjami:

- Pomiar True-RMS
- Interfejs Bluetooth do połączenia ze smartfonem z systemem Android



Nr zam.: PEAKTECH 3445

Zł 384,42 **NOWOŚĆ**

PeakTech®



Dostawa bez smartfona

Oscyloskop cyfrowy

Oscyloskop cyfrowy najnowszej generacji z kolorowym i podświetlanym wyświetlaczem o wysokiej rozdzielczości, dużą szerokością pasma oraz częstotliwością pomiaru, dużą wewnętrzną pamięcią danych oraz złączem USB.

- Złącze VGA do podłączenia zewnętrznego monitora
- Przyłącze LAN do zdalnego odczytu danych przez sieć
- 20 cm (8") kolorowy wyświetlacz TFT

30 MHz
2 KANAŁY



Nr zam.: PEAKTECH 1265

Zł 1.104,89 **BEST SELLER**

PeakTech®



Firmy prezentujące swoje oferty w niniejszym wydaniu EdW



ARTRONIC..... 1



ELMAX..... 75

ELPIN PCB s.c.

ELPIN..... 75



EX-OR..... 75



FERYSTER..... 55



GTB-SOLARIS..... 75



PIEKARZ..... 1, 41



PRODUCENT AUTOMATYKI GRZEWCZEJ

PW KEY..... 75



REICHELT ELEKTRONIK..... 3



SEMICON..... 17



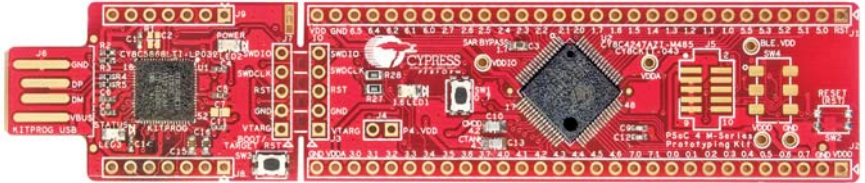
str. 15

Alexander Amplifier, czyli wzmacniacz stałoprądowy

Tematyka wzmacniaczy audio jest bardzo obszerna, ale też kontrowersyjna. Przedstawiany projekt i artykuł to bardzo interesujący wkład do dyskusji na temat poszczególnych koncepcji i rozwiązań.

str. 24 Układy PLD

Układy programowalne, w przeciwieństwie do mikroprocesorów, nadal uważane są za bardzo drogie i bardzo trudne do wykorzystania. Czy jest to słuszny pogląd?



```

Blink | Arduino 1.8.5 (Windows Store 1.8.1)
Plik Edytuj Szukaj Narzędzia Pomoc

Blink
void setup() {
  pinMode(LED_BUILTIN, OUTPUT);
}

void loop() {
  digitalWrite(LED_BUILTIN, HIGH);
  delay(1000);
  digitalWrite(LED_BUILTIN, LOW);
  delay(1000);
}

```

str. 32

Kurs Arduino – Pierwsze kroki

Jeśli jeszcze nie miałeś kontaktu z platformą Arduino, przekonaj się, że początki są wręcz dziecinnie proste. Kurs Arduino w EdW zaczynamy od „kanonicznego” programu Blink.

str. 57

Regulator mocy z wyświetlaczem LCD

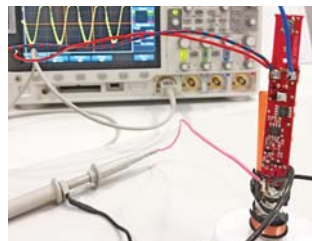
Prezentowany regulator przeznaczony jest do sterowania fazowego urządzeń zasilanych z sieci energetycznej 230V. Znakomicie nadaje się do regulacji grzałek, żarówek oraz wielu innych źródeł światła, które oznaczone są „dimmable”.



str. 62

Latarka do... zębów?

Interesująca historia przeróbki elektrycznej szczoteczki do zębów z bezprzewodowym ładowaniem na bardzo wygodną w użyciu latarkę.



Copyright AVT-Korporacja Sp. z o.o., Warszawa, ul. Leszczyńska 11. Projekty publikowane w „Elektronice dla Wszystkich” mogą być wykorzystywane wyłącznie do własnych potrzeb. Korzystanie z tych projektów do innych celów, zwłaszcza do działalności zarobkowej, wymaga zgody redakcji „Elektroniki dla Wszystkich”. Przedruk oraz umieszczanie na stronach internetowych całości lub fragmentów publikacji zamieszczanych w „Elektronice dla Wszystkich” jest dozwolone wyłącznie po uzyskaniu pisemnej zgody redakcji. Redakcja nie odpowiada za treść reklam i ogłoszeń zamieszczanych w „Elektronice dla Wszystkich”.

Miesięcznik



www.elportal.pl

(12 numerów w roku) jest wydawany we współpracy z kilkoma redakcjami zagranicznymi.

Wydawca:

Wiesław Marciniak

Adres Wydawcy:
AVT-Korporacja sp. z o.o.
ul. Leszczyńska 11
03-197 Warszawa
tel.: (22) 257 84 99
fax: (22) 257 84 00

Redaktor Naczelny:
Piotr Górecki, pg@elportal.pl

Redaktorzy Działów:

Andrzej Janeczek
sp5aht@swiatradio.com.pl

Opracowanie graficzne, skład:

Ewa Górecka-Dudzik

Okladka, zdjęcia, skanowanie:

Piotr Górecki jr

Sekretarz Redakcji

Ewa Górecka-Dudzik
ewa.dudzik@elportal.pl
tel.: (22) 783 00 20
(w godzinach 10:00 – 15:00)

Dział Reklamy:

Katarzyna Gugala
katarzyna.gugala@elportal.pl
tel.: (22) 257 84 64

Klasyczne listy i paczki (projekty i Szkoła Konstruktorów) prosimy adresować:

AVT – EdW
ul. Leszczyńska 11
03-197 Warszawa
(+dopisek określający zawartość)

Korespondencja elektroniczna:

e-maile do Redakcji EdW:
edw@elportal.pl

e-maile do Szkoły Konstruktorów:
szkola@elportal.pl

rozwiązania konkursów – e-maile:
konkursy@elportal.pl

uwagi do rubryki Errata:
errata@elportal.pl

Prenumerata:

W Wydawnictwie AVT
tel: (22) 257 84 22
e-mail: prenumerata@avt.pl

W RUCH S.A.
tel: 801 800 803, (22) 717 59 59
e-mail: prenumerata@ruch.com.pl
www.prenumerata.ruch.com.pl

Stali współpracownicy:

Michał Adamus
Arkadiusz Bartold
Aleksander Bernaczek
Szymon Janek
Krzysztof Kawa
Rafał Orodziński
Michał Pędzimaż
Michał Stach
Szymon Trygar
Adam Sobczyk
Piotr Świerczek
Piotr Wójtowicz

eWydanie dla: Michał Sas-Uhrynowski (msas)

Wydanie elektroniczne przeznaczone wyłącznie do użytku własnego bez prawa do rozpowszechniania.

Projekty

Projekty AVT

Alexander Amplifier, czyli stałoprądowy wzmacniacz słuchawkowy.....	15
Zegar Nixie z „kukulką”, część 3	21

Elektronika 2000

Regulator mocy z wyświetlaczem LCD	57
Gasnąca lampka LED	60

Forum Czytelników

Latarka do... zębów?.....	62
Energooszczędny termometr diodowy.....	64

Szkoła Konstruktorów

Zadanie główne 265

Zaproponuj rozwiązanie elektroniczne, zmniejszające skutki braku energii w sieci 230V.....	44
--	----

Rozwiązanie zadania głównego 260

Zaproponuj nietypowe, „elektroniczne” wykorzystanie fabrycznego kalkulatora lub jego części	45
---	----

Druga klasa Szkoły Konstruktorów Co tu nie gra? 265, 260.....	49
--	----

Trzecia klasa Szkoły Konstruktorów Policz 265, 260.....	53
--	----

Artykuły różne

Smog.....	14
Układy PLD	24
DipTrace – trójwymiarowy podgląd płytki drukowanej i tworzenie dokumentacji produkcyjnej, część 8	29
Arduino. Odcinek 1 – pierwsze kroki	32
Kurs C – lekcja 23	39
Szumy stabilizatorów i zasilaczy, część 2	66
Taśmy LED. Wskazówki praktyczne, część 2	68

Rubryki stałe

Nowości, ciekawostki	6
Poczta	10
Skrzynka porad	12
Prenumerata	8
Księgarnia AVT.....	78
Miniankieta	74
Reklamy	76
Oferta handlowa AVT	81

Konkursy

Jak to działa?.....	70
Czego tu brak?	72
Krzyżówka	73



Marzec i Arduino

W tym numerze mamy na okładce bardzo interesujący projekt stałoprądowego wzmacniacza audio. Jednak najważniejszym wydarzeniem jest rozpoczęty właśnie **kurs Arduino**. Arduino to dojrzałe, a dziś wręcz jedyne sensowne narzędzie do rozpoczęcia przygody z programowaniem mikroprocesorów.

Tak. Bardzo łatwo zacząć i cieszyć się pierwszymi sukcesami, ale z kilku powodów można zostać na etapie „dziecięcych gadżetów” i „gotowców”. Dużo trudniej przejść na wyższy poziom umiejętności i samodzielności. Lektura wielu książek i kursów wskazuje, że często ani Autor, ani Czytelnicy nie mają wiedzy o elektronice. Problem jest też ze stroną informatyczną: w książkach podawane są gotowe programy – szkice i skrótowe objaśnienia tych listingów. Bez głębszej wiedzy informatycznej Czytelnik może wprowadzać co najwyżej drobne modyfikacje, bo większe powodują, że program przestaje działać.

Kurs Arduino w EdW przeznaczony jest dla bardziej i mniej zaawansowanych elektroników, dlatego nie będziemy traktować płytki jako tajemniczej czarnej skrzynki, tylko dość dokładnie zbadamy także jej właściwości „elektroniczne”. Nadal będziemy też w możliwie przystępny sposób stopniowo wyjaśniać trudniejsze zagadnienia programistyczne.

Drugim bardzo ważnym artykułem w tym numerze jest wprowadzenie do PLD – współczesnych układów programowalnych. Przeczytajcie i swoimi głosami (edw@elportal.pl) zdecydуйте, czy chcielibyście w przyszłości zobaczyć w EdW obszerniejszy cykl nauki programowania PLD, prowadzony przez tego Autora.

Serdecznie pozdrawiam.

Piotr Górecki



**Prenumerata
– naprawdę warto!**

FUJIFILM X-H1

Nie tylko tanie fotograficzne aparaty kompaktowe prawie zniknęły z rynku. Aparaty i kamery w smartfonach sprawiły, że niewiele firm produkuje amatorskie kamery cyfrowe, a sprzedaż tych urządzeń spadła drastycznie. Szybko rozwija się jednak rynek uniwersalnych bezlusterkowych aparatów fotograficznych do zastosowań profesjonalnych i półprofesjonalnych. Zarówno Panasonic z serią Lumix GH jak i Sony z A7s wiodą prym wśród producentów urządzeń do rejestrowania wideo. Pozwalają między innymi na nagrywanie 4K 60 klatek na sekundę, czy zewnętrzny zapis w wysokiej jakości kompresji 4:2:2. Do tego dochodzi stabilizacja sensora i cała rodzina obiektywów z szybkim i cichym fokusem.

W ostatnich dniach do Panasonic a Sony dołączył Fujifilm z nowym aparatem X-H1. X-H1 to ten sam sensor o rozmiarze APS-C, znany z X-T2, X-T20 czy X-E3. Podstawowa różnica to stabilizacja matrycy w korpusie, możliwość zapisu wideo o profilu f-log (odpowiednik plików RAW w fotografii) i dodatkowe przyciski, które przydają się podczas sesji wideo. Urządzenie najprawdopodobniej będzie droższe niż model X-T2, ale znacznie tańsze niż średnioformatowy Fujifilm GFX 50S.

Zastosowany sensor jak zwykle ma rozmiar APS-C, wykonany został w technologii X-Trans CMOS III i ma rozdzielczość

24,3Mpix. Zintegrowany 5-osiowy system stabilizacji obrazu ma według producenta skuteczność na poziomie 5,5EV. Usprawniono też sprężynowy mechanizm migawki, który ma być cichszy i powodować mniej drgań. Na górnym panelu znalazł się ekranik pomocniczy, którego estetyka bardzo przypomina te znane z aparatów marki Leica, szczególnie z modelu SL.



MICROSOFT ANDROMEDA

Windows 10 Mobile możemy uznać za twór martwy. Nikt, łącznie z Microsoftem, nie produkuje już smartfonów z tym systemem, a wersja Mobile, mimo ciągłego wsparcia, przestała być aktywnie rozwijana. Czy to koniec mobilnych podrygów firmy z Redmond? Podobno nie.

Co najmniej kilka niezależnych źródeł donosi o Projekcie Andromeda, czyli inicjatywie wewnątrz działu Windows i Surface, której rezultatem ma być stworzenie mobilnego hybrydowego systemu operacyjnego i mobilnego urządzenia ze składanym na pół ekranem. O planowanym oprogramowaniu znamy już wiele szczegółów – nowy Windows będzie składał się z tego samego jądra NT, ale podstawową różnicą będą różne powłoki graficzne, w zależności od trybu, w jakim dane urządzenie pracuje. Do dyspozycji ma być między innymi powłoka mobilna i desktopowa.

Według bieżących informacji system nie pozwoli na uruchamianie starych aplikacji desktopowych Win32 i skupi się na nowej platformie promowanej przez Microsoft, czyli Universal Windows Platform i aplikacjach dostępnych w Sklepie Microsoft. Tylko czy ktoś będzie chciał Windowsa bez aplikacji desktopowych? Firma będzie sama musiała odpowiedzieć sobie na to pytanie i zaproponować odpowiednie rozwiązania.

Według niektórych źródeł urządzenie o kodowej nazwie Andromeda (działające pod kontrolą procesora ARM) umożliwi uruchamianie starszych programów przy pomocy wirtualizacji i dostępu zdalnego. Niestety na tę chwilę nie znamy szczegółów, jak takie rozwiązania miałyby działać.



GALAXY S9 ZBLIŻA SIĘ WIELKIMI KROKAMI

Wizualnie Samsung Galaxy S9 będzie bardzo podobny do swojego poprzednika. Jednak niektórymi funkcjami ma bardziej przypominać iPhone'a X od Apple, a jakością zdjęć na nowo zdefiniować rynek smartfonów.

Kilka zwiastunów dotyczących Samsunga Galaxy S9 pojawiło się już w kanałach społecznościowych tego producenta, a klipy wideo dają niewielki wgląd w to, czego można się spodziewać na premierze nowego flagowca Koreańczyków podczas targów Mobile World Congress w Barcelonie, 25 lutego.

Koreańczycy z Południa nie mogli sobie również darować zaprezentowania własnego odpowiednika Animoji – zabawnych emoji imitujących ruchy twarzy użytkowników iPhone'ów X. Producent zapowiada, że ich nowy model zmieni podejście ludzi do mobilnych aparatów. Zarówno robienie zdjęć w zwolnionym tempie, jak też w sla-

nych warunkach oświetleniowych, nie powinno być problemem dla S9-tki.

Pomoc w tym ma nowy procesor, który bije rekordy w benchmarkach. AnTuTu podaje, że Galaxy S9+ ma wyświetlacz o rozdzielczości Full HD+, a dokładniej 2220x1080 pikseli, co przekłada się na proporcje 18,5:9 i 398ppi. Najpewniej mamy tu jednak powtórkę z Galaxy S8 i to domyślnie ustawiona rozdzielczość z możliwością zwiększenia jej do QHD+, czyli 2960x1440 pikseli, co przełoży się na 531ppi.

Pozostałe elementy specyfikacji, wymienione przez AnTuTu, to 6GB RAM i 64GB pamięci wbudowanej oraz system operacyjny Android 8.0.0 Oreo. W teście procesora i pamięci Galaxy S9+ wykręcił w AnTuTu 265267 punktów i to w wersji ze Snapdragonem 845. Wersja z Exynosem, kierowana na rynki europejskie i koreański, będzie jeszcze szybsza.



APPLE HOMEPOD Z PROBLEMAMI

Od samego początku inteligentny głośnik od Apple miał dużego pecha. Najpierw kilkakrotnie przesuwano datę premiery, później okazało się, że kiepsko współpracuje z głosowym asystentem Siri, aż w końcu zauważono, że silikonowa podstawka zaczyna trwale brudzić drewniane powierzchnie.

Podczas konferencji, na której zaprezentowano HomePoda wszystko wyglądało świetnie. Głośnik wyglądał bardzo estetycznie, menagerowie Apple'a zachwalali jakość dźwięku urządzenia, a współpraca z inteligentnym asystentem Siri zdawała się ustawiać HomePoda w jednym szeregu z głośnikami od Amazona czy Google'a. W praktyce okazało się, że jedynym realnym atutem jest dobre brzmienie. Choć i w tym przypadku analitycy zaczęli zwracać uwagę, że zauważalnie taniej można nabyć głośniki Sonosa, które brzmią jeszcze lepiej.

Braki pod względem funkcjonalności Apple oczywiście potencjalnie może w przyszłości wypełnić poprzez aktualizacje oprogramowania, ale jest jedna cecha, której żadna programowa poprawka nie usunie. Przy okazji pierwszych testów niektórzy recenzenci zauważyli coś bardzo niepokojącego. Otóż postawiony na drewnianym podłożu głośnik Apple HomePod zostawia pod sobą białe ślady. Stwierdzono, że HomePod już po 20 minutach zaczyna na lakierowanym drewnie dębowym lub drewnie potraktowanym olejem (np. mineralnym) zostawiać białe ślady. Apple w odpowiedzi stwierdziło, że ślady powinny same zniknąć w ciągu paru dni. Przedstawiciel firmy (nie wiemy czy żartobliwie) dodał, że jeśli te nie znikną, to... pozostanie na nowo wykończyć powierzchnię drewna.



KONSOLA OD GOOGLE?

Jeśli informacje, na które się natknęliśmy, są prawdziwe, Google może w tym momencie pracować nad sprzętem, który miałby stać się kolejną siłą na rynku konsol, zaraz za PlayStation, Xboxem i Switchem od Nintendo.

Google opracowuje usługę o nazwie kodowej „Yeti”, która wraz z równoległe projektowaną konsolą i kontrolerem, miałaby wyzwać na pojedynek gigantów branży – Sony, Microsoft i Nintendo. Wszystko wskazuje jednak na to, że byłoby to urządzenie bardziej podobne do strumieniującej rozgrywkę Nvidii Shield TV niż do stacjonarnych konsol z przepastnymi dyskami, takich jak PlayStation i Xbox.

Jednocześnie Google prowadziło już rozmowy z twórcami gier, orientując się, czy któryś z nich byłby zainteresowany stworzeniem gry przeznaczonej tylko dla Yeti oraz wykonywaniem portów gier już dostępnych na rynku.

Niedawne zatrudnienie w Google weterana gier wideo, Phila Harrisona mogłoby sugerować, że Google chce zmierzyć się z tą wymagającą branżą, jednak nie powinniśmy się spodziewać niczego tak widowiskowego. Wprowadzenie na arenę kolejnego zawodnika w postaci własnej konsoli stacjonarnej pochłonęłoby ogromne koszty i czas. A przecież Google w dalszym ciągu próbuje przekonać ludzi do platformy wirtualnej rzeczywistości Daydream. Wobec planów związanych z segmentem VR i AR, próba zaistnienia w domach użytkowników usług Google w postaci podłączanej do telewizora konsoli wydałaby się nie tylko krokiem w tył, ale i działaniem wbrew obecnemu rynkowemu trendom.



INTELIAGENTNE OKULARY OD INTELA

Intel, choć stracił wizerunkowo przez afery z Meltdown i Spectre oraz dał się wyprzedzić Samsungowi na rynku półprzewodników, to nadal jest technologicznym gigantem. Stać go nawet na opracowanie swojej wersji okularów rozszerzonej rzeczywistości, podobnych do Google Glass. I robi to w całkiem niezłym stylu.

Inteligentne okulary Intelu nazywają się Vaunt, mają plastikowe ramki i ważą mniej niż 50 gramów, a zatem są cięższe od zwykłych okularów, ale jednocześnie lżejsze od rozwiązań konkurencji. Elektronika wepchnięta jest w zauszniki i steruje laserem o niskiej mocy, który odpowiada za projekcję czerwonego, monochromatycznego obrazu wprost do oka. Rozdzielczość obrazu to maksymalnie 400 x 150 pikseli.



„Laser w oku” może brzmieć odstrasząco, jednak według Intelu jego moc jest tak niska, że nie jest on w stanie w żaden sposób wpłynąć na wzrok, a tym bardziej go uszkodzić. Obraz jest rzucany bezpośrednio na tylną część siatkówki, dzięki czemu nie ma znaczenia, czy jesteśmy dalekowidzami, czy nosimy „minusy”.

Okulary nie są wyposażone w aparat. Ich podstawową funkcją jest wyświetlanie prostych powiadomień, dzięki czemu dowiemy się w dyskretny sposób o nowej wiadomości, nawet bez potrzeby zerkania na smartwatch. W jednym z pokazów, na których prezentowano możliwość smartokularów, użytkownik otrzymywał informacje dotyczące rozmowy podczas rozmowy telefonicznej lub listę zakupów podczas wizyty w sklepie.

PRENUMERUJ za darmo lub półdarmo..

3 pierwsze numery
za darmo
roczna prenumerata startowa

Bezpłatna prenumerata startowa (3 miesiące), powiązana z późniejszą prenumeratą płatną (9 miesięcy). Aby rozpocząć prenumeratę startową, wpłać 108,00 zł tytułem prenumeraty płatnej. Jeśli przez 3 miesiące nie zdołamy Cię przekonać do kontynuacji prenumeraty, możesz wycofać całą swoją wpłatę w terminie 75 dni od jej wniesienia.

do 50% zniżki
za lojalność

Prenumerujesz nieprzerwanie od minimum roku? Przedłużaj prenumeratę ze zniżką lojalnościową (po zalogowaniu na www.avt.pl)

prenumerata		roczna	dwuletnia
jeśli jeszcze nie jesteś Prenumeratorem		132 zł zniżka 8%	
jeśli prenumerujesz nieprzerwanie od:	roku	120 zł zniżka 16%	192 zł zniżka 33%
	2 lat	108 zł zniżka 25%	
	3 lat		168 zł zniżka 41%
	5 lat	96 zł zniżka 33%	144 zł zniżka 50%

Ceny prenumerat standardowych:

prenumerata roczna
1 wydanie gratis
132 zł

prenumerata dwuletnia
8 wydań gratis
192 zł

e-prenumerata roczna
zniżka 15%
87,70 zł

e-prenumerata dwuletnia
zniżka 30%
144,40 zł

prenumerata łączona:
prenumerata papierowa (startowa, standardowa lub ze zniżką lojalnościową) + równoległa e-prenumerata EdW **ze zniżką 80%**
roczna e-prenumerata równoległa 20,60 zł
dwuletnia e-prenumerata równoległa 41,20 zł

Nie lubisz płacić wszystkiego na raz?
Załóż „teczkę” na www.UlubionyKiosk.pl/teczka

prenumeratę zamówisz:

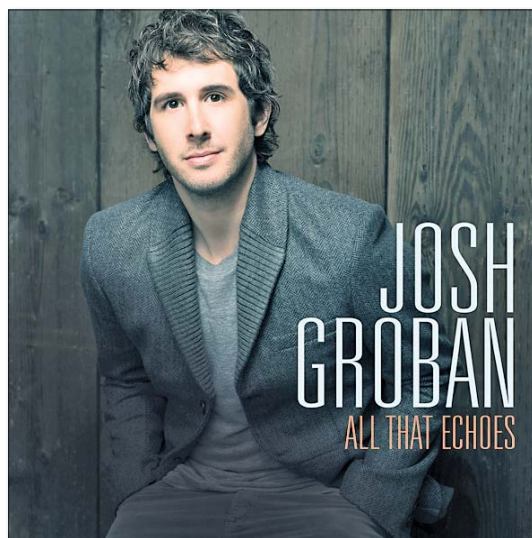
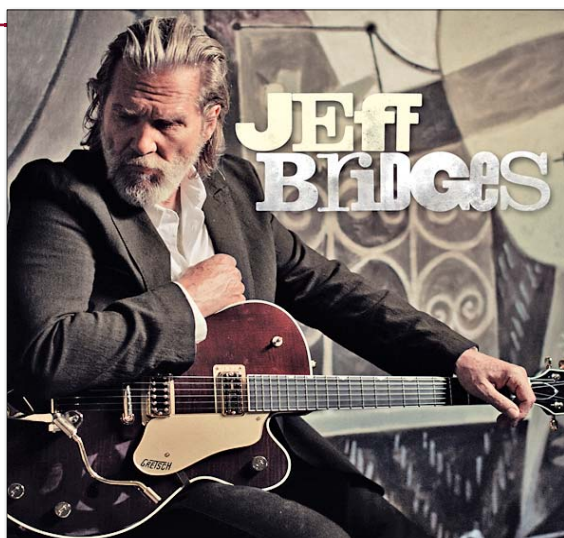
- na www.avt.pl • mailowo - prenumerata@avt.pl • telefonicznie - 22 257 84 22
- wpłacając na konto: AVT Korporacja sp. z o.o., ul. Leszczyńska 11, 03-197 Warszawa, ING Bank Śląski 18 1050 1012 1000 0024 3173 1013

eWydanie dla: **Michał Sas-Uhrynowski (msas)**

Wydanie elektroniczne przeznaczone wyłącznie do użytku własnego bez prawa do rozpowszechniania.

...i korzystaj Z PRZYWILEJÓW

prezent za zaprenumerowanie magazynu



Każdorazowo opłacenie prenumeraty jest premiowane prezentem. W tym numerze są to płyty:

- Jeff Bridges „Jeff Bridges”
- Josh Groban „All that echoes”

Zamów swój prezent mailowo - prenumerata@avt.pl.

jeśli zamawiasz prenumeratę po raz pierwszy lub jeśli zamówisz ją
po zalogowaniu na www.avt.pl, otrzymasz

kody na bezpłatne e-wydania

dowolnych naszych czasopism:

	jeśli przedłużasz prenumeratę	jeśli jesteś nowym Prenumeratorem
krok 1:	zaloguj się na www.avt.pl	zamów prenumeratę EdW na www.avt.pl
krok 2:	przedłuż swoją prenumeratę	utworzymy Twoje konto Prenumeratora
krok 3:	po odnotowaniu wpłaty przyznamy Ci pulę kodów na darmowe e-wydania do wykorzystania na www.UlubionyKiosk.pl (kody będą dostępne po zalogowaniu na www.avt.pl)	

rabaty i gratisy

w Klubie AVT Elektronika

- do 50% zniżki na www.sklep.avt.pl
- do 50% zniżki na www.UlubionyKiosk.pl
- bezpłatne czasopisma dla prenumerujących minimum dwa tytuły Wydawnictwa AVT (szczegóły na avt.pl/klub)

eWydanie dla: **Michał Sas-Uhrynowski (msas)**

Wydanie elektroniczne przeznaczone wyłącznie do użytku własnego bez prawa do rozpowszechniania.

W rubryce „Poczta” zamieszczamy fragmenty Waszych listów oraz nasze odpowiedzi i komentarze. Prosimy o listy dotyczące bieżących wydań EdW, a także o listy z Waszymi komentarzami, propozycjami, problemami, pytaniami, oczekiwaniami względem nas,

z propozycjami tematów do opracowania, itp. Autorzy najciekawszych, wartościowych listów otrzymują upominki, najczęściej w postaci drobnych kitów AVT. Piszcie do nas, bardzo cenimy Wasze listy, choć nie wszystkie prośby możemy zrealizować.

UWAGA! UWAGA!

Potwierdzamy otrzymanie każdego e-maila. Zachęcamy do wykorzystywania opcji: *Żądaj potwierdzenia doręczenia*. Jeśli ktoś nie otrzyma potwierdzenia w ciągu tygodnia, proszony jest o wysłanie swojej wiadomości jeszcze raz – do skutku. A gdyby przypuszczalnym powodem skasowania e-maila przez serwery poczty były potencjalnie groźne załączniki (np. typu .exe, .bas, itp.), bardzo prosimy wysłać informację o tym bez żadnych załączników.

Witam serdecznie.

*Nawiązując do listu Pana Piotra Wyderskiego, bardzo chętnie dalej bym poczytał na temat PLD (w ramach jakiegoś kursu lub chociaż większego artykułu)!
Także apel Pana Wiktora Panka odnośnie do LTSpice pokrywa się z moją prośbą sprzed kilku lat...*

*Pozdrawiam,
Maciej Zimniak*

Dzień dobry

1. Popieram idee artykułów na temat PLD.

Nawet jeśli nie będzie o programowaniu, to porządny materiał teoretyczny byłby mile widziany.

2. Wykorzystanie tranzystorów MOSFET/IGBT do sterowania urządzeniami 230V.

Jeśli tylko mają lepsze właściwości od triaków, to bardzo chętnie bym się o nich dowiedział.

W szczególności, jeśli lepiej sobie radzą z obciążeniem o charakterze indukcyjnym i/lub pojemnościowym.

Przez „lepiej” rozumiem brak potrzeby doboru gasika oraz „pewne” wyłączenie w każdych warunkach.

*Pozdrawiam
Andrzej Telszewski*

Witam serdecznie!

„Elektronikę dla Wszystkich” czytam od bardzo dawna.

Wspaniałą rzeczą, która się wydarzyła, było pojawienie się artykułów na temat zasilaczy impulsowych. Jest to temat dla mnie o tyle ważny, że w pracy zawodowej zajmuję się tego typu zasilaczami, a wiedzy w tym zakresie nie mam zbyt dużej, gdyż nigdy nie miałem potrzeby zajmowania się tymi zagadnieniami.

Co do kursu PLD – kiedyś myślałem o zajęciu się układami FPGA, jednak było to chwilowe zainteresowanie. Myślę jednak, że pojawienie się kursu PLD byłoby dobrym pomysłem. Czasem istnieje potrzeba zrealizowania szybkiego układu typowo logicznego, którego budowa z wykorzystaniem klasycznych układów CMOS czy TTL byłaby trudna, a wciągnięcie do tej pracy mikrokontrolera nie dałoby gwarancji szybkości działania. Programowalne układy logiczne są tu dobrym rozwiązaniem.

*Pozdrawiam serdecznie
Jarosław Tarnawa*

Mówisz – i masz!

W styczniowej Poczcie zamieściliśmy e-mail Piotra Wyderskiego, który zasygnalizował potrzebę i możliwość szerszego przedstawienia układów programowalnych PLD. Autor ewentualnego przyszłego cyklu zdążył napisać artykuł wprowadzająco-zachęcający, z którego treścią zapoznać się na stronach 24–28. O tym, czy szerszy cykl o PLD pojawi się już niedługo na łamach EdW, zadecyduje to, czy oprócz powyższych trzech listów otrzymamy więcej podobnych zgłoszeń. Czekamy więc na Wasze e-maile (edw@elportal.pl), dotyczące zainteresowania szerszym praktycznym kursem programowania PLD.

Dzień dobry

Jestem studentem elektrotechniki w Akademii Morskiej w Gdyni i póki co z elektroniką działam (raczkuję) hobbystycznie. Do zakupu EdW przymierzalem się od jakiegoś czasu, gdy podczas szukania w sieci odpowiedzi na nurtujące pytania, często trafiałem na wzmianki o czasopiśmie lub nawet znajdowałem jego pojedyncze strony.

Do prenumeraty przekonał mnie znaleziony w sklepie pakiet EdW 05+06/2017. Dwa numery czasopisma w okazynnej cenie nie pozwoliły przejść obok siebie obojętnie. Zderzenie z jednym z artykułów poświęconych przetwornicom impulsowym, pióra pana Piotra Góreckiego, oraz serwisowaniu zasilaczy impulsowych, pana Szymona Trygara, które zbiegły się w czasie z moim aktualnym obszarem zainteresowań (a raczej zaległości do nadrobienia jeszcze z uczelni), zaowocowały zrobieniem sobie prezentu noworocznego w postaci prenumeraty czasopisma oraz zakupieniem kilku archiwalnych numerów.

W tym miejscu chciałbym zapytać, czy planowany jest cykl artykułów dotyczących podstaw działania sprzętu audio oraz jego poszczególnych komponentów, jak głośniki dynamiczne, mikrofony pojemnościowe itd.? Czy może jednak mógłbym takie kompendium znaleźć w archiwalnych numerach EdW? Jeżeli tak, proszę redakcję o wskazanie konkretnych numerów.

Odpowiedziałem już na niektóre z aktualnie trwających konkursów i przymierzam się (bardzo) powoli do udziału w Szkole Konstruktorów, gdy tylko nabiorę więcej wprawy w rozumieniu tego i owego...

Szczerze żałuję, że nie zdecydowałem się na prenumeratę EdW wcześniej, bo z pewnością ułatwiłaby mi rozumienie tematu, który tak pokrętnie wyklada się na studiach. Na zakończenie chciałbym życzyć całej redakcji samych sukcesów i wciąż przybywających czytelników zafascynowanych EdW, tak jak ja.

*Pozdrawiam
Mateusz Dziuba*

Spis treści wszystkich numerów EdW od roku 1996 do 2017 można znaleźć pod adresem: <https://elportal.pl/spistresci.zip>. Zachęcamy też do korzystania z prenumeraty EdW i jej licznych zalet oraz do udziału w Szkole Konstruktorów EdW, gdzie każdy numer przynosi zadania z trzech klas, różniące się między innymi stopniem trudności.

*Dzień dobry
W grudniowym numerze EdW, w rubryce Poczta, Pan Sebastian Jarmosiewicz prosi o pomoc dotyczącą minimalizacji funkcji logicznych.*

Jeśli to możliwe, proszę o podanie mi jego adresu e-mail, gdyż chciałbym mu pomóc. Ewentualnie proszę o przekazanie mu informacji, że chętnie wytłumaczę te zagadnienia, jeśli będę potrafił.

*Dziękuję i pozdrawiam
Piotr Gajdosz*

Informację przekazaliśmy.

*Witam,
chciałbym uzupełnić informacje zawarte w artykule ze Skrzynki Porad ze styczniowej EdW. Otóż warto wiedzieć, że oprócz gotowych modułów złożonych z diod i sterownika (WS2812B) na rynku są też dostępne same układy sterownika, zgodne z opisanym w Skrzynce protokołem (np. WS2811B). Jest to istotne o tyle, że na rynku jest dostępnych wiele elementów zawierających same wbudowane diody RGB (w moim przypadku był to przycisk RESET z takim podświetlaniem). Podłączając je do wyjść WS2811B i dołączając układ do już istniejącego łańcucha, uzyskujemy możliwość jednolitego sterowania całością, bez potrzeby tworzenia specjalnych wyjątków dla takich elementów dodatkowych.*

*Pozdrawiam
Piotr Wyderski*

PS. Dziękuję za upominek. Bałwanek bardzo córce się spodobał. :-)

*Witam
Jestem czytelnikiem EdW od ponad 3 lat, dzięki temu moja wiedza o elektronice uległa znacznemu rozszerzeniu. Mam pewną propozycję a zarazem prośbę. Od kilku miesięcy w III klasie Szkoły Konstruktorów ukazują się artykuły dotyczące termiku, obliczania strat cieplnych etc. Temat ten według mnie jest już „przewalkowany” dość solidnie (grono moich znajomych, którzy czytają EdW, też tak uważa). Istniałaby możliwość zmiany tematu zadań III klasy Szkoły? Można byłoby w niej zacząć poruszać tematy związane z przetwornicami DC/DC, aby na podstawie not katalogowych obliczać parametry takich przetwornic, projektować własne? Wtedy czytelnicy EdW mogliby wykorzystać wiedzę z cyklu o przetwornicach.*

*Pozdrawiam
Wiktor Panek*

Elementarne zagadnienia dotyczące radiatorów rzeczywiście są „solidnie przewalkowane”. Ale dotyczą one tej łatwiejszej części problemu chłodzenia. W III klasie Szkoły Konstruktorów mamy jeszcze do przerobienia bardzo ważny temat radiatorów do wzmacniaczy audio. Wbrew pozorom, nie są to sprawy oczywiste, o czym przekonuje choćby bieżący wykład Policz260 (a kluczowe informacje podane będą za miesiąc). Pokrewna sprawa to obliczanie radiatorów do elementów pracujących impulsowo.

Ponadto do omówienia mamy kolejny niełatwy aspekt problemu chłodzenia: współczesne elementy montowane są powierzchniowo i wtedy funkcję radiatora, a przynajmniej pośrednika, pełni płytka drukowana. Ten bardzo aktualny i praktyczny temat też wymaga szerszego omówienia, tym bardziej że dotyczy parametrów podawanych we współczesnych katalogach.

Trudno dobrać materiał do publikacji, żeby zadowolić wszyst-

kich Czytelników. Odnosnie do tematyki radiatorów jeden z uczestników konkursu *Co tu nie gra?* dodał do nadesłanego rozwiązania poniższy dopisek:

*Ps. Przy okazji chciałbym Panu Redaktorowi bardzo podziękować za cykl odcinków o radiatorach w cyklu **Policz**. Jest to bardzo wartościowe źródło informacji dla mnie o kwestiach termicznych (które staram się wykorzystywać w praktyce), w zasadzie nie znalazłem nic, co mogłoby dorównać temu, poza artykułami w internecie, które również są Pana autorstwa.*

Dzięki!!!!!!

Czasami w EdW (która jest reklamówką) uda się coś zamieścić... (zwykle termometr). Gdzie są czasy, gdy w EP było 7 pomysłów w jednym piśmie (reklama na okładce).

(Czy są starsi prenumerujący??) jestem ur. w 1945, posiadam wszystkie numery (...) wnioski: Projekt realnie do zrobienia, tzn. jak najmniej SMD. Hobbysta tego nie polutuje, a jak nie robi, to się nie nauczy i się zniechęci. Więcej o programowaniu i mniej reklam. Zatrudnić projektantów, niestety trzeba podnieść cenę pisma (mniej reklam) ... (płaciłem za tranzystor TG5 tyle, ile kosztował chleb, 8 zł).

bronekp

Są to fragmenty e-maila przysłanego na przełomie roku do Działu Prenumeraty AVT. Skupiają się tu jak w soczewce opinie starszych wiekiem elektroników. W związku ze zmianami, jakie nastąpiły we wszystkich dziedzinach elektroniki, nie wrócą lata, z sentymentem wspominane przez mniej młodych Czytelników (niestety da jednych, na szczęście dla innych).

Witam

Chciałbym zapytać, czy możecie mi pomóc w odszukaniu oprogramowania. Zakupiłem czujnik poziomu cieczy AVT 2822. Mam tam informację, że programowanie znajduje u Państwa. Niestety nie udało mi się znaleźć na Waszej stronie. Bardzo proszę o informacje w tym temacie lub wskazanie miejsca, gdzie się znajduje.

Zbigniew

Dzień dobry

W numerze 02/2016 EdW został opisany projekt mikrooomierza akustycznego. Niestety nie wszedł on do produkcji i nie mogę go kupić. Czy jest możliwość otrzymania rysunku ścieżek do termotransferu?

*Pozdrawiam
Artur*

Przypominamy, że materiały dodatkowe do projektów publikowanych w EdW zamieszczane są w Elportalu na stronie <https://elportal.pl/materiały-dodatkowe/> W pierwszym przypadku chodzi o projekt z kwietnia roku 2007, więc po przewinięciu strony w dół trzeba znaleźć ten numer. Jest tam kilka pozycji do pobrania – między innymi: https://elportal.pl/ftp_05/200704avt2822.zip zawierający dokumentację płytki i program w języku C. W drugim przypadku trzeba przewinąć stronę do numeru 2/2016 i pobrać plik oznaczony *Mikrooomierz 800kB*: https://elportal.pl/wp-content/uploads/2016/02/201602_Mikrooomierz.zip.

Upominki za listy do Poczty otrzymują w tym miesiącu:

Mateusz Dziuba, Piotr Gajdosz oraz bronekp.

Skrzynka Porad

W rubryce przedstawiane są odpowiedzi na pytania nadesłane do Redakcji. Są to sprawy, które, naszym zdaniem, zainteresują szersze grono Czytelników.

Jednocześnie informujemy, że Redakcja nie jest w stanie odpowiedzieć na wszystkie nadesłane pytania, dotyczące różnych drobnych szczegółów.



(...) interesuje mnie nagrywanie i odtwarzanie krótkich fragmentów muzyki lub mowy (...) Kiedyś w EdW były artykuły o mówiących kostkach (...) odtwarzania, ale też nagrywania (...) Czy redakcja mogłaby odświeżyć ten temat? (...)

Rzeczywiście, wiele lat temu przedstawialiśmy układy scalone rodziny ISD, które w tamtym czasie były dużą atrakcją, stanowiąc prościutką odmianę półprzewodnikowego magnetofonu bez taśmy. Od tego czasu wiele się zmieniło, między innymi zupełnie znikły z rynku magnetofony, wyparte przez rozmaite odtwarzacze, magazynujące w postaci cyfrowej informacji o dźwięku w półprzewodnikowej pamięci FLASH. Dziś układy scalone ISD nie są już taką atrakcją z uwagi na postęp techniczny i wprowadzenie nowych elektronicznych sposobów zapisu i odtwarzania dźwięku. Niemniej nadal są dostępne, także w postaci gotowych modułów z kostkami ISD o różnej pojemności, czyli czasie rejestracji. Również w sklepie AVT, gdzie można kupić zarówno same kostki „40minutowe”, jak też moduł:

<https://sklep.avt.pl/modul-isd1820-z-glosnikiem.html>

w skrócie <https://goo.gl/9wi8sg>.

Inny przykład do modułu: www.gotronik.pl/modul-odtwarcza-dzwiekow-isd1760-magnetofon-cyfrowy-p-2396.html

w skrócie: <https://goo.gl/ThJNkh>.

Nie trzeba wglębiać się w szczegóły działania układów scalonych – zakupione moduły są od razu gotowe do użycia, a ich obsługa jest bardzo łatwa.

Należy jednak pamiętać, że układy ISD, z uwagi na zasadę działania, dają dźwięk o niezbyt dobrej jakości, a czas nagrywania jest krótki, co najwyżej rzędu kilku minut. Dlatego w niektórych zastosowaniach trzeba poszukać innego rozwiązania, zapewniającego lepsze parametry. Z tego powodu nie przewidujemy w EdW szerszych materiałów na ten temat.

Trzeba więc wiedzieć, że dostępne są także tanie moduły odtwarzaczy MP3, choćby w sklepie AVT:

sklep.avt.pl/dekoder-mp3-z-czytnikiem-kart-microsd.html

w skrócie: <https://goo.gl/Erboqe>.

To akurat jest moduł tylko do odtwarzania nagrań z karty SD, jednak można kupić moduły wyposażone w mikrofon, pozwalające też na nagrywanie. Przy zakupie bezpośrednio w Chinach (np. Aliexpress) ich ceny okażą się porównywalne lub tylko trochę wyższe niż najprostszymi modułów na krajowym rynku.

Ponieważ takie bardzo tanie moduły mogą się zepsuć, realizując urządzenie, warto kupić na zapas dodatkową sztukę. Gdyby za jakiś czas układ odmówił posłuszeństwa, nie trzeba będzie żmudnie poszukiwać zamiennika, tylko można będzie błyskawicznie wymienić cały moduł na egzemplarz rezerwowany.



Jakie elementy elektroniczne warto wymontować ze starego sprzętu przed jego wyrzuceniem na śmietnik?

Takie i podobne pytania co jakiś czas pojawiają się w e-mailach przysyłanych do redakcji EdW.

Niestety, nie ma dobrej, jedynie słusznej odpowiedzi na tak postawione pytanie.

W takich kwestiach każdy powinien postąpić według swego rozeznania i upodobania. Redakcja EdW nie chce ingerować w te sprawy: nie chce ani namawiać, ani zniechęcać do gromadzenia detali wymontowanych ze starszego sprzętu. Wiele, a właściwie prawie wszystko zależy od nastawienia pytającego oraz jego sytuacji magazynowo-mieszkaniowej. Wiadomo, że istnieje duża grupa elektroników, którzy zbierają wszelkie możliwe „przydasie” i robią to nie z konkretnej potrzeby, tylko „dla Miecia” (no... żeby mieć). Wychodzą oni z założenia, że nie ma potrzeby kupowania, jeżeli pełnowartościowe elementy są dostępne za darmo. Chcą mieć pod ręką nie tylko podzespoły elektroniczne, ale też wszelkie materiały pomocnicze, w tym obudowy i części mechaniczne. W razie potrzeby chcą mieć maksimum potrzebnych części pod ręką, żeby nie jeździć po sklepach i nie szukać tego, co aktualnie potrzebne. Prawie każdy elektronik może ze swego doświadczenia podać przykład, że gdy coś „elektronicznego” wyrzucił na śmietnik, to średnio po dwóch tygodniach okazywało się, iż bardzo by się to przydało. Tak często bywa. Ale praktyka pokazuje, że przytłaczająca większość tak gromadzonych „skarbów” nigdy nie zostanie wykorzystana.

Z drugiej strony istnieje grupa praktykujących elektroników, których zapasy elementów elektronicznych i modułów są niemal zerowe. Osoby takie wychodzą z założenia, że dziś wszystko jest bez problemu dostępne w sklepach internetowych. Uznają, że elektronika jest bardzo tania i oszczędności wynikające z zastosowania starszych, być może gorszych elementów z odzysku są znikome. Ponadto uważają, że lepiej jest cały czas nadażać za aktualnym stanem techniki i wykorzystywać elementy nowoczesne, a nie przestarzałe. Skrajnym przypadkiem jest wykorzystanie jedynie modułów wyposażonych w połączenia stykowe, w ogóle niewymagające lutowania.

Nie trzeba chyba dodawać, że do pierwszej postawy składają się głównie elektrony starsi wiekiem, którzy mają cały czas w pamięci czasy słusznie minione, gdy był kłopot z zakupem zarówno części elektronicznych, jak też obudów i innych detali pomocniczych. Drugą postawę wykazują przede wszystkim młodzi elektrony, którzy znają tylko aktualny stan rzeczy.

Nie można powiedzieć, że któraś z tych postaw jest lepsza. Przede wszystkim trzeba pamiętać, że nie mówimy o profesjonalnym podejściu do elektroniki, tylko o sprawach związanych z uprawianiem hobby. A tu nie ma „jednej jedynej prawdy”. Ogromną, a właściwie najważniejszą rolę odgrywa satysfakcja i zadowolenie.

A co wtedy, gdy ktoś ma pasję polegającą nie na wykorzystaniu, a jedynie na gromadzeniu, magazynowaniu starych podzespołów i czerpaniu radości tylko z posiadania i oglądaniu swoich zbiorów?

Takich skrajnych postaw jest niewiele. Ale niezależnie od swojego „kierunku” uprawiania elektronicznego hobby

i nastawienia do gromadzenia podzespołów z odzysku, warto się chwilę zastanowić. I może coś zmienić w swoich przyzwyczajeniach...

Bo często mamy utarte ścieżki, wręcz głębokie koleiny i nawet do głowy nam nie przyjdzie, żeby z nich wyjść.

Może warto najpierw przeanalizować przeszłość, by decydującym kryterium stało się doświadczenie: czy i kiedy korzystałem ze zgromadzonych elementów albo też kiedy były potrzebne, a ich nie było?

Młodzi, którzy nie mają żadnych zapasów podzespołów i kupują wszystko na bieżąco, mogliby się zastanowić, czy nie przydałby im się zestaw najczęściej używanych elementów czy modułów?

Z kolei osoby gromadzące wszystko, co się da, mogą przejrzeć swoje obfite zbiory i zastanowić się, które detale nie okazały się przydatne w ciągu ostatnich powiedzmy pięciu czy nawet dziesięciu lat?

Czy naprawdę warto je nadal przechowywać?

Odrębną sprawą jest przechowywanie elementów retro, pochodzących sprzed kilkudziesięciu lat. Tu sprawa jest zupełnie inna. Jest szansa, że niedługo znów okażą się bardzo atrakcyjne. Należy się jednak bardzo poważnie zastanowić, czy zamiast rozbierać kilkudziesięcioletnie urządzenia, nie należałoby raczej przechować ich w całości?

Decyzje w kwestii odzyskiwania elementów ze zużytego sprzętu każdy powinien więc podjąć indywidualnie, kierując się nie tyle sentymentem, co rozsądkiem.

(...) Co to są tranzystory do przetwornic Gan i Sic?

Autorowi pytania chodzi o tranzystory polowe MOSFET, po angielsku nazywane: *gallium nitride transistors*, wykonywane z azotku galu, który ma symbol chemiczny GaN oraz tranzystory wykonywane z węglika krzemu (SiC – *silicon carbide*). Zagadnienie jest bardzo szerokie.

Aktualnie coraz częściej w kontekście przetwornic indukcyjnych słyszy się o tranzystorach GaN, mniej o tranzystorach SiC. Temat dotyczy jednak także innych elementów elektronicznych, w tym diod, wykonywanych z półprzewodników o tak zwanej szerokiej przerwie energetycznej (WBG – *Wide Band Gap*). Inny aspekt to fakt, że wysoka jest w nich ruchliwość nośników prądu, przez co określane są skrótowo HEMT (*high electron mobility transistors*). Przerwa energetyczna szersza niż w krzemie i germanie pozwala zrealizować tranzystory wysokonapięciowe. W połączeniu z dużą ruchliwością nośników pozwala to na pracę przy dużych częstotliwościach i na szybkie przełączanie. O ile tranzystory zbudowane z SiC i GaN już od lat stosowane są w układach w.cz., o tyle nowością jest wprowadzanie tranzystorów MOSFET GaN i SiC, działających praktycznie tak samo jak najpopularniejsze krzemowe MOSFET-y z kanałem N, przeznaczonych głównie do pracy w przetwornicach. Przy takich samych rozmiarach mają one dużo mniejszą rezystancję R_{DSon} niż tranzystory krzemowe MOSFET i mniejsze szkodliwe pojemności. Mogą pracować przy większych częstotliwościach, co pozwala budować mniejsze przetwornice o większej sprawności.

100% elektroniki

Prenumerujesz
Elektronikę Praktyczną +
Elektronikę dla Wszystkich?
Skorzystaj z promocji
1 + 1 = 3
i zamów bezpłatną
prenumeratę Elektronika



na www.avt.pl/prenumerata

Smog

Już w ubiegłym sezonie grzewczym media w całym kraju alarmowały, że wielokrotnie zostały przekroczone normy emisji szkodliwych gazów i pyłów do atmosfery. Szczególnie zła sytuacja miała miejsce w dużych aglomeracjach miejskich, gdzie ludzie oddychają mieszaną powietrza oraz gazów, pyłów i substancji szkodliwych. Jednak podobna sytuacja jest także w małych miastach, a nawet w miejscowościach mających status uzdrowiska. Nawet tam zostały przekroczone dopuszczalne normy zanieczyszczeń w powietrzu. Media natychmiast wskazały głównych winowajców takiego stanu rzeczy, czyli przede wszystkim spalanie węgla kamiennego oraz pojazdy z silnikiem Diesla. Polska została wskazana jako kraj Unii Europejskiej, który emituje najwięcej zanieczyszczeń.

Jednak sytuacja nie jest tak oczywista, jak przedstawili to politycy i media, publikując tendencyjne informacje. Na stronie internetowej Międzynarodowej Agencji Energetycznej znajdziemy informacje, że państwem emitującym najwięcej zanieczyszczeń w Europie są... Niemcy. Najnowsze dane są za 2015 rok i wynika z nich, że Niemcy wyemitowały ogółem 1187G (1 gigagram to miliard gramów, czyli milion kilogramów i tysięcy ton) zanieczyszczeń, podczas gdy Polska wyemitowała w tym czasie 714Gg zanieczyszczeń, czyli o 40% mniej¹. Widać to poglądowo na rzucie ekranu na **rysunku 1**. W Internecie znajdziemy także informacje, że Niemcy, pomimo odchodzenia od energetyki jądrowej i przejścia na energię odnawialną, nadal stawiają na energetykę opartą na węglu². Powodem tego jest cena, ponieważ węgiel jest jednym z tańszych źródeł energii.

Skąd zatem smog w czasach, kiedy przemyśl ciężki jest w naszym kraju czynny w mniejszym stopniu niż w czasach poprzedniego ustroju politycznego? Źródłem widocznego smogu może być inwersja temperatury, która powoduje takie zjawiska atmosferyczne jak mgła. Pozornie wygląda to jak zanieczyszczenie powietrza smogiem. Inwersja temperatury może powodować intensyfikowanie się zanieczyszczeń nad obszarami o dużej ich emisji. Nie musi to oznaczać wzrostu emisji zanieczyszczeń, lecz trudność w ich rozproszeniu na skutek inwersji temperatury. Źródeł zanieczyszczeń jest więcej niż tylko spalanie węgla i samo-

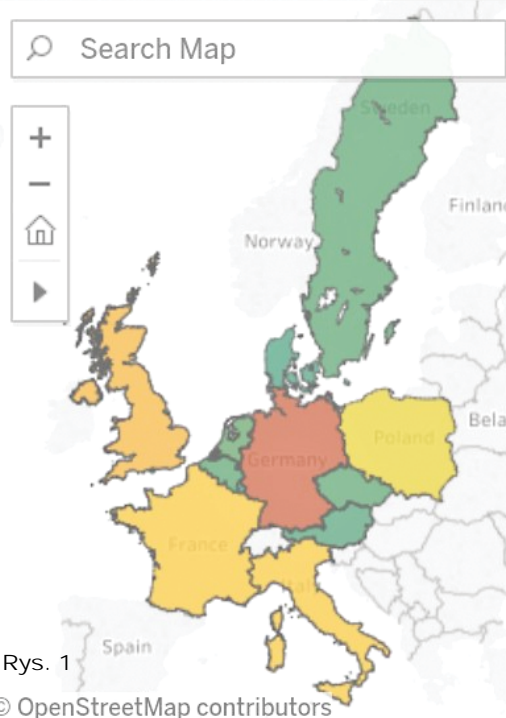
chody z silnikiem Diesla. W transporcie spora część zanieczyszczeń pochodzi nie ze spalania paliw w samochodach, ale ze zużytych klocków hamulcowych i opon. Powstawanie smogu tłumaczy się też teorią chemtrails, według której samoloty cywilne i wojskowe celowo rozpylają w powietrzu różnego rodzaju szkodliwe substancje. Koncepcję chemtrails uważa się za teorię spiskową.

Powodem smogu jest głównie energetyczne i ekonomiczne ubóstwo społeczeństwa, czego nie chcą zauważyć politycy. Przekłada się to na spalanie nie tylko paliw niskiej jakości, ale także śmieci i różnego rodzaju odpadów. Ludzie z przyczyn ekonomicznych, a nie z przekory i przesadnej oszczędności, brudzą się węglem, sadzą i popiołem, spalają wszystko, co się da i jeżdżą starymi samochodami. Wiele osób nie ma też świadomości, że węgiel można spalać w sposób, który nie tylko znacząco zmniejsza emisję szkodliwych substancji, ale pozwala także zaoszczędzić opał. Mowa o tak zwanym górnym spalaniu³. Palenie w tradycyjnym piecu to nie tylko wrzucenie opału do pieca i jego rozpalenie. Taki piec w zależności od jakości opału i sposobu palenia trzeba co kilka dni czyścić z sadzy i pyłów. Na jakość i efektywność spalania ma również wpływ czystość przewodu kominowego. Jeśli zaniedbamy te czynniki, to efektywność spalania spada, a zwiększa się emisja szkodliwych spalin i zużycie opału. Piece wymagają też odpowiedniej obsługi i nie zawsze wystarczającym rozwiązaniem jest sterownik mikroprocesorowy.

Problem spalania śmieci i odpadów można przynajmniej częściowo rozwiązać, stosując system kaucyjny na butelki PET czy inne plastikowe opakowania, ewentualnie wprowadzając punkty skupu surowców wtórnych. Podobne rozwiązania sprawdzają się u naszych zachodnich sąsiadów. Można wprowadzić je również u nas, nie tylko z korzyścią dla środowiska, ale i ludzi.

O tym wszystkim media milczą, prezentują tylko dane na temat smogu i emisji zanieczyszczeń oraz wskazują gotowe rozwiązania. Polegają one na zmianie źródeł energii do ogrzewania domów: zamiast węgla inne paliwa, bardziej ekologiczne, ale i droższe, takie jak gaz, olej opałowy, pompy ciepła czy energia elektryczna. Oczywiście wiąże się to z kosztowną modernizacją istniejących instalacji grzewczych. Politycy obiecują częściowe dofinansowania wymiany pieców na nowe, droższe. Tylko czy ubogie społeczeństwo będzie stać na nowe piece piątej

European
Environment
Agency



Rys. 1

© OpenStreetMap contributors

generacji, przystosowane do spalania często tego samego węgla, ale mające „eko” w nazwie, a co za tym idzie także wyższą cenę? Czy w tak wymienionych nowoczesnych piecach nadal będą spalane śmieci, odpady i opał niskiej jakości? Wszak niektórzy producenci w materiałach reklamowych swoich pieców informują, że mają one możliwość zamontowania ruszta, a tym samym tradycyjnego spalania. Poza tym inne źródła energii, gaz czy olej opałowy, też nie są idealne i emitują w czasie spalania szkodliwe substancje. Nie uwzględnia się również faktu, że gazownice i elektryczne sieci przesyłowe w kraju często są w kiepskim stanie i zwyczajnie nie wytrzymają zwiększonego obciążenia, gdyby nagle większość społeczeństwa zaczęła ogrzewać dom za pomocą gazu i energii elektrycznej. Szczególnie w czasie zdarzających się co jakiś czas zim stulecia. Czy w takiej sytuacji brak alternatywnego źródła ogrzewania spowoduje, że ludzie w swoich mieszkaniach i domach zwyczajnie będą marzli w imię ekologii?

Krzysztof Kawa
kaawa@wp.pl

Źródła informacji:

- 1) <http://tiny.pl/gpd6p>
- 2) <http://tiny.pl/gpd68>
- 3) <http://tiny.pl/gplng>



Alexander Amplifier, czyli stałoprądowy wzmacniacz słuchawkowy



Wzmacniacz „napędzający” słuchawki, funkcjonujący jako niezależne urządzenie lub jako moduł do wbudowania, zwiększa walor użytkowy i stanowi cenne uzupełnienie posiadanej aparatury audio.

Większość popularnych zestawów audio umożliwia dołączenie doń słuchawek elektrodynamicznych. Często realizacja wyjścia słuchawkowego sprowadza się do włączenia rezystorów mocy między końcówkę mocy a gniazdo słuchawkowe, co nie jest rozwiązaniem optymalnym, ponieważ parametry wzmacniaczy o mocach powyżej kilku watów optymalizowane są do sterowania zestawami głośnikowymi. Końcówka mocy współpracując ze słuchawkami za pośrednictwem szeregowych rezystorów pracuje zwykle ze zredukowaną mocą, co pogarsza parametry dynamiczne i szumowe. Poprawę parametrów można uzyskać wzmacniając sygnał liniowy dedykowanym wzmacniaczem słuchawkowym. Na łamach czasopism elektronicznych dość często publikowane są opisy wzmacniaczy słuchawkowych. Mnogość rozwiązań rozciąga się od prostych do bardziej skomplikowanych. Od układów opartych na elementach dyskretnych, lampach, specjalizowanych układach scalonych, układów powalających pomysłową prostotą i innych dziwolągów w postaci „diamentowych tranzystorów”, do wzmacniaczy mających integrator kasujący składową stałą na wyjściu, gdy nie występuje kondensator (kondensatory) wyjściowy itp. Z doświadczenia wiem, że nie istnieje rozwiązanie idealne,

choć rozwiązanie wzmacniacz operacyjny plus wtórnik tranzystorowy wydaje mi się najbardziej „słuszne”. Może warto rozważenia są najprostsze wzmacniacze pracujące w klasie A (tranzystor, źródło prądowe). Przy mocy maksymalnej wyrażanej w mW koszarne straty cieplne, tj. mała sprawność jest do zaakceptowania. Czy do „napędzania” słuchawek warto stosować specjalizowane układy scalone, nie tracąc na jakości? To pytanie bez odpowiedzi...

Jakaś część użytkowników ma „uczulenie na wszystko, co scalone”. Na pewno brakuje wiarygodnego i rzetelnego artykułu (cyklu), prezentującego rozwiązania (różne topologie) wzmacniaczy słuchawkowych z informacjami, co w danej topologii jest zaletą, a co wadą, oraz prezentującego nie tyle rozwiązania mechaniczne słuchawek, co ich cechy czy parametry, jakie warto brać pod uwagę przy ich zakupie. Ale wiadomo, że w dziedzinie audio wiele zależy od subiektywnych odczuć i trudno o 100-procentowo pewne opinie czy stwierdzenia.

Ja uważam, że decydującym kryterium decydującym o zakupie jest odsłuch. Dobrze byłoby, by był to odsłuch podbudowany „teorią” z takiego właśnie artykułu. Słuchawki elektrodynamiczne mimo zasady działania identycznej jak głośnik pracują w innych warunkach (przy mniejszych napięciach, natężeniach prądu). Inne są zatem wymagania co do wzmacniacza i nie jest to tylko problem skali. Inaczej odczuwa się szum z oddalonego głośnika, a inaczej ze słuchawki przy uchu. Rodzaj obudowy słuchawki ma także istotne znaczenie. Czy słuchawki elektrostacyjne można

potraktować tylko jako kosztowną ciekawostkę? Słuchawki trzeba jakoś napędzać, a często spotykane rozwiązanie, polegające na dołączeniu ich do kilku- czy kilkunastowatowej końcówki mocy (przez rezystory) przeznaczonej do „sterowania” głośników, wydaje się mało optymalne i niecelowe.

W moim wzmacniaczu zastosowałem nietypową, nieczęsto spotykaną topologię. Subiektywnie na ucho odczuwa się, że osiągnięte parametry są dobre (może dlatego, że w konstrukcję wzmacniacza włożyłem dużo serca). Mam niezbyt duże doświadczenie w budowaniu układów audio, choć dobrze radzę sobie ze wzmacniaczami operacyjnymi i wzmacniaczami mocy opartymi na układach scalonych. Budowa „słuchawkowca” była dla mnie lekcją, z której dużo się nauczyłem. Zniekształceń nieliniowych wzmacniacza nie mierzyłem, ponieważ zintegrowana karta dźwiękowa mojego komputera nie dałaby miarodajnych wyników. Zdaję sobie sprawę, że rozróżnienie na ucho zawartości harmonicznych 0,01%..0,1% nawet dla zdrowego człowieka jest zupełnie niewykonalne. Zwykle ich zawartość podawana przez producentów sprzętu jest jedynie czynnikiem „marketingowym”, zresztą powszechnie chwalone wzmacniacze lampowe mają zaskakująco duże zniekształcenia. Pasma przenoszenia mojego wzmacniacza mogłoby być znacznie mniejsze. W tym przypadku wyszedłem z założenia, że stosując wzmacniacz operacyjny z górnej półki (wiem, że są jeszcze szybsze np. ze sprzężeniem prądowym, tyle że albo kosztowne lub nieosiągalne i chyba ich użycie byłoby niecelowe w takim układzie), należy wykorzystać jego mocne strony tj. szybkość narastania napięcia wyjściowego (SR) do uzyskania dobrej odpowiedzi impulsowej. To w praktyce sprowadziło się do ograniczenia pasma



od góry tak, by sygnał prostokątny nie powodował oscylacji przy osiągnięciu wartości szczytowych (aby nie było podbicia na krańcu pasma). Co do stałoprądowego toru sygnałowego, to powiedzmy, że taki jest mój kaprys. Jednak powszechnie wiadomo, że wzmacniacze i przedwzmacniacze stałoprądowe cieszą się zainteresowaniem wielu audiofilów. Zastosowanie kondensatorów wejściowych i wyjściowych, odcinających składową stałą, nie miałyby istotnego, zauważalnego wpływu, prócz zawężenia pasma od dołu, nierejestrowanego na nośnikach audio. Zastosowanie kondensatorów zwolniłoby także od dbałości o jak najmniejszą składową stałą na wyjściu – cóż, przyjmijmy, że jest to kaprys amatora konstruktora. Jednak w przypadku zastosowania kondensatorów elektrolitycznych pojawiłby się problem z ich polaryzacją przy zasilaniu symetrycznym. Być może eksperci stwierdzą, że mój wzmacniacz „wyszedł” zwyczajny, ale mam satysfakcję że to mój wzmacniacz i jego budowa była dla mnie kolejną porcją praktycznych doświadczeń.

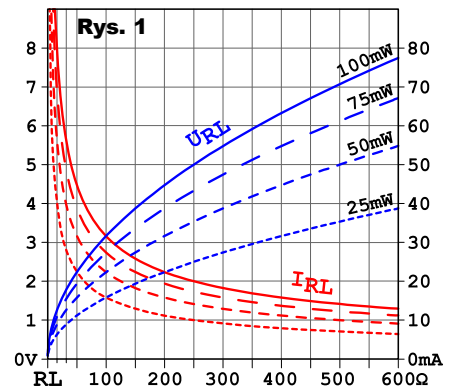
Wzmocnienie napięciowe, choć niekonieczne dla słuchawek niskoimpedancyjnych, w prezentowanym układzie jest zapewnione (6,8x≈16dB) i to nie tylko z powodu zagwarantowania wystereowania słuchawkom wysokoimpedancyjnym, ale też z przyczyn związanych z zastosowaniem szybkiego OP37, który nie może pracować przy wzmocnieniu równym jedności. Parametry szumowe, w szczególności stosunek sygnał/szum też nie był mierzony, ale znowu na ucho wydaje się bardzo dobry, choć przypuszczam, że to zasługa wzmacniacza OP37 i odpowiednio małych wartości rezystorów polaryzujących jego wejścia. „Wysyłając” słuch, można usłyszeć szum potencjometru regulacji głośności przy zmianie położenia jego suwaka na węglowej ścieżce, ale tylko przy braku dołączonego sygnału, w czasie trwania ciszy. Potencjometr, ten „element standardowy”, jest najsłabszym ogniwem w układzie. Po wymianie poten-

cyjometru na taki z wyższej półki i ewentualnie mniejszej rezystancji, efekt ten powinien wyraźnie osłabnąć.

Ale w praktyce regulację głośności przeprowadza się przy dołączonym sygnale audio, który całkowicie maskuje takie delikatne szumy. Ponadto proponowany układ to końcówka mocy i ważne jest, by jak najmniej szumiące było dołączone do niej źródło sygnału (ewentualnie pierwszy stopień przedwzmacniacza). W każdym razie niski poziom szumów w „słuchawkowcu” jest istotniejszy niż we wzmacniaczu głośnikowym.

Stosunkowo niska impedancja wejściowa wzmacniacza podyktowana jest tym, by do niesymetrycznego bądź co bądź wejścia toru sygnałowego nie przedstawiał się brum sieciowy przy niepodłączonych kablach RCA. Przy włączeniu wzmacniacza słychać cichutkie „pyk”, a przy wyłączeniu sygnał się wycisza i gdy już nie jest słyszalny (po ok. 5s od wyłączenia z sieci energetycznej) słychać cichsze „pyk”. Zjawiska te nie są uciążliwe dla słuchacza (choć bardziej istotne jest, że nie są „uciążliwe” dla przetworników w słuchawkach). Dlatego też nie przewidziałem – nie było potrzeby zastosowania – dodatkowego układu opóźnionego dołączania słuchawek do wejścia wzmacniacza przy włączeniu i natychmiastowego ich odłączania przy wyłączeniu zasilania. Zastosowanie nawet prostego układu sterującego przekażnikiem z napięcia AC transformatora byłoby chyba przerostem formy nad treścią.

Zdecydowałem się na zastosowanie często spotykanego we wzmacniaczach słuchawkowych rozwiązania, tj. słuchawki dołączone są przez szeregowo rezystory, które pełnią funkcję zabezpieczenia przeciwzwarceniowego i „wyrównania” mocy dla różnych słuchawek w szerokim zakresie ich impedancji znamionowej. Jaki wpływ mają na brzmienie, tego nie jestem w stanie rozstrzygnąć. Audiofilskie „mity” o lepszym brzmieniu takiego rozwiązania jakoś mnie nie przekonują, ponieważ zdrowy rozum podpowiada, że nie polepsza to właściwości wzmacniacza (rezystory te nie są objęte pętlą sprzężenia zwrotnego) i zwiększają rezystancję wyjściową wzmacniacza, która według powszechnych opinii powinna

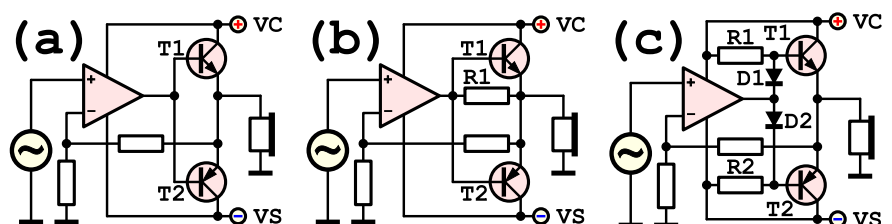


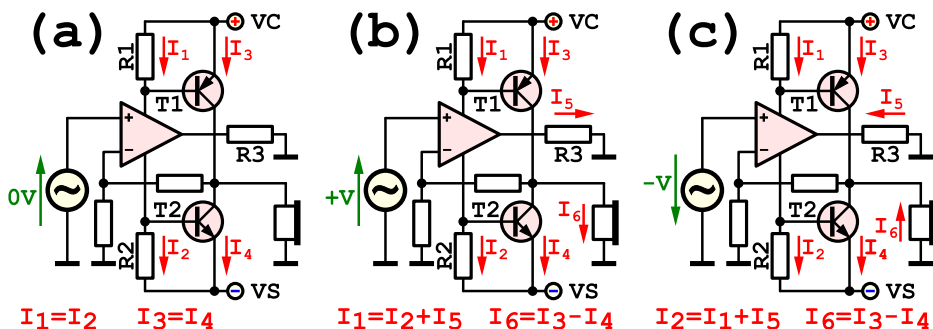
być jak najmniejsza. Z drugiej strony mają na pewno wpływ na właściwości wzmacniacza i jego dynamiczną współpracę ze skomplikowanym układem RLC, jakim są przetworniki słuchawek.

Wzmacniacz słuchawkowy powinien zapewnić poprawne wystereowanie słuchawek, tj. zapewnić dynamikę odsłuchiwanej audycji (dla zakresu spotykanych impedancji znamionowych słuchawek). Na rysunku 1 widać amplitudy napięcia i prądu na słuchawce (przetworniku) potrzebne do uzyskania stałej mocy szczytowej (linie ciągłe dla stosunkowo dużej mocy 100mW), zależnie od jej impedancji znamionowej. Przy wyższych impedancjach słuchawek amplitudy napięcia i prądu można zasadniczo uzyskać wprost z wyjść większości popularnych wzmacniaczy operacyjnych. Przy dołączeniu słuchawek „niskoimpedancyjnych” potrzebna amplituda napięcia ulega zmniejszeniu i wymagane jest zapewnienie większej amplitudy prądu, znacznie przekraczającej wydajność prądową wyjść popularnych wzmacniaczy operacyjnych. Wzmacniacz oprócz zapewnienia napięć umożliwiających wystereowanie słuchawek „wysokoomowych” powinien również mieć odpowiednią wydajność prądową dla wystereowania słuchawek „niskoimpedancyjnych”. Wyłączając stosowanie specjalizowanych układów scalonych, najprostszym sposobem na spełnienie tych wymagań jest użycie wzmacniacza operacyjnego, zapewniającego wzmocnienie napięciowe do sterowania wtórnikiem, buforem tranzystorowym zapewniającym potrzebny prąd. Najczęściej spotykane topologie wzmacniaczy słuchawkowych przedstawia rysunek 2. Typ najprostszy (a) jest nieużyteczny w zastosowaniach



Rys. 2





audio ze względu na nieakceptowalne zniekształcenia spowodowane nieprzewodzeniem tranzystorów przy małych sygnałach. W uproszczeniu: silne sprzężenie zwrotne jest w stanie zredukować zniekształcenia, lecz wymagane do tego jest znaczny iloczyn pasma i wzmocnienia GBW oraz duża szybkość narastania napięcia wyjściowego SR wzmacniacza. Lepsza jest topologia (b), która ze względu na prostotę jest często spotykana w mniej wymagających zastosowaniach. Przy małych sygnałach prąd obciążenia zapewnia wzmacniacz operacyjny, a gdy sygnał ma większy poziom, wtedy na R1 powstaje spadek napięcia otwierający tranzystory wtórnika, które zaczynają dostarczać prąd do obciążenia. Najlepsze rezultaty zapewnia wzmacniacz (c), gdzie wstępnie spolaryzowane tranzystory końcowe zapewniają przepływ określonego prądu spoczynkowego. Różnie mogą wyglądać obwody ich polaryzacji, na przykład można zastosować źródła prądowe zamiast rezystorów R1, R2, a zamiast diod D1, D2 obwód polaryzujący z „potencjometrycznym tranzystorem”. Niemniej wadą tej topologii jest moc strat ciepłych w spoczynku przy ustawionych relatywnie dużych prądach

spoczynkowych. Opisana Rys. 3 w następnym śródtytułe koncepcja układowa prezentowanego wzmacniacza jest inna i bazuje na układzie nazywanym po angielsku Alexander Amplifier (od nazwiska twórcy). Podział urządzenia na trzy płytki drukowane, tj. płytkę wzmacniacza, zasilacza i wskaźnika przesterowania, umożliwia budowę niezależnego urządzenia lub wbudowanie samego wzmacniacza do posiadanego sprzętu audio. Wzmacniacz ma stałoprądowy tor sygnałowy. Oznacza to, że pasmo przenoszenia zaczyna się od zera – brak kondensatorów odcinających składowe stałe na wejściach i wyjściach. Nadmiar mocy, na który składa się wydajność prądowa i wzmocnienie napięciowe przy zasilaniu symetrycznym $\pm 15V$, sprawia, że wzmacniacz jest w stanie wysterować nawet „toporne” słuchawki o małej skuteczności.

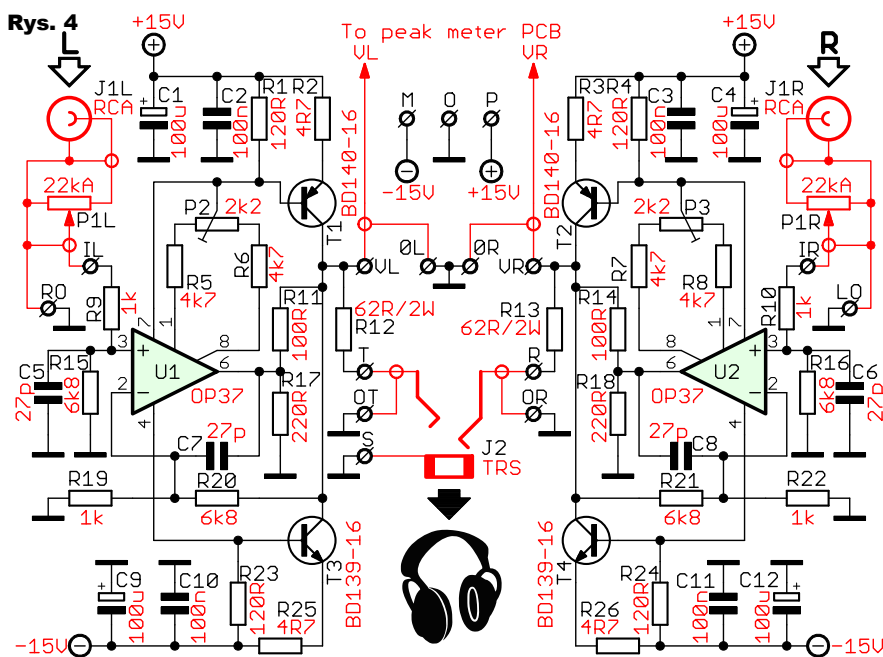
Opis układu

Zasada działania wzmacniacza przedstawiona jest na rysunku 3. Można tu zauważyć nietypowe sterowanie tranzystorami końcowymi. W spoczynku (a), gdy wejście nieodwracające wzmacniacza operacyjnego jest na potencjale masy,



spoczynkowy prąd zasilania wzmacniacza operacyjnego ($I_1=I_2$), przepływając przez odpowiednio dobrane rezystory R1 i R2, powoduje powstanie na nich spadku napięcia, skutkując częściowym otwarciem tranzystorów T1, T2 i przepływem niewielkiego prądu spoczynkowego ($I_3=I_4$). Dzielnik pętli ujemnego sprzężenia zwrotnego sprawia, że wzmacniacz pracuje w konfiguracji nieodwracającej, gdzie „stara” się utrzymać napięcie na wejściu odwracającym równe napięciu na wejściu nieodwracającym, czyli napięciu sygnału wejściowego. Brak różnicy potencjałów na wyprowadzeniach słuchawki oznacza brak przepływu prądu. W przypadku pojawienia się dodatniego napięcia wejściowego (b) z wyjścia dążącego do równowagi wzmacniacza operacyjnego znacznie wypływać prąd I_5 , zwiększając o swą wartość prąd I_1 , co powoduje zwiększenie spadku napięcia na R1. Większe napięcie na R1 oznacza większe otwarcie T1. Wzrost wartości prądu I_3 przy niezmiennym prądzie I_4 umożliwia powstanie wyższego potencjału na kolektorach T1 i T2 i przepływ prądu I_6 przez obciążenie. Dla ujemnego napięcia wejściowego (c) sytuacja jest odwrotna, tj. prąd I_5 wpływa do wzmacniacza i zwiększa o swą wartość prąd I_2 , zwiększając spadek napięcia na R2, co oznacza większe otwarcie T3 i przepływ prądu I_6 przez przetwornik w drugą stronę. Nietypowe sterowanie tranzystorów wyjściowych ma na celu w głównej mierze zmniejszenie mocy strat wzmacniacza zamienianej na ciepło, gdy jest on w spoczynku.

Rysunek 4 przedstawia schemat ideowy wzmacniacza. Wizualną analizę zasady działania wzmacniacza ułatwia rysunek 5, gdzie widoczne są przebiegi w jego kluczowych punktach (analiza przy obciążeniu 16Ω i 600Ω). Wykresy (a) dotyczą



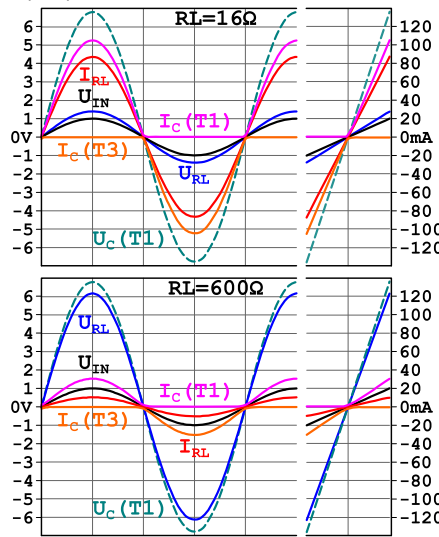
rezystorów „spadkowych” R17 i R18 o wartościach 120Ω, które dają prąd spoczynkowy 2mA. W prototypie zdecydowano się jednak na pracę z większym prądem spoczynkowym, około 20mA, wyznaczonym wartością 180Ω rezystorów „spadkowych”, co pokazują wykresy (b).

Decydujący wpływ na redukcję zniekształceń nieliniowych do akceptowalnego poziomu ma silna pętla ujemnego sprzężenia zwrotnego redukująca wzmocnienie napięciowe wzmacniacza, przy sporym iloczynie wzmocnienia i pasma przenoszenia wzmacniacza operacyjnego GBW=63MHz. Oba kanały wzmacniacza mają niezależne odprężanie szyn zasilania kondensatorami C1...C4, C9...C12. Ponieważ oba kanały są identyczne, poniższy opis dotyczy jednego z nich. Wybór typu wzmacniacza operacyjnego U1 (OP37) sterującego buforem prądowym T1, T3 podyktowany jest małym dryfem temperaturowym napięcia niezrównoważenia, co jest istotne w stałoprądowym torze sygnałowym. Duża wartość GBW, zapewniająca nadmiar wzmocnienia, wymagana do redukcji zniekształceń przez pętlę sprzężenia zwrotnego (szerokość pasma przenoszenia przy ustalonym wzmocnieniu). Mniejsze znaczenie ma jego optymalizacja szumowa do zastosowań audio przy współpracy z źródłami sygnału o rezystancji w zakresie od 500Ω do kilkudziesięciu kΩ. Ze względów „szumowych” popularna kostka TL071 równie dobrze sprawdziłaby się w układzie, jednak jej GBW i szybkość narastania napięcia wyjściowego (SR) zdecydowanie przemawia na jej niekorzyść w stosunku do OP37. Zastosowanie OP37 korzystnie przekłada się na szybkość wzmacniacza, a tym samym na jego odpowiadając impulsową.

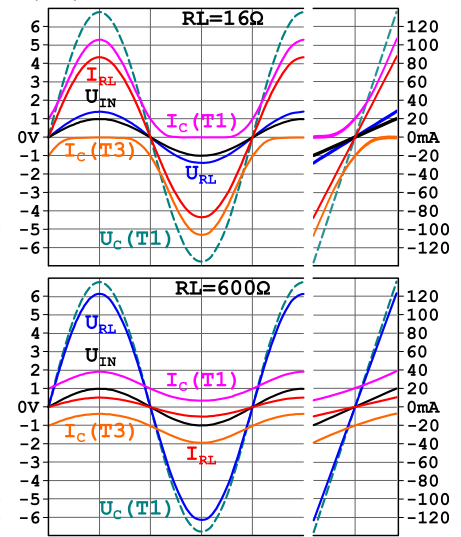
Liniowy sygnał audio ze źródła dołączanego do gniazda J1L (RCA) trafia na tłumiący potencjometr regulacji głośności P1. Dzięki dołączeniu szeregowo połączonych R9, R15 między suwak P1 (o charakterystyce liniowej) a masę regulacja głośności jest nieliniowa i zbliżona do stosowanej w takich zastosowaniach charakterystyki typu B. Rezystor R15 polaryzuje wejście nieodwracające U1, a R9 z C5 tworzą filtr tłumiący składowe sygnału o częstotliwościach radiowych, mogące indukować się w kablach przyłączeniowych. Dzielnik R19, R20 stałoprądowej pętli ujemnego sprzężenia zwrotnego wyznacza wzmocnienie napięciowe ($K=R20/R19+1$) równe 7,8x. Ponieważ sygnały wejściowe są osłabiane dzielnikiem R9, R15 sumaryczne wzmocnienie napięciowe wzmacniacza wynosi 6,8x ($\approx 16\text{dB}$).

Wyjście wzmacniacza ma charakter źródła prądowego, więc konieczne jest

(a) $R_1, R_2=120\Omega$



(b) $R_1, R_2=180\Omega$

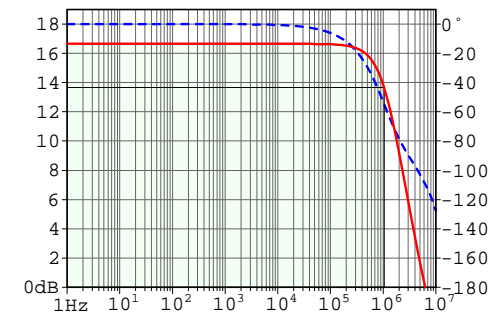


Rys. 5

sprężenie stałoprądowe między wyjściem U1 i wyjściem wzmacniacza. Jest ono realizowane za pomocą rezystora R11, bez którego układ stałby się generatorem pracującym w zakresie częstotliwościach ponadakustycznych. Kondensator C7 ogranicza pasmo przenoszenia i wyrównuje charakterystykę w zakresie częstotliwości ponadakustycznych, tłumiąc podbicie na końcu pasma przenoszenia. Tak uzyskana górna granica pasma przenoszenia (-3dB) jest zbliżona do 1MHz, co ilustruje rysunek 6. Rezystory emiterowe R2, R25 eliminują niekorzystną zależność prądu spoczynkowego tranzystorów T1, T3 (ich punktu pracy) od temperatury, minimalizując dodatnie sprzężenie termiczne.

Potencjometr P2 służy do „kasowania” szczałkowego wychylenia membrany przetwornika spowodowanego przepływem składowej stałej prądu, wynikającej ze „wzmacniania niedoskonałości” użytych wzmacniaczy operacyjnych, tj. napięcia niezrównoważenia oraz napięć wynikających z różnic prądów polaryzacji wejść.

Choć zastosowany wzmacniacz operacyjny charakteryzuje się małym napięciem niezrównoważenia (typ. 10uV), to możliwy rozrzut produkcyjny, do 100uV, zależnie od wersji, przy stałoprądowym sprzężeniu zwrotnym, w najmniej korzystnym przypadku zaowocuje pojawieniem się na wyjściu wzmacniacza napięcia stałego o wartości $100\text{uV} \cdot 7,8x = 0,78\text{mV}$. Już tak niewielkie napięcie spowoduje przepływ prądu, odchylenie membrany od położenia neutralnego i teoretycznie może mieć negatywny wpływ na jakość „wytwarzanego” dźwięku. W tym przypadku zastosowanie spotykanego we wzmacniaczach słuchawkowych aktywnego układu, tzw. DC serwa, zwykle jest to układ integra-



Rys. 6

torą, służącego do aktywnego kasowania składowej stałej na przetworniku, nie jest konieczne. Zmiany prądów płynących z/do wyjścia U1 nie przekraczają $\pm 2\text{mA}$ – suma prądów płynących przez wzmacniacz zmienia się od 3mA do 5mA. Dryft temperaturowy, spowodowany nagrzewaniem struktur, można uznać za nieistotny przy typowym dla OP37 dryfie temperaturowym napięcia niezrównoważenia równym $0,2\text{uV}/^\circ\text{C}$.

Rezystor R12, tworzący dzielnik napięciowy z rezystancją obciążenia, prócz zabezpieczenia przeciwzwarciowego odpowiedzialny jest za „wyrównanie” mocy wyjściowej wzmacniacza dla słuchawek o impedancji znamionowej z zakresu 16...600Ω. Wykres mocy wyjściowej w funkcji rezystancji obciążenia dla sygnału wejściowego o amplitudzie 1V ($0,71\text{RMS}$) widoczny jest na rysunku 7. Moc tracona wydzielana w pojedynczym tranzystorze T1 i T3 nawet podczas zwarcia wyjścia do masy nie przekracza 1W, co jest w stanie rozproszyć obudowa TO126 bez radiatora. Wzmacniacz daje się wysterować blisko napięć zasilania i pomimo wymaganych na wyjściu niedużych amplitud, zdecydowano się na zasilanie „standardowymi” w aparaturze audio napięciami $\pm 15\text{V}$. Dzięki temu



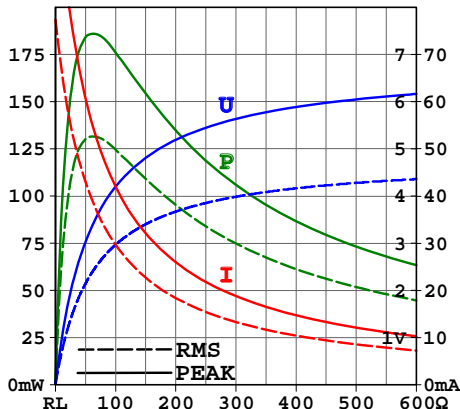
waniu (zielony) wzmacniacza. Układ U1 (LM339) realizuje dwa komparatory okienkowe odpowiednio U1A, U1B dla kanału lewego i U1C, U1D dla prawego. Bezpośrednie połączenie wyjść poszczególnych komparatorów jest możliwe dzięki wyjściom typu otwarty kolektor układu U1. Wskaźnik reaguje na przekroczenie napięcia w obu kanałach, zarówno dla dodatniej, jak i ujemnej połówki sygnału ($\pm 7V$). Przeznaczone do monitorowania sygnały z płytki wzmacniacza doprowadzono do kółkow...

w przypadku współpracy urządzenia ze słuchawkami wysokoomowymi o małej skuteczności (czułości) możliwe jest zwiększenie wzmocnienia napięciowego (amplitudy napięcia na słuchawce) przez wymianę rezystorów R20, R21 na odpowiednio większe.

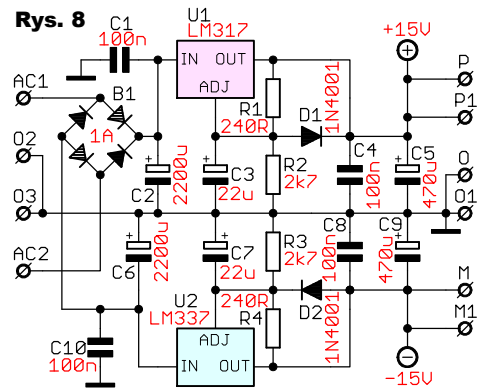
Zasilanie zapewnia symetryczny zasilacz stabilizowany, którego schemat przedstawiono na **rysunku 8**. Klasyczna aplikacja stabilizatorów U1, U2 o zmniejszonej wartości tętnień nie wymaga szerszego komentarza. Mostek Graetzta B1 wymagany jest przy zasilaniu układu transformatorem o dwóch uzwojeniach wtórnych, czyli z uzwojeniem wtórnym z odczepem. Przy zasilaniu z transformatora o jednym uzwojeniu wtórnym zamiast mostka B1 należy włutować dwie diody prostownicze, np. 1N4001, zaznaczone na schemacie „kreskowym” deseniem. W takim przypadku jeden koniec uzwojenia wtórnego należy dołączyć do punktu O2, drugi koniec do punktów AC1 i AC2. Nadmiarowo szacowana moc zasilającego transformatora wynosi około 7VA.

Schemat wskaźnika przesterowania widoczny jest na **rysunku 9**. Funkcję optycznego wskaźnika pełni dwustrukturowa LED D1. Jej świecenie sygnalizuje obecność napięć zasilania, kolor świecenia informuje o przesterowaniu (czerwony) bądź nieprzesterowa-

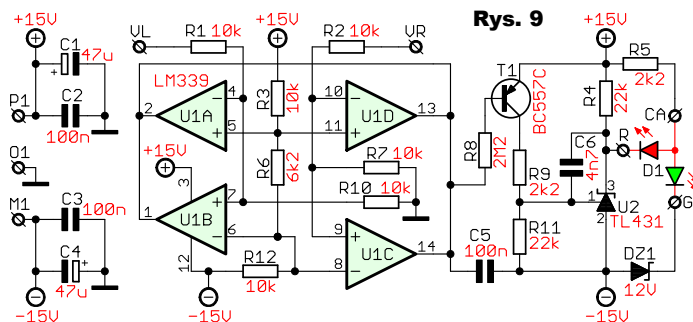
Rys. 7



ni (zielony) wzmacniacza. Układ U1 (LM339) realizuje dwa komparatory okienkowe odpowiednio U1A, U1B dla kanału lewego i U1C, U1D dla prawego. Bezpośrednie połączenie wyjść poszczególnych komparatorów jest możliwe dzięki wyjściom typu otwarty kolektor układu U1. Wskaźnik reaguje na przekroczenie napięcia w obu kanałach, zarówno dla dodatniej, jak i ujemnej połówki sygnału ($\pm 7V$). Przeznaczone do monitorowania sygnały z płytki wzmacniacza doprowadzono do kółkow...



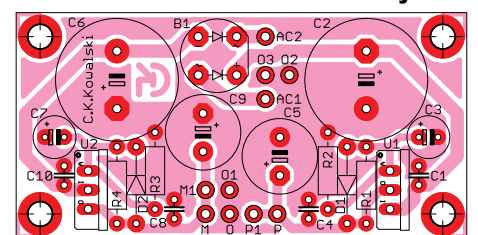
ture LED D1, realizując tym sposobem detektor szczytu. Po naładowaniu C5 prąd kolektora T1 przestaje płynąć. Napięcie na wejściu referencyjnym U2 jest mniejsze od wewnętrznego napięcia odniesienia (V_{REF}) i na katodzie panuje napięcie zbliżone do plusa zasilania. Zaświecona zostaje zielona struktura D1, natomiast czerwona przestaje świecić. Wymagany do prawidłowego zasilania U2 prąd ($>1mA$) zapewnia rezystor R4. W trakcie uruchamiania okazało się, że w czasie przełączania struktur LED D1 powstałe na katodzie U2 oscylacje przenikają do toru sygnałowego. Aby temu przeciwdziałać, zastosowano kondensator C6 dolutowany od spodu PCB. Stąd rysunek PCB (z uwzględnionym C6) wskaźnika przesterowania różni się od jej fotografii.



Rys. 9



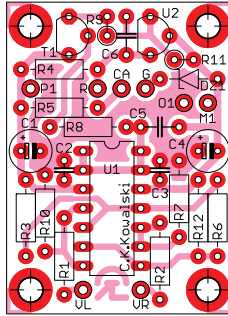
Rys. 10



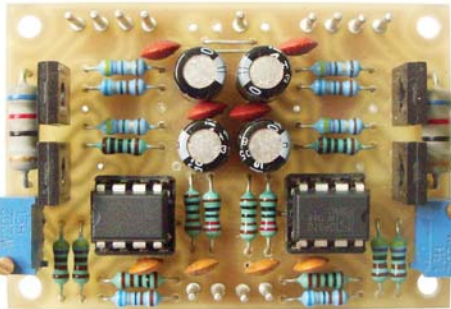
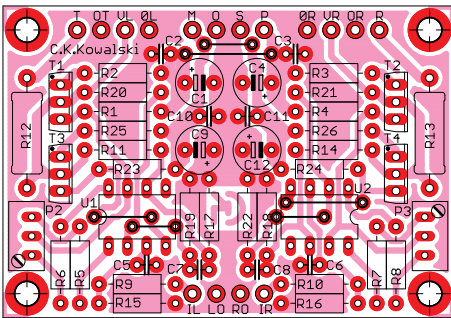
Montaż i uruchomienie

Poszczególne płytki najlepiej montować i uruchamiać w kolejności: zasilacz **rysunek 10**, wskaźnik przesterowania **rysunek 11**, wzmacniacz **rysunek 12**. Montaż elementów po wcześniejszym sprawdzeniu jakościowym PCB należy przeprowadzić w kolejności wyznaczanej przez kryterium gabarytowe lutowanych podzespołów, tj. od najmniejszych do największych. Na płycie wzmacniacza jako pierwsze należy włutować sześć zwór. Uruchomienie zasilacza sprowadza się do pomiaru napięć wyjściowych. W przyłączeniowe punkty lutownicze warto włutować kołki lutownicze. W przypadku bliskiego rozmieszczenia płytek wzmacniacza i zasilacza w obudowie, punkty P, O, M na obu płytkach można połączyć bezpośrednio zworami. Okablowanie urządzenia sprowadza się do połączenia przewodami jednakowo oznaczonych punktów poszczególnych płytek drukowanych. Trzymanie się oznaczeń punktów lutowniczych pozwoli uniknąć szkodliwych pętli mas, tj. ekran przewodów sygnałowych łączony jest z masą jedynie

Rys. 11



Rys. 12



na płycie wzmacniacza (jak na schemacie z **rysunku 4**). Zasada ta nie dotyczy gniazd wejściowych RCA, gdzie ekran ma być dołączony również od strony gniazd, oddzielnie dla prawego i lewego kanału. Wymagana jest izolacja mas gniazd wejściowych RCA i wyjściowego TRS od obudowy, gdy jest ona metalowa.

Przewody będące na potencjale sieci energetycznej 230V należy wykonać w podwójnej izolacji. Styk ochronny PE gniazda sieciowego koniecznie połączyć z obudową (gdy jest metalowa) i masą układu (punkt w PCB zasilacza). Zasilając wskaźnik przesterowania, sprawdzić reakcję LED, zwierając wejścia VR, VL do dodatniej i ujemnej szyny zasilania. Reakcja LED powinna być zgodna z opisem z poprzedniego śródtytułu. Uruchomienie płytki wzmacniacza można przeprowadzić „na ucho”, dołączając do gniazd źródło sygnału audio i słuchawki. Przed dołączeniem słuchawek warto sprawdzić woltomierzem wyjścia wzmacniacza pod kątem występowania składowej stałej. Jedyłą wymaganą regulacją jest korekcja napięcia niezrównoważenia wzmacniacza operacyjnych. Po ustabilizowaniu temperatur półprzewodników i przy zwartych do masy wejściach, potencjometrami montażowymi P2, P3 należy ustawić napięcie na obciążeniu (współpracujących słuchawkach) jak najbardziej zbliżone do zera. Zastosowana w prototypie obudowa pochodzi ze starego napędu CD-ROM (2001r.) z odzysku, gdy producenci nie stosowali jeszcze przetłoczeń usztywniających. Przetłoczenia w nowszych napędów (mniejsza grubość blachy) stawiają pod znakiem zapytania estetykę tak wykonanej obudowy. Ponieważ obudowa ma charakter jednostkowy, nie powstała do niej dokumentacja rysunkowa. Szczegóły obudowy mogące zainspirować Czytelników widoczne są na fotografiach. Kilka dodatkowych fotografii zostało też umieszczonych w Elporcie wśród materiałów dodatkowych do tego numeru EdW. Jak wykazują statystyki, **głośne i długie słuchanie**

Wykaz elementów

Wzmacniacz:

Wszystkie metalizowane o tolerancji 1%

R2,R3,R25,R26	4,7kΩ
R12,R13	62kΩ/2W
R11,R14	100kΩ
R1,R4,R23,R24	120kΩ..180kΩ
R17,R18	220kΩ
R9,R10,R19,R22	1kΩ
R5,R6,R7,R8	4,7kΩ
R15,R16,R20,R21	6,8kΩ
P1	22kΩ-A pot. obrotowy stereo
P2,P3	2,2kΩ pot. montażowy RJ9W
C5,C6,C7,C8	27p ceramiczny
C2,C3,C10,C11	100n ceramiczny
C1,C4,C9,C12	100u/25V
T1,T2	BD140-16
T3,T4	BD139-16
U1,U2	OP37
J1	gniazda 2xRCA
J2	gniazdo TRS φ 6,3mm (1/4")

Zasilacz:

R1,R4	240kΩ
R2,R3	2,7kΩ
C1,C4,C8,C10	100n ceramiczny
C3,C7	22u/25V
C5,C9	470u/25V
C2,C6	2200u/25V
B1	RB-1A lub 2x 1N4001
D1,D2	1N4001
U1	LM317
U2	LM337

Detektor szczytu:

R5,R9	2,2kΩ
R6	6,2kΩ
R1,R2, R3,R7,R10,R12	10kΩ
R4,R11	22kΩ
R8	2,2MΩ
C5	100n foliowy
C6	4,7n foliowy
C2,C3	100n ceramiczny
C1,C4	47u/25V
D1	LED φ 5mm RG wspólna anoda
DZ112V
T1	BC557C
U1	LM339
U2	TL431

Komplet płytek jest dostępny w Sklepie AVT jako AVT3217

audycji przy użyciu słuchawek może powodować znaczne ubytki słuchu, dlatego zalecane jest umiarkowanie (którego życzę Szanownym Czytelnikom) przy ustalaniu głośności potencjometrem regulacyjnym.

Cyprian Kamil Kowalski
c4v2@o2.pl

Zawsze znajdziesz, przejrzysz i kupisz aktualny numer „Elektroniki dla Wszystkich” (zarówno w wersji papierowej, jak i elektronicznej) na www.UlubionyKiosk.pl

Zegar Nixie z „kukułką”

Zegar Nixie, tykający jak klasyczny zegar mechaniczny, wyposażony w kukułkę lub inne zwierzątko sygnalizujące pełną godzinę. Modułowa budowa ułatwia dostosowanie zegara do własnych upodobań.

W dotychczas przedstawionych dwóch częściach artykułu opisane były elektroniczne aspekty projektu.

Całość projektu została wykonana w darmowym środowisku CoIde CoCoX (v 1.7.8). Można je pobrać ze strony: <http://www.cocox.org/>, a dodatkowo należy ściągnąć pliki kompilatora GCC np, z adresu: <https://developer.arm.com/open-source/gnu-toolchain/gnu-rm/downloads>. Po zainstalowaniu oprogramowania uruchamiamy środowisko CoCox CoIde. Możemy wybrać lokalizację, gdzie będzie znajdował się nasz projekt. Następnie musimy ustawić ścieżkę do wcześniej zainstalowanego kompilatora. W tym celu klikamy: „Project”, a następnie „Select Toolchain Path” i wpisujemy przykładowo:

```
„E:\GNU Tools ARM Embedded\62017-q2-update\bin”, w zależności od tego, gdzie mamy zainstalowany kompilator. Teraz przystępujemy do utworzenia nowego projektu. Klikamy Project → New Project i wpisujemy nazwę np. „zegar_3”, klikamy „next”, wybieramy w kolejnej zakładce „chip”, znowu „next”, wybieramy nasz mikrokontroler, czyli w zakładce „ST” szukamy STM32F303VB i klikamy „finish”. Następnie wybieramy zakładkę „Configuration” (jeżeli jej nie widać, to klikamy „View”, i „Configuration”) i zakładkę „Link”. W polu „Linked Libraries” ustawiamy ścieżki do plików „libgcc” i „libm”(przycisk „Add”), przykładowo:
```

```
„E:\GNU Tools ARM Embedded\62017-q2-update\lib\gcc\arm-none-eabi\6.3.1\libgcc.a”
```

```
oraz:
„E:\GNU Tools ARM Embedded\62017-q2-update\arm-none-eabi\lib\libm.a”
```

Z kolei w zakładce „compile” zmieniamy ustawienie z „Not use FPU” na „FPU_hard”.

Teraz w zakładce „Repository” zaznaczamy pola przy następujących peryferiach: RCC, GPIO, DAC, TIM, MISC. Z drzewa projektu po lewej stronie wybieramy i otwieramy plik „main.c”, a następnie usuwamy całą jego zawartość i wklejamy zawartość pliku „main.c” przygotowanego w folderze „plik_źródłowy”. Zawarte tam pozostałe trzy pliki (beep.c, kukulka.c, tick.c) kopiujemy do folderu, gdzie znajduje się nasz nowy plik „main.c”.

Po tych zabiegach kompilacja powinna przebiegać prawidłowo. Aby to sprawdzić, klikamy „Project”, a następnie „Rebuild” i w konsoli sprawdzamy, czy kompilacja zakończyła się sukcesem. Pliki wynikowe znajdują się w lokalizacji: „zegar_3\zegar_3\Debug\bin”. Możemy pliki te wgrywać za pomocą wspomnianego wcześniej programu ST LINK Utility, ale możliwe jest także programowanie z poziomu CoCoxa („Flash→Program Download”), możliwe jest także debugowanie. W przypadku błędów podczas prób programowania należy spróbować podmienić plik „STLinkUSBDriver.dll” znajdujący się w lokalizacji: „CooCox\CoIDE\bin” na zawarty w folderze „ST_LINK_driver” (w materiałach do tego numeru). Teraz możemy przystąpić do modyfikacji programu.

W przypadku, gdy po uruchomieniu oryginalnego programu okaże się, że liczba kroków wykonywanych przez silnik krokowy jest za duża lub za mała, należy zmodyfikować deklarację „zakres_wysuniecie” o domyślnej wartości 42. Druga modyfikacja może dotyczyć zmiany dźwięków wydawanych przez zegar i „kukułkę”. Dyskretne wartości sygnałów dźwiękowych są zawarte w jednowymiarowych tablicach: „beep” i „tick”, umieszczonych w plikach o takich samych nazwach. W oryginalnej wersji zegara



„kukułka” jest... owcą, więc i wydawany przez nią dźwięk to beczenie. Aby zmienić rodzaj wydawanego dźwięku, należy zastąpić wspomniane wcześniej pliki innymi. W folderze zawierającym plik „main.c” znajduje się dodatkowo plik „kukulka.c” zawierający tablicę z dyskretnymi wartościami dźwięków kukania i bicia dzwonu. Jeżeli chcemy go wykorzystać, to zmieniamy instrukcję „include” w pliku main.c oraz częstotliwość próbkowania i nazwę wykorzystywanej przez przetwornik DAC zmiennej – jest to opisane w komentarzach w kodzie programu. Jeżeli ktoś chce wykorzystać jeszcze inne dźwięki, to musi najpierw znaleźć lub nagrać odpowiedni plik wav (dużo plików, w tym wykorzystane w oryginalnej wersji zegara, można znaleźć na stronie: www.findsounds.com), a następnie przekonwertować go do postaci tablicy. Pamiętać należy o kilku sprawach. Najważniejsze parametry pliku dźwiękowego w postaci cyfrowej to częstotliwość próbkowania i rozdzielczość. Przykładowo plik „kukulka” zapisany jest z częstotliwością próbkowania 22,050kHz i rozdzielczością 8 bitów. Należy jednak pamiętać, że im plik jest zapisany z wyższą częstotliwością próbkowania, tym jego objętość jest większa, a dostępna ilość pamięci Flash w mikrokontrolerze to 128kB. Rozdzielczość przetwornika DAC zawartego w układzie STM32F303 to 12 bitów. Można wykorzystać pliki zapisane z większą rozdzielczością, ale wtedy obniżeniu ulegnie ich jakość, a także pliki o niższej rozdzielczości – wtedy nie ulegnie ona zmianie. Dźwięki tykania zegara



Fot. 10

i wydawane przez kukułkę nie muszą być zapisane z taką samą częstotliwością próbkowania. Szczegóły dotyczące jej ustawiania zawarte są w komentarzach w listingu programu. W folderze ...**obrobka_plików** znajduje się plik „wav2hex.m” konwertujący plik typu „wav” na plik zawierający tablicę, gotową do wykorzystania w programie mikrokontrolera. Uwaga! Jest on napisany w programie Matlab, który jest trudno dostępny dla większości użytkowników (choć można poszukać darmowej, ograniczonej czasowo wersji). Napisany program najpierw wczytuje plik typu „wav”, odczytuje częstotliwość próbkowania i rozdzielczość, z jaką został zapisany (w bitach), zapisuje te dane w zmiennych: „y” (wektor próbek), „fs” (częstotliwość próbkowania), „bit” (liczba bitów na próbkę). Następnie skaluje odpowiednio wartości (typowy plik „wav” przyjmuje wartości ułamkowe od -1 do 1, a my potrzebujemy, aby wartości zmieniały się od 0 do 4095,



Fot. 11

co zapewnia maksymalne wykorzystanie rozdzielczości przetwornika) i zapisuje w postaci tablicy. Wzorując się na tym programie, można napisać własną aplikację konwertera np. w języku C.

Inne możliwości modyfikacji programu to np. godziny, w których kukułka wyśpiewuje godzinę oraz częstotliwość z jaką zegar wydaje dźwięki tyknięcia. Wykorzystanie wewnętrznego zegara RTC do odmierzenia czasu i tym samym zabezpieczenie przed zanikiem zasilania, wymaga dość istotnej modyfikacji programu, gdyż w obecnej wersji za odmierzenie czasu odpowiadają timery, a po zaniku zasilania następuje reset mikrokontrolera. W związku z tym baterijka nie pełni żadnej funkcji i jeżeli nie chcemy dodawać obsługi RTC, to nie należy jej montować.

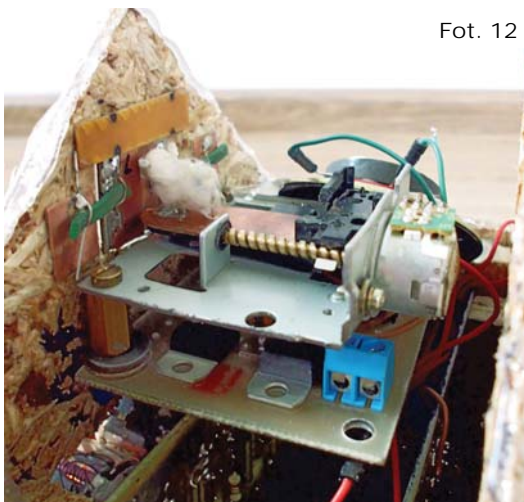
Nie mniej ważną sprawą są kwestie wykonania obudowy i mechanicznego montażu elementów zegara np. drzwiczek kukułki. W wersji pokazanej na **fotografii 10** obudowa wykonana jest z płyty paździerzowej, daszek z panelu podłogowego, drzwiczki z cienkiego laminatu z przylutowanymi „zawiasami”, wykonanymi z obciętych końcówek elementów, zamykają się samoczynnie dzięki gumce recepturce. Z kolei owieczka wykonana została z... kondensatora elektrolitycznego z przylutowaną diodą LED 3mm, pokrytych wata. Pokazują to **fotografie 11...13**. Przyciski sterujące i potencjometr siły głosu umieszczono z tyłu urządzenia.

Na koniec wskazówki dla tych, którzy chcieliby płytki drukowane wykonać samodzielnie: dwie z trzech płytek drukowanych wykonane są jako dwuwarstwowe, zatem ich zrobienie może sprawić problem początkującym. Szczególnie trudna może okazać się płyta główna, która zawiera mikrokontroler

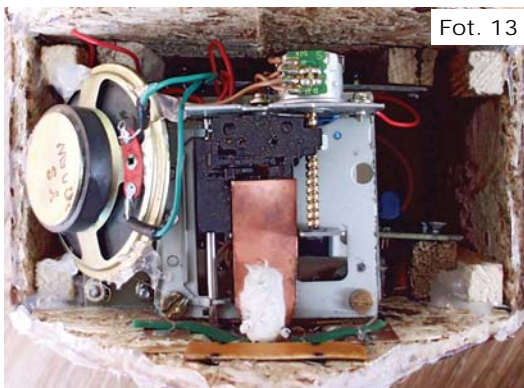


w obudowie TQFP64 i musi być wykonana precyzyjnie. Jest to jednak możliwe w warunkach domowych – amatorskich, metodą termotransferową („żelazkową”) i bardzo niskim kosztem. Taki też sposób wykorzystałem przy konstrukcji pierwszej wersji urządzenia. W tym celu zresztą zastosowane zostały spore otwory na przelotki, aby można je było wykonać standardowym wiertłem bez obawy uszkodzenia miedzi.

Ponieważ tematem artykułu nie jest metodyka wykonywania płytek drukowanych, ograniczę się do kilku wskazówek co do wykonywania płytek dwustronnych. Należy zaopatrzyć się w papier kredowy (możliwie najcieńszy) i wydrukować na dwóch kartkach warstwy „top” i „bottom”. Następnie należy zrobić z wydrukowanych stron „kopertę”, do której włożymy laminat. W tym celu jedną kartkę odkładamy na bok, a drugą docinamy do rozmiarów wydrukowanego obszaru, ale dodajemy margines ok. 5mm z każdego boku. Następnie musimy idealnie spasować dwie kartki (tonerem do środka), można to zrobić pod światło, np. na szybie oświetlonej od drugiej strony. Dopasowanie najlepiej poznać po pokryciu się otworów przelotek leżących w sporej odległości od siebie. Po spasowaniu należy dociętą kartkę przykleić na krawędziach taśmą klejącą do kartki niedociętej, pozostawiając jeden bok nieprzyklejony. Następnie docinamy laminat dwustronny do wymiarów nieco większych niż zaprojektowana płytka (ok. 3mm dodajemy do każdego boku), oczyścimy, odtuszczymy i wkładamy do koperty. Odprasowywanie polecam wykonać tradycyjnie żelazkiem, a nie za pomocą laminatora, gdyż ruch wałków może spowodować wzajemne przesunięcie wydruków. Przed przystąpieniem do wytrawiania należy sprawdzić prawidłowość przeniesienia ścieżek i przewiercić co najmniej dwie przelotki położone możliwie daleko od siebie. Jeżeli z obydwu



Fot. 12



Fot. 13

Wykaz elementów

Płyta główna:

wszystkie rezystory SMD o wymiarze 1206

R1	470Ω
R2,R3,R5	4,7kΩ
R4	270Ω
R6	330Ω
R7, R10	22kΩ
R8,R9	2,2kΩ
R11	potencjometr 5kΩ (może być o zbliżonej wartości)
C1	4,7u/6,3V (SMD, obudowa A)
C2,C3,C4,C5,C6,C7,C8,C13	100n (SMD, obudowa 0603)
C9	47u/6,3V (SMD, obudowa A)
C10	10n (SMD, obudowa 0603)
C11	1u/6,3V (SMD, obudowa A)
C12	10u/6,3V (SMD, obudowa A)
C14,C15	22p (SMD, obudowa 0603)
U1	STM32F303RBT6
IC1	LM317 (obudowa SOT223)
IC2	L272
LED1	dioda LED 3mm
S1,S2,S3	przyciski

(przycisku S1 „RESET” nie trzeba koniecznie montować)

Q1	kwarc 8MHz
X1,X2	złącze ARK podwójne
JP1	listwa goldpin 1x4
JP2	listwa goldpin 2x3
JP3	listwa goldpin 2x8
G1	złącze baterii CR2032

Uwaga, przyciski i potencjometr można montować nie bezpośrednio na płytce, ale za pomocą przewodów.

Płyta wyświetlacza:

wszystkie rezystory 0,125W

R1	680kΩ
R2	2,2kΩ
R3	4,7kΩ
R4	330Ω
R5,R7,R9,R11,R12,R13,R16,R18,R19,R20,R24,R25,R26,R28	22kΩ
R6,R14,R21,R27	470kΩ
R8,R15,R22,R29	100kΩ
R10,R17,R23,R30	10kΩ
C1	1000u/25V
C2,C3	100n
C4	1n

C5	47u /350V
T1,T2,T4,T5,T6,T7,T9,T10,T11,T12,T14,T15,T16,T17	MPSA42
T3,T8,T13,T18	MPSA92
Q1	IRF840
U1	MC34063
D1	BYV29/400 (lub podobna)
JP1	listwa goldpin 2x8
X1	złącze ARK podwójne
L1	cewka 300uH, 1,5A
V1,V2,V3,V4	lampa Nixie IN-2
N1,N2,N3,N4	lampa Nixie IN-14

Płyta wykonawcza sterująca silnikiem krokowym:

C1,C2,C3,C4,C7	100n
C5,C6	47u/16V
IC1,IC2	7805
IC3	L293D
JP1,JP2	listwa goldpin 1x4
X1	złącze ARK podwójne

Płytką drukowaną jest dostępna w Sklepie AVT jako AVT3213

stron otwór pojawił się w odpowiednim miejscu, to znaczy, że ścieżki są przeniesione prawidłowo. W przeciwnym wypadku należy całą procedurę powtórzyć.

Początkującym w pierwszej kolejności polecam wykonać płytkę modułu wyświetlacza, ponieważ jest ona prostsza i można nabrać nieco wprawy. Po wytrawieniu płytkę należy pokryć kalafonią rozpuszczoną w denaturacie (lub

pastą lutowniczą), a następnie pocynować i za pomocą rozcieńczalnika usunąć pozostałości kalafonii. W tak przygotowanej płytce trzeba wykonać w pierwszej kolejności wszystkie przelotki. Można wykorzystać w tym celu srebrzanek o średnicy 0,6mm, obcięte końcówki elementów elektronicz-



nych (niestety pasują tylko te najcieńsze) lub dowolny inny rodzaj drutu. Lepiej jest najpierw przylutować przelotkę z obydwu stron, a dopiero później dociąć druk.

Michał Raczyński
rm23892@zut.edu.pl

R E K L A M A

EDW A10 Elektronika dla juniora (i seniora) - zestaw elementów do kursu EdW2016

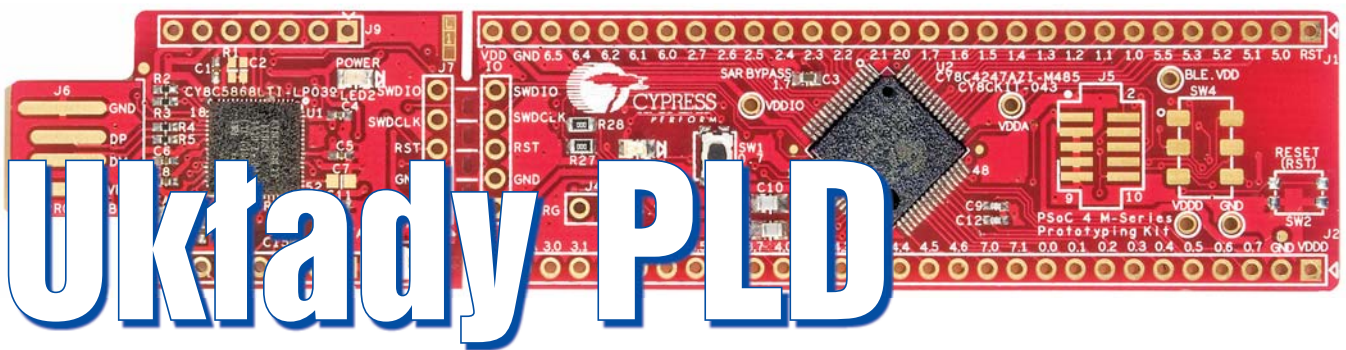
W Elektronice dla Wszystkich 1/2016, w związku z jubileuszem 20-lecia czasopisma rozpoczęliśmy elementarny kurs podstaw elektroniki dla najmłodszych i starszych, którzy chcieliby przypomnieć sobie podstawy. Założeniem jest, że młodziutki „uczeń” nie zostanie pozostawiony sam sobie, bo proponowane ćwiczenia ma wykonywać z kimś choć troszkę starszym i doświadczonym.

Oprócz rodziców może to być starsze rodzeństwo albo ktoś z rodziny lub przyjaciół.

kod handlowy:
EDWA10
45 zł

POBIERZ KURS





Logiczne układy programowalne jak dotąd nie były szerzej opisywane na łamach EdW. Najwyższa pora, żeby to zmienić i zainteresować Czytelników tą tematyką, która nie do końca słusznie uchodzi za tajemniczą i trudną. Celem niniejszego artykułu jest stosunkowo zwężone i przystępne wprowadzenie Czytelników

Przeglądając projekty układów cyfrowych publikowanych w *EdW*, wyraźnie widzimy, że procesory zadomowiły się na dobre w świadomości hobbystów, czemu sprzyjają zarówno serie artykułów wyjaśniających zasadę ich działania od strony czysto sprzętowej, jak i sposób ich programowania, np. w języku C. Widzimy też jednak, że wielu rzeczy za pomocą procesora zrobić się nie da (lub jest to co najmniej bardzo trudne albo nienaturalne) i dlatego jest on otoczony wianuszkami elementów wspomagających. O ile na obecność elementów mocy i sporej części układów analogowych wiele poradzić się nie da, o tyle perspektywa sprowadzenia całej części cyfrowej do jednego układu logicznego i kondensatorów odsprężających jest bardzo kusząca. Zwłaszcza biorąc pod uwagę fakt, że dziś nie jest to domena fantastyki naukowej, lecz rzeczywistość. Szkoda więc z tej możliwości nie skorzystać lub choćby o niej nie wiedzieć.

Cóż, *na układach PLD ciężki niestety odium niedostępności*, co w znacznej części wynika z historii rozwoju technologicznego w dziedzinie półprzewodników. Układy takie miały albo niewielką pojemność (jak np. układy PAL), albo były bardzo drogie i znajdowały zastosowania ważne, ale niszowe, jak np. stacje bazowe telefonii komórkowej. Do tego wymagały kosztownych sprzętowych narzędzi wspomagających (programatory, emulatory wewnątrzukładowe) i jeszcze droższego oprogramowania, o specjalistycznej wiedzy nie wspominając. Ostatnio *sytuacja uległa radykalnej zmianie*. Spadły ceny samych układów (do poziomu nawet kilkunastu złotych za sztukę w detalu!), narzędzia wspomagające zestandaryzowały się i są powszechnie dostępne na portalach aukcyjnych, a oprogramowanie często jest dostępne za darmo. Wejście w ten świat jest ułatwione również dlatego, że nawet bardzo pojemne układy

PLD, zdolne do pracy pod kontrolą Linuksa, są dziś bez problemu dostępne w postaci gotowych płytek ewaluacyjnych podobnych do Raspberry (np. płytki ZynqBerry) w cenie poniżej tysiąca złotych. Oczywiście jest to kwota stanowczo zbyt duża jak na początek, zwłaszcza dla hobbystów! Dlatego godząc się na pewne ograniczenia (np. zamiast 23 000 komórek programowalnych w nawet najprostszym Zynq, będziemy mieli ich 192) my nasze eksperymenty będziemy prowadzić na układach PSoC5LP firmy Cypress, a konkretnie na zestawie CY8CKIT-059, przedstawionym na **fotografii 1** (natomiast fotografia tytułowa pokazuje inny zestaw: Cypress CY8CKIT-043 PSoC® 4 M-Series Prototyping Kit).

Ponieważ rodzina PSoC5LP jest stosunkowo mało znana, a producent bardzo chciałby to zmienić, zestaw z fotografii 1 jest przez niego dotowany w taki sposób, by w sprzedaży detalicznej kosztował tylko 10 dolarów netto. Doliczając do tego koszty przesyłki (np. z *Element14*) oraz cenę znacznie poprawiającą wygodę przedłużacza USB, **przygodę z PLD można rozpocząć już za mniej niż sto złotych**. Potrzebne będzie tylko jeszcze pobranie ze strony producenta w pełni darmowego, lecz wymagającego rejestracji oprogramowania (PSoC Creator w najnowszej wersji), bo wszystkie narzędzia sprzętowe są już częścią zestawu.

Dokładniejszy rzut oka na fotografię 1 pokazuje, że tak naprawdę zestaw składa się z dwóch części. „Górny” fragment zawierający wytrawiony wtyk USB jest programatorem i debuggerem, a pozostała część płytki to właściwy układ programowalny. Obie płytki są połączone perforowanym przewężeniem, więc zestawu można używać albo w postaci kompletnej pojedynczej płytki, albo odłamać programator i połączyć go z układem PSoC przewodami, co może być wygodne przy wykorzystaniu zestawu jako gotowego

w podstawy PLD, szczególnie w kontekście ich praktycznego wykorzystania przez hobbystów. Forma taka wymaga jednak celowego uproszczenia lub wręcz pominięcia wielu zagadnień, więc jeśli temat spotka się z szerszym zainteresowaniem Czytelników, istnieje możliwość jego szczegółowego rozwinięcia.

modułu sterującego większym urządzeniem. Oprócz tego płytka zawiera dwa rzędy otworów pasujących do standardowych listew goldpin, które z powodu późniejszej wygody użytkowania polecam Czytelnikom wlotować, co pozwoli choćby na eksperymenty na płytce stykowej – **rysunek 2**.

W odróżnieniu od typowych układów FPGA, seria **PSoC5LP** (oraz Zynq), to **układy hybrydowe**. Oznacza to, że w ich skład wchodzi nie tylko najistotniejsze dla nas programowane bloki logiczne PLD, lecz również zwykły, 32-bitowy procesor ARM z rdzeniem Cortex M3 wraz z pamięciami programu oraz RAM. Od strony technicznej programowanie tych układów nie różni się niczym istotnym od typowego mikrokontrolera, więc w świat PLD można wchodzić stopniowo i w niezbędnym zakresie. Ciekawą własnością układów PSoC (na omówienie której niestety nie ma tu miejsca) jest to, że oprócz programowalnych bloków cyfrowych dostępny jest też bardzo bogaty zestaw programowalnych bloków analogowych, których wzajemne połączenia i funkcje można modelować myszką: jest to Field Programmable Analog Array, FPAA. Na życzenie Czytelników temat można będzie rozwinąć również i w tę stronę.

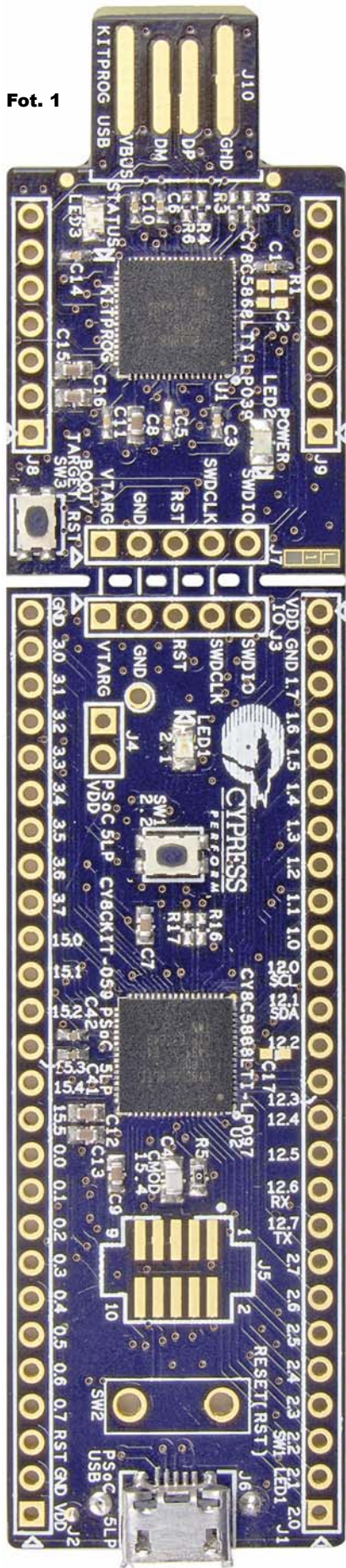
Warstwa sprzętowa

Skoro mamy już na czym prowadzić eksperymenty, przyjrzyjmy się, czym ogólnie są układy PLD. Niezależnie od producenta i modelu, układ taki składa się zawsze z następujących bloków:

Połączenia (ang. *routing resources*). Mówiąc obrazowo, jest to zestaw przewodów dostępnych dla użytkownika (oraz kompilatora, w tym kontekście zwane go syntezatorem), których używa się do łączenia wejść i wyjść pozostałych bloków w dokładnie taki sam sposób, jak ścieżek na płytce drukowanej. Z powodu zło-

zoności sprzętowej nie wszystkie z tych połączeń mają jednakowe możliwości: istnieją połączenia lokalne, służące do łączenia bloków sąsiadujących ze sobą fizycznie w strukturze układu PLD (jest ich dużo, ale mają ograniczony zasięg), są też połączenia globalne (łączą odległe bloki w układzie oraz piny wejścia-wyjścia); wyróżniona grupa połączeń służy też do dystrybucji sygnałów zegarowych. Liczba mnoga, bo – w odróżnieniu od małych mikrokontrolerów – niezależnych sygnałów zegarowych można mieć wiele (w PSoC5LP do 8 cyfrowych i jeszcze 4 specjalizowane dla bloków analogowych), co daje szerokie możliwości.

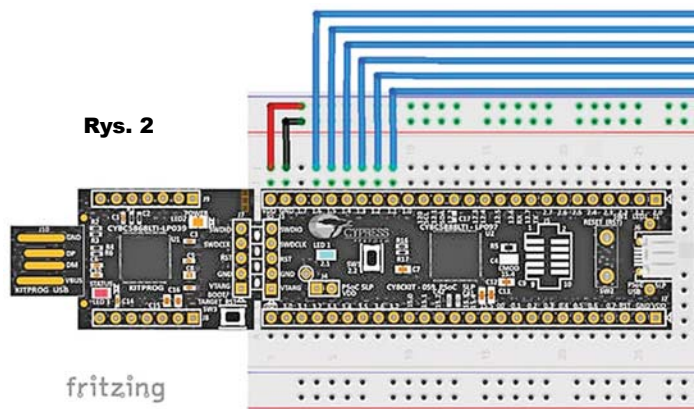
Programowalne komórki logiczne, potocznie są zwane makrocelami (*macrocell*). Każdy układ cyfrowy można zbudować z odpowiednio połączonych bramek, a zasadę działania każdej bramki da się w pełni opisać poprzez jej tzw. tabelę prawdy. **Rysunek 3** pokazuje taką tabelkę dla dwuwejściowej bramki NOR. Tabela ta wiąże wartość logiczną sygnału wyjściowego bramki z zestawami wartości podanych na jej wejścia. W standardowych układach logicz-



Fot. 1

Rys. 3

X_1	X_2	Y
0	0	1
0	1	0
1	0	0
1	1	0



Rys. 2

fritzing

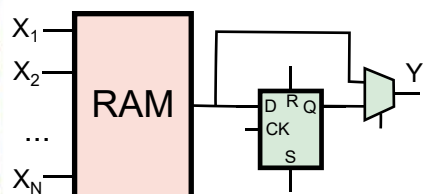
nych (np. serii TTL lub CD4000) „tabela” ta jest wytrawiona w krzemie w postaci odpowiedniej struktury połączonych tranzystorów i z tego powodu funkcja układu nie podlega jakiegokolwiek zmianie. Jest to jednak ograniczenie tylko technologiczne, a nie fundamentalne, stąd gdyby dało się w jakiś sposób zmieniać zawartość tej tabeli, uzyskalibyśmy bramkę uniwersalną. I dokładnie to robią układy PLD. W praktyce można to osiągnąć na wiele sposobów, ale obecnie mamy *dwie* znaczące różnice kategorii rozwiązań.

Pierwsza, stosowana w większych układach PLD (FPGA, Field Programmable Gate Array), polega na tym, że „bramka” taka ma postać miniaturowej pamięci RAM, w której funkcję wejść komórki logicznej pełnią linie adresowe (zazwyczaj jest to 4–6 linii), a jednobitowa zawartość tej pamięci pod wskazanym przez

wejścia adresem opisuje wartość wyjściową bramki – **rysunek 4**. Zawartość tej pamięci jest ładowana na starcie urządzenia przez wyspecjalizowany blok, nadając bramce „tożsamość”. W nowszych układach PLD zawartość tę można również zmieniać bez przerywania pracy urządzenia, w ramach tzw. rekonfiguracji dynamicznej.

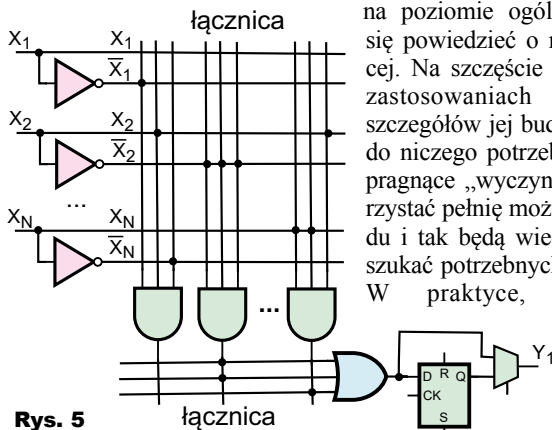
Druga to rozwiązanie uproszczone i historycznie pierwsze, stosowane m.in. w układach CPLD, Complex Programmable Logic Device. Pozwalające określić *dowolną* funkcję logiczną N zmiennych bloki RAM są konstrukcyjnie stosunkowo złożoną strukturą, więc w celu jej uproszczenia godzimy się na rezygnację z pełnej elastyczności. Polega to na tym, że komórce można wskazać ograniczony zbiór (np. nie więcej niż 12) *mintermów*, czyli połączonych operatorem *AND* wyrażeń logicznych złożonych ze wszystkich parametrów wejściowych (lub ich negacji), a następnie wyniki tych częściowych wyrażeń połączyć w finalny rezultat spójnikiem *OR*. Obliczanie wartości funkcji ma więc charakter trójwarstwowy. Warstwa pierwsza nie jest konfigurowalna. Na wejściu otrzymuje ona zbiór wszystkich N sygnałów wejściowych bramki i na swoich $2N$ wyjściach powtarza otrzymane wartości wejściowe **oraz** wytwarza ich negację. Para linii wyjściowych warstwy pierwszej (tj. sygnał i jego negacja) trafia do warstwy złożonej z bramek *AND* poprzez programowalną łącznicę. Tu już ma znaczenie konfiguracja podana przez użytkownika: określa ona, który sygnał (lub jego negacja) ma zostać uwzględniony przy obliczaniu

Rys. 4



iloczynu logicznego. Na każdym przecięciu szyny poziomej z pionową znajduje się programowalny przetwornik (najczęściej w postaci pojedynczego tranzystora *MOSFET*), który zwiera obie szyny (tranzystor włączony) lub

izoluje je od siebie (tranzystor wyłączony). Zarówno liczba bramek AND, jak i maksymalna liczba sygnałów wejściowych dla pojedynczej bramki, jest podstawową cechą konkretnego układu PLD i jest podana w jego dokumentacji. Różne wejścia tej samej bramki mogą być



Rys. 5

podłączone od tego samego sygnału wejściowego, co zamienia bramkę np. fizycznie pięciowejściową w efektywnie dwuwejściową. Ostatnia warstwa wybiera pewną liczbę częściowych wyników obliczonych przez bramki AND, również na podstawie konfiguracji użytkownika, i poprzez drugą łącznicę przekazuje ją do pojedynczej bramki OR. Uzyskany wynik to wartość naszej funkcji logicznej. Cały ten proces obrazowo pokazuje rysunek 5. Ponieważ często obliczamy więcej niż jedną funkcję logiczną dla danego zbioru sygnałów wejściowych (np. dla najprostszego sumatora ripple-carry jest to wartość sumy częściowej i wartość bitu przeniesienia dla następnego półsumatora), producenci korzystają z możliwości łatwego zysku, zarówno na polu złożoności układowej, jak i poboru energii, poprzez wielokrotne wykorzystanie zbioru wyników pośrednich, dodając więcej niż jedną bramkę OR w ostatniej warstwie. Jest to jednak tylko ciekawy szczegół techniczny, formalnie są to osobne funkcje kombinacyjne. Tak jest m.in. w omawianych układach PSoC5LP.

A oto dalsze istotne zagadnienia: zazwyczaj układy logiczne budowane są w postaci sekwencyjnej, więc na wyjściu komórki logicznej znajduje się też przerzutnik D umożliwiający „zatrzaśnięcie” wartości obliczonej przez komórkę do następnego cyklu zegarowego. Podobny efekt dałoby oczywiście połączenie wyjścia komórki z jednym z jej wejść, ale wówczas nie byłoby ono dostępne do innych celów. A że jest to typowy przypadek użycia, również i on został odpowiednio wsparty sprzętowo przez producentów, bez konieczności zużywania cennych zasobów programowalnych. Przerzutnik ten można oczywiście opcjonalnie ominąć multiplexerem, jeżeli ktoś nie życzy sobie go mieć w tym miejscu, pragnąc uzyskać układ czysto kombinacyjny.

Szczegółowa struktura komórki zmienia się zależnie od układu i producenta, więc

na poziomie ogólnym nie da się powiedzieć o niej nic więcej. Na szczęście w typowych zastosowaniach znajomość szczegółów jej budowy nie jest do niczego potrzebna, a osoby pragnące „wyczynowo” wykorzystać pełnię możliwości układu i tak będą wiedziały, gdzie szukać potrzebnych informacji. W praktyce, jeśli nie jest ważna oszczędność każdego możliwego

go centa (gdy nie produkujemy urządzeń w milionach sztuk), to w sytuacji podbramkowej znacznie prościej i taniej jest po prostu wykorzystać większy/szybszy układ programowalny. Szczegóły konstrukcyjne są oczywiście opisane w dokumentacji układu i warto zgrubnie zapoznać się, z czym mamy do czynienia, ale zazwyczaj wystarczy wiedza o tym, co komórka robi, a nie to, w jaki sposób to robi. Wyobrażenie, że jest to maleńki kawałek pamięci RAM, jest bardzo bliskie rzeczywistości i jako pierwsze przybliżenie jest uniwersalnie dobre dla różnych układów PLD. Natomiast szczegółami niech zajmie się syntezytor.

Niezależnie od sposobu realizacji bramki uniwersalnej możliwości pojedynczej komórki są dość duże, ale oczywiście ograniczone jej konstrukcją sprzętową. Nie wszystkie funkcje logiczne da się zapisać za pomocą kombinacji np. pięciu zmiennych wejściowych, bo może się zdarzyć, że stan układu zależy akurat od siedmiu parametrów i żadne przekształcenia nie pozwalają pozbyć się nadmiarowych dwóch. Pozornie sytuacja wygląda beznadziejnie, ale wówczas robimy dokładnie to samo, co w układach budowanych ze zwykłych bramek logicznych: musimy użyć większej liczby prostych bramek i odpowiednio je ze sobą połączyć. Brzmi to wszystko strasznie skomplikowanie, ale... to też nie jest nasze zadanie i wykona je za nas automat. My musimy tylko określić, co układ logiczny ma robić (na jeden z wielu sposobów, o czym dalej), a o to, jak to osiągnąć, zatroszczy się syntezytor. Weźmie on na siebie minimalizację funkcji logicznych, wyliczenie zawartości pamięci konfiguracyjnych poszczególnych komórek logicznych oraz określi ich wzajemne połączenia. Na koniec uraczy nas raportem wskazującym zużycie poszczególnych zasobów (komórki logiczne, połączenia) oraz przeprowadzi statyczną analizę czasu propagacji sygnałów w będącej wynikiem

jego pracy sieci logicznej, czyli poda maksymalną częstotliwość jego poprawnej pracy. Oczywiście biorąc przy tym pod uwagę przyjęty zakres temperatur i napięcie zasilania.

Stale bloki funkcjonalne (ang. *fixed-function blocks*). O ile wszystko można zbudować z uniwersalnych bloków logicznych, z pewnością nie wszystko warto. Wiele standardowych bloków albo powtarza się w niezmienniczej postaci w większości projektów (np. porty SPI i I2C, Ethernet, USB) albo jest bardzo zasobochonna (np. układy mnożące), albo też ma unikatowe i trudne do zastąpienia elementami logicznymi własności (np. kilkukilobajtowa dwuportowa pamięć RAM), więc producenci często „profilaktycznie” dodają ich zoptymalizowane, gotowe realizacje do struktur krzemowych swoich układów. Panuje tu pełna dowolność zarówno co do funkcjonalności, jak i liczby takich dodatkowych bloków, ale zazwyczaj to też nie jest nasze zmartwienie. Syntezytor zna właściwości danego układu PLD i albo będzie starał się automatycznie wykorzystywać gotowe bloki funkcjonalne w miarę ich dostępności, uzupełniając ewentualne braki zasobami logiki uniwersalnej, albo będzie oczekiwał od nas wskazania, że pragniemy użyć danego bloku, a resztę, w tym konfigurację połączeń, wykona samodzielnie.

Nie chciałbym straszyć niepotrzebnymi szczegółami, ale muszę dorzucić kilka informacji. Otóż dla potrzeb najbardziej wymagających projektów istnieje też możliwość ograniczenia lub zablokowania syntezytorowi swobody wyboru zasobów. Dla uzupełnienia obrazu dodam, że można również wskazać to, które fizycznie komórki mają być wykorzystane do zrealizowania jakiejś funkcji i jak mają przebiegać połączenia pomiędzy nimi, ale tzw. *floorplanner* nie jest narzędziem przeznaczonym dla początkujących. Funkcje ręcznego *routingu* bywają natomiast przydatne przy budowaniu struktury bloków analogowych, ze względu na minimalizację zniekształceń sygnału i przenikania doń zakłóceń wytwarzanych przez szybkie bloki cyfrowe.

Piny wejścia/wyjścia. Układy elektroniczne zazwyczaj wchodzą w interakcję ze swoim otoczeniem i robią to za pomocą wyprowadzeń. Jednak w odróżnieniu od sporej części mikrokontrolerów, wyprowadzenia te najczęściej nie mają ściśle przypisanych funkcji (choć bywają wyjątki – patrz dokumentacja!) i z punktu widzenia układu PLD są kolejnym zasobem, do którego można poprowadzić połączenie. Główna różnica jest tylko

taka, że bieżący stan tego zasobu widać z zewnątrz układu.

To była dobra wiadomość. Ale w układach hybrydowych (PSoC5LP) sprawa się nieco komplikuje, bo piny są dostępne również dla procesora. Z punktu widzenia części PLD mimo wszystko można przypisać dowolnemu pinowi dowolną funkcję, ale z punktu widzenia procesora już niekoniecznie: bit 2 portu nr 5 jest dostępny na dokładnie tym pinie i nigdzie indziej. Ma to znaczenie w przypadku magistral wielobitowych i trzeba zwracać na to uwagę. Alternatywnie, kosztem zasobów programowalnych, można zbudować układ, który będzie sprzętowo zmieniał kolejność bitów na porcie i wówczas dowolność odwzorowań zostaje przywrócona.

Powyższe możliwości mają bardzo duże znaczenie przy projektowaniu płytek drukowanych, bo często udaje się przenieść wszystkie (lub co najmniej sporą część) problematycznych połączeń z płytki do struktury układu PLD. Potrzebne nam sygnały znajdują się wówczas tam, gdzie jest to wygodne **nam**, a nie projektantowi procesora, co znacznie upraszcza płytke drukowaną.

Możliwość konfigurowania pinów nie kończy się na ich fizycznej lokalizacji. Zazwyczaj można też wybrać strukturę stopnia końcowego (*push-pull*, albo otwarty dren od strony napięcia zasilania lub masy), opcjonalnie włączyć wewnętrzne rezystory podciągające *pull-up/pull-down* i określić szybkość pracy pinu (preferujemy bardziej strome zbocza sygnałów czy mniejsze zakłócenia i niższe zużycie energii?). Czasami można też określić stan pinów w chwili resetu urządzenia, bo nie zawsze musi odpowiadać nam domyślny stan wysokiej impedancji. Rodzina PSoC5LP ma tę możliwość.

Należy też dodać, że piny są często pogrupowane w tzw. banki, z których każdy może w razie potrzeby być zasilany innym napięciem. Załatwia to bez dodatkowych narzutów sprawę konwersji poziomów logicznych w układach mieszanych. PSoC5LP może jednocześnie

pracować z logiką na przykład 5V oraz 3,3V, nie potrzebując do tego zewnętrznych translatorów sygnału. Może wręcz zaabsorbować translatory lub inne proste bramki występujące w różnych miejscach układu, bo układ PLD nie musi przecież mieć postaci monolitycznego bloku. Jeśli potrzebujemy kilku logicznie niezwiązanych bloków i wszystkie w sumie mieszczą się w zasobach jednego układu, to nic nie stoi na przeszkodzie takiego właśnie jego wykorzystania.

Warstwa programowa

Tak w przybliżeniu ma się sytuacja w warstwie sprzętowej, natomiast teraz warto zapoznać się ze sposobem programowania takich układów. Od czasów pradawnych niemal niepodzielnie panują tu dwa języki opisu sprzętu (HDL, *hardware description language*): VHDL i Verilog. Oba oferują podobne możliwości, więc często zdarza się, że oprogramowanie do syntezy logiki rozumie oba. W takim przypadku wybór któregoś z nich jest kwestią indywidualnych upodobań. W przypadku PSoC5LP wspierany jest jednak tylko Verilog, co dość skutecznie zawęża wachlarz możliwości. W niniejszym artykule nie ma miejsca na kurs tego języka, więc **listing 1**

```
// Moduł multiplexera 2:1
module mux 2 to 1(
    x1, // wejście nr 1
    x2, // wejście nr 2
    sel, // sygnał wyboru wejścia
    y // wyjście
);
// wejścia
input x1, x2, sel;
// wyjście
output y;
// opis działania modułu
wire y;
assign y = (sel) ? x2 : x1;
endmodule
```

jedynie informacyjnie pokazuje, jak wygląda „program” opisujący sprzęt – tu dwuwejściowy multiplexer. Znow wygląda to na bardzo skomplikowane i znow tylko pozornie. Prawdą jest, że Verilog wygląda na język stosunkowo trudny, ale jest to głównie wina tego, że specyfika jego zastosowań nie jest podobna do niczego wcześniej znanego,

nawet jeżeli ktoś potrafi programować. Z odrobiną praktyki i przyzwyczajenia wrażenie pryska i język ten staje się bardzo użytecznym narzędziem. Nie zmienia to faktu, że jest to narzędzie dla nieco bardziej zaawansowanych użytkowników i zdecydowanie nie należy zaczynać przygody z PLD od jego nauki. Tu znow przychodzi nam z pomocą automaty. Oto bardzo dobra wiadomość: **konieczność programowania PLD w np. Verilogu wcale nie oznacza konieczności pisania w tym języku**. Każde znane mi oprogramowanie do syntezy PLD (w tym PSoC Creator) zawiera zbiór narzędzi do graficznego modelowania struktury układu logicznego. Czasami jest to tylko konwerter z postaci graficznej na pliki Veriloga, ale częściej praca nad układem od samego początku przebiega w postaci graficznej i pośrednich plików verilogowych w ogóle nie widać, jeśli ktoś ich aktywnie nie szuka. W związku z tym jeszcze lepsza wiadomość jest następująca: **jedynym wymaganiem do zabawy z PLD jest umiejętność narysowania swojego układu logicznego w postaci schematu złożonego ze standardowych bramek**. Umiejętność pracy z tekstem w HDL zaczyna być przydatna dopiero przy większych projektach. Należy przy tym zaznaczyć, że układy „narysowane” nie są w żaden sposób ułomne w stosunku do układów „napisanych”, zwłaszcza z punktu widzenia ich wydajności. Verilog i VHDL są z założenia przenośne na inne układy PLD (przynajmniej teoretycznie...), więc zadaniem człowieka jest czytelne wyrażenie intencji, a zadaniem syntezyatora przeniesienie ich optymalnie na konkretny układ. Dlatego nie ma sensu wykonywanie ręcznej minimalizacji wyrażeń logicznych itp. czynności, by zaoszczędzić kilka bramek. „Pod spodem” i tak nie ma żadnych bramek, tylko zawartość wielu małych RAM-ów, z których każdy z osobna ma się bardzo luźno do narysowanej funkcji! Zaoszczędzony czas najlepiej przeznaczyć na zapewnienie czytelności schematu układu i jego poprawności.

R E K L A M A

AVT 3128 DEVASTATOR, czyli bombowy zegarek

Estetyczny, praktyczny, efektowny, niezwykle pomysłowy zegar, który zawsze wywołuje efekt WOW wśród rodziny i znajomych. Ponadto DEVASTATOR to pierwszy naprawdę uniwersalny projekt DIY. Jest odbierany równie entuzjastycznie przez osoby zupełnie niezwiązane z techniką, jak i przez doświadczonych elektroników.



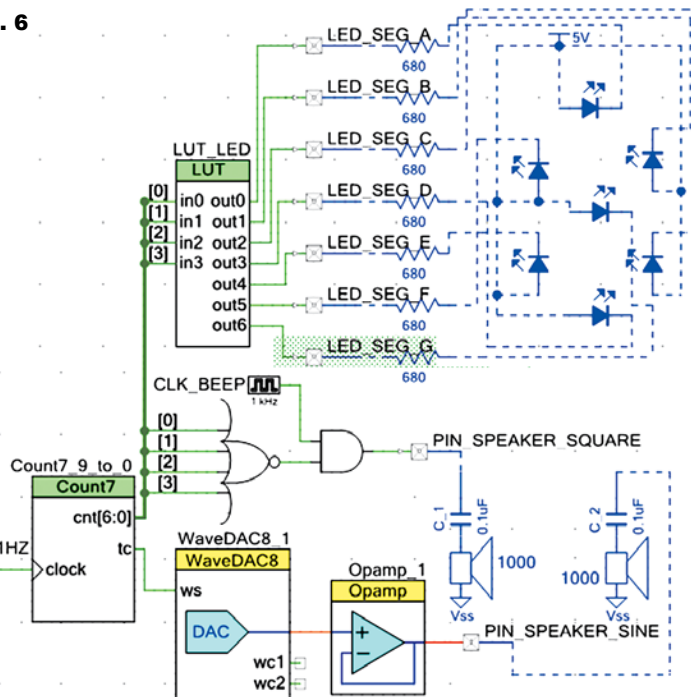
Znajdź nas na



Jeżeli zaś chodzi o **czytelność**, to tu znów nie ma jednego sposobu na wymodelowanie układu. Narzędzia graficzne umożliwiają wybór reprezentacji najlepiej dostosowanej do danego podproblemu, a nawet wielu jednocześnie na tym samym schemacie. Można narysować sobie układ z ogólnie znanych symboli bramek, można „wyklikać” myszką jego tabelę prawdy i pozostawić narzędziu znalezienie struktury realizującej tę tabelę, układy sekwencyjne można graficznie modelować na poziomie maszyn stanów, można sobie układ wreszcie napisać w Verilogu, a nawet wszystko to jednocześnie, jeśli z jakiegoś powodu tak właśnie będzie najlepiej. Dekodera 4-bit na sygnały sterujące wyświetlaczem segmentowym nie ma sensu składać z bramek i właściwym rozwiązaniem będzie tablica prawdy funkcji o 4 wejściach i 7 wyjściach. Prosty operacji na sygnale nie należy zapisywać tablicą prawdy, bo nic z niej w sposób oczywisty nie wynika, lecz lepiej użyć bramek. Niech zawsze decyduje czytelność intencji twórcy układu.

Ważną sprawą jest też możliwość **modularyzacji** układu. Paleta dostępnych komponentów nie ogranicza się do imponującej liczby bloków dostarczonych fabrycznie przez producenta oprogramowania. Użytkownik może sobie tworzyć własne komponenty, a już utworzonych używać na schemacie jako pojedynczych bloków, a nie złożonych sieci bramek (np. wspomniany dekodery wyświetlacza), które trzeba powielać w każdym miejscu użycia. Komponenty te mogą być również użyte do zbudowania większych komponentów; postać

Rys. 6



i stopień zagnieżdżenia ogranicza tylko wyobraźnia użytkownika. Szczególnie dobrze widać to w zestawie gotowych bloków dostarczanych przez Cypress w PSoC Creator: zaczynając od bloczka bramki AND, przez gotowe przerzutniki, na blokach sterownika wentylatora opartych na regulatorze PID kończąc. Z komponentów tych można tworzyć biblioteki (tj. wyodrębnić dobrze określone podsystemy z domeny projektu i stosować je w innych projektach), a bibliotekami wymieniać się z innymi lub je sprzedawać. Są to właśnie słynne IP Cores, czyli gotowe bloczki zbudowane i przetestowane przez innych, abyśmy my już nie musieli na nowo wymyślać kontrolera pamięci, Ethernetu czy magistrali SATA. Tworzenie ich to gałąź przemysłu sama w sobie, ale co istotne, każda platforma ma też dodatkowo

swoje grono entuzjastów *open source*. Nie inaczej jest w przypadku układów PSoC, gdzie użytkownicy za darmo dodają swoje komponenty do ogólnie dostępnej kolekcji, z czego sam kilkakrotnie skorzystałem.

Funkcję przykładu praktycznego wykorzystania PLD i ilustrację opisanych technik pełni układ pokazany na **rysunku 6**. Jego zadaniem jest cykliczne odliczanie od 9 do 0, zmniejszenie wartości licznika co sekundę, wyświetlenie tej wartości na wyświetlaczu siedmio-segmentowym o wspólnej anodzie i włączenie sygnału dźwiękowego po osiągnięciu przez licznik wartości 0. Celowo skorzystałem z możliwości wyrażenia w PSoC Creatorze poszczególnych składowych układu różnymi technikami, a także wplotem konfigurowalne bloki analogowe, by pokazać prostotę ich integracji z logiką programowalną. Dzięki temu sygnał dźwiękowy o częstotliwości 1kHz może mieć zarówno postać przebiegu prostokątnego, wygenerowanego na drodze czysto cyfrowej, jak i przebiegu sinusoidalnego, wytworzonego techniką DDS. Układ po zsyntezowaniu zajął 11 komórek programowalnych, a więc jedynie 5,7% dostępnych w układzie zasobów. Projekt jest udostępniony w *Elportalu*, po jego skompilowaniu można załadować go na wspomnianą na początku płytkę ewaluacyjną CY8CKIT-059.



Piotr Wyderski
piotr.wyderski@gmail.com

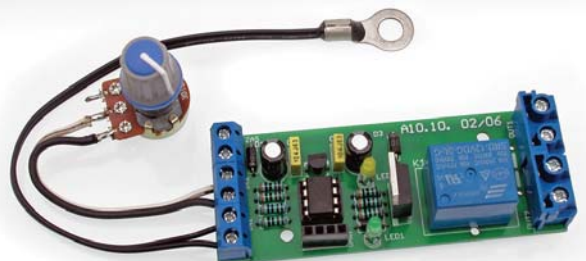
R E K L A M A

AVT 1878 Prosty termostat cyfrowy

Urządzenie jest nieskomplikowanym termostatem – wykorzystuje termistor jako czujnik temperatury. Temperaturę załączania ustawia się potencjometrem. Elementem wykonawczym jest przekaźnik.



Znajdź nas na



DipTrace – trójwymiarowy podgląd płytki drukowanej i tworzenie dokumentacji produkcyjnej

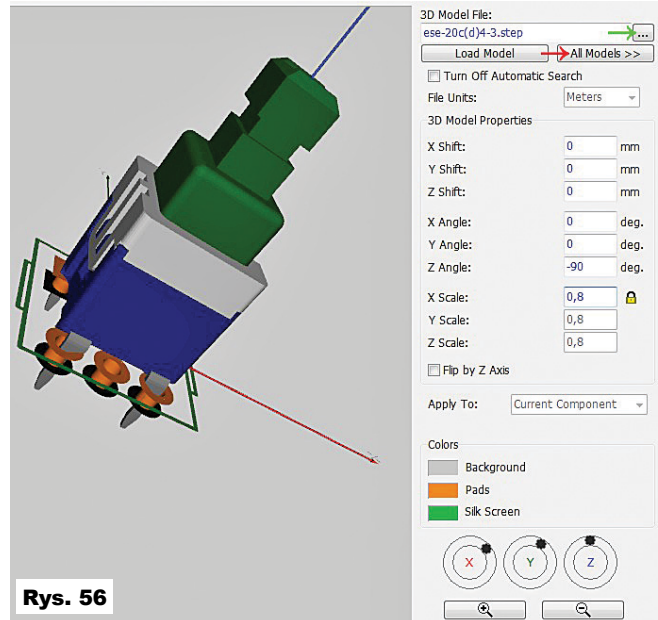
część 8

W ósmej części kursu zajmiemy się tworzeniem widoku 3D płytki drukowanej, wydrukiem oraz dokumentacją produkcyjną.

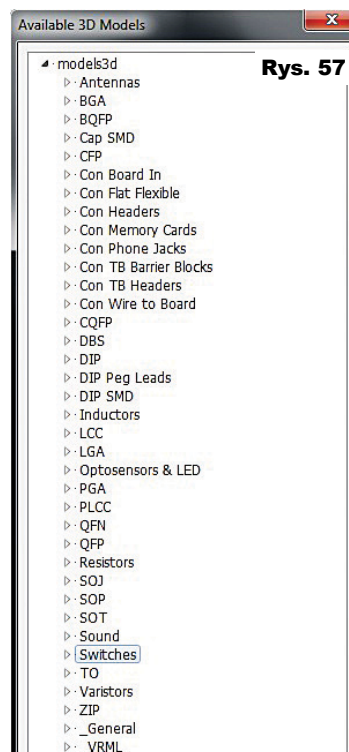
Widok 3D

Coraz więcej edytorów płytek drukowanych ma wbudowany moduł do trójwymiarowego podglądu płytki drukowanej. Taki moduł jest też wbudowany w edytor PCB Layout pakietu DipTrace. Aby móc skorzystać z tego modułu, muszą być zainstalowane biblioteki modeli 3D elementów płytkowych, które instaluje się odrębnie od głównego programu. Moduł ten uruchamia się ikonką 3D na pasku narzędziowym lub z menu **Tools** → **3D Preview** → **3D Visualization...** Otworzy się wówczas okno widoczne na rysunku 55 (wszystkie rysunki z artykułu w dużej rozdzielczości dostępne są w Elportalu wśród materiałów dodatkowych do tego numeru). Mamy w nim listę elementów płytkowych naszego projektu i przypisane im modele 3D. Jednak na liście elementów naszej płytki brakuje dwóch modeli 3D dla elementów znajdujących się na płytce. Zaznaczyłem to czerwonymi strzałkami na rysunku 55. Jeden brakujący element to złącze baterii zasilającej **J1**. Brak modelu 3D tego złącza zignorujemy, zakładając, że przewody baterii zasilającej zostaną wlutowane prosto w płytkę. Drugi brakujący model 3D to przełącznik **S1**. Aby dodać odpowiedni model 3D dla przełącznika **S1**, klikamy na przycisku z trzema kropkami wskazanym zieloną

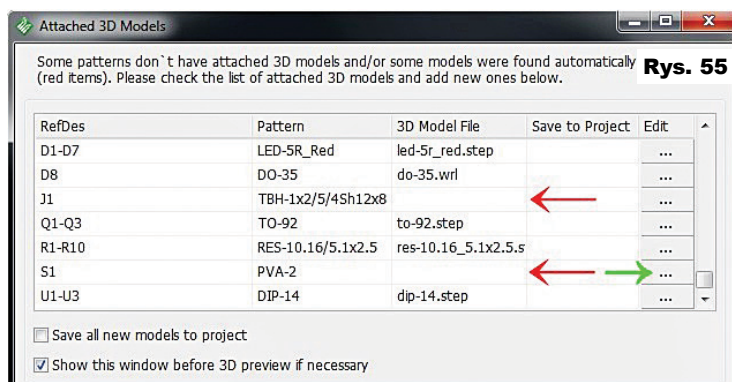
strzałką na rysunku 55. Otworzy się duże okno widoczne na rysunku 56, w którym widzimy dwuwymiarowy zarys przełącznika **S1**. W oknie tym klikamy na przycisku **All Models >>** wskazany czerwoną strzałką. Otworzy się kolejne okno widoczne na rysunku 57, gdzie w postaci drzewa mamy pogrupowane modele elementów płytkowych. Niestety nie ma innej możliwości jak ręczne przeglądanie tych elementów w celu wyboru właściwego. Nie znalazłem dokładnego odpowiednika modelu 3D dla przełącznika **S1**. Wybrałem podobny model płytkowy przełącznika o nazwie **ese-20c(d)4-3.step** z grupy **Switches**. Ma on tylko cztery wprowadzenia, ale podobny jest pod względem wyglądu do zastoso-



Rys. 56



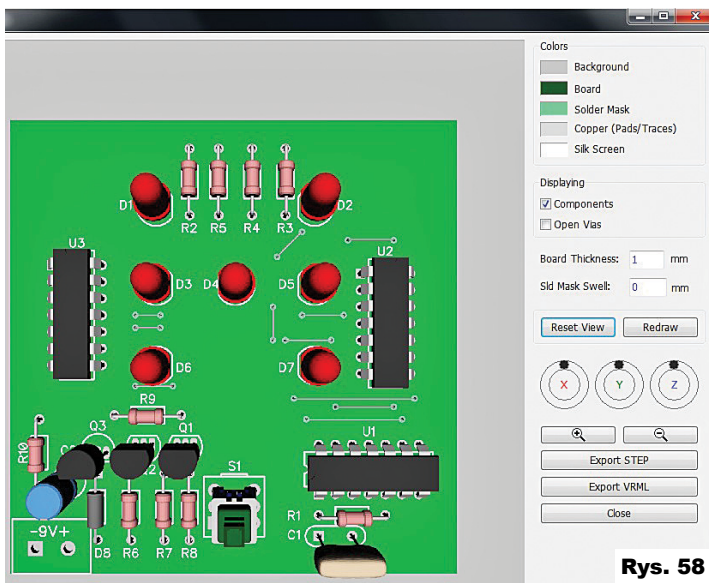
Po wybraniu odpowiedniego modelu jego podgląd pojawi się w oknie z rysunku 56. W oknie tym mamy dostępne właściwości modelu, gdzie możemy wprowadzić korektę wymiarów, pozycjonowania oraz skali wybranego modelu 3D. W naszym przypadku zmniejszyłem skalę wybranego modelu do 0,8. Pomocne w dobraniu skali jest obserwowanie, czy wyprowadzenia danego modelu pokrywają się z polami lutowniczymi jego zarysu. Dobranie skali modelu ma znaczenie w przypadku innych modeli elementów niż te które



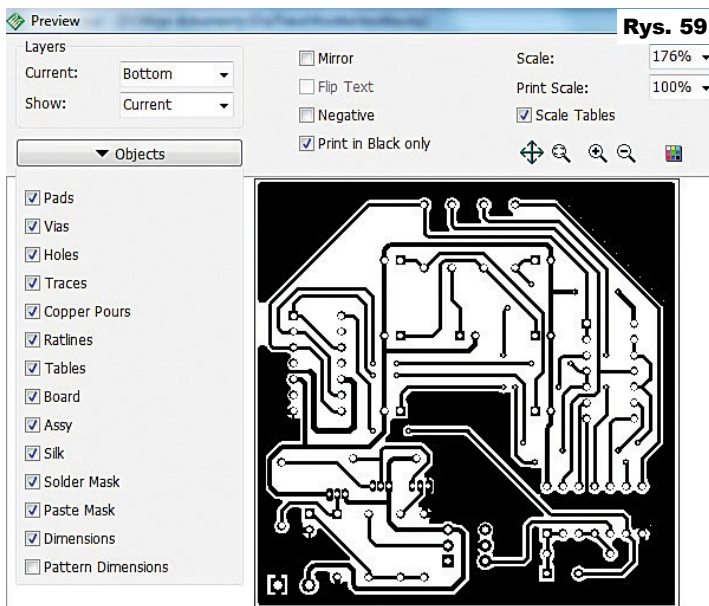
są dołączone do programu DipTrace. Między innymi pasują modele 3D elementów płytkowych utworzone dla programu KiCad. Możemy je dodać, klikając na przycisku oznaczonym zieloną strzałką na rysunku 55. Po zatwierdzeniu wybranego modelu w oknach z rysunków 55 i 56 rozpocznie się generowanie obrazu 3D zaprojektowanej płytki drukowanej. Okno z utworzonym obrazem 3D naszej płytki drukowanej możemy zobaczyć na **rysunku 58**. Pomimo, że przełącznik S1 ma niedokładnie dobrany model obudowy 3D, pasuje on dość dobrze w miejsce oryginalnego przełącznika. Obraz płytki możemy obracać, klikając na nim lewym klawiszem myszki i trzymając wciśnięty klawisz, przesuwać myszkę. Możliwe jest też obracanie płytką za pomocą pokręteł osi X, Y, Z po lewej stronie okna z rysunku 58 oraz dobranie kolorystyki obrazu płytki i okna z rysunku 58. Dostępny jest też eksport obrazu płytki w formacie STEP i VRML. Nie ma możliwości eksportu obrazu 3D płytki drukowanej bezpośrednio do postaci pliku graficznego. Najprościej jest go wykonać poprzez tradycyjny zrzut ekranu. Gdyby zaszła potrzeba zmiany modelu elementu płytkowego, jego skalowania lub dobrania wysokości ustawienia nad płytką drukowaną, to w oknie z rysunku 55 wystarczy kliknąć na przycisku wskazanym zieloną strzałką przy danym modelu. Następnie w oknie z rysunku 56 odpowiednio zmodyfikować ustawienia danego modelu.

Drukowanie rysunku płytki

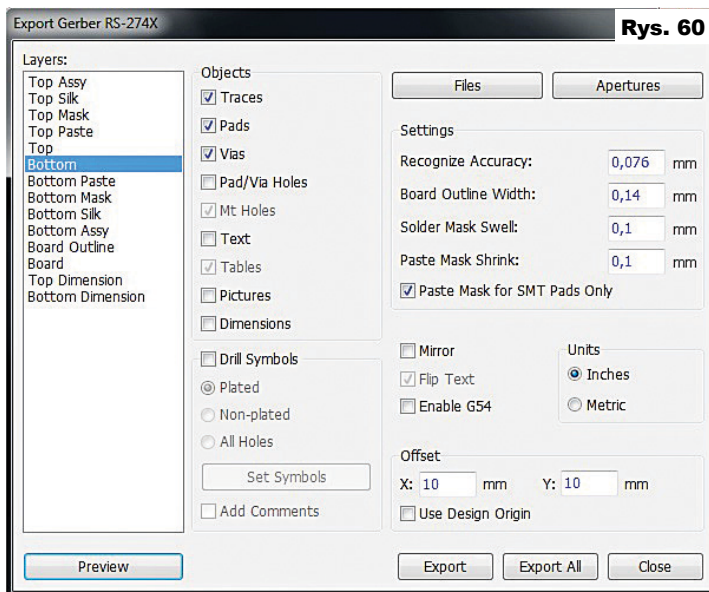
Zanim przystąpimy do drukowania płytki, trzeba ustawić, jakie warstwy mamy wydrukować. W tym celu wybieramy



Rys. 58



Rys. 59



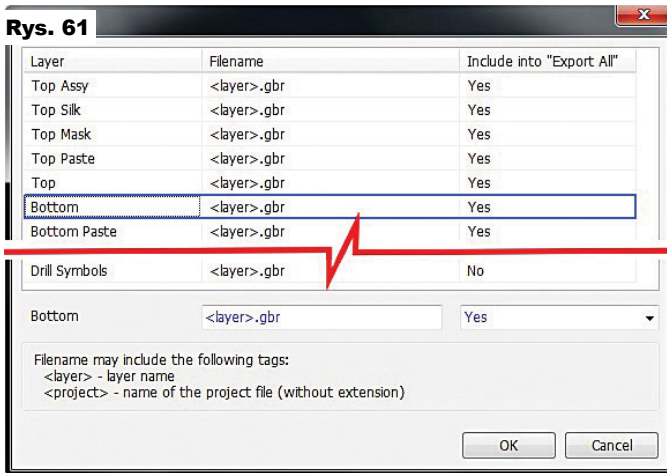
Rys. 60

z menu **File** → **Preview...** Otworzy się okno z **rysunku 59**, w którym mamy możliwość wyboru warstwy do wydruku. Możemy wybrać czy wydruk ma być w kolorze, odbiciu lustrzanym lub negatywie. Rozwijane menu **Objects** pozwala wybrać, jakie obiekty płytki drukowanej mają znaleźć się na wydruku. Możliwa jest też kalibracja drukarki – niewidoczna na rysunku 59 z uwagi na duży rozmiar tego okna. Widzimy, że drukowanie z edytora PCB Layout jest rozwiązane prosto i przejrzysto. Drukować możemy bezpośrednio na fizycznej drukarce lub wirtualnej, jak choćby wspomniany w czwartej części tego kursu PDF Creator.

Eksport danych

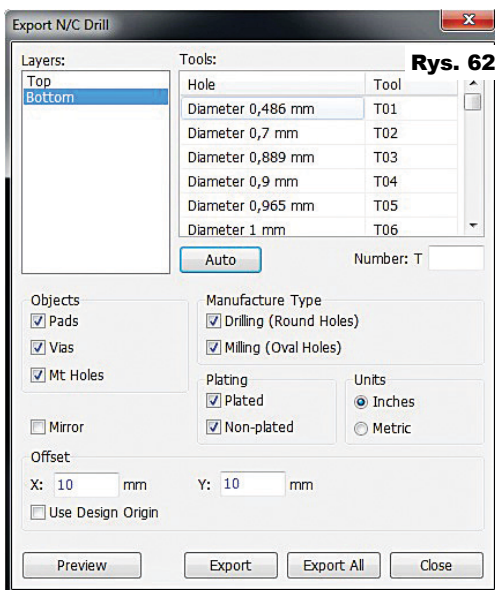
Z menu **File** → **Export** można wyeksportować rysunek płytki drukowanej w postaci pliku DXF. Z tego menu możliwy jest eksport w wielu formatach do różnych programów EDA i CAD. Również możemy wygenerować pliki produkcyjne w formacie Gerbera. Przed wygenerowaniem plików w formacie Gerber należy uzgodnić szczegóły z zakładem, gdzie będą wykonywane płytki (można dostarczyć pliki z domyślnymi ustawieniami z pytaniem, czy jest dobrze). Po wybraniu menu **File** → **Export** → **Gerber...** otworzy się okno z **rysunku 60**. W oknie tym po lewej stronie wybieramy warstwy, dla których mają zostać wygenerowane pliki produkcyjne. Dla płytki jednostronnej z elementami na warstwie górnej potrzebne będą warstwy: **Board Outline** – krawędzie płytki, **Bottom** – ścieżki warstwy dolnej, **Top Silk** – opis elementów na warstwie górnej, **Bottom Mask** – maska lutownicza warstwy dolnej. Wybieramy kolejno daną warstwę i klikamy na przy-

Rys. 61



warstwy. Pozostałe parametry w oknie z rysunku 60 pozostawiamy niezmiennie. Ewentualne zmiany należy prowadzić w uzgodnieniu z zakładem, który będzie wykonywał zaprojektowaną płytę. Do wykonania płytki w zakładzie produkcyjnym, oprócz plików

U mnie został utworzony plik o nazwie **Bottom_Bottom.drl**. Także w przypadku generowania pliku wierceń być może trzeba będzie pewne szczegóły uzgodnić z producentem płytki. Choćby fakt, że w przypadku metalizacji otworów ich średnica się zmniejsza i w skrajnym przypadku wyprowadzenia elementów mogą się nie zmieścić w pomniejszych przez metalizację otworach. Utworzone w ten sposób dane produkcyjne wysyłamy do zakładu zajmującego się wykonywaniem płytek drukowanych. Jak już wspomniałem, zakłady te często mają określone swoje wymagania dotyczące dostarczanej dokumentacji.



Rys. 62

Gerber, potrzebny jest tak zwany plik wierceń. Zawiera on informacje o średnicach otworków w płycie oraz ich współrzędnych. Dane te są niezbędne do wykonania wierceń płytki na wiertarce numerycznej. W celu wygenerowania pliku wierceń wybieramy menu **File** → **Export** → **N/C Drill...** Otworzy się okno z **rysunku 62**, gdzie wybieramy warstwę, dla której ma być utworzony plik wierceń. Dla płytki jednowarstwowej wystarczy wybrać warstwę Bottom. W przypadku płytek wielowarstwowych trzeba wybrać pozostałe warstwy. Poszczególne średnicom przypisujemy narzędzia przyciskiem **Auto** i klikamy przycisk **Export**.

Dokumentacja ta, a nawet sam projekt płytki powinien być wykonany z uwzględnieniem tych wymagań, aby nie trzeba było później nanosić dużej liczby zmian w projekcie.



W kolejnych częściach naszego kursu zajmiemy się modyfikowaniem i edycją bibliotek schematowych i płytkowych programu DipTrace.

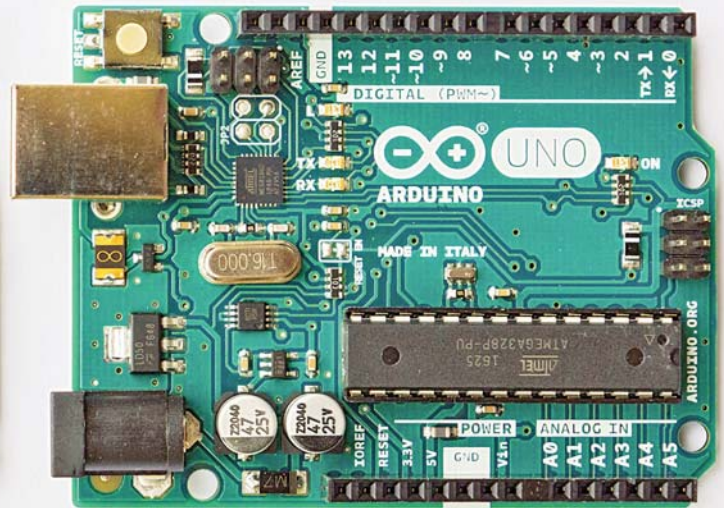
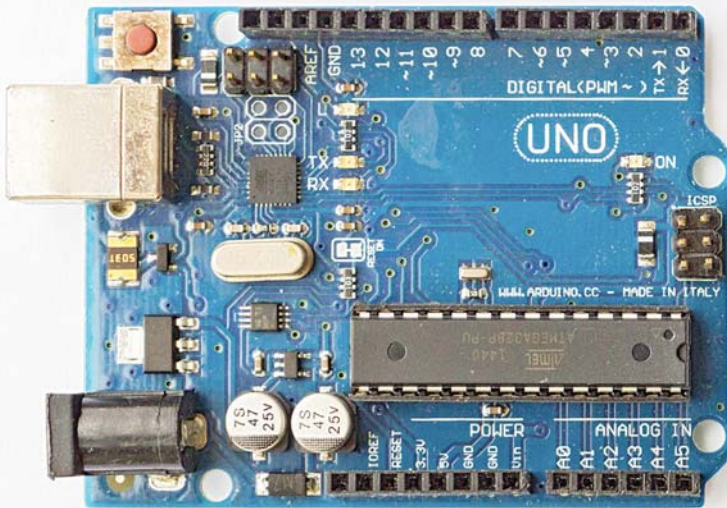
Krzysztof Kawa
kaawa@wp.pl

cisku **Export**. Dla każdej z wybranych warstw zostanie utworzony odrębny plik o nazwie takiej jak wybrana warstwa i rozszerzeniu pliku *.gbr. Przykładowo dla warstwy Bottom będzie to plik **Bottom.gbr**. Wybraną warstwę możemy podglądać, klikając na przycisku **Preview** w oknie z rysunku 60. Natomiast klikając na przycisku **Files**, otworzymy okno widoczne na **rysunku 61**, w którym możemy ustawić, czy w nazwach plików Gerber ma się pojawić nazwa danej warstwy, czy też nazwa projektu, a także zmienić rozszerzenie pliku dla danej warstwy. Zdarza się, że producenci płytek drukowanych mają w tym zakresie swoje wymagania. Możliwe jest też ustawienie, czy pliki danych warstw będą tworzone od razu po kliknięciu na przycisku **Export All** w oknie z rysunku 60. Opcje zawarte w ramce **Objects** z okna rysunku 60 decydują, jakie obiekty danej warstwy zostaną uwzględnione przy generowaniu pliku Gerber dla danej

AVT 3160 Pisanka LED

Pisanki to barwna tradycja, która pierwotnie symbolizować miała rodzącą się do życia przyrodę. Dziś ta tradycja trochę wymyka się Wielkanocy i staje się popularnym hobby. Włączając się w proces podtrzymywania tego zwyczaju, zachęcamy do wykonania ciekawego gadżetu LED-owego. Pisanka jest efektem świetlnym, którego płytka przeznaczona jest do kolorowania bądź zdobienia innymi technikami.





Arduino odcinek 1 – pierwsze kroki

Rozpoczęcie przygody z Arduino jest bardzo proste. Potrzebne są trzy rzeczy:

Płytką (Arduino Uno R3) – dwa wykonania widzisz na fotografii tytułowej.

Kabel USB AB – zwykle kupowany wraz z płytką – **fotografia 1**.

Pakiet oprogramowania – Arduino Software IDE (darmowy).

Wprawdzie możliwe jest zdalne (online) korzystanie z pakietu IDE, jednak proponuję, żebyś ściągnął wersję „stacjonarną” odpowiednią dla swojego komputera i systemu operacyjnego ze strony:

<https://www.arduino.cc/en/Main/Software>.

Dla Windows 8.1 i 10 możesz pobrać tę aplikację ze sklepu Microsoft Store (**rysunek 2**), co ma dość istotne zalety. Zostaniesz zapytany, czy chcesz finansowo wesprzeć projekt (*contribute*). Jeśli nie, wybierzesz **JUST DOWNLOAD**. Potem w sklepie jedno kliknięcie spowoduje zainstalowanie pakietu.

Gdy uruchomisz pakiet Arduino IDE, zobaczysz ekran jak na **rysunku 3**.

Trzeba też dołączyć płytkę Arduino za pomocą kupionego wraz z płytką kabla USB, co umożliwi przesłanie programu do mikroprocesora na płytce. Kabel zasi-

ła też płytkę z portu USB komputera. Na płytce trwale świecić się zielona kontrolka LED.

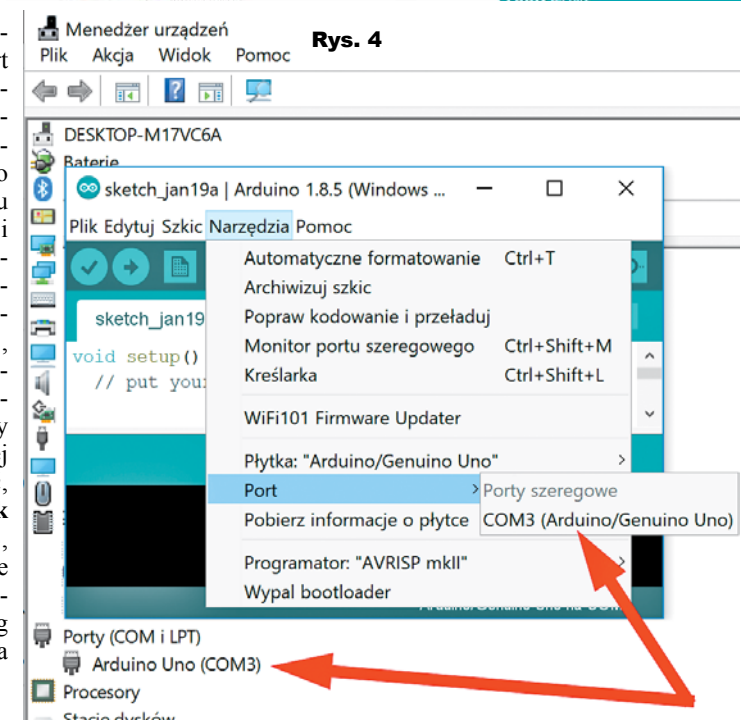
Tu trzeba wyjaśnić, że komputer sam wykryje sprzęt i zainstaluje sterownik USB (przydzielając mu numer portu COM), konieczny do komunikacji komputera PC z płytką Arduino.

Ale komputer nie konfiguruje automatycznie pakietu Arduino IDE, który musi wiedzieć, z jaką płytką współpracuje i który port COM został przydzielony do komunikacji. Aby to uzupełnić, trzeba w Arduino IDE otworzyć menu **Narzędzia**. W pozycji **Płytką** zapewne pojawi się domyślna płytką „Arduino/Genuino Uno”, ale niżej, w pozycji **Port**, trzeba wybrać „podpowiadany” port, który przydzielił wcześniej **Menedżer urządzeń**, jak pokazuje **rysunek 4** (tu jest to COM3, ale u Ciebie zapewne będzie inny). Konfiguracja według **rysunku 5** umożliwia dalsze działania.

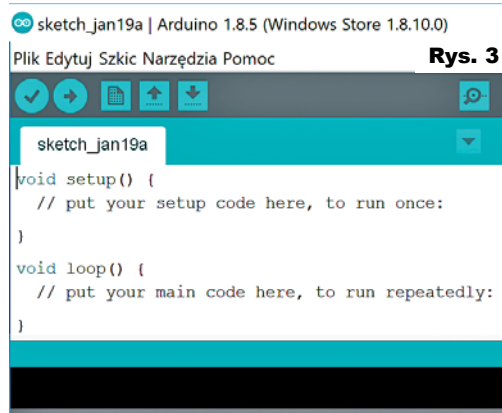
Fot. 1



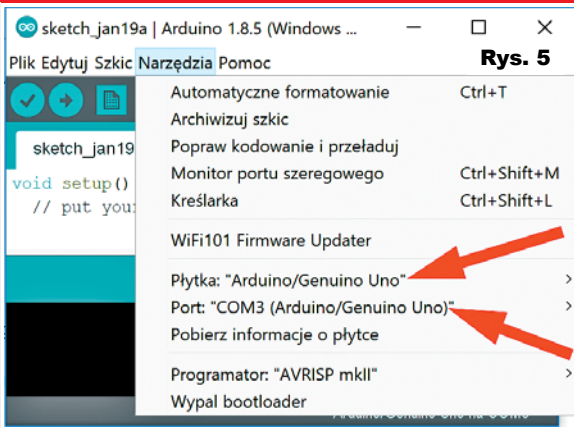
Rys. 2



Rys. 4



Rys. 3



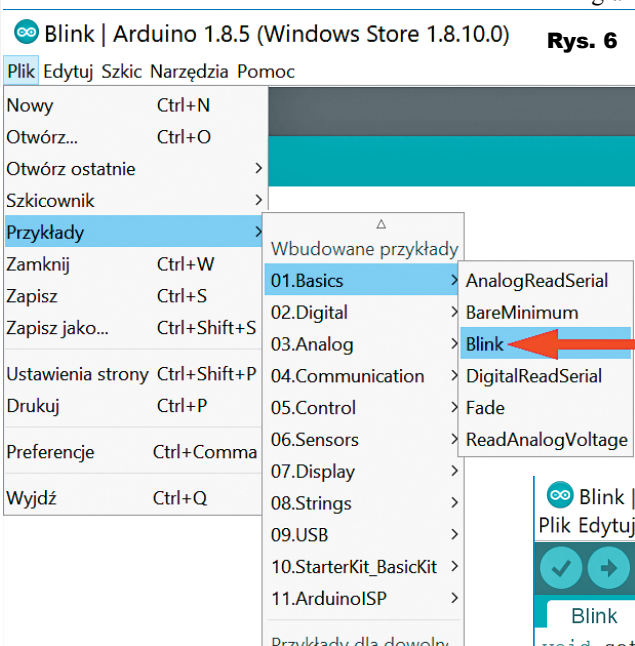
Rys. 5

program po usunięciu komentarzy wygląda jak na **rysunku 7**.

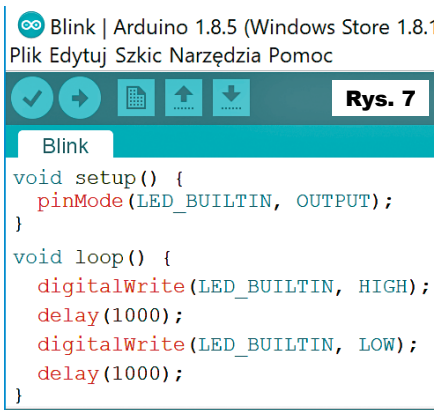
W funkcji *setup()* konfigurowane jest wyjście, współpracujące z diodą LED wbudowaną w płytke (LED_BUILTIN). A w pętli *loop()* dioda ta cyklicznie będzie zaświecana stanem wysokim (HIGH), po odczekaniu 1000 milisekund wyłączana (LOW), po kolejnych 1000ms zaświecana i tak w kółko.

Na razie jest to tylko program na ekranie komputera PC. Aby zaprogramować procesor umieszczony na płytce Arduino, musisz wykonać dwa proste kroki.

Rys. 6



Aby sprawdzić, czy program jest poprawny, trzeba kliknąć ikonkę „z ptaszkiem”, opisaną uspokajającym określeniem „Zweryfikuj”. W rzeczywistości, o czym świadczy napis po zakończeniu procesu (**rysunek 8**), następuje nie tylko weryfikacja, ale i kompilacja szkicu *Blink* do kodu, użytecznego dla zastosowanego procesora ATmega328. Co istotne, kompilator wypisuje na dole okna informacje,



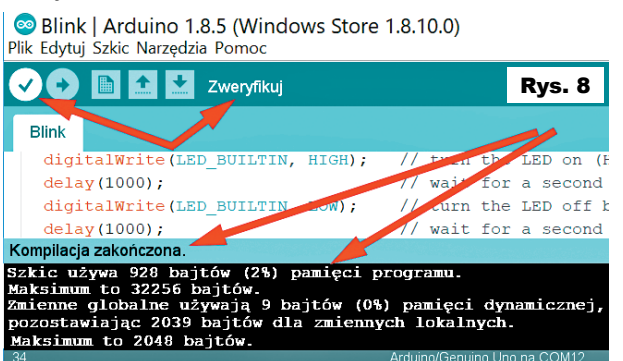
Rys. 7

Pierwszy program

Gdy mamy wszystko przygotowane, moglibyśmy napisać pierwszy program, tak zwany *szkic* (*sketch*) dla Arduino, który ma rozszerzenie *.ino*. Na rysunku 3 masz szkielec programu – szkicu, a jego nazwa wskazuje na datę utworzenia. Każdy szkic Arduino musi zawierać te dwie funkcje: wykonywaną jednorazowo po uruchomieniu *setup()* oraz gotową pętlę *loop()*.

Zamiast pisać program, łatwiej jest skorzystać z „gotowca”. W pakiecie Arduino IDE masz ich mnóstwo. Zgodnie z arduinową tradycją, odszukaj i otwórz szkic (program) *Blink.ino*. W tym celu w menu wybierz kolejno *Plik* – *Przykłady* – *01.Basic* – *Blink*, jak pokazuje **rysunek 6**.

Otworzy się nowe okno ze szkicem *Blink.ino* (miganie). Jest to plik tekstowy, gdzie najpierw masz komentarz – opis, a właściwy



Rys. 8

Gotowy program trzeba jeszcze przesłać przez łącze USB do mikroprocesora, co nastąpi po kliknięciu przycisku ze strzałką opisanego „Wgraj” – **rysunek 9**.

Po zaprogramowaniu procesora żółta dioda na płytce Arduino zacznie rytmicznie migać: świeci przez 1000ms (1 sekundę) i nie świeci przez 1000ms, zgodnie z programem.

W szkicu *Blink* zmieniaj teraz dowolnie wyrażony w milisekundach czas świecenia i przerwy:

```

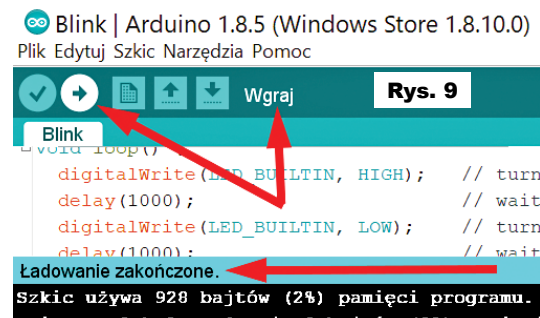
void loop() {
  digitalWrite(LED_BUILTIN, HIGH);
  // tu wstaw czas świecenia [ms]
  delay(XXXX);
  digitalWrite(LED_BUILTIN, LOW);
  // tu wstaw czas przerwy [ms]
  delay(XXXX);
}
    
```

Po każdej zmianie musisz kliknąć najpierw ikonkę „Zweryfikuj”, a po skompilowaniu ikonkę „Wgraj” według rysunków 8, 9, by wgrać program do procesora. Możesz odłączyć płytke od komputera i zasilać ją nie z portu USB, tylko z zasilacza 7...12V przez umieszczone na płytce gniazdo.

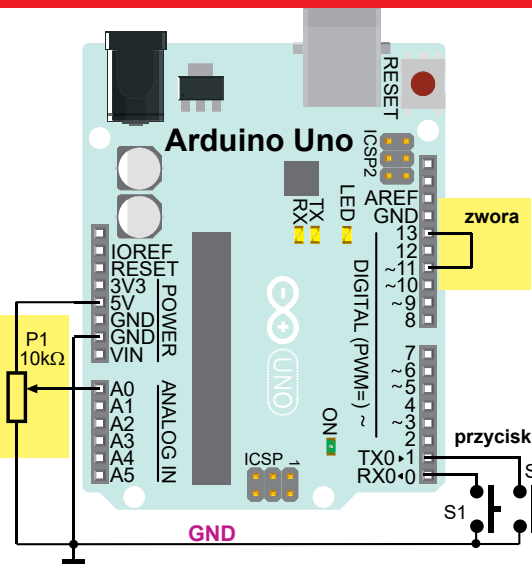
Właśnie z powodzeniem zacząłeś programować Arduino!

Jak na razie wykorzystaliśmy wbudowaną diodę LED, podłączoną na stałe do „cyfrowej” nóżki 13. Takich „cyfrowych” nóżek mamy na płytce Arduino czternaście – są numerowane od 0 do 13. Mogą one być *dwustanowymi wyjściami* o wydajności prądowej do 40mA albo *dwustanowymi wejściami*. Sześć z nich o numerach 3, 5, 6, 9, 10, 11 oznaczonych jest na płytce znakiem tyldy: (~). Mogą one być tak zwane *analogowymi wejściami*, co nie jest do końca prawdą – ale może tam występować przebieg PWM o wypełnieniu skokowo regulowanym od zera do 255. Te „wyjścia analogowe” można wykorzystywać do skutecznej pseudoanalogowej regulacji jasności diod LED, regulacji prędkości silników i podobnych celów.

Na płytce Arduino mamy też sześć końcówek oznaczonych A0...A5, które też mogą być „zwykłymi”, cyfrowymi wejściami i wyjściami. Ale mogą również być *wejściami analogowymi*. Wbudowany



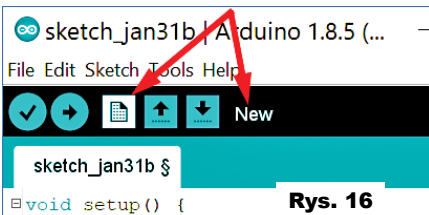
Rys. 9



Rys. 14

szkicu w danym odcinku. Powyższy szkic jest oznaczony **A0101**.

Zawiera też komentarze i omówione dalej modyfikacje. Możesz korzystać z tego „gotowca”, ale **gorąco zachęcam, żebyś sam pisał proponowane programy**. To bardzo ważne, bo dopiero samodzielnie modyfikując programy i własnoręcznie wklepując kod za pomocą klawiatury, prze-



Rys. 16

staniesz być teoretykiem i nauczysz się unikać powszechnych błędów. **Do najczęstszych błędów należy brak średników oraz nawiasów.**

Jeśli więc podczas weryfikacji kompilator podświetli na różowo jakąś linię i wypisze słabo zrozumiałą komunikat, w pierwszej kolejności sprawdź, czy nie brakuje średnika lub nawiasu w tej albo na końcu poprzedniej linii. Nasz program może też wyglądać tak:

```
void setup() {}
void loop() {analogWrite(11,analogRead(0)/4);} i wgraj do procesora na płytce. Sukces!
```

Zauważ, że usunąłem prawie wszystkie spacje. Taka wersja króciutkiego programiku może nawet jest bardziej czytelna. Jednak w przypadku bardziej rozbudowanych programów jest odwrotnie i wtedy zdecydowanie nie zachęcam do takich oszczędności i zwięzłości ze względu na przejrzystość i czytelność kodu. Wiedz, że takie usuwanie spacji i innych tzw. białych znaków robione jest zawsze na początku kompilacji, więc **jeżeli polepsza to czytelność szkicu, to nie bój się wstawiać w szkicu dowolnej liczby dodatkowych spacji i przejść do nowej linii (enter)**. W Arduino IDE masz też możliwość zwiększenia czytelności kodu przez automatyczne „standardowe” formatowanie programu według ogólnych zasad. Aby tak sformatować napisany szkic wybierz: *Tools – AutoFormat* albo naciśnij **Ctrl+T**.

Nie szcędź też komentarzy, które powinny możliwie jasno opisywać zarówno działanie programu, jak i poszczegól-

nych instrukcji. Jasne i zwięzłe komentarze są naprawdę bardzo ważne zarówno dla innych, którzy zajrzą do Twojego programu, jak i dla Ciebie, gdy za jakiś czas będziesz próbował przypomnieć sobie jego działanie. Jednoliniowy komentarz zaczyna się dwoma znakami ukośnika //, a dowolnie długi komentarz blokowy zawarty jest między parami znaków ukośnika i gwiazdki: **/* komentarz blokowy */**

Teraz skompiluj nasz pierwszy program i wgraj do procesora na płytce. **Sukces!**

Obracanie suwaka potencjometru zmienia jasność żółtej diody LED na płytce!

Program działa następująco: „arduinowa” funkcja **analogRead(0)** odczytuje wartość napięcia podawanego na wejście analogowe A0. W rzeczywistości jest to dość skomplikowany odczyt właściwie skonfigurowanego przetwornika A/D (ADC) i Arduino robi to bez naszego udziału. Ponieważ odczytana z ADC wartość jest 10-bitowa (0...1023), dzielimy odczytaną liczbę przez 4 i otrzymujemy liczbę 8-bitową z zakresu 0...255, reprezentującą wartość napięcia (0...5V) z suwaka potencjometru.

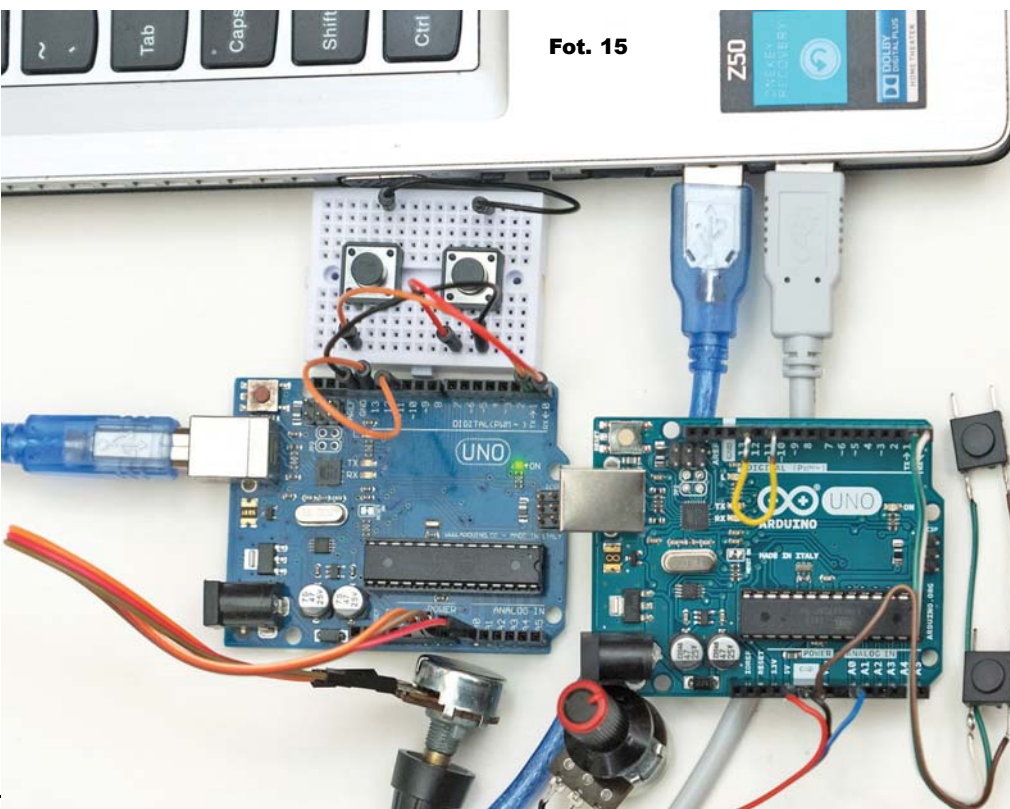
Ten sam efekt uzyskamy, pisząc: **analogWrite(11, analogRead(0) >> 2);** Zamiast dzielenia przez 4 (całkowitego, z pominięciem reszty), odczytaną liczbę dziesięciobitową przesuwamy tu bitowo w prawo o dwa miejsca (**>> 2**), czyli gubimy dwa najmniej znaczące bity.

W każdym razie uzyskana ośmiobitowa liczba (0...255) reprezentująca napięcie zostaje przekazana do funkcji **analogWrite(11, napięcie)**, która na nóżce 11 daje sygnał PWM o wypełnieniu 0...255, odpowiadającym wartości zmierzonego napięcia na nóżce A0. Wbudowana w Arduino funkcja **analogWrite()**, dająca na wyjściu przebieg impulsowy PWM, też jest dość skomplikowana, a ponieważ wykorzystuje wbudowane timery procesora, działa tylko na pinach oznaczonych tyldą (~). Szczegółów możesz poszukać w dokumentacji mikrokontrolera ATmega328P.

Ponieważ funkcji **analogWrite()** nie możemy zastosować do pinu 13, do którego dołączona jest dioda LED, wykonaliśmy zworę między pinami 11 i 13. Możemy zrobić takie zwarcie, gdyż (nieużywane) piny procesora domyślnie są konfigurowane i ustawiane jako (niepodciągnięte) wejścia.

Nasz program działa w pętli **loop()**, więc mierzy napięcie i aktualizuje współczynnik wypełnienia sygnału na pinie 11 tysiące razy na sekundę, ale taka niepotrzebna pracowitość nam nie przeszkadza.

Fot. 15



Łączy szeregowo

A teraz zmierzoną wartość napięcia dodatkowo zobrazujemy w postaci liczb. Wykorzystamy fakt, że płytka cały czas połączona jest kablem z komputerem. Łączy USB (właściwie łączy RS-232 skonwertowane na USB) służy nie tylko do programowania i do zasilania, ale jest też wygodnym i łatwo dostępnym połączeniem szeregowym naszej płytki z komputerem. Na komputerze pracuje pakiet Arduino IDE, gdzie mamy możliwość otwarcia konsoli komunikacyjnej przez kliknięcie ikonki *Serial Monitor*, jak na **rysunku 17** (także z menu: *Tools – Serial-Monitor* albo klawiszami Shift+Ctrl+M). Konsola jest cały czas gotowa do wymiany informacji z płytką Arduino, ale musimy uzupełnić nasz program – szkic. W tym celu skorzystamy z gotowej biblioteki *Serial*, która jest integralną częścią pakietu Arduino IDE i bez żadnego wysiłku z naszej strony zapewni komunikację z komputerem. Na początku naszego szkicu w funkcji *setup()* dopiszemy polecenie konfigurujące szybkość łącza szeregowego RS-232, równą 9600 bitów na sekundę, bo tak skonfigurowana jest domyślnie konsola na komputerze. Dalej w programie w pętli *loop()* dodamy instrukcję wysłania do konsoli na komputerze zmierzonej wartości napięcia oraz wprowadzimy opóźnienie. Program, który w Elportalu ma nazwę **A0102**, może wyglądać następująco:

```
// A0102
void setup() {
  Serial.begin(9600);
}
void loop() {
  analogWrite(11, analogRead(0) / 4);
  Serial.println(analogRead(0));
  delay(1000); // opóźnienie 1s
}
```

Właściwy program jest króciutki. Prócz zmiany jasności diody LED, poleceniem *Serial.println()* w każdym obiegu pętli wysyłamy odczytaną wartość napięcia przez łączy szeregowo COM i wyświetlamy na konsoli. Aby nie następowało to wiele razy na sekundę, dodaliśmy w pętli sekundowe opóźnienie *delay(1000)*. Mniej więcej co sekundę płytka wysyła więc do komputera wartość napięcia z suwaka potencjometru jako liczbę z zakresu 0...1023, co związane jest z mrugnięciem żółtej diody LED oznaczonej TX. Liczby te wyświetlane są na ekranie komputera. **Rysunek 18** pokazuje, jak zmieniały się u mnie przy kręceniu suwakiem potencjometru. Zauważ, że z uwagi na duże

opóźnienie *delay(1000)* pokręcanie suwakiem daje teraz skokowy efekt zmiany jasności LED na płytce, co jest niezbyt pożądaną cechą. Jeśli chcesz, możesz dowolnie zmniejszyć wartość opóźnienia.

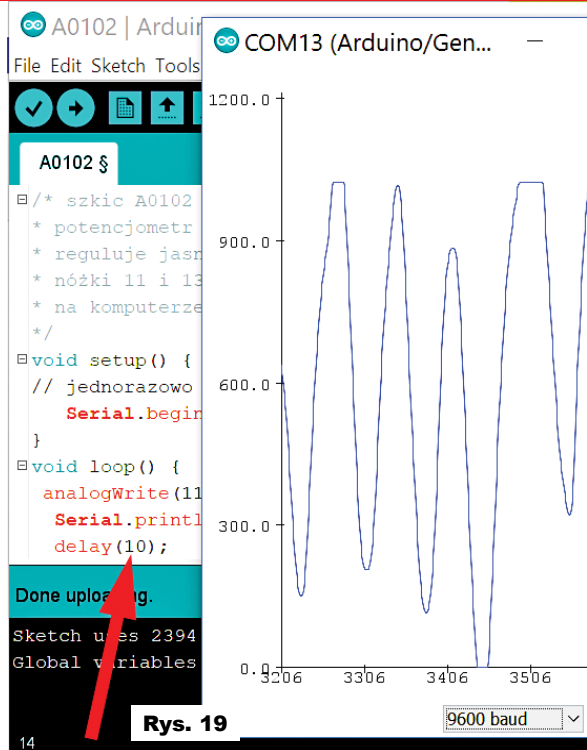
Może od razu zasignalizują też, że w pakiecie Arduino IDE oprócz konsoli tekstowej wprowadzono drugi monitor graficzno-analogowy.

Zamknij konsolę tekstową i albo w menu wybierz *Tools – Serial-Plotter*, albo naciśnij Shift+Ctrl+L. **Rysunek 19** pokazuje wykres zmian napięcia w czasie pokręcania suwaka potencjometru (przy 10-milisekundowym opóźnieniu w pętli).

Zwróć też uwagę, że w każdej pętli naszego programu dwa razy odczytujemy napięcie z suwaka, co z różnych powodów nie jest eleganckim rozwiązaniem. Bardziej eleganckie i szybsze jest zapisanie raz odczytanej wartości w jakiejś zmiennej. Może to wyglądać następująco:

```
void setup() {
  //jednorazowo skonfiguruj
  // i ustaw szybkość portu
  Serial.begin(9600);
}
void loop() {
  //zmierz napięcie na pinie A0
  // i wpisz do zmiennej
  int napiecie = analogRead(0);
  // teraz ustaw PWM
  analogWrite(11, napiecie / 4);
  // i wyślij
  Serial.println (napiecie);
  delay(100); // opóźnienie 0,1s
}
```

W pętli *loop()* deklarujemy „w locie” zmienną typu *int* o nazwie *napiecie* i od razu wpisujemy do niej 10-bitową liczbę reprezentującą napięcie z pinu A0. Potem korzystamy już tylko ze zmiennej



Rys. 19

napięcie, a nie tracimy czasu na stosunkowo długie mierzenie napięcia za pomocą przetwornika ADC.

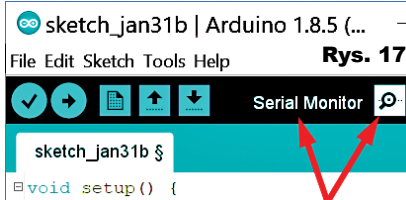
Przyciski

Wykorzystaliśmy bardzo pożyteczne funkcje, jakie oferuje Arduino: *analogWrite()*, *analogRead()* i *Serial.println()*. Skorzystajmy też z poleceń bardziej podstawowych: *digitalWrite()* i *digitalRead()*, które działają na wszystkich pinach, także analogowych A0...A5. Najpierw zrealizujemy coś bardzo prostego: przerzutnik RS. Niech przycisk S1 włącza, a S2 niech wyłącza żółtą diodę LED dołączoną do nóżki 13. Potencjometru nie wykorzystujemy.

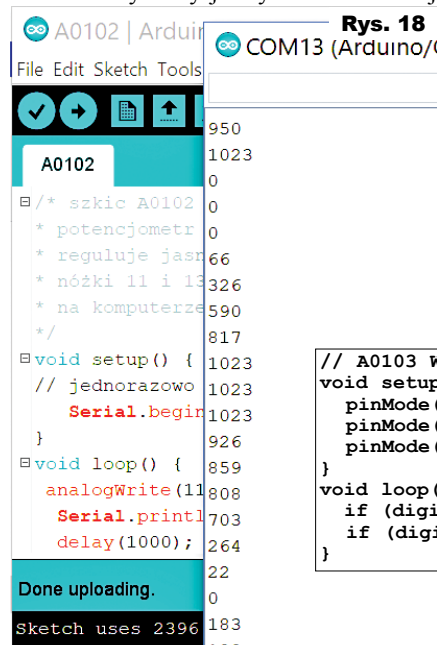
Zasadniczo przy nauce Arduino proponuje się dołączanie przycisków „od góry”, jak na **rysunku 20a**. Z kilku względów bardziej praktyczne jest dołączanie przycisków „od dołu” według **rysunku 20b**. My stosujemy ten drugi sposób, dodatkowo wykorzystując fakt, że w mikroprocesorze można włączyć kilkudziesięciokiloomowe rezystory podciągające piny do plusa zasilania według **rysunku 20c**. Stąd na rysunku 14 mamy tylko przyciski dołączone do masy. Program może wyglądać następująco:

```
// A0103 Wykorzystanie przycisków
void setup() {
  pinMode(0, INPUT_PULLUP);
  pinMode(1, INPUT_PULLUP);
  pinMode(11, OUTPUT);
}
void loop() {
  if (digitalRead(0) == LOW) digitalWrite(11, HIGH);
  if (digitalRead(1) == LOW) digitalWrite(11, LOW);
}
```

Domyślnie wszystkie piny procesora, a tym samym Arduino skonfigurowane



Rys. 17



Rys. 18

są jako wejścia, ale „zwykle” wejścia niepodciągnięte. Trzeba więc włączyć podciąganie na pinach cyfrowych 0 i 1. Musimy też skonfigurować pin 11 jako wyjście. Wcześniej tego nie robiliśmy, bo korzystaliśmy ze znacznie bardziej złożonej „arduinoowej” funkcji bibliotecznej *analogWrite()*, która konfigurację robiła automatycznie, w sposób dla nas niewidoczny. Natomiast funkcja *digitalWrite()* jest prostsza i nie konfiguruje pinu, tylko wpisuje wartość binarną do jednej z komórek rejestru procesora.

Jednorazowo w funkcji *setup()* włączamy więc podciąganie, a potem w pętli *loop()* nieustannie sprawdzamy stan pinów 0, 1. Za pomocą dwóch warunków *if* sterujemy wyjście cyfrowe 11 oraz połączony z nim pin 13 i diodę LED. Łatwizna! A teraz spróbuj zakomentować, czyli wyłączyć linię konfigurującą wyjście, dodając dwa ukośniki:
`// pinMode(11, OUTPUT);`
 W ten sposób zarówno pin 11, jak i 13 będą wejściami. Spowoduje to ciągle świecenie diody LED. Nie można jej wyłączyć.

Jeżeli jednak rozłączyłbyś piny 11, 13 i włączyłbyś diodę LED między pin 11 i masę według **rysunku 21**, układ znów zaczął działać, tylko dołączona dioda LED będzie świecić bardzo słabo.

Wprawdzie pin 11 jest teraz *wejściem*, ale polecenie *digitalWrite(11, HIGH)*, czyli wpisanie jedynki do wejścia powoduje... włączenie wbudowanego rezystora podciągającego – jest to samo, co polecenie konfiguracyjne *pinMode(11, INPUT_PULLUP)*. Tak po prostu działają piny portów mikrokontrolera.

Ale dlaczego świeci, i to pełnym blaskiem, dioda LED dołączona do pinu 13? Odpowiedź znajdziesz na schemacie wewnętrznym płytki Arduino UNO R3. Otóż dioda LED nie jest dołączona wprost do pinu 13, tylko za pośrednictwem wzmacniacza LMV358, którego wpływający prąd wejściowy (typowo 11nA, maksymalnie 250nA) okazuje się „czynnikiem podciągającym”. Wystarczy, że pomiędzy pin 13 i masę włączysz rezystor 1MΩ...4,7MΩ, a dioda ta zgaśnie

i układ według **rysunku 22** będzie jak najbardziej prawidłowo zaświecał diodę także z zakomentowaną linią

```
// pinMode(11, OUTPUT);
```

Czy w działaniu zrealizowanego w ten sposób przerzutnika RS zauważamy jeszcze coś dziwnego?

Tak! Podczas naciskania S2 zaświeca się dioda TX.

Aby to wyjaśnić, do programu dodamy polecenia obrazujące stan diody LED na konsoli komputera za pomocą liczb 0 (nie świeci), 1 (świeci). Skompiluj i wgraj do procesora poniższy program:

```
void setup() {
    pinMode(0, INPUT_PULLUP);
    pinMode(1, INPUT_PULLUP);
    pinMode(11, OUTPUT);
    // jednorazowo skonfiguruj i ustaw szybkość portu
    Serial.begin(9600);
}
void loop() {
    if (digitalRead(0) == LOW) digitalWrite(11, HIGH);
    if (digitalRead(1) == LOW) digitalWrite(11, LOW);
    Serial.println (digitalRead(11)); // wyślij
    delay(300); // opóźnienie 300ms
}
```

Podczas wgrzywania programu żółta lampka TX będzie świecić ciągle, ale gdy otworzysz konsolę na komputerze i gdy płytka Arduino „dogada się” z konsolą, żółta lampka TX będzie migać mniej więcej co 0,3 sekundy. Wprawdzie naciskanie przycisków S1, S2 zaświeca i gasi kontrolkę połączoną z pinem 13, ale długie naciśnięcie S2 gasi lampkę TX i w tym czasie do konsoli nie są przekazywane informacje. Dlaczego?

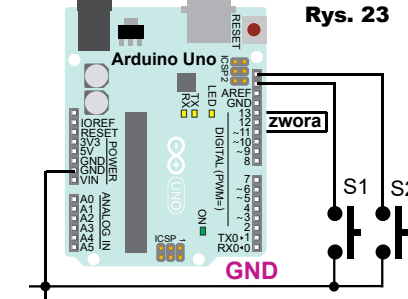
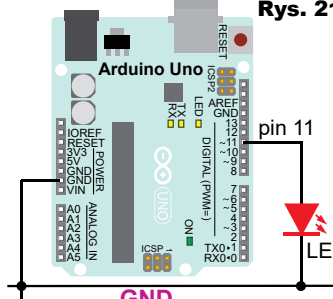
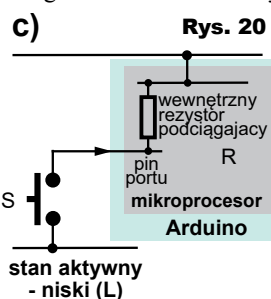
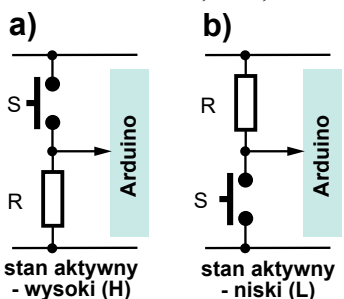
Wejścia cyfrowe 0, 1 są współdzielone z łączem szeregowym COM (RS-232). Pełną odpowiedź znajdziesz na schemacie płytki Arduino. W każdym razie gdy w procesorze pracuje *sprzętowe* łącze szeregowo, nie powinniśmy korzystać z cyfrowych pinów 0, 1. Ale warto wiedzieć, że za pomocą odpowiedniej biblioteki łącze szeregowo można też zrealizować *programowo* na dowolnych pinach procesora, oczywiście licząc się ze znacznym zwiększeniem rozmiarów programu.

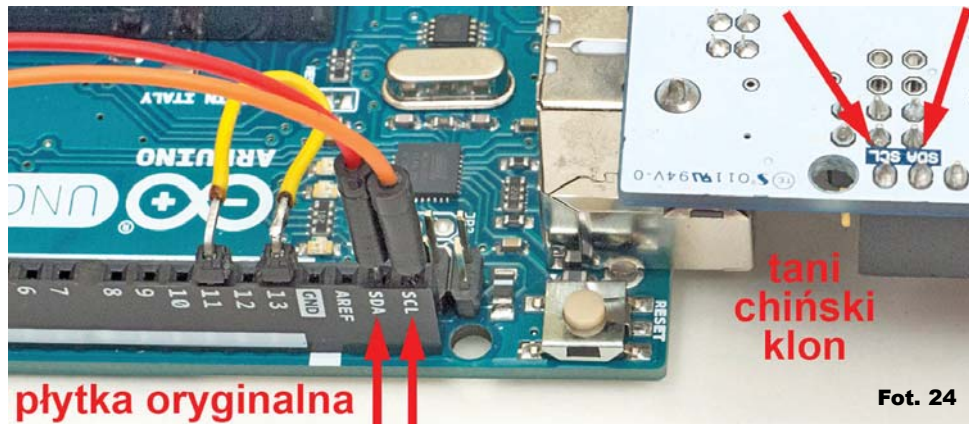
Aby zamknąć temat współdzielenia pinów, przełącz przyciski S1, S2 do skrajnych gniazd płytki Arduino według **rysunku 23**. W orygi-

nalnej płytce są one wyraźnie oznaczone SCL, SDA (**fotografia 24**), w klonach oznaczenie jest tylko na spodzie płytki. Program trzeba zmienić następująco:

```
void setup() {
    pinMode(A4, INPUT_PULLUP);
    pinMode(A5, INPUT_PULLUP);
    pinMode(11, OUTPUT);
    // jednorazowo skonfiguruj i ustaw szybkość portu
    Serial.begin(9600);
}
void loop() {
    if (digitalRead(A4) == LOW) digitalWrite(11, HIGH);
    if (digitalRead(A5) == LOW) digitalWrite(11, LOW);
    Serial.println (digitalRead(11)); // wyślij
    delay(300); // opóźnienie 300ms
}
```

Teraz wszystko pracuje prawidłowo. Warto tylko zmniejszyć czas opóźnienia z 300ms do na przykład 50ms. Zwróć uwagę, że przyciski S1, S2 dołączyliśmy do nóżek oznaczonych SDA, SCL, a w programie wpisaliśmy numery pinów analogowych A4, A5. W ten sposób przekonaaliśmy się, że po pierwsze mamy bezpośrednie połączenie nóżek A5 – SCL oraz A4 – SDA. We wcześniejszych wersjach Arduino „zdublowanych” gniazd oznaczonych SDA, SCL nie było. Zostały dodane dla kompatybilności z przyszlými wersjami Arduino, z procesorami, gdzie nie ma współdzielenia A4, A5 ze sprzętowym portem I2C (TWI – Two Wire), czyli końcówkami SDA SCL. Jeśli w Arduino Uno R3 będziesz wykorzystywał sprzętowy port TWI/I2C (i bibliotekę Wire), nie powinieneś wykorzystywać pinów A4, A5. Możesz też na dowolnych dwóch pinach zrealizować port I2C programowo.





Fot. 24

Przekonaliśmy się też, że tak zwane wejścia czy piny analogowe (A0...A5) mogą bez problemu pełnić funkcję pinów cyfrowych. Zachęcam do analizy schematu płytki Arduino i karty katalogowej procesora ATmega328P.

Na koniec jeszcze jeden dość istotny szczegół. Czy zauważyłeś, że po dołączeniu przycisków do innych pinów musimy przyszucać cały program i zmienić go w kilku miejscach? Gdyby program był bardziej rozbudowany, byłoby to uciążliwe, a przeoczenie spowodowałoby błędy. Dlatego powszechnie stosuje się lepsze sposoby. Takim lepszym sposobem jest wykorzystanie dyrektywy `#define`. Przykład poniżej:

```
#define pinWyjscioowy 11
#define pinWeS1 A4
#define pinWeS2 A5
void setup() {
    pinMode(pinWeS1, INPUT_PULLUP);
    pinMode(pinWeS2, INPUT_PULLUP);
    pinMode(pinWyjscioowy, OUTPUT);
    //jednorazowo skonfiguruj i ustaw szybkość portu
    Serial.begin(9600);
}
void loop() {
    if (digitalRead(pinWeS1) == LOW)
        digitalWrite(pinWyjscioowy, HIGH);
    if (digitalRead(pinWeS2) == LOW)
        digitalWrite(pinWyjscioowy, LOW);
    Serial.println (digitalRead(pinWyjscioowy));
    delay(300); // opóźnienie 300ms
}
```

Dyrektywa preprocesora (`#define`) jeszcze przed kompilacją dokona prostego zastąpienia, podstawienia: wszędzie zamiast `pinWyjscioowy` wpisze `11`, zamiast `pinWeS1` wpisze po prostu `A4` i zamiast `pinWeS2` wpisze `A5`.

Ponieważ dyrektywa `#define` dokonuje tylko „bezmądnego” podstawienia przed kompilacją, preferowany jest inny, lepszy sposób pokazany na listingu obok:

Tu na pierwszy rzut oka wydaje się, że na początku, zamiast dyrek-

tyw preprocesora, definiowane są trzy zmienne globalne: `pinWyjscioowy`, `pinWeS1` oraz `pinWeS2`. Wszystkie są typu `int` (integer) i od razu przypisywane są im określone wartości, którymi są numery pinów Arduino. Przy definiowaniu i inicjalizowaniu zmiennych, na końcach linii musimy postawić średnik.

Osoby nieco bardziej zorientowane zapewne będą się zastanawiać, dlaczego definiujemy tu *zmienne*. Jak wiadomo, zdefiniowanie w programie zmiennej określonego typu oznacza zarezerwowanie w pamięci danych procesora odpowiedniej ilości miejsca, stosownie do „wielkości” danego typu. Dla typu `int`, czyli liczb całkowitych –32768...32767 należy w pamięci RAM zarezerwować dwa bajty (16 bitów).

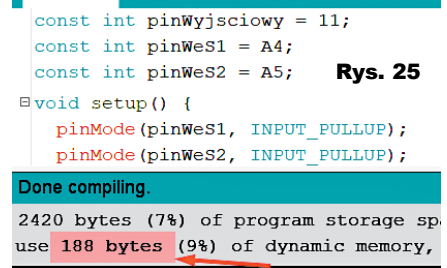
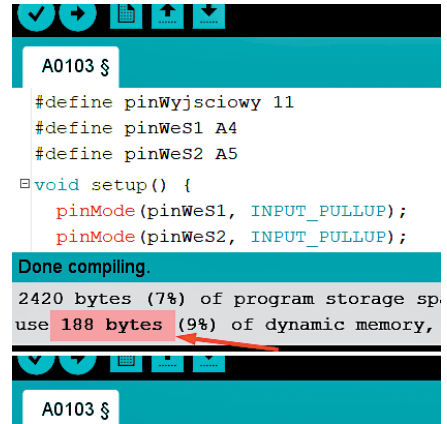
Ale ściślej biorąc, w tym przypadku nie są to zmienne, tylko *stałe*, a to dzięki użyciu słowa kluczowego `const`.

Ogólnie biorąc, *stałe* zasadniczo też finalnie stają się komórkami pamięci RAM, tylko takimi, których zawartości nie wolno modyfikować w trakcie programu.

Jednak o wszystkim decyduje kompilator. A jest on wystarczająco inteligentny, by „wiedzieć”,

że w omawianym teraz przypadku *stałe* nie mają być komórkami pamięci, tylko „pomocniczymi stałymi”, potrzebnymi jedynie na czas kompilacji. Dla-

```
const int pinWyjscioowy = 11;
const int pinWeS1 = A4;
const int pinWeS2 = A5;
void setup() {
    pinMode(pinWeS1, INPUT_PULLUP);
    pinMode(pinWeS2, INPUT_PULLUP);
    pinMode(pinWyjscioowy, OUTPUT);
    // jednorazowo skonfiguruj i ustaw szybkość portu
    Serial.begin(9600);
}
void loop() {
    if (digitalRead(pinWeS1) == LOW) digitalWrite(pinWyjscioowy, HIGH);
    if (digitalRead(pinWeS2) == LOW) digitalWrite(pinWyjscioowy, LOW);
    Serial.println (digitalRead(pinWyjscioowy)); // wyślij
    delay(300); // opóźnienie 300ms
}
```



Rys. 25

tęgo kompilator nie zarezerwuje dla tych stałych pamięci RAM. Udowadnia to porównanie objętości zajmowanej pamięci RAM dla skompilowanych obu wersji programu: **rysunek 25** pokazuje, że są one jednakowe (po 188 bajtów).

Ten drugi sposób ze stałymi jest lepszy, ponieważ dyrektywa preprocesora `#define` dokonuje tylko „bezmądnego” podstawienia, a teraz definiujemy stałe określonego typu, więc kompilator będzie pilnował też pewnych dodatkowych szczegółów i wykryje więcej ewentualnych błędów.

Nie musisz tego do końca rozumieć ani zagłębiać się we wszystkie szczegóły. Ale w treści programów szkieletów nie używaj numerów pinów. Na początku programu definiuj stałe. Wtedy zmiana pinów wiązać się będzie ze zmianą stałych tylko na początku programu.

Tyle w pierwszym odcinku cyklu, w którym zapoznaliśmy się z podstawowymi poleceniami wejścia/wyjścia: `digitalRead()`, `digitalWrite()`, `analogRead()` i `analogWrite()`.

Jak widzisz, korzystanie z Arduino jest zaskakująco łatwe.

Jestem przekonany, że już teraz napiszesz swoje własne programy. Nie przejmuj się, jeśli coś nie będzie działać według oczekiwań. Stopniowo będziemy poznawać tajniki

Arduino oraz języków C i C++.

Piotr Górecki

Kurs AVR – lekcja 23

Rozwiązanie zadania z poprzedniego odcinka

W poprzedniej lekcji tematem zadania domowego było rozbudowanie rejestratora temperatury, opisanego w lekcji 18, o możliwość wysyłania zapisanych danych do komputera. Założmy, że wysyłanie będzie się odbywało po odebraniu komendy tekstowej z komputera. Do kodu rejestratora dodajemy naszą bibliotekę `usart.c/h`, a w funkcji `main()` dodajemy wywołanie funkcji `usartInit()` i włączamy globalnie przerwanie za pomocą `sei()`. Włączenie przerwania dla odbioru danych z USARTu możemy umieścić również na początku `main()` lub wewnątrz `usartInit()`. Dodajemy funkcję obsługi przerwania, która będzie buforowała przychodzące znaki w tablicy oraz ustawiała zmienną po odebraniu pełnej linijki komendy. Wszystko tak jak w poprzednich przykładach z komunikacją tekstową. W głównej pętli dodajemy sprawdzanie, czy przyszła nowa komenda. Tradycyjnie będzie to sygnalizowała zmiana `newCommand` (listing 1).

Gdy odebrana zostanie komenda, sprawdzamy, jaka to była komenda. Założmy, że będą obsługiwane dwie komendy: `DATAGET` będzie żądaniem wysłania danych o zapisanych temperaturach, a `DATETIMESET` pozwoli nam na ustawianie daty i czasu na płytce testowej. Ta druga komenda będzie wygodną alternatywą dla ustawiania czasu za pomocą klawiatury matrycowej. Oczywiście nazwy komend są dowolne. Komendę rozpoznajemy, porównując początkowe bajty odebranej linijki z daną nazwą. W naszym

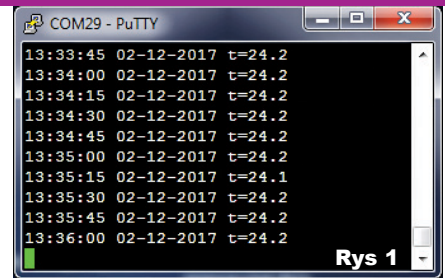
przykładzie nazwa pierwszej komendy ma 7 znaków, a drugiej 11 znaków.

Jeśli wydana została komenda `DATAGET`, wysyłamy zapisane dane. Realizowane jest to za pomocą funkcji `sendRecords()`, która w pętli wysyła rekord po rekordzie, korzystając z funkcji `sendRecord()` (listing 2).

Aby wysłać wszystkie rekordy z pamięci EEPROM, zaczynamy od adresu 2, ponieważ pierwsze dwa bajty tej pamięci są zarezerwowane do przechowywania adresu najnowszego rekordu. Następnie przeskakujemy do kolejnych rekordów, za każdym razem przeskakując tyle bajtów, ile wynosi rozmiar rekordu. Kończymy gdy natrafimy na adres początku ostatniego rekordu. Przy każdym rekordzie wczytujemy go z EEPROM-u i poprzez wskaźnik kierujemy do funkcji `sendRecord()`.

W funkcji tej tworzymy linijkę zawierającą datę, czas oraz zmierzoną temperaturę. Linijka tekstu przechowywana jest w tablicy o nazwie `str` i tworzona funkcją `sprintf()`. Odbywa się to etapami: najpierw czas (godzina, minuta, sekunda), następnie data (dzień, miesiąc, rok) i na koniec wartość temperatury. Aby takie tworzenie linijki tekstu było możliwe etapami, korzystamy z faktu, że `sprintf()` zwraca liczbę wpisanych znaków. Po pierwszym wywołaniu więc wiemy, ile liter/znaków trafiło do tablicy `str` i od którego miejsca tablicy zacząć kolejne wywołanie.

Dlatego przy pierwszym wywołaniu jako pierwszy parametr przekazujemy po prostu adres tablicy `str` (jej pierwszego elementu) a przy drugim wywołaniu dodajemy przesunięcie zapamiętane w zmiennej `pos`. Linijkę do wysłania moglibyśmy stworzyć za pomocą pojedynczego wywołania funkcji `sprintf()`, ale byłaby ona dosyć długa i przez to mniej czytelna. Gdy linijka jest gotowa, wysyłamy ją por-



Rys 1

```
void getSeparatedNumbers(char str[], uint16_t outArray[],
uint8_t outArraySize) {
    uint8_t strPos = 0;    uint8_t outPos = 0;
    for (uint8_t i=0; i<outArraySize; i++) outArray[i]=0;
    while ((str[strPos] != 0) && (outPos < outArraySize)) {
        if (str[strPos] < '0' || str[strPos] > '9') {
            outPos++;
        } else {
            outArray[outPos] = outArray[outPos]*10 + (str[strPos]-'0');
            strPos++;
        }
    }
}
```

Listing 3

tem szeregowym za pomocą funkcji `usartSendString()` (rysunek 1).

Ustawianie czasu realizowaliśmy już w poprzednim zadaniu domowym. Tutaj dodano także ustawianie daty, stąd komenda nazywa się `DATETIMESET`. Aby można było podawać rok w postaci czterocyfrowej, liczby wczytywane są do tablicy typu `uint16_t`. Ponadto zmodyfikowano funkcję `getSeparatedNumbers()`, aby można było stosować dowolny znak jako separator (listing 3). Wynika to z tego, że zwykle dla czasu jako separator używany jest dwukropki, a dla daty znak minus/myślnik. W każdym razie separatorem jest pojedynczy znak, nie można użyć np. dwóch spacji. Można więc np. wydać komendę:

```
DATETIMESET 17 18 00 2017 11 28
lub
```

```
DATETIMESET 17:18:00 2017-11-28
```

Pełny projekt rozwiązania zadania domowego znajduje się w materiałach dodatkowych do tego odcinka kursu AVR.

Callback – wskaźnik na funkcję

Przypomnijmy: aby za pomocą funkcji `printf()` wysłać tekst na wyświetlacz lub do portu szeregowego, konfigurujemy najpierw bibliotekę standardową, podając jej adres naszej funkcji, wysyłającej pojedynczy znak właśnie do LCD lub portu szeregowego. Wykonujemy to odpowiednio w funkcjach `lcdInitPrintf()` i `usartInitPrintf()`, które wywołując funkcję `fdevopen()`, przekazują odpowiednio wskaźnik na funkcję `put()` z `lcd.c` lub `usartPut()` z `usart.c`. Gdy wywołamy funkcję `printf()` generującą żądany ciąg znaków, będzie ona wywoływać nasze funkcje w celu wysłania kolejnych znaków. Jest to więc wywołanie zwrotne (ang. `callback`): nasz kod (np. funkcja `main()`) wywołuje funkcję z biblioteki standardowej języka C, a ta funkcja z kolei wywołuje z powrotem nasz kod (np. funkcję `usartPut()`).

Autorzy biblioteki standardowej, w tym funkcji `printf()`, nie wiedzą nic o naszych

```
if (newCommand) {
    newCommand = 0;
    if (strncmp(buffer, "DATAGET", 7) == 0) {
        sendRecords();
    }
    if (strncmp(buffer, "DATETIMESET", 11) == 0) {
        DateTime newDateTime;
        uint16_t datetimestr[6];
        getSeparatedNumbers(buffer + 12, datetimestr, 6);
        newDateTime.hours = datetimestr[0];
        newDateTime.minutes = datetimestr[1];
        newDateTime.seconds = datetimestr[2];
        newDateTime.year = datetimestr[3];
        newDateTime.month = datetimestr[4];
        newDateTime.dayOfMonth = datetimestr[5];
        rtcWriteDateTime(&newDateTime);
    }
}
```

Listing 1

```
void sendRecord(TemperatureRecord * temperatureRecord) {
    char str[64];
    DateTime * dateTime = &temperatureRecord->dateTime;
    uint8_t pos = sprintf(str, "%02d:%02d:%02d ", dateTime->hours, dateTime->minutes,
dateTime->seconds);
    pos += sprintf(str + pos, "%02d-%02d-%4d ", dateTime->dayOfMonth, dateTime->month,
dateTime->year);
    sprintf(str + pos, "t=%.1f\r\n", temperatureRecord->temperature);
    usartSendString(str);
}

void sendRecords(void) {
    uint16_t recordAddress = 2;
    while (recordAddress <= 1023 - sizeof(TemperatureRecord)) {
        TemperatureRecord historicRecord;
        eepromReadBytesInternal(recordAddress, (uint8_t *) &historicRecord,
sizeof(historicRecord));
        sendRecord(&historicRecord);
        recordAddress += sizeof(TemperatureRecord);
    }
}
```

Listing 2

funkcjach, takich jak `put()` czy `usartPut()`. Nie wywołują ich bezpośrednio w swoim kodzie, np. w funkcji `printf()`. Mogą jednak umieścić wywołanie funkcji na podstawie wskaźnika, który zostanie wcześniej dostarczony. Dla funkcji `printf()` taki wskaźnik przekazujemy za pomocą `fdevopen()`. Jak widać, mechanizm wskaźników na funkcje bardzo zwiększa elastyczność danej biblioteki. Autor biblioteki nie musi z góry wiedzieć, jaka funkcja ma zostać wywołana. Zostanie to określone dopiero po uruchomieniu mikrokontrolera, nie na etapie kompilacji.

Wspomnieliśmy wyżej o bibliotece standardowej języka C, zawierającej funkcję `printf()`. My też piszemy swoje biblioteki, np. `sw_usart.c/h`. W bibliotece tej mamy programowy UART, który przy odebraniu bajtu ustawia zmienną globalną `softwareUartNewData`. Dzieje się to w przerwaniu timera. Zmienna ta musi być więc obserwowana w głównym programie, aby mógł on odbierać dane z programowego UART-u. Jeśli główny program będzie zajęty jakąś dłuższą operacją, może przeoczyć część przychodzących danych. Przydałoby się więc coś w rodzaju funkcji obsługi przerwania, tak jak przy zwykłym Usarcie. Funkcja ta byłaby wywoływana dla każdego odebranego bajtu i mogłaby te bajty np. buforować. Możemy założyć, że np. wołana byłaby funkcja `softwareUsartCallback()`. Byłaby ona we fragmencie kodu odpowiedzialnym za odbiór danych w programowym Usarcie, przy ustawianiu zmiennej `softwareUartNewData`. Funkcja musiałaby być zdefiniowana też w głównym kodzie, inaczej cały program by się nie skompilował, a mówiąc ściślej, nie zostałaby zlinkowany. Używanie biblioteki `sw_usart.c/h` wymagałoby więc tworzenie takiej funkcji w głównym kodzie, np. w pliku z funkcją `main()`. Nawet jeśli funkcja ta byłaby nam niepotrzebna. Nie jest to duży problem, ale wygodniej byłoby, gdyby była możliwość wskazywania lub nie funkcji, która ma być wywoływana. W tym celu możemy skorzystać ze wskaźnika na funkcję.

Tak jak zwykle wskaźniki są zmiennymi przechowującymi adresy innych zmiennych, tak wskaźniki na funkcje są zmiennymi przechowującymi adresy funkcji. Deklaracja wskaźnika na funkcję wygląda jednak nieco inaczej niż zwykłego wskaźnika. Kompilator przy skoku do funkcji musi wiedzieć, jakie ona pobiera argumenty i co zwraca. Normalnie te informacje są zawarte w deklaracji funkcji. Muszą być one też zawarte w deklaracji wskaźnika na funkcję. Stąd deklaracja zmiennej będącej wskaźnikiem na funkcję będzie wyglądała trochę jak deklaracja funkcji:

```
uint8_t (*functionPtr)(uint16_t, int8_t);
```

W powyższym przykładzie deklarujemy zmienną o nazwie `functionPtr` służącą jako wskaźnik

na funkcję, która zwraca wartość typu `uint_8` i pobiera dwa argumenty: jeden typu `uint16_t` i drugi typu `int8_t`. Analogicznie do zwykłych wskaźników, które inicjujemy, przypisując do nich adresy zmiennych, na które mają wskazywać, tak samo przy wskaźnikach na funkcję przypisujemy im adresy funkcji:

```
functionPtr = &jakasFunkcja;
```

Należy tutaj zwrócić uwagę, że po nazwie funkcji nie umieszczamy nawiasów. Nie chcemy jej bowiem wywołać (uruchomić), a tylko pobrać jej adres operatorem `&` i następnie przypisać ten adres do zmiennej wskaźnikowej.

Aby wywołać funkcję pokazywaną przez wskaźnik, korzystamy z operatora wyluskania (gwiazdki), tak jak wtedy, gdy chcemy odczytać wartość z adresu pokazywanego przez wskaźnik:

```
uint8_t r = (*functionPtr)(212, 34);
```

Przed wywołaniem funkcji przez wskaźnik musimy upewnić się, że wskaźnik ten pokazuje na prawidłowe miejsce w pamięci. Inaczej zamiast do właściwej funkcji mikrokontroler skoczy w jakieś losowe miejsce w pamięci Flash lub poza nią, co oczywiście doprowadzi do zawieszenia się mikrokontrolera. Zwykle robimy tak jak z zwykłymi wskaźnikami: w momencie deklaracji przypisujemy wartość 0 a przed użyciem sprawdzamy, czy wskaźnik jest różny od zera.

Zalóżmy, że chcemy do naszej biblioteki dodać funkcję ustawiającą zmienną będącą wskaźnikiem na funkcję. Dodawana funkcja musi więc jako parametr przyjmować wskaźnik na funkcję. W jaki sposób zdefiniować taki parametr? Tak samo jak deklarowaliśmy zmienną wskaźnikową na funkcję:

```
void setCallback(uint8_t (*function)(uint16_t, int8_t)) {
    functionPtr = function; }
```

Taka deklaracja funkcji jest jednak nieco nieczytelna. Możemy ją uprościć, definiując swój własny typ wskaźnikowy:

```
typedef uint8_t (*FunctionPtr)(uint16_t, int8_t);
```

W ten sposób definiujemy typ `FunctionPtr` będącym typem zmiennej wskaźnikowej na funkcję pobierającą parametry `uint16_t` i `int8_t` oraz zwracającej `uint8_t`. Za pomocą tego typu możemy teraz zadeklarować zmienną wskaźnikową oraz funkcję pobierającą taką zmienną:

```
FunctionPtr functionPtr = 0;
void setCallback(FunctionPtr function) {
    functionPtr = function; }
```

Wywołanie zwrotne w bibliotece programowego portu szeregowego

Wiedząc już, jak korzystać ze wskaźników na funkcje, możemy dodać obsługę wywołania zwrotnego (`callback`) w naszej bibliotece programowego UART-u. To wywołanie będzie służyło do przekazywania odebranych bajtów do programu głównego.

Wywoływana funkcja nie musi nic zwracać, może być typu `void`. Musi mieć natomiast parametr typu `uint8_t`, aby można było przekazać odebrany bajt. Typ dla potrzebnego wskaźnika zdefiniujemy więc następująco:

```
typedef void (*CallbackFunction)(uint8_t);
```

Wykorzystamy go do zadeklarowania zmiennej przechowującej wskaźnik:

```
static CallbackFunction callback = 0;
```

Deklarujemy tutaj zmienną wskaźnikową o nazwie `callback` i inicjujemy ją wartością zero. Jako że jest to zmienna globalna, mogłaby być modyfikowana z innych plików `.c` z kodem źródłowym, co byłoby niepożądane. Chcielibyśmy, aby tę zmienną dało się modyfikować tylko z pliku `sw_uart.c`. W tym celu dodajemy modyfikator `static`. Następnie stwórzmy funkcję do ustawiania wskaźnika:

```
void softwareUartSetCallback(CallbackFunction call) {
    callback = call; }
```

Funkcja jest bardzo prosta, ustawia nasz wskaźnik na adres przekazany w parametrze `call`. Przejdźmy teraz do wykorzystania wskaźnika. Gdy odebrany zostanie nowy bajt, wykonywane są linijki:

```
softwareUartInputData = inData;
softwareUartNewData = 1;
```

Dopiszmy po nich wywołanie funkcji wskazywanej przez wskaźnik:

```
if (callback) (*callback)(inData);
```

Sprawdzamy tutaj, czy zmienna `callback` jest różna od zera. Jeśli jest różna, to znaczy, że została wcześniej ustawiona funkcja `softwareUartSetCallback`, która przypisała do niej adres funkcji, która ma zostać wykonana. Wywołujemy więc docelową funkcję, przekazując do niej wartość zmiennej `inData`, do której zapisane zostały odebrane bity.

Jak ma wyglądać główny program? Analogicznie do korzystania z przerwania generowanego przez sprzętowy USART, tworzymy funkcję odbierającą kolejne bajty. W przypadku przerwania była to funkcja o ściśle określonej nazwie, zdefiniowana z pomocą makra `ISR`. W przypadku wywołania zwrotnego nazwa może być dowolna, ale zwracany typ i parametry muszą zgadzać się z typem używanego w bibliotece wskaźnika na funkcję. Zwracany więc będzie typ `void`, a parametr będzie jeden, typu `uint8_t`. Przykładowo:

```
void rxCallback(uint8_t data) { }
```

Przy inicjalizacji programowego UART-u, czyli obok wywołania funkcji `initSoftwareUart()` musimy jeszcze zainicjalizować wskaźnik na naszą funkcję:

```
softwareUartSetCallback(&rxCallback);
```

Aby było to możliwe, deklaracja tej funkcji oraz typu `CallbackFunction` powinny być umieszczone w pliku `sw_uart.h`, nie `sw_uart.c`. Inaczej nie będą widoczne w `main.c`. Co natomiast powinno być wewnątrz funkcji `rxCallback()`? Używamy jej tak samo jak funkcji obsługi przerwania


```
#include <avr/io.h>
#include <avr/interrupt.h>
#include "sw_uart.h"
#include "lcd.h"
```

```
void rxCallback(uint8_t data);
char buffer[64];
uint8_t bufferIndex = 0;
volatile uint8_t newLine = 0;
```

```
int main(void) {
    initSoftwareUart();
    softwareUartSetCallback(&rxCallback);
    lcdInit();
    sei();
    softwareUartSendString("Elektronika\r\n");
    while (1) {
        if (newLine) {
            newLine = 0;
            lcdString(buffer);
        }
    }
}
```

```
void rxCallback(uint8_t data) {
    if (data != '\r') {
        if (bufferIndex < sizeof(buffer)) {
            buffer[bufferIndex++] = data;
        }
    }
    else {
        buffer[bufferIndex] = 0;
        bufferIndex = 0;
        newLine = 1;
    }
}
```

Listing 4

ze zwykłego USART-u. Możemy np. buforować przychodzące bajty lub wykonywać na nich inne proste operacje, niezajmujące zbyt wiele czasu. Obsługa przerwania powinna trwać jak najkrócej. Tutaj bowiem też mamy przerwanie, funkcja, której adres został przekazany do softwareUartSetCallback jest wywoływana z przerwania od Timer1, na którym opiera się nasz programowy UART. Listing 4 przedstawia przykładowy plik main.c z programem korzystającym z wywołania zwrotnego. Funkcja wołana przez wskaźnik buforuje nadchodzące bajty, aż zostanie odebrana cała linia tekstu. Wtedy linia ta jest wyświetlana na

LCD. Pełny projekt zawierający zmodyfikowaną bibliotekę programowego UART-u znajduje się w materiałach dodatkowych.

Funkcja scanf()

W wielu programach korzystamy z funkcji printf() służącej do wyświetlania użytkownikowi sformatowanych danych. W bibliotece standardowej języka C istnieje też funkcja przeciwna: scanf(). Służy ona do pobierania danych od użytkownika. Normalnie printf() wysyła znaki na standardowe wyjście, a scanf() czytuje je ze standardowego wejścia. Zarówno standardowe wyjście, jak i wejście obsługiwane są przez system operacyjny, a urządzeniami wyjścia i wejścia są monitor i klawiatura. Na naszym mikrokontrolerze nie mamy jednak systemu operacyjnego i musieliśmy napisać funkcję put(), którą printf() wywoływał, aby wyświetlić tekst na LCD. Napisałiśmy też analogiczną funkcję usartPut(), aby tekst mógł być wysyłany na port szeregowy. Aby czytać coś od użytkownika funkcją scanf(), również musimy napisać funkcję pośredniczącą, która będzie czytywała znaki z określonego urządzenia. Funkcja ta musi mieć poniższą deklarację:

```
int get(FILE * file);
```

Jak widzimy, funkcja zwraca wartość typu int i pobiera wskaźnik na zmienną typu FILE. Typ ten jest zdefiniowany w standardowej bibliotece stdio.h. Założymy, że naszym urządzeniem wejściowym będzie USART. Funkcja get() może mieć wtedy następującą postać:

```
int usartGet(FILE * file) {
    while (!(UCSRA & _BV(RXC)));
    return UDR;
}
```

Funkcja czeka na pojawienie się nowego bajtu, sygnalizowanego flagą RXC, a nastę-

nie pobiera ten bajt z rejestru odbiorczego UDR i zwraca. Parametr file nie jest używany. Jak widać, funkcja działa w sposób blokujący – nie zakończy się, dopóki nie zostanie odebrany bajt.

Jak wspomnieliśmy, scanf() czytuje dane z wprowadzonego tekstu. Wykonuje to w ten sposób, że pobiera kolejne znaki i próbuje dopasować je do podanego wzorca. Tak jak w przypadku ciągu formatującego w funkcji printf(), we wzorcu mogą znaleźć się znaki traktowane literalnie (jako one same) lub też specyfikatory poprzedzone znakiem %, które powodują podstawienie.

Założmy, że spodziewamy się od użytkownika jednego ciągu znaków (bez spacji) oraz dwóch liczb całkowitych:

```
wyraz 123 45
```

Aby czytać te dane, postępujemy tak, jak gdybyśmy chcieli je wyświetlić funkcją printf(). Jako pierwszy argument funkcji scanf() podajemy ciąg formatujący ze specyfikatorami %s, %d, %d (lista obsługiwanych specyfikatorów zawarta jest w tabeli 1). Kolejne argumenty to adresy zmiennych, do których chcemy czytać dane z analizowanego tekstu. Jest to różnica w stosunku do printf(), bo tam przekazywaliśmy same zmienne, a dokładniej ich wartości (z wyjątkiem tablic znaków). scanf() jednak nie pobiera danych do wyświetlenia tak jak printf(), ale zwraca dane. Stąd musi pobierać adresy zmiennych, aby móc do nich zapisać. Wyjątkiem są tutaj tablice znaków, przekazujemy je zawsze przez wskaźnik. Przykładowe użycie scanf() będzie wyglądało następująco:

```
int a, b; char txt[16];
scanf("%s %d %d", txt, &a, &b);
```

Po wykonaniu tej linijki na podanym wyżej ciągu znaków otrzymamy w zmiennej txt ciąg „wyraz”, w zmiennej a wartość 123, a w zmiennej b wartość 45. Jak widać, scanf() pozwala nam w wygodny sposób czytać dane z tekstu. Ze względu jednak na modyfikowanie przez tę funkcję pamięci, trzeba zachować ostrożność. Jeśli czytujemy wyraz do tablicy znaków, korzystając ze specyfikatora %s, to scanf() nie zna rozmiaru tej tablicy. Jeśli wyraz ten będzie miał więcej znaków, niż ma tablica, nadpisane zostaną inne zmienne. Z kolei specyfikator %d powoduje, że przekazany wskaźnik na zmienną liczbową będzie traktowany jako wskaźnik na typ int, który ma w przypadku kompilatora AVR GCC dwa

Specyfikator		Tabela 1
%	Dopasowuje znak %, nie wykonuje podstawienia do zmiennej	
d	Liczba dziesiętna całkowita ze znakiem, podstawienie do zmiennej typu int (int16_t)	
i	Liczba całkowita ze znakiem. Jeśli ciąg zaczyna się od 0X lub 0x, wczytywana jest liczba szesnastkowa. Jeśli ciąg zaczyna się od 0, wczytywana jest liczba ósemkowa. W pozostałych przypadkach wczytywana jest liczba dziesiętna	
o	Liczba ósemkowa, podstawienie do zmiennej typu unsigned int	
u	Liczba dziesiętna bez znaku, podstawienie do zmiennej typu unsigned int	
x	Liczba szesnastkowa bez znaku, unsigned int	
f	Liczba zmiennoprzecinkowa, podstawienie do zmiennej typu float	
e, g, F, E, G	Jak f	
s	Ciąg znaków, bez znaków białych (tabulacja, spacja), podstawienie do tablicy typu char	
c	Dopasowuje znak. Podobnie jak w przypadku s można podać długość i czytać więcej znaków, ale już bez wstawiania znaku NUL na końcu	
[Dopasowuje znaki z zestawu określonego między nawiasem otwierającym [a zamykającym]. Np. %[abc] będzie dopasowywać znaki, dopóki będą to litery a, b lub c. Można też użyć myślnika, aby określić zakres: %[a-zA-Z0-9] dopasuje małe i duże litery alfabetu angielskiego oraz cyfry. Jeśli na początku zestawu podamy znak ^, będzie to oznaczało przeczenie, np. %[!0-9] dopasuje wszystkie znaki oprócz cyfr. Znak] można umieścić w zestawie, jeśli będzie jako pierwszy, zaraz za [lub za ^. Znak - można umieścić w zestawie, jeśli będzie na końcu. %[!a-z] dopasuje znaki], - oraz małe litery. Znaki podstawione są do tablicy typu char, podobnie jak przy %s	
p	Dopasowuje wskaźnik, typ void*. Specyfikator sporadycznie używany	
n	Nie czytuje niczego, ale podstawia liczbę dotychczas wczytanych elementów do zmiennej typu int	

bajty (jest tożsamy z `int16_t`). Stosując `%d` nie możemy więc przekazać adresu zmiennej typu `int8_t`, `uint8_t` lub `char`, bo oprócz tej zmiennej może zostać nadpisana inna, sąsiadująca z nią w pamięci. Czy to znaczy, że `scanf()` można używać dla liczb całkowitych tylko ze zmiennymi typu `int`? Na szczęście nie. Dla zmiennych jednobajtowych możemy użyć specyfikatora `%hhhd` zamiast `%d`, a dla czterobajtowych (`long int`, `uint32_t`, `int32_t`) użyjemy `%ld`. Istnieje także specyfikator `%hd`, ale na platformie AVR jest to samo co samo `%d`, ponieważ zarówno `int`, jak i `short int` mają rozmiar 2 bajtów.

Z kolei dla ciągów znaków możemy podać wielkość docelowej tablicy za symbolem `%`. Jeśli więc mamy np. zmienne liczbowe typu `uint8_t` i tablicę znaków o rozmiarze 16 bajtów, napiszemy:

```
scanf("%15s %hhhd %hhhd", txt, &a, &b);
```

Podaliśmy tutaj liczbę 15, ponieważ nie jest liczony znak NUL kończący ciąg, a dla niego też musi znaleźć się miejsce.

Jak wspomnieliśmy, w ciągu przekazywanym do `scanf()` oprócz specyfikatorów poprzedzonych znakiem `%` mogą znajdować się inne znaki, których pojawienia się oczekujemy, ale których nie chcemy konwertować. Załóżmy, że chcemy wczytać godzinę, minutę i sekundę rozdzielone dwukropkami. Użyjemy wtedy ciągu „`%d:%d:%d`”. Jeśli wpisany zostanie ciąg „11:12:13”, `scanf()` wykona trzy podstawienia. Jeśli jednak użytkownik użyje spacji zamiast dwukropków, `scanf()` wykona tylko jedno podstawienie a potem przerwie działanie, gdyż spacja po pierwszej liczbie nie jest dwukropkiem. Możemy w ten sposób kontrolować występowanie elementów wpisanego ciągu bez wczytywania ich do zmiennych. Wyjątkiem jest tu spacja, gdyż oznacza ona dopasowanie nie tyle pojedynczego znaku spacji, ale ogólnie jednego lub więcej tzw. znaków biały (spacje, tabulacje).

Wartość zwracana przez `scanf()` to liczba poprawnie przekonwertowanych elementów. Dzięki temu wiedząc, ile wartości ma być podstawionych do zmiennych, możemy określić, czy wykonane zostały wszystkie podstawienia. Jeśli zwrócona wartość jest mniejsza, oznacza to, że `scanf()` napotkał znak niepasujący do wzorca i wczytywanie zakończyło się niepowodzeniem.

Uwaga! W momencie gdy `scanf()` napotka znak niepasujący do wzorca, nie tylko przestanie wczytywać kolejne znaki, ale zwróci niepasujący znak z powrotem do strumienia wejściowego. Oczywiście nie będzie to oznaczało fizycznego wysłania przez port szeregowy, ale zapamiętanie znaku w wewnętrznej strukturze strumienia. Z tego względu ponowne wywołanie `scanf()` spowoduje, że funkcja ta znów natknie się na ten nieprawidłowy znak. Jest tutaj nie-

bezpieczeństwo zapełnienia się. Kolejne znaki nie będą bowiem odbierane z portu szeregowego, ale ciągle będzie pobierany ten sam zapamiętany znak. W takiej sytuacji można np. wczytywać znaki, aż osiągnięty zostanie koniec linii (Enter) i dopiero wtedy uruchomić `scanf()` ponownie.

Jako ilustrację działania `scanf()` napiszmy prosty kalkulator działający przez port szeregowy. Użytkownik będzie mógł wykonywać cztery podstawowe działania arytmetyczne, wpisując programem PuTTY wyrażenia takie jak `32 + 72` czy `-234 / 24`. `scanf()` dopasowuje znaki tak długo jak jest to możliwe, potem próbuje z kolejnym elementem. Dlatego jeśli chcemy wczytać ciągi znaków specyfikatorem `%s`, to muszą być one oddzielone spacją. Natomiast operator matematyczny taki jak `+` czy `-` stojący między liczbami nie musi mieć koniecznie dodatkowych spacji oddzielających. Popatrzmy na przykładowy program (listing 5).

Po inicjalizacji portu szeregowego uruchamiamy funkcję `usartInitPrintfScanf()`, aby standardowe wejście i wyjście skojarzyć z naszymi funkcjami obsługującymi port szeregowy. Funkcja ta jest bardzo prosta:

```
void usartInitPrintfScanf(void) {
    fdevopen(&usartPut, &usartGet);
}
```

W głównej pętli programu funkcją `scanf()` wczytujemy tekst od użytkownika. Mają się w nim znaleźć dwie liczby rozdzielone pojedynczym znakiem. Sprawdzamy zatem, czy udało się wczytać dokładnie 3 elementy. Ponieważ chcemy obsługiwać możliwie duże liczby (`-2 147 483 648 - +2 147 483 647`), stosujemy dla zmiennych liczbowych typ `long int` (`int32_t`). W zależności od wpisanego między liczbami znaku, wykonujemy odpowiednie działanie na liczbach: dodawanie, odejmowanie, mnożenie lub dzielenie. W naszym przykładzie operujemy na liczbach całkowitych, stąd również wynik dzielenia będzie liczbą całkowitą. Wynik wysyłany jest do użytkownika funkcją `printf()`. Korzystając z tego kalkulatora przez program PuTTY, dobrze jest w jego ustawieniach włączyć opcje echa lokalnego oraz lokalnego edytowania linii, tak jak było to opisane w poprzedniej lekcji.

Jeśli wypiszemy tekst, którego `scanf()` nie będzie umiał przeanalizować, wczytujemy znaki aż do końca linii. Wykonywane jest to funkcją `getchar()`. Jest to standardowa funk-

```
#include <avr/io.h>
#include <util/delay.h>
#include <stdio.h>
#include "usart.h"

int main(void) {
    usartInit();
    usartInitPrintfScanf();
    while(1){
        long int a, b;
        char c;
        if (scanf("%ld %c %ld",
            &a, &c, &b) == 3) {
            long int r;
            switch(c) {
                case '+':
                    r = a + b;
                    break;
                case '-':
                    r = a - b;
                    break;
                case '*':
                    r = a * b;
                    break;
                case '/':
                    r = a / b;
                    break;
                default:
                    r = 0;
            }
            printf("%ld\r\n", r);
        } else {
            while (getchar() != '\r');
        }
    }
}
```

Listing 5

cja języka C pobierająca jeden znak ze strumienia standardowego wejścia. Wlicza się w to nieprawidłowy znak nieprzetworzony przez `scanf()` i zapamiętany w strumieniu. Nie możemy więc tutaj użyć `usartGet()`, bo funkcja ta nie wyczyści strumienia z owego nieprawidłowego znaku.

Jeśli chcielibyśmy korzystać z ułamków, musimy dla liczb użyć zmiennych typu `float` oraz specyfikatora `%f`. Wpisując liczby ułamkowe, trzeba pamiętać, że symbolem dziesiętnym jest kropka. Tak samo jak w przypadku funkcji `printf()` konieczne jest zmodyfikowanie ustawień linkera, aby były obsługiwane liczby

zmiennoprzecinkowe. We właściwościach projektu, w opcjach linkera dopisujemy następujące parametry:

```
-Wl,-u,vfscanf -lscanf_flt
```

Ponieważ będziemy chcieli nie tylko wczytywać liczby zmiennoprzecinkowe, ale także wyświetlać za pomocą `printf()`, całość parametrów linkera będzie wyglądała następująco:

```
-Wl,-u,vfprintf -lprintf_flt -Wl,-u,vfscanf -lscanf_flt
```

To, co jeszcze możemy zmienić w tym prostym przykładzie, to specyfikator `%c`. Ponieważ spodziewamy się tylko 4 możliwych znaków działań arytmetycznych, możemy je podać już w samym specyfikatorze. Będzie to wtedy specyfikator `%[` zamiast `%c` i w całości będzie wyglądał tak: `%1[+*/-]`. Minus musi być na końcu, gdyż w przeciwnym wypadku będzie oznaczał przedział znaków. Jedyńka oznacza, że wczytujemy tylko jeden znak.

W materiałach dodatkowych znajduje się pełny projekt kalkulatora, z opcjami linkera ustawionymi dla liczb zmiennoprzecinkowych.

Funkcja `sscanf()`

Oprócz funkcji `scanf()` istnieje też funkcja `sscanf()`. Różni się ona tym, że nie pobiera znaków z urządzenia wejściowego, ale ze zmiennej tablicowej. Możemy z niej skorzystać, jeśli dane odbieramy w przerwaniu. Z przerwania portu szeregowego korzystamy zwykle wtedy, gdy nie chcemy blokować głównego programu oczekiwaniem na kolejne bajty. A funkcja `scanf()` jest funkcją blokującą, wstrzymuje program do czasu odebrania całego tekstu pasującego

```
#include <avr/io.h>
#include <stdio.h>
#include "usart.h"

int main(void) {
    usartInit();
    FILE usartStream = FDEV_SETUP_STREAM
    (&usartPut, &usartGet, _FDEV_SETUP_RW);
    while (1) {
        int a = 0;
        if (fscanf(&usartStream, "%d", &a) == 1) {
            fprintf(&usartStream, "%d\r\n", a * 2);
        } else {
            while (getc(&usartStream) != '\r');
        }
    }
}
```

Listing 6

do wzorca, co przy ręcznym wpisywaniu może trochę potrwać. Jeśli więc korzystamy z przerwania i buforujemy linijki tekstu w tablicy, przydatna będzie właśnie funkcja `sscanf()`. Wywołujemy ją jak `scanf()`, ale dodatkowo jako pierwszy parametr podajemy adres tablicy z odebrany tekstem:

```
sscanf(buffer, "%ld %c %ld", &a, &c, &b);
```

Jak widać, `sscanf()` działa analogicznie do `sprintf()`, która to funkcja także działała na tablicy znaków zamiast wysyłać znaki do urządzenia wyjściowego, np. LCD.

Strumienie wejścia/wyjścia

Wywołując funkcje `printf()` i `scanf()`, odwołujemy się do strumienia standardowego wejścia/wyjścia, który wcześniej skojarzyliśmy z funkcjami służącymi do komunikacji z określonym urządzeniem. Dobrze byłoby jednak mieć możliwość wykorzystywania `printf()` i `scanf()` np. do dwóch urządzeń/peryferiów jednocześnie. Przykładowo przydatne byłoby móc wysyłać tekst zarówno na LCD, jak i port szeregowy. Tutaj z pomocą przychodzi definiowanie własnych strumieni.

Definicję strumienia przechowujemy w zmiennej typu `FILE`. Możemy ją zainicjalizować na dwa sposoby: makrem `FDEV_SETUP_STREAM()` lub `fdev_setup_stream()`. Pierwsze działa na zasadzie przypisania:

```
FILE usartStream = FDEV_SETUP_STREAM
(&usartPut, &usartGet, _FDEV_SETUP_RW);
```

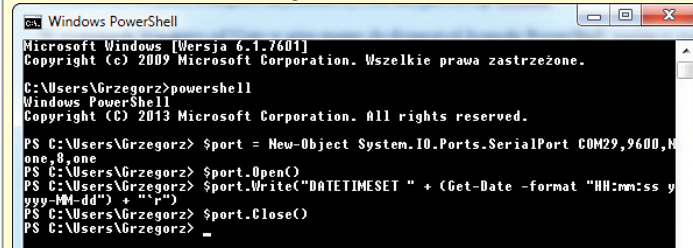
natomiasz drugie pobiera wskaźnik do wcześniej utworzonej zmiennej:

```
FILE usartStream;
fdev_setup_stream(&usartStream,
&usartPut, &usartGet, _FDEV_SETUP_RW);
```

Inicjalizacja wygląda więc podobnie jak przy standardowym, domyślnym strumieniu inicjalizowanym funkcją `fdevopen()`, której podajemy wskaźniki na dwie funkcje: zapisującą oraz odczytującą jeden znak. Tutaj dodatkowo mamy zmienną identyfikującą dany strumień. Zmienna ta jest strukturą i przechowuje skojarzone ze strumieniem wskaźniki na funkcje. Dodatkowo określamy, czy będzie możliwy odczyt ze strumie-

Aby z komputera móc komunikować się z mikrokontrolerem przez port szeregowy, potrzebny jest odpowiedni program. Często wystarczy program terminalowy, taki jak PuTTY, ale nieraz zachodzi potrzeba, aby program działał samodzielnie i nie wymagał wpisywania komend. Wtedy musimy go napisać samemu. Można to zrobić np. w języku C#, tak jak w przypadku prostego programu pomocniczego wykorzystywanego przez nas w poprzednich dwóch lekcjach, lub w dowolnym innym. Nie musi być to język kompilowany, może być też skryptowy, np. Python.

W systemach Windows od Visty w górę mamy do dyspozycji konsolę PowerShell, następczynię starego wiersza poleceń (`cmd.exe`). W XP nie ma PowerShella, ale można go zainstalo-



nia, zapis lub obie te funkcje. W powyższych przykładach podaliśmy wskaźniki na funkcje do obsługi portu szeregowego oraz opcję dostępu zarówno do zapisu, jak i do odczytu. Jeśli będziemy tylko odczytywać, zamiast `_FDEV_SETUP_RW` podamy `_FDEV_SETUP_READ`, a jeśli będziemy tylko zapisywać, użyjemy makra `_FDEV_SETUP_WRITE`. Naturalnie wtedy też jeden ze wskaźników będzie pusty, przekażemy zamiast niego wartość `NULL`.

Zmienną przechowującą skonfigurowany strumień możemy następnie przekazywać do funkcji, które mają dodaną na początku literkę `f`: `fprintf()` i `fscanf()`. Jako pierwszy parametr pobierają one wskaźnik na zmienną strumienia:

```
fscanf(&usartStream, "%d", &a);
fprintf(&usartStream, "%d", a);
```

Na **listingu 6** zaprezentowany został prosty przykład wykorzystania strumienia skojarzonego z portem szeregowym. Program wczytuje liczbę z portu szeregowego i odsyła liczbę dwa razy większą. W przypadku wpisania czegoś, co nie jest liczbą, program za pomocą standardowej funkcji `getc()` wczytuje znaki aż do Entera.

Zadania

W ramach zadania domowego proponuję napisanie zegara korzystającego z RTC oraz LCD. Data i godzina powinny być ustawiane z komputera, przez port szerego-

wać. Za pomocą PowerShella możemy nie tylko pisać skrypty wykonujące różne rzeczy w naszym systemie, ale też komunikujące się z portem szeregowym (rysunek 2). Możemy np. ustawić aktualny czas w rejestratorze temperatury z zadania domowego, wydając poniższe komendy. `COM29` zastępujemy posiadanym portem COM.

```
$port = New-Object System.IO.Ports.SerialPort COM29,9600,None,8,one
$port.Open()
$port.Write("DATETIMESET " + (Get-Date -format "HH:mm:ss yyyy-MM-dd") + "\r")
$port.Close()
```

Pisanie programów lub skryptów na PC nie jest w zakresie tego kursu, ale na pewno warto poznać przynajmniej podstawy, korzystając z dostępnej literatury i stron internetowych.

Nie trzeba przy tym nawet uczyć się koniecznie nowego języka, można skorzystać z C.

Wykorzystana powinna zostać funkcja `printf()` do obsługi LCD, `fprintf()` do wysyłania danych przez port szeregowy, a `sscanf()` do odczytywania komend odbieranych z wykorzystaniem buforowania w przerwaniu.

Jako że zajmujemy się teraz wymianą danych z komputerem, dobrze byłoby się nad tą kwestią zastanowić bardziej całościowo. Rejestrator temperatury z zadania domowego wysyła dane, które możemy obejrzeć w PuTTY. To jednak trochę mało. Może chcielibyśmy je zaimportować do arkusza kalkulacyjnego i narysować wykres temperatury? Jak to zrobić? Czy trzeba coś zmieniać w programie na mikrokontrolerze? Co trzeba zrobić po stronie komputera? Obecnie dużo urządzeń współpracuje z komputerami lub łączy się z serwerami w Internecie. Dlatego nie możemy się skupiać wyłącznie na samym mikrokontrolerze i potrzebne jest szersze spojrzenie. Zachęcam, aby się nad tym chwilę zastanowić, przynajmniej ogólnie, jako dodatkowe zadanie.

W materiałach dodatkowych znajdują się projekty rejestratora temperatury, kalkulatora, programu pokazującego działanie dodatkowych strumieni oraz programu demonstrującego programy UART z wywołaniem zwrótnym.



Grzegorz Niemirowski
grzegorz@grzegorz.net

Szkoła Konstruktorów



Szkoła Konstruktorów ma trzy klasy (Zadanie główne, Co tu nie gra? i Policz). Każdy Czytelnik może w terminie dwóch miesięcy nadesłać rozwiązanie jednego, dwóch lub wszystkich trzech zadań Szkoły z danego numeru, zwykłą pocztą lub w postaci e-maila. Paczki z modelami i koperty zawsze adresujcie: **AVT – EdW, ul. Leszczynowa 11, 03-197 Warszawa** i **koniecznie** podawajcie na kopercie czy paczce zawartość, np. *Szko265, NieGra265, Policz265* (na innych analogicznie *Jak3, #3, Brak3, Projekt*, itd.).

Rozwiązania nadsyłane e-mailem powinny być kierowane na adres: szkola@elportal.pl (*szkola*, a nie *szkoła*). Bardzo proszę: w tytule e-maila i w nazwie każdego złącznika, oprócz **nazwy konkursu** i numeru zadania, umieśćcie **swoje nazwisko** (najlepiej bez typowo polskich liter), na przykład: *Szko265Kowalski, Policz265Zielinski, NieGra265Malinowski, Jak3Krzyzanowski*. Chodzi o to, żeby w tytule e-maila i w nazwach wszystkich załączników była zarówno informacja o zadaniu, jak i o Autorze. Bardzo też proszę, żeby jeden e-mail zawierał rozwiązanie tylko jednego konkursu, a nie kilku, co mi znacznie ułatwi segregowanie poczty.

Regularnie potwierdzam otrzymanie rozwiązań, nadsyłanych e-mailem. Jeśli w terminie dziesięciu dni nie otrzymacie mojego potwierdzenia, prześlijcie pliki jeszcze raz (do skutku).

Bardzo proszę, by każdy uczestnik zadania głównego podawał **imię, nazwisko, adres zamieszkania oraz rok urodzenia, a w przypadku uczniów także informacje o szkole i klasie, do której uczęszcza**. Jest to pomocne przy opracowywaniu rozwiązań, ocenie prac oraz wysyłce upominków, nagród i dyplomów (dane osobowe będą wykorzystane wyłącznie w związku z oceną prac i nagrodami). Jeśli na łamach czasopisma nie chcecie ujawniać swoich danych – napiszcie, a zachowam dyskrecję, podając albo pseudonim, albo imię i pierwszą literę nazwiska, ewentualnie miejscowość zamieszkania. Autorzy rozwiązań zadania głównego, jeśli chcą, mogą też przysłać fotografie swojej osoby (portret), które będą zamieszczone przy rozwiązaniu zadania.

Mam też prośbę dotyczącą kwestii technicznych. Nie umieszczajcie ilustracji w tekście! Wszystkie ilustracje (fotografie i rysunki) powinny być przesłane jako oddzielne pliki. Bardzo proszę też o przysyłanie schematów, projektów płytek i wszelkich innych rysunków w popularnych formatach, na przykład PDF, JPG, GIF czy PNG, i to także wtedy, gdy przysyłacie oryginalny, źródłowy plik z danego programu projektowego (sch, pcb, brd, ddb, itp.).

Uwaga! Osoby nadsyłające opisy swoich rozwiązań praktycznych proszone są, żeby tekst redagowały według wskazówek z EdW 7/2011. Pomocny szablon oraz inne materiały pomocnicze są umieszczone pod adresem: www.elportal.pl/szablon.

Wystarczy przysłać e-mailem postać elektroniczną rozwiązania, nie jest konieczny papierowy wydruk ani płyta CD/DVD. Ale jeżeli ktoś pisze tekst na komputerze i przysłał do mnie wydruk w kopercie, to niech także przysła e-mail z plikiem tekstowym (.DOC, .TXT, .ODT), co znacznie ułatwi zacytowanie całości lub fragmentu rozwiązania oraz przygotowanie do ewentualnej publikacji. Jeśli jednak nadsyłacie w paczce model lub płytę z dokumentacją, zawsze dołączajcie papierowy wydruk własnoręcznie podpisanego i opatrzonogo datą oświadczenia: *Ja, niżej podpisany, oświadczam, że projekt/artykuł pt.:, który przysyłam do redakcji „Elektroniki dla Wszystkich”, jest moim osobistym opracowaniem i nie był wcześniej nigdzie publikowany.*

Jeśli natomiast przysyłacie fotografie modelu pocztą elektroniczną, takiej samej treści oświadczenie powinno się znaleźć w treści e-maila.

Zadanie główne nr 265

Propozycję zadania nadesłał kilka miesięcy temu **Rafał Jasiński**. Napisał między innymi: (...) *proponuję zadanie dla domu. U nas na wsi w okolicach Płocka co jakiś czas zdarzają się wyłączenia i przerwy w dostawie energii. Można postawić zadanie Szkoły Konstruktorów, żeby uczestnicy zaproponowali, jak można zmniejszyć nieprzyjemne skutki braku prądu w sieci (...)*

Zaproponuj rozwiązanie elektroniczne, zmniejszające skutki braku energii w sieci 230V.

Znów zadanie jest bardzo szerokie. Z jednej strony mogą to być pomysły, dotyczące awaryjnego oświetlenia. Można zaproponować budowę dodatkowej instalacji. Ale pomysły niekoniecznie mają dotyczyć całego mieszkania. A może dobrym rozwiązaniem byłoby posiadanie pewnej liczby lampek z diodami LED i akumulatorami z odzysku? Trzeba byłoby tylko zadbać, żeby akumulatory były zawsze odpowiednio naładowane. A to już zadanie dla elektronika.

Pokrewnym tematem jest zapewnienie ładowania i zasilania smartfonów, table-

tów i laptopów. W wielu wypadkach sprawy nie rozwiąże niewielki PowerBank. Co proponujecie w tym zakresie?

Uwaga!

Każdy Autor, nadsyłając rozwiązanie zadania głównego, może dołączyć też swoją fotografię (portret). Fotografia zostanie opublikowana w artykule, omawiającym nadesłane rozwiązania.

W EdW zaczęliśmy omawiać tematykę inteligentnego domu. I właśnie treść zadania 265 ściśle wiąże się też z inteligentnym domem. Inteligentny dom musi mieć rezerwowe źródło zasilania. Możliwości są różne. Jedną kwestią to akumulator czy akumulatory. Ale druga nie mniej ważną to pytanie, jak te akumulatory ładować i jak utrzymywać w gotowości. Wbrew pozorom, nie są to zadania najłatwiejsze.

Oczywiście Wasze pomysły mogą zawierać propozycje kompleksowego roz-

wiązania problemu przerw w dostawie energii, łącznie z podtrzymaniem pracy lodówki, zamrażarki i pieca centralnego ogrzewania. Mogą to być tylko pomysły. Ale jeżeli ktoś ma jakiegokolwiek, dobre lub złe doświadczenia w tym zakresie, niech przedstawi je w ramach tego zadania.

Rozwiązanie zadania 265 może też dotyczyć tylko jednego, nawet bardzo wąskiego aspektu, związanego z brakami energii w sieci. Oprócz pomysłów związanych z wykorzystaniem rezerwowych

źródeł energii, można też przedstawić propozycje, które pośrednio wiążą się z brakiem prądu w sieci. Na przykład można zaproponować układ informujący nieobecnych domowników, że nastąpiło wyłączenie prądu. Pomyślcie też nad innymi kwestiami, wiążącymi się z brakiem prądu w sieci. Jak zawsze, czekam też na wszelkie propozycje teoretyczne, za które także przydzielam punkty, nagrody i kupony.

Zapraszam do udziału!

Nadsyłajcie propozycje zadań!

Autorzy propozycji zadań, które zostaną wykorzystane w Szkole, otrzymują jako nagrodę kupon 100zł na zakupy w sklepie AVT: www.sklep.avt.pl.

Koszty przesyłki pokrywa AVT.

Dobra propozycja nie powinna być ani zbyt trudna, ani zbyt ogólna, ani zbyt wąsko ukierunkowana.

Dobre zadanie Szkoły powinno mieć na tyle szeroki zakres, żeby mogli w nim wziąć udział zarówno doświadczeni elektronicy, jak i początkujący, w tym najmłodszy.

Zachęcam do nadsyłania propozycji następnych zadań Szkoły!

Rozwiązanie zadania głównego 260

Temat październikowego zadania 260 brzmiał: **Zaproponuj nietypowe, „elektroniczne” wykorzystanie fabrycznego kalkulatora lub jego części.**

Zadanie było specyficzne, miało dość wąski zakres. Tym bardziej cieszę się z nadesłanych propozycji i rozwiązań.

Na początek krótki e-mail od **Macieja Kowalczyka** z Gdańska: (...) można wykorzystać kalkulator jako licznik o bardzo dużej pojemności do różnych zadań. Wystarczy dodawać jeden. Jak kto chce, to można liczyć do tyłu.

Rzeczywiście, to chyba najprostszy sposób wykorzystania kalkulatora.

Krzysztof Kasprzak napisał: Witam. Przeglądając EdW, natknąłem się na to zadanie z kalkulatorem. Przypomniałem sobie, że posiadam urządzenie DIY, które zrobiłem lata temu. Pierwszym zastosowaniem kalkulatora jest, cóż by innego, jak nie nawijarka do transformatorów. Jak to działało? Wraz z karkasem transformatora obracał się element przechodzący przez transoptor szczelinowy. Sterował on przekaźnikiem kontaktowym zwierającym klawisz „=” w kalkulatorze. W kalkulatorze wcześniej wciskaliśmy „1+”. Sam mózg nawijarki to sterownik wykonany na układach TTL, regulacja mocy na tyrystorze. Dodatkowo także złącze do sterowania z PC przez port równoległy, ale szczerze, to nie pamiętam, czy to kiedykolwiek uruchomiłem od strony PC.

Drugim zastosowaniem starego kalkulatora jest użycie go jako dawcy części. Tu akurat miałem spalony polski kalkulator z wyświetlaczem lampowym. Część płytki z tego kalkulatora wraz z przetwornicą, wyświetlaczem i konwerterem poziomów użyłem w częstościomierzu TTL (schemat pochodził z „Radioelektronika”). Włączyłem go po latach odpoczynku i... działa :) Pozdrawiam

St erownik nawijarki pokazany jest na **fotografii 1**. Natomiast **fotografia 2** przedstawia dwa ujęcia częstościomierza. Choć układy powstały już wcześniej, przydzielam Autorowi i punkty, i kupon.

St ały uczestnik Szkoły, **Daniel Turbasa** z Krakowa, tym razem napisał: Zadanie pierwszej klasy Szkoły Konstruktorów dotyczące wykorzystania kalkulatorów lub jego podzespołów z jednej strony było bardzo proste, z drugiej jednak okazało się niezmiernie skomplikowane (...) poprzez



Fot. 1 Sterownik nawijarki Krzysztofa Kasprzaka

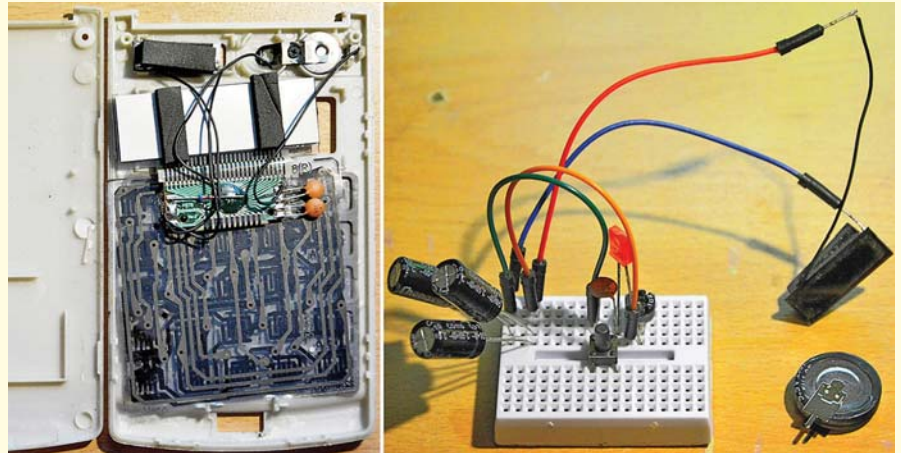
Fot. 2 Częstościomierz Krzysztofa Kasprzaka



miniaturyzację aktualnie wytwarzanych kalkulatorów (...) Niestety płytki z mikrokontrolerem, ze względu na brak informacji, nie można wykorzystać w dalszych projektach. Podobnie rzecz się ma z wyświetlaczem (...) pozostaje klawiatura oraz ogniwo słoneczne. Ja skupiłem się na tym drugim. (...) sprawdziłem parametry ogniwa (...) z krzemu amorficznego (...) o sprawności kilka procent, jednak zaletą tego typu ogniw jest możliwość pracy w szerokim zakresie długości fali światła (...) przy sztucznym oświetleniu prąd zwarcia ogniwa wynosi około 1mA, napięcie (...) 3V. Moc ogniwa (...) około 3mW. Nie jest to dużo, ale...

(...) Ogniwo zamontowane jest równolegle do baterii zasilającej, która ma dodatkowo diodę zabezpieczającą przed przepływem prądu z ogniwa fotowoltaicznego do baterii. Samo ogniwo również powinno zostać zabezpieczone w ten sposób, gdyż nieoświetlone będzie stanowić dodatkowe obciążenie dla zastosowanej baterii. Część ogniwa ma taką diodę w swojej strukturze, toteż nie jest ona widoczna (...) energia słoneczna nie może zostać w żaden sposób zgromadzona i wykorzystana w późniejszym terminie. Moją propozycją (...) jest zastosowanie dodatkowego kondensatora, który gromadziłby energię w chwili, gdy urządzenie nie jest aktualnie używane, a panują dobre warunki oświetleniowe. (...) można właściwie zrezygnować z dodatkowego zasilania w postaci baterii, wykorzystując jedynie energię zgromadzoną w kondensatorze oraz dostarczaną przez ogniwo fotowoltaiczne.

Podczas testów natknąłem się na kilka ważnych aspektów problemu. (...) szybki spadek napięcia kondensatora (...) wymaga stosowania elektroniki o szerokim zakresie napięć zasilania (...) opcjonalne źródło zasilania w postaci baterii powinno mieć napięcie znacznie niższe od maksymalnego napięcia, które może wystąpić na kondensatorze. (...) rozwiązaniem [jest] zwiększenie pojemności kondensatora dołączonego równolegle do ogniwa, jednak nie można przesadzać z jego wartością. Rozładowany kondensator praktycznie stanowi zwarcie (...) jego napięcie w początkowej fazie ładowania będzie bliskie zeru. (...) Proste testy (...) pozwoliły na zasymulowanie pracy okresowej z różną wartością pobieranego prądu. Dla małych wartości prądu oraz w odpowiednio dużym okresie układ do pracy nie wymaga dodatkowego zasilania. Wzrost prądu lub skrócenie impulsów powoduje szybsze rozładowanie kondensatora oraz wymaga dodatkowego zasilania w postaci baterii. Również w przypadku braku oświetlenia wymagane jest dodatkowe źródło energii. Jednak urządzenie

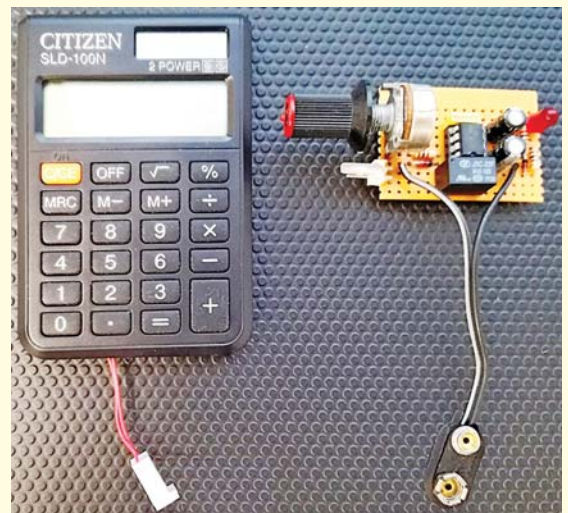


Fot. 3 Układ testowy Daniela Turbasy

powinno pozwolić na zaoszczędzenie czasu pracy na baterii nawet dwukrotnie (...) Jako przykład wykorzystania można podać układ z czujnikiem, który okresowo zbiera informacje. Przez większość czasu kondensator może się ładować, by co pewien okres oddać część energii dla aktywnie pracującego urządzenia. Większość nowoczesnych czujników oraz procesorów ma odpowiednie tryby uśpienia oraz tryby niskiego poboru energii podczas pracy, co pozwala na znaczną minimalizację zużycia energii. (...) Fotografia 3 pokazuje kalkulator i układ testowy zawartego w nim miniogniwa słonecznego. Rzeczywiście, ogniwa fotowoltaiczne z powodzeniem wykorzystywane są do zasilania różnego rodzaju nowoczesnych czujników. Autorowi przydzielam punkty i kupon.

Krzysztof Smoliński z Poznania tym razem napisał: Witam, z racji tego, że nie chciałem zniszczyć całkowicie kalkulatora, który jest mi czasem przydatny, postawiłem sobie dodatkowe założenie konkursu: ingerencja w kalkulator nie może pozbawić go podstawowych funkcji. Założenie to pozbawiło mnie możliwości realizacji kilku pomysłów, które rzucam jako teoretyczne:

- urządzenie do nauki liczenia: wyświetlany jest wynik oraz jeden z czynników lub składników (w zależności, czy uczymy się tabliczki mnożenia, czy dodawania). Zadaniem użytkownika jest podanie drugiego składnika.



Fot. 4 Eksperymenty Krzysztofa Smolińskiego

Fot. 5 Kalkulator z impulsatorem



- Mastermind – urządzenie losuje hasło, a użytkownik próbuje je zgadnąć. Po każdej próbie jako ciąg zer i jedynek wskazane jest, które cyfry są na właściwych pozycjach.

- urządzenie do ćwiczenia pamięci: komputer losuje coraz dłuższe hasło,

a zadaniem użytkownika jest powtórzenie go. W ten sposób należy zapamiętać rosnący ciąg cyfr:

Wszystkie powyższe propozycje wymagałyby dużej ingerencji w kalkulator i w zasadzie wykorzystanie jedynie jego wyświetlacza i klawiatury.

Moja propozycja to **uniwersalny licznik**. Może on służyć jako licznik zdarzeń, krokomierz, licznik przebytej trasy na rowerze lub licznik czasu. W kalkulatorze wyprowadziłem na zewnątrz tylko styki odpowiedzialne za klawisz „=”. Tak więc jego podstawowa funkcjonalność nie została zachwiana. Resztę załatwia operator lub odpowiednia przystawka. Oto zastosowania:

- licznik zdarzeń (...) Wystarczy, by każde zdarzenie wyzwalalo zwarcie wyprowadzonych styków. (...)

- krokomierz/licznik rowerowy – montujemy magnes na kole oraz kontaktny na ramie roweru (...) Uruchamiając kalkulator, podajemy, ile wynosi obwód okręgu, po jakim będzie się poruszał magnes i wciskamy „+”. (...) Kalkulator ma zasilanie słoneczne, więc nie ma potrzeby podłączania dodatkowego zasilania!

- licznik czasu – do tego zastosowania przygotowałem przystawkę. Jest to prosty generator oparty na kostce 555 z miniaturowym przekładnikiem podłączonym do jego wyjścia. Potencjometr umożliwi regulację w zakresie kilkunastu ms do kilku s, jeżeli wymagane byłoby zliczanie impulsów innych niż 1s. Dodatkowo na kalkulatorze przy uruchomieniu możemy podać odpowiednią podziałkę, by wynik podawany był w innych jednostkach (godzinach, dniach, tygodniach).

Rozwiązanie z pomiarem czasu jest całkiem przyjemne, natomiast wykorzystanie kalkulatora w rowerze pozostawiam sobie do wiosny, kiedy to ponownie wyciągnę rower z garażu :)

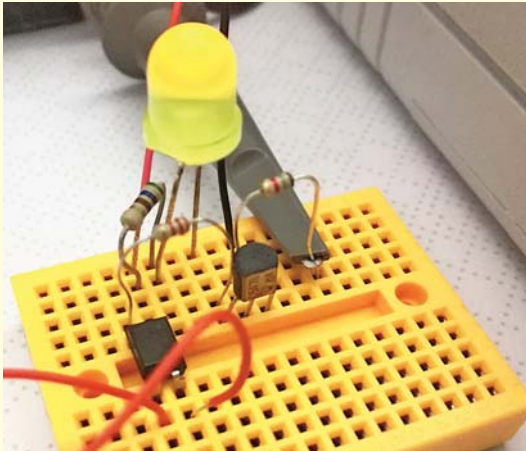
Na **fotografii 4** widoczna jest wspomniana przystawka i zmodyfikowany kalkulator. Autorowi przydzieliłem punkty oraz kupon.

Kolejny stały uczestnik Szkoły, **Dawid Placha**, napisał: (...) Pomysł wykorzystania kalkulatora (...) wydawał się bardzo interesujący. Niestety, rekonesans wykazał że lepszym rozwiązaniem jest zakup telefonu komórkowego z większymi

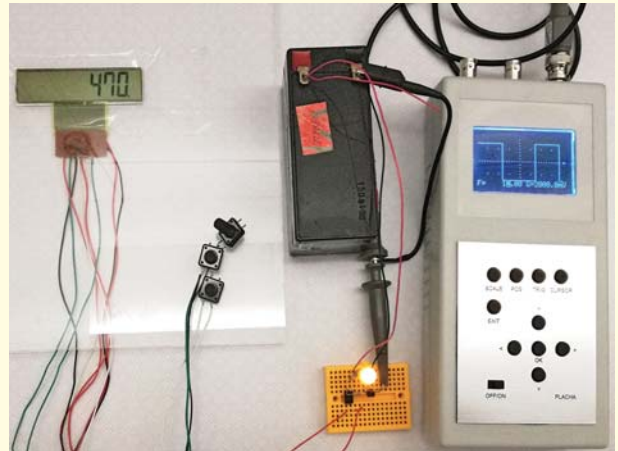
możliwościami. Najtańszy nowy aparat możemy kupić za kilkanaście złotych, to praktycznie tyle samo, ile wydalibyśmy na kalkulator. Testy:

Pierwszym krokiem było rozkręcenie obudowy kalkulatora i podpięcie impulsatora do styków sumy (=) [**fotografia 5**]. Zadałem wartość 1+, i kręcąc osią impulsatora, odczytywałem zliczane impulsy. Test wykazał że nie każdy impuls jest zliczany, co miało związek z prędkością kręcenia. W dalszym etapie, na płycie prototypowej zainstalowałem transceptor PC817 używany tranzystorem NPN [**fotografia 6**] sterowanym z wyjścia generatora („Kombajn pomiarowy” AVT 2999) [**fotografia 7**]. Testy wykazały, że maksymalną częstotliwością zliczania impulsów jest 11Hz. Powyżej tej wartości liczydło gubi kroki, a przy częstotliwości powyżej 18Hz układ już nie zlicza.

Graniczna częstotliwość 11Hz wyklucza budowę projektów pomiarowych, takich jak kątomierz lub miernik odległości liniowej zbudowany na suwaku, połączony sznurkiem z impulsatorem. Ten ostatni pomysł próbowałem zasymulować, korzystając z przerzutników i bramek, ponieważ przy każdej zmianie kierunku należy zadać znak i wartość sumy. Jeżeli kalkulator ma tylko sumować, to śmiało można wykorzystać go w formie krokomierza, dodając magnes z kontraktronem lub w formie licznika osób, wykorzystując optobarierę. (...)



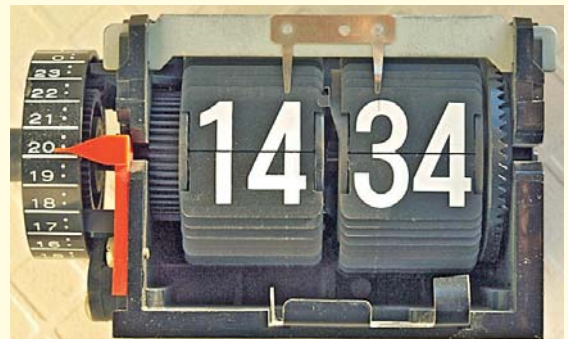
Fot. 6 Układ sterujący Dawida Placha



Fot. 7 Eksperymenty Dawida Placha

Dodatkowe fotografie dokumentujące przeprowadzone eksperymenty dostępne są w Elportalu wśród materiałów dodatkowych do tego numeru EdW (260_placha.zip 1,82MB).

A oto treść e-maila, nadesłanego przez **Michała Stacha** z Kamionki Małej: Dzień dobry! Temat SzK260 dotyczący wykorzystania kalkulatora mocno nie pasował do bieżących spraw. Do kalkulatora, który mam, czuję duży sentyment. (...) Przeznaczenie go na eksperymenty byłoby nadużyciem. Do tematów okołokalkulatorowych można z trudem zaliczyć zagospodarowanie wyświetlacza kłapkowego.

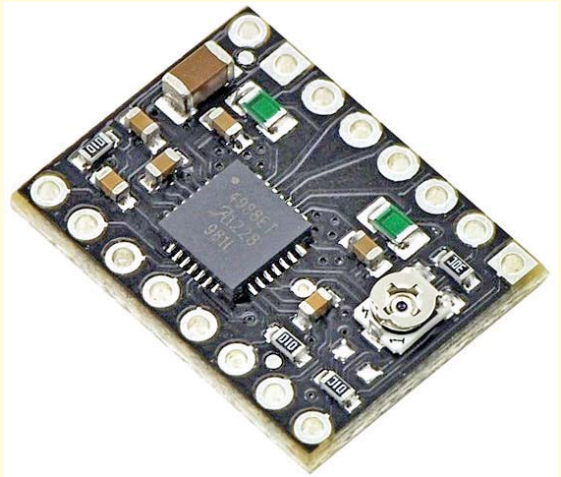


Fot. 8 i 9 Wyświetlacz „kłapkowy”





Publika-cja	Nag-roda	Talon AVT PLN	Imię	Nazwisko	Miejscowość	Punkty
-	-	100	Rafał	Jasiński	Płock	-
-	-	-	Maciej	Kowalczyk	Gdańsk	1
-	-	80	Krzysztof	Kasprzak	?	6
-	-	80	Daniel	Turbasa	Kraków	5
-	-	80	Krzysztof	Smoliński	Poznań	5
-	-	180	Dawid	Placha	Rdzawa	6
?	-	80	Michał	Stach	Kamionka Mała	4



Fot. 10 Moduł sterowania silnikiem

Pokazany na **fotografii 8** wyświetlacz (jak to inaczej nazwać – wskaźnik mechaniczny?) ma tyle wspólnego z kalkulatorem, że pokazuje cyfry. Jednak cel jest szczytny: przywrócić zegar do życia. Poza oryginalnym wyglądem, taki wskaźnik ma kilka ciekawych cech (...) unikalny dźwięk mijających minut (...) sposóbysterowania kłapek godzinowych. Klapka następnej godziny zwalniana jest wcześniej, ale do końca przytrzymuje ją wypustka na klapce 59 minuty. Szczegóły pokazane są na **fotografii 9**. Zegary tego typu napędzane i synchronizowane są siecią. (...) Chciałbym wykorzystać silnik krokowy z przekładnią, sterowany z układu pokazanego na **rysunku 1**. Tranzystor T1 odzyskuje podwojoną częstotliwość sieci i podaje ją na wejście modułu A4988 (**fotografia 10**) (...) Moduł udostępnia kilka funkcji, które można konfigurować poprzez DIP-SW1. Zapewne wybrane

zostanie sterowanie mikrokontrolerowe 1/16. Przy założeniu, że na wejściu STEP sterownika pojawi się 6000 imp./min, a na wyjściu oczekujemy 1 obr./min, to przy pracy w 1/16 kroku przekładnia silnika krokowego będzie musiała mieć (rozsądne i dostępne) przełożenie 1:375. Inna opcja to wstawienie pomiędzy emiter T1 a wejście modułu A4988 licznika, najlepiej „epokowego” z serii UCY (żadnych mikroprocesorów w sprzeczności retro). (...)

Rozwiązanie traktujemy na razie jako teoretyczne i nie do końca spełniające warunki zadania, niemniej przydzielam i punkty, i kupon, który pomoże zrealizować finalną wersję sterownika. A swoją drogą, warto byłoby zastanowić się, czy nie zapewnić rezerwowego zasilania i zrezygnować z sieci jako źródła częstotliwości wzorcowej? Takie rozwiązanie miałoby

zalety – autonomię i odporność na zaniki napięcia sieci, ale dokładność mogłaby być gorsza i potrzebne byłyby baterie rezerwowe (lub akumulator) o pojemności odpowiedniej nie tylko do zasilania elektroniki, ale i silnika krokowego.

Niech rozwiązania zadania 260 będą zachętą do nietypowego wykorzystywania kalkulatorów. Jeśli ktoś z powodzeniem wykorzysta tak kalkulator, niech przedstawi swoje dzieło na łamach EdW.

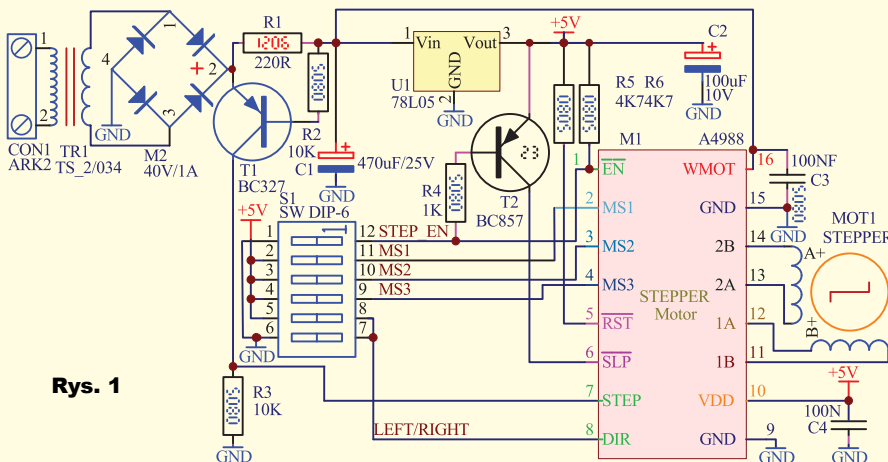
Aktualne informacje o punktacji oraz rozdziale nagród, upominków i kuponów podane są w tabelkach. Znak zapytania oznacza, że ewentualna publikacja nastąpi dopiero po nadesłaniu ostatecznych materiałów (przygotowanych według wskazówek ze strony www.elportal.pl/szablon).

Osoby nagrodzone kuponami otrzymują z naszej redakcji stosowny e-mail z informacją i wskazówkami, a dopiero potem zamawiają w sklepie AVT (wrzucają do koszyka pod adresem www.sklep.avt.pl) towary za przydzieloną sumę, a w uwagach piszą, że jest to kupon ze Szkoły Konstruktorów. Kupon za zadania z kolejnych miesięcy można sumować, by kupić sprzęt o większej wartości. Istnieje też możliwość dopłaty różnicy cen w przypadku zamówienia na sumę większą niż przydzielony kupon. Ale uwaga: Kupon ważny jest tylko 12 miesięcy – po tym terminie traci ważność i przepada.

Serdecznie zapraszam do udziału w zadaniu głównym 265, a także w drugiej i trzeciej klasie naszej Szkoły Konstruktorów! Zachęcam uczestników, żeby praktyczne rozwiązania zadań Szkoły przygotowywali według Szablonu ze strony

<http://elportal.pl/zostan-wspolautorem-elektroniki-dla-wszystkich/>

Wasz instruktor
Piotr Górecki



Rys. 1

Punktacja Szkoły Konstruktorów

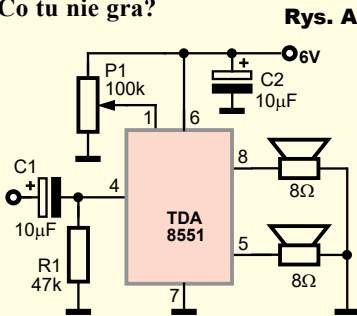
Antoni Pajewski Warszawa.....	101	Adam Sobczyk Warszawa.....	48	Andrzej Adameczek Ostrowiec Św.	7
Jacek Konieczny Poznań.....	95	Lukasz Olszok Tarnowskie Góry.....	45	Tomasz Zaorski Kalinówka.....	7
Lukasz Dachowski Cymbark.....	72	Paweł Hoffmann Wrocław.....	44	Tadeusz Suszał Warszawa.....	7
Dawid Placha Rdzawa.....	70	Szymon Czepiel Pisarzowice.....	43	Radosław Borowicz Poznań.....	7
Aleksander Bernaczek Magnuszowice.....	69	Michał Pędzimaz Stara Słupia.....	41	Teodor Woźniak Łódź.....	7
Daniel Turbasa Kraków.....	67	Piotr Gajdosz Grybów.....	41	Piotr Olszewski Goldap.....	6
Szymon Trygar Szczecin.....	66	Krzysztof Kawa Lubcza.....	38	Adam Czerniawski ?.....	6
Slawomir Węgrzyn Dziekanowice.....	64	Krzysztof Smoliński Poznań.....	25	Lukasz Seweryn Wodzisław Śl.....	6
Artur Bereit Barcin.....	60	Marcin Dobrogowski Gajowniki.....	18	Dariusz Zbyrad Tamobrzeg.....	6
Robert Szolc Bytom.....	58	Marian Caruk Luban.....	17	Michał Świątański Nakło.....	6
Michał Stach Kamionka Mała.....	56	Jakub Gajda Kraków.....	15	Artur Gubernat ?.....	6
Paweł Sablik Pisarzowice.....	52	Przemysław Michałak Kępkowo.....	14	Radosław Rongers Warszawa.....	6
Andrzej Herbut Siekierzyn.....	52	Marcin Malich Wodzisław Śl.....	13	Marian Gabrowski Polkowice.....	6
Sebastian Jarmosiewicz Motwica.....	50	Roman Braumberger Bytom.....	12	Piotr Świerczek Bielsko-Biała.....	6
Radosław Smałek Zabrze.....	49	Rafał Rówiak Słaboszów.....	7	Krzysztof Kasprzak ?.....	6

Co tu nie gra?

NieGra265

Na **rysunku A** przedstawiony jest schemat wzmacniacza. Jak zwykle pytanie brzmi:

Co tu nie gra?



Nawet gdy w układzie jest kilka usterek, wystarczy zgłosić jedną. Bardzo proszę o możliwie krótkie odpowiedzi. Kartki, listy i e-maile oznaczcie dopiskiem *NieGra265* i nadeślijcie w terminie 60 dni od ukazania się tego numeru EdW. W e-mailach podawajcie też od razu swój adres pocztowy, żebym nie musiał pisać, gdy przydzielę upominek. Można też jeszcze przysyłać rozwiązania poprzedniego zadania 264. Autorzy najlepszych odpowiedzi otrzymają upominki, a najaktywniejsi uczestnicy są okresowo nagradzani bezpłatnymi prenumeratami EdW lub innego wybranego czasopisma AVT.

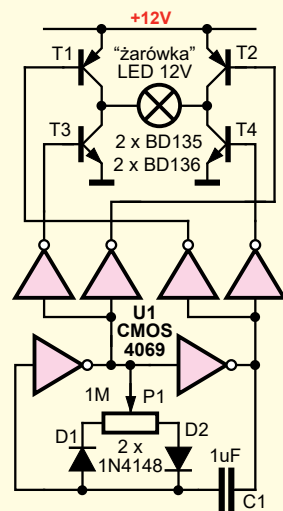
Rozwiązanie zadania NieGra260

Na **rysunku B** pokazany jest zamieszczony w EdW 10/2017 schemat regulatora do 12-woltowej „żarówki” LED, zawierający generator o zmiennym wypełnieniu i mostek czterech tranzystorów dla uzyskania napięcia zmiennego na „żarówce”.

Oczywiście na schemacie doszukaliście się szeregu usterek.

Zacznijmy od dołu schematu. Mamy tu generator o zmiennym współczynniku wypełnienia. Wypełnienie można zmieniać potencjometrem w bardzo szerokim zakresie, praktycznie od zera do 100%. W tym na pozór prawidłowym generatorze

Rys. B



śluszenie doszukaliście się dwóch usterek. Po pierwsze wartości elementów P1 i C1: rezystancja 1 megaohm i pojemność 1 mikrofarada niewątpliwie dadzą bardzo niską częstotliwość generatora rzędu 1 herca. W najlepszym razie otrzymamy migacz, ale nie regulator jasności. Przy impulsowej regulacji jasności źródeł światła częstotliwość sygnału PWM powinna wynosić co najmniej 100Hz, a lepiej więcej (ale nie za dużo z uwagi na zakłócenia impulsowe).

Dla porządku można nadmienić, że regulacja potencjometrem współczynnika wypełnienia będzie też powodować niewielkie zmiany częstotliwości, ale to nie ma znaczenia.

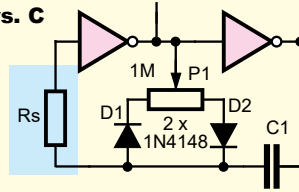
Nie jest tu potrzebne wykorzystanie popularnej kostki CMOS 40106 z obwodami histerezy na wejściach. To byłoby wręcz błędem, bo byłby kłopot z regulacją wypełnienia o skrajnych wartościach. Nie jest błędem użycie niemal zapomnianych dziś inwerterów z kostki 4069 i to także ich popularniejszej wersji 4069UB, czyli bez buforowania (gdzie każdy inwerter składa się tylko z dwóch tranzystorów komplementarnych CMOS).

Drugą usterką w układzie dwubramkowego generatora z bramkami CMOS jest brak rezystora R_s „na wejściu pierwszej bramki” według **rysunku C**.

Słabo zorientowani mogą zaprotestować: *po co szeregowy rezystor na wejściu CMOS, skoro wiadomo, że wejścia CMOS praktycznie nie pobierają prądu?*

Owszem, w warunkach ustalonych podczas „normalnej pracy” w obwodach wejść CMOS prądy nie płyną. W przypadku tego rodzaju generatora jest inaczej, ponieważ gwałtowna zmiana, skok napięcia w punkcie Y „wypycha” drugi zacisk

Rys. C



naładowanego kondensatora C1 (punkt X) „poza szyny zasilania”. I wtedy impulsy prądu rozładowującego kondensator płyną do szyn zasilania przez struktury diodo-

we zabezpieczające wejścia przed ładunkami statycznymi, jak to jest pokazane na **rysunku D**. Amplituda takich impulsów prądowych jest wyznaczona głównie przez wydajność

wyjścia drugiej bramki.

Taki rezystor szeregowy R_s nie jest potrzebny w układach CMOS rodziny 4000 przy napięciach zasilania do 5V z uwagi na małą wydajność wyjścia. Przy wyższym napięciu zasilania wydajność ta jest większa niż podany w katalogu maksymalny prąd wejść CMOS. Dlatego przy napięciu zasilania powyżej 5V dobrym zwyczajem jest stosowanie takiego rezystora R_s , by ograniczyć prąd, płynący w obwodach wejściowych. W rzeczywistości nie występują tam diody, tylko dość skomplikowane struktury tranzystorowe, podobne do układu zastępczego tyrystora. Zbyt duży impuls prądu może włączyć taki „tyrystor”. W tym wypadku napięcie zasilania generatora niewątpliwie wynosi 12V, jednak praktyka pokazuje, że układ generatora zapewne będzie też pracował bez rezystora (bowiem we współczesnych układach CMOS 4000 próg włączenia struktur tyrystorowych jest znacznie wyższy niż podany w katalogu dopuszczalny prąd wejść).

Omówiliśmy generator, który ma wytwarzać przebieg prostokątny o zmiennym wypełnieniu. Kolejną sprawą to działanie mostka tranzystorowego. Zasada jest taka, że na pewno nie mogą jednocześnie przewodzić tranzystory „w jednej kolumnie”, czyli T1+R3 ani T2+T4. Podstawowe trzy przypadki do analizy to:

- zatkanie wszystkich tranzystorów (**rysunek E**),
- przewodzenie „po przekątnej” T1 + T4 (**rysunek F**),
- przewodzenie „po przekątnej” T2 + T3 (**rysunek G**).



Można powiedzieć, że tranzystory z jednej strony mostka sterowane są takim samym stanem logicznym. Oczywiście ewidentnym błędem byłoby zwarcie baz według **rysunku H**, bo spowodowałoby to otwarcie wszystkich czterech tranzystorów i zwarcie zasilania. Problemu takiego nie byłoby w wersji z wtórnymi według **rysunku J**, ale tu wadą byłaby niemożliwość pełnego otwarcia tranzystorów i w efekcie napięcie na obciążeniu byłoby znacząco niższe od napięcia zasilania, zapewne znacznie więcej niż o 2 volty.

Rysunki F, G pokazują w uproszczeniu, jakie mają być napięcia (raczej stany) na bazach tranzystorów. Oczywiście w rzeczywistości na przewodzącym złączu baza-emiter tranzystora wystąpi napięcie co najwyżej 1V. Właśnie z tego powodu nie można wykorzystać wersji z rysunku H, tylko bazami trzeba sterować indywidualnie. Dlatego też obecność aż czterech inwerterów sterujących nie jest błędem. Słusznie

zauważyliście jednak, że w obwodach baz nie ma rezystorów ograniczających, co powoduje, że wyjścia tych czterech inwerterów pracują praktycznie w stanie zwarcia, płynie tam dość duży prąd, co może spowodować przegrzanie kostki 4069, ale nie musi: to będzie zależać od właściwości konkretnego egzemplarza układu scalonego. W niektórych katalogach przedstawione są wydajności prądowe wyjść wskazujące, że przy zasilaniu 12V typowy prąd wyjściowy w takich warunkach wyniesie około 20mA. To nie byłoby groźne. Ale w pewnych egzemplarzach niektórych producentów prąd ten może być znacznie

większy, że brak rezystorów szeregowych w obwodach baz jest więc problemem z uwagi na ryzyko przegrzania. Ale z drugiej strony prąd baz tranzystorów mostka nie może być za mały, żeby tranzystory te zostały skutecznie nasycone. Tranzystory BD135/BD136 mają maksymalny prąd kolektora 1,5 ampera. Warto jednak zauważyć, że już przy prądzie 0,5A gwarantowana wartość wzmocnienia prądowego wynosi tylko 25x. Jeśli chcielibyśmy pracować z prądem kolektora 500mA, to prąd bazy powinien być znacznie większy od 20mA. A część egzemplarzy kostki 4069 może przy napięciu zasilania 12V mieć mniejszy maksymalny prąd wyjściowy.

Przed chwilą martwiliśmy się, że niektóre egzemplarze kostki 4069 o większej wydajności prądowej mogą ulec przegrzaniu, a teraz zastanawiamy się nad problemem odwrotnym: obawiamy się, że inne egzemplarze będą mieć wydajność prądową za małą do prawidłowegoysterowania tranzystorów. Na rysunku B nie ma informacji o prądzie płynącym przez żarówkę i tranzystory, więc trudno ocenić, na ile problem jest realny. Część uczestników podkreślała, że większość 12-woltowych „żarówek LED” ma moc 3...5W, a wtedy

prądy są mniejsze niż 0,5A. W każdym razie takie rozwiązanie jest niepewne i grozi komplikacjami.

To nie koniec problemów związanych z obwodem sterowania. Jak pokazuje **rysunek K**, w jednej fazie cyklu generatora przewodzą tranzystory T1, T3. T2, T4 są zatkane. Prąd obciążenia nie płynie, ale za to T1, T3 praktycznie robią zwarcie w obwodzie zasilania. Analogicznie jest w drugiej fazie cyklu pracy generatora – **rysunek L**.

Połączenia bramek sterujących bazy są błędne. **Rysunek M** pokazuje połączenia po korekcie.

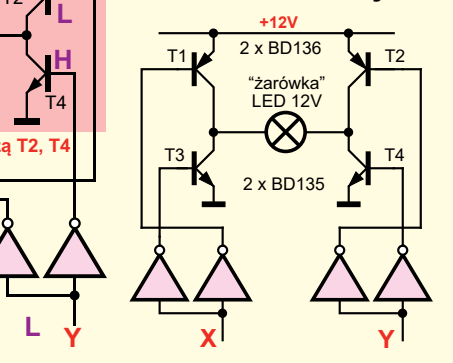
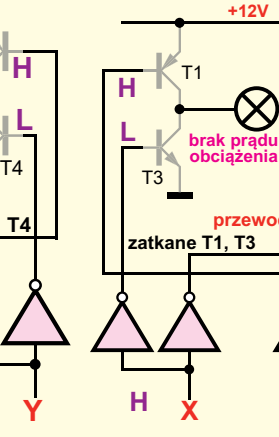
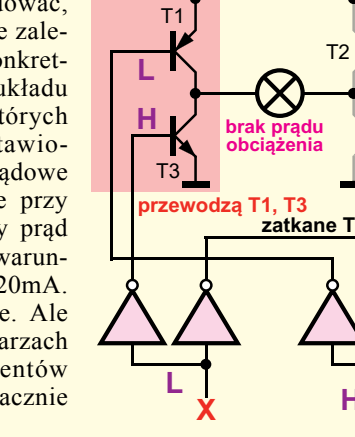
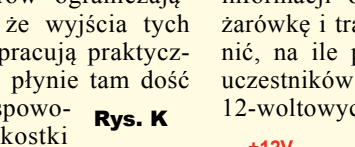
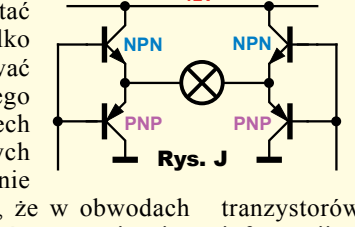
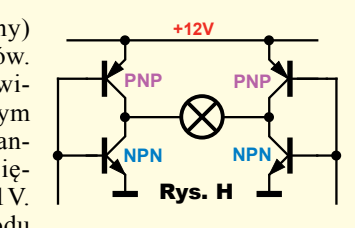
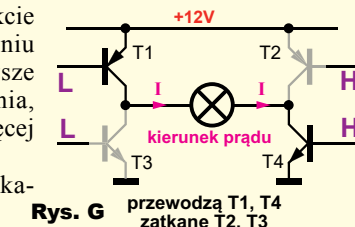
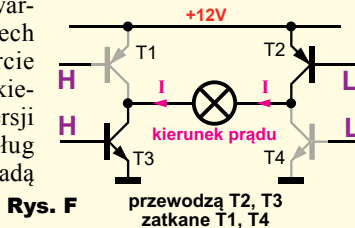
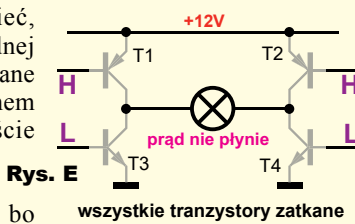
I wreszcie doszliśmy do najpoważniejszego błędu. Niektórzy uczestnicy mieli poważniejsze wątpliwości. Oto dwa przykłady:

- Nie do końca rozumiem intencję autora, dlaczego przelączenie kierunku zasilania żarówki (zakładam, że to (...) element niebiegunowy) ma powodować regulację jej świecenia (...).

- Przede wszystkim nie bardzo rozumiem sens zasilania „żarówki” 12V napięciem zmiennym, mając do dyspozycji odpowiednie napięcie stałe. Chyba nikt na świecie nie produkuje „żarówek” na 12V z kondensatorem na wejściu, jak to bywa w wypadku tanich „żarówek” zasilanych napięciem sieci. No dobrze, jak mówią starzy Ślązacy: po co lekko, jak ciężko też idzie.

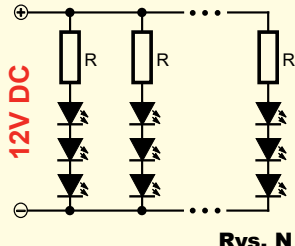
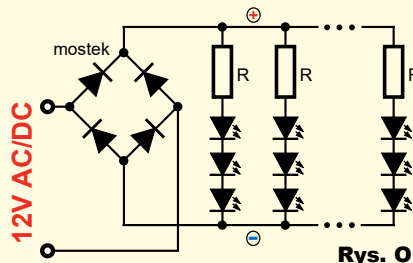
Tu muszę wyjaśnić, że analizowany schemat został przeze mnie przygotowany tylko na potrzeby tego zadania Szkoły. Niejako skondensowałem w tym prostym schemacie szereg błędów, często popełnianych przez hobbystów. Trudno sobie wyobrazić, że ktoś w jednym układzie popełni wszystkie „grzechy”, jakie zawiera układ, niemniej wszystkie omówione tu usterki i błędy można spotkać na schematach tworzonych nie tylko przez zupełnie początkujących.

W przedstawionych właśnie cytatach **Rys. L** zasygnalizowany jest kluczowy problem z rysunkiem B, mianowicie sens wykorzystania mostka steru-



jącego, składającego się z czterech tranzystorów. W omawianym przypadku obecność mostka jest ilustracją starego przysłowia: *dzwonią, tylko nie wiadomo, w którym kościele*. Mostki są bardzo przydatne, wręcz niezbędne, przy sterowaniu silników. Po pierwsze mostek pozwala w prosty sposób zmienić kierunek obrotów silnika, jak wskazują rysunki F i G. Po drugie, przy odpowiednim sterowaniu (modulacji PWM) można też płynnie regulować jego prędkość obrotową. Na przykład w ten sposób, że kierunek obrotów ustalić trwale włączając jeden z „górných” tranzystorów T1, T2, a szybkość regulować, doprowadzając przebieg o zmiennym wypełnieniu (PWM) do jednego z „przeciwstawnych” tranzystorów (T4, T3).

Mostki są wykorzystywane w różnego rodzaju konwerterach napięcia stałego na zmiennie, na przykład w UPS-ach i falownikach. W takich urządzeniach


Rys. N

Rys. O

mostek zasilany pojedynczym napięciem stałym pozwala wytworzyć napięcie przemiennie, które jest niezbędne dla wielu urządzeń, w szczególności dla klasycznych zasilaczy, zawierających transformator sieciowy.

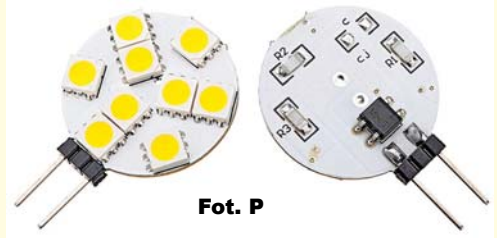
Ma to sens, gdy obciążeniem jest silnik lub jakieś urządzenie biegunowe, „dwukierunkowe”. W przypadku 12-woltowej żarówki nie można „zmienić kierunku świecenia” i już to stawia pod znakiem zapytania sens wykorzystania mostka.

Nasuwa się też pytanie, czy istnieją 12-woltowe żarówki lub elementy „żarówkopodobne”, które wymagają napięcia zmiennego,

a które nie będą pracować przy napięciu stałym albo pulsującym przebiegu jednokierunkowym?

Raczej nie!

Niektóre 12-woltowe żarówki z cokołem G4 (MR16) świecą tylko przy zasilaniu napięciem jednej biegunowości. Są zbudowane analogicznie do taśm LED: w szereg połączone są po trzy białe diody i rezystor ograniczający – przykład na **rysunku N**. Inne mają taką samą budowę obwodów LED, ale zawierają dodatkowo prostownik mostkowy według **rysunku O** i **fotografii P**. Jeszcze inne 12-woltowe „żarówki LED” zawierają


Fot. P

R E K L A M A

Magnetyzer - demagnetyzer

Niezbędny w każdym warsztacie!

- nie wymaga zasilania
- wymiary: 50x50x30mm


VTMD
7zł

Opaska magnetyczna na nadgarstek

Zwiększa wygodę i bezpieczeństwo pracy. Pozwala na szybki dostęp do bitów, wkrętów, gwoździ lub innych drobnych metalowych elementów.



Zapięcie na rzep.
Powierzchnia magnesu ok. 13cm²

HPUT4
10zł

Zestaw narzędzi precyzyjnych do napraw sprzętu GSM



- 5 wkrętaków precyzyjnych: gwiazdka 0.8mm i 1.2mm, PH 1.2mm i , 1.5mm, płaski 2mm
 - dwustronny otwierak plastikowy
 - 3 dwustronne otwieraki metalowe
 - pęseta
- Całość zapakowana w etui zamykane na magnes


GSM_KAISI
55zł

sklep.avt.pl handlowy@avt.pl tel.: 22 257 84 50



TRANSPORT z IQ czyli jazda bez trzymanki

„Sama się obsłuży, zasili i naprawi” – głosi śródtytuł w jednym z artykułów najnowszego Tematu numeru, poświęconego nowym modelom motoryzacji i transportu. Ową Zosią Samosią jest nie co innego, jak... droga. Tak, ten zwykły dotąd „kawałek asfaltu”, po którym jeździmy autem czy rowerem, a który prawdopodobnie już niedługo będzie współpracował z poruszającymi się po nim pojazdami nowej ery. Jak widać, inżynierowie i wynalazcy pracują dziś nie tylko nad konstrukcjami aut elektrycznych i autonomicznych, ale także nad całą służącą im inteligentną infrastrukturą. Czy zmieni to również zachowania użytkowników pojazdów, wpływając wręcz na wzorce cywilizacyjne? Zapraszamy do lektury „Transportu z wysokim IQ”!

Nowy numer już w sprzedaży
www.ulubionykiosk.pl
 Koniecznie odwiedź mobilny serwis na smartfony i tablety MT NEWS 24/7:
m.mlodytechnik.pl

przetwornicę i też mają „na wejściu” prostownik mostkowy – przykład na **fotografii R**. Wszystkie mogą pracować przy napięciu stałym. Ale **żadna nie wymaga zasilania napięciem zmiennym**.

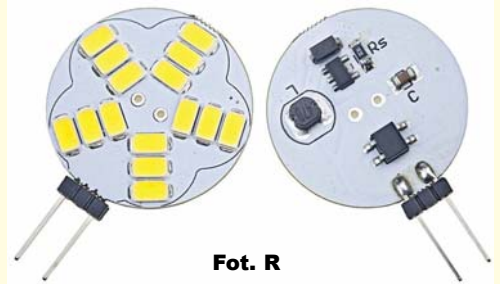
Upewnia nas to, że na rysunku B mostek z czterech tranzystorów w ogóle nie jest potrzebny.

Mało tego: on uniemożliwia regulację! Problem w tym, że w układzie z mostkiem regulacja potencjometru P1 nie zmienia wartości skutecznej napięcia wyjściowego. Niezależnie od kształtu przebiegu, na żarówce cały czas występuje napięcie praktycznie równe napięciu zasilania, tylko o różnej biegunowości. Wartość skuteczna czterech przykładowych przebiegów prostokątnych pokazanych na **rysunku S** jest... jednakowa. W tym przypadku zmiana „wypełnienia” nic nie daje.

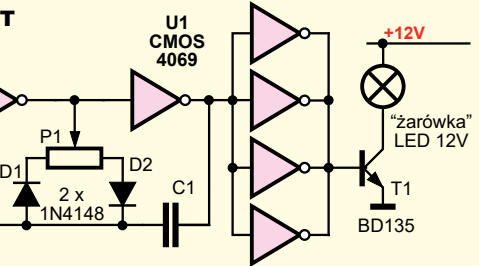
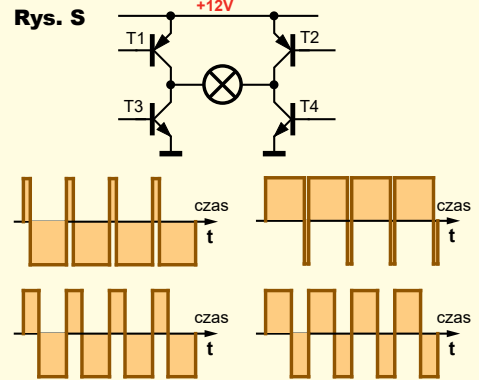
Paradoksalnie, układ z mostkiem tranzystorowym mógłby regulować jasność najprostszymi, „jednokierunkowymi żarówkami LED” bez mostka o schemacie według rysunku N. Tylko po co komplikować układ?

Do regulacji jasności 12-woltowych „żarówek ledowych” wystarczyłby układ według **rysunku T**.

Na pewno wersji zbudowanych tak, jak na rysunkach N, O. Jeżeli jednak „żarówka LED” zawiera przetwornicę, to pojawia się problem. Mianowicie taka wewnętrzna przetwornica, która ma za zadanie utrzymać niezmienny prąd diod LED, często zawiera kondensator o stosunkowo dużej pojemności (przykład na **fotografii U**). Sterowanie przebiegiem impulsowym może zakłócić pracę takiej przetwornicy i efekty mogą być dalekie od oczekiwanych. Zależy to od kilku czynników, w tym od częstotliwości pracy generatora w takim regulatorze oraz od budowy i właściwości przetwornicy. Między innymi dlatego na opakowaniach 12-woltowych „żarówek LED” zamieszczona jest informacja, czy nadają się one do ściemniania (*dim-able/not dimmable*). I to jest kolejny potencjalny problem: zastosowanie „żarówki” w wersji *not dimmable* może uniemożliwić regulację impulsową. Niektórzy uczestnicy wyraźnie napisali, że większość żarówek LED nie nadaje się do ściemniania, więc



Fot. R



Fot. U



w ich przypadku regulacja impulsowa nie spełni oczekiwań.

Wszystkie bez wyjątku odpowiedzi mogłem uznać za prawidłowe, bo każdy z uczestników wskazał co najmniej jeden istotny błąd.

Upominki za rozwiązanie zadania *Co tu nie gra?* numer 260 otrzymują:

Radosław Smalec – Zabrze,
Jarosław Tarnawa – Zasole,
Maciej Zieliński – Kraków.

Wszystkich uczestników zadania dopisuję do listy kandydatów na bezpłatne prenumeraty.

Policz

Policz260

W ostatnich miesiącach zajmowaliśmy się zaawansowanymi systemami odprowadzania ciepła, między innymi chłodzeniem wymuszonym. W tym odcinku wracamy do podstaw i zajmiemy się kwestią radiatorów przeznaczonych do klasycznych wzmacniaczy mocy audio.

Zagadnienie nie jest łatwe z kilku powodów. Ale podstawowa zasada także i tu jest bardzo prosta: do (scalonego) wzmacniacza mocy audio trzeba dobrać taki radiator, by w najgorszych spodziewanych warunkach pracy nie przegrzać półprzewodnikowej struktury (nie przekroczyć dopuszczalnej temperatury złącza). Także i tu trzeba wziąć pod uwagę klasyczny model termiczny z trzema rezystancjami termicznymi według **rysunku 1**.

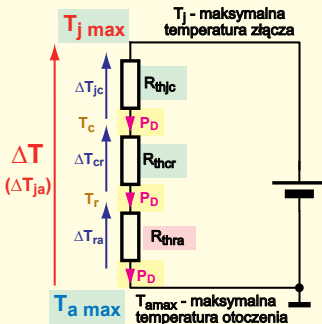
Finalne obliczenia są proste. Znając dopuszczalną (minimalną) wartość różnicy temperatur między złączem i otoczeniem ΔT_{ja} oraz moc strat P_D , z termicznego prawa Ohma możemy obliczyć całkowitą rezystancję termiczną:

$$R_{thja} = \Delta T_{ja} / P_D = (T_{jmax} - T_{amax}) / P_D$$

wiedząc, że $R_{thja} = R_{thjc} + R_{thcr} + R_{thra}$, możemy obliczyć:

$$R_{thra} = R_{thja} - (R_{thjc} + R_{thcr})$$

Maksymalną temperaturę złącza T_{jmax} (najczęściej $+150^\circ\text{C}$) i wartość rezystancji R_{thjc} scalonego wzmacniacza mocy odczytamy z katalogu. Wartość maksymalnej temperatury otoczenia T_{amax} możemy realistycznie przyjąć w granicach $40..60^\circ\text{C}$. Wartość „przejściowej” rezystancji styku układu scalonego z radiatorem R_{thcr} też możemy oszacować z sensowną dokładnością. Dla niewielkiej obudowy TO-220 dla przypadku bez przekładki, ze smarem termicznym przyjmujemy zwykle $0,5^\circ\text{C/W}$. Dla większych rozmiarów obudów TOP-3, TO-247, TO-218 możemy przyjąć $0,3..0,4^\circ\text{C/W}$. Scalone wzmacniacze większej mocy zwykle umieszczane są w obudowach o jeszcze większej powierzchni styku z radiatorem. Przykładem może być popularna kostka TDA7294, której obudowa jest pokazana na **rysunku 2**. Dla tej wielkości obudów rezystancję termiczną R_{thcr} , bez przekładki ze smarem, możemy przyjąć $0,2^\circ\text{C/W}$, natomiast z przekładką siliko-


Rys. 1

nową albo miko-
wą ze smarem
o $0,5..0,6^\circ\text{C/W}$.

Naszym celem jest dobranie radiatora, czyli obliczenie wartości R_{thra} , a do tego potrzebna jest jeszcze wartość mocy strat P_D . I tu zaczyna się problem!

Trzeba mocno podkreślić, że chodzi o **moc strat P_D , traconą we wzmacniaczu w postaci ciepła**.

A nie o moc wyjściową wzmacniacza (zwykle oznaczaną P_O), która doprowadzona jest do obciążenia (głośnika).

Gdy mamy do czynienia ze wzmacniaczami mocy audio, najbardziej interesuje nas moc wyjściowa P_O , a nie moc strat P_D . Potrzebna nam do obliczeń moc strat P_D to różnica między mocą pobieraną ze źródła zasilania, którą oznaczymy P_Z , i mocą wyjściową wzmacniacza P_O , (gdzie $P_D = P_Z - P_O$). Wartość mocy zasilania P_Z nie jest podawana, ale dość popularnym i „dostępnym” parametrem jest sprawność energetyczna wzmacniacza, zwykle oznaczana małą grecką literką eta (η), która jest stosunkiem mocy wyjściowej P_O i mocy zasilania P_Z , jak ilustruje to **rysunek 3**.

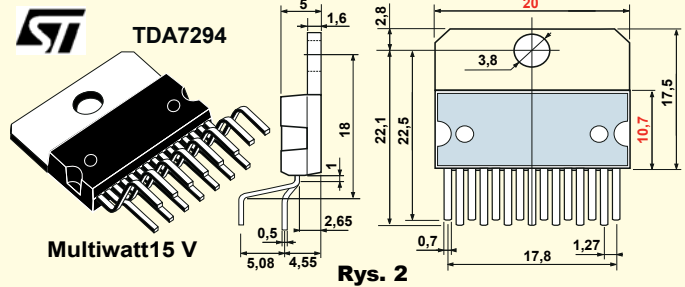
Mając sprawność ($\eta = P_O / P_Z$) oraz moc wyjściową wzmacniacza P_O , możemy łatwo obliczyć najpierw moc zasilania:

$$P_Z = P_O / \eta$$

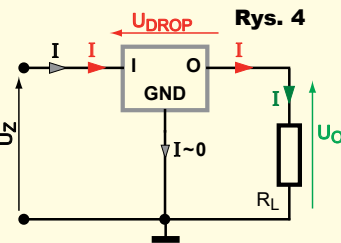
a potem moc strat P_D :

$$P_D = P_Z - P_O$$

Wyznaczenie sprawności (η) oraz wielkości mocy strat P_D jest łatwe na przykład w układzie liniowego stabilizatora. Tam mamy napięcie wejściowe U_Z , napięcie


Rys. 2

$$\text{sprawność } \eta = \frac{P_O}{P_Z}$$


Rys. 3


wyjściowe U_O i spadek napięcia na stabilizatorze U_{DROP} , jak ilustruje to **rysunek 4**. Są to napięcia i prądy stałe, więc nie ma kłopotu, a jeśli napięcie wejściowe zawiera tętnienia, to bierzemy średnią wartość tego napięcia. Zaniedbujemy małe prądy, pobierany przez stabilizator. Możemy obliczyć moce z zależności:

$$P = U * I$$

W przypadku wzmacniacza mocy audio sprawa jest zdecydowanie bardziej skomplikowana.

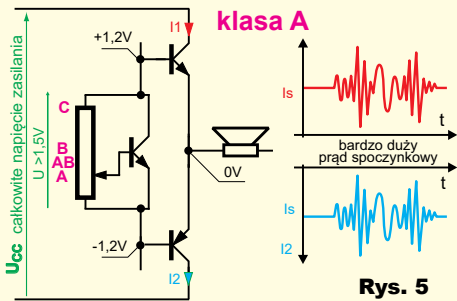
Okazuje się, że zagadnienie ma niejako dwie warstwy. Jedna warstwa

to matematyczne obliczenia dotyczące teoretycznego modelu i ściśle określonych, ale sztucznych założeń i warunków pracy. Druga warstwa związana jest głównie z faktem, że realne przebiegi audio to skomplikowana mieszanina składowych o różnej częstotliwości i różnych chwilowych amplitudach.

W tym odcinku zajmiemy się tylko warstwą pierwszą. Zaczniemy od tego, że mamy do czynienia z różnymi wzmacniaczami mocy audio. Jeśli chodzi o problem mocy strat, wspomnijmy tylko o wzmacniaczach klas A, B, AB oraz D.

Jak wiadomo, wzmacniacze mocy klasy A charakteryzują się tym, że przez tranzystory wyjściowe cały czas płynie bardzo duży prąd spoczynkowy I_S i sygnał audio moduluje wartość prądu, który nigdy nie zmniejsza się do zera, co w uproszczeniu pokazane jest na **rysunku 5**.

W spoczynku moc wyjściowa jest równa zero, a więc sprawność jest równa zero, a moc strat P_D jest równa iloczynowi prądu spoczynkowego I_S oraz całkowitego napięcia zasilania U_{CC} . Przy pełnym wysterowaniu idealnego wzmacniacza klasy A sprawność energetyczna η wynosi 25%. To oznacza, że moc strat przy



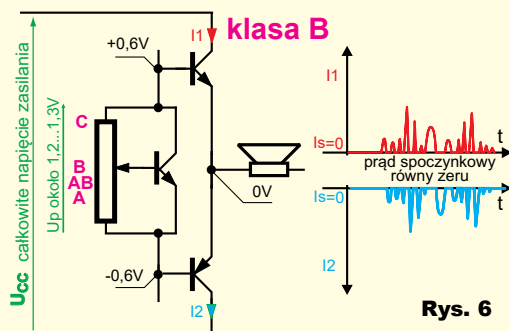
Rys. 5

pełnymysterowaniu jest równa 75% maksymalnej „spoczynkowej” mocy strat: $P_D = 0,75 * I_S * U_{CC}$.

Zupełnie inaczej jest w teoretycznym wzmacniaczu, pracującym w idealnej klasie B. Czysta klasa B oznacza, że w spoczynku przez tranzystory wyjściowe nie płynie żaden prąd. A jeśli prąd w spoczynku jest równy zero, moc strat też jest równa zero. Gdy pojawia się sygnał użyteczny, zaczyna płynąć prąd. Zasadę działania wzmacniacza klasy B pokazuje rysunek 6. Wzmacniacz zaczyna oddawać do obciążenia jakąś moc P_O i w tranzystorach wyjściowych wydziela się jakaś moc strat P_D .

Problem w tym, że sygnał użyteczny jest przebiegiem zmiennym (przemiennym) i obliczenia nie są już tak łatwe, jak w przypadku stabilizatora z rysunku 4. Wróć do rysunków 3, 4. Aby obliczyć moc strat P_D , trzeba obliczyć lub oszacować moc wyjściową P_O i moc zasilania P_Z .

Możemy to obliczyć stosunkowo łatwo, jeśli założymy, że sygnał użyteczny jest czystym przebiegiem sinusoidalnym. Otóż idealizowany wzmacniacz klasy B wytwarza na swym wyjściu sygnał sinusoidalny, którego międzyszczytowa amplituda jest równa całkowitemu napięciu zasilania U_{CC} , jak pokazuje to rysunek 7.



Rys. 6

Moc zasilania to iloczyn napięcia zasilającego i prądu: $P = U_{CC} * I$. Zakładamy, że napięcie zasilające jest niezmiennie (U_{CC}), a do obliczeń trzeba przyjąć średnią wartość prądu. W przypadku wzmacniacza audio pewną komplikacją jest to, że średnia wartość przebiegu sinusoidalnego jest... równa zero. Ale przecież wcale nie znaczy to, że średnia wartość prądu zasilającego jest równa zero. Na

przykładzie wzmacniacza zasilanego napięciem symetrycznym i rysunku 9 jasno widać, że w „dodatniej” połowie przebiegu sinusoidalnego prąd pobierany jest tylko „z górnej połówki napięcia zasilania”. Przy ujemnej połowie – tylko „z dolnej połówki napięcia zasilania”.

Przeanalizuj rysunek 9. Czy już widzisz, że zarówno podczas „górnej”, jak i „dolnej” połówki sinusoidy, prąd obciążenia będzie miał taki sam kształt, właśnie połówki sinusoidy?

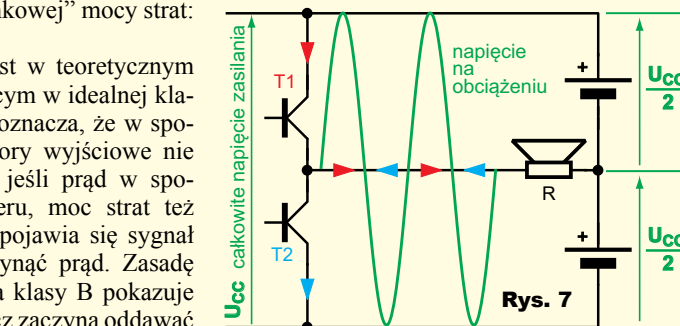
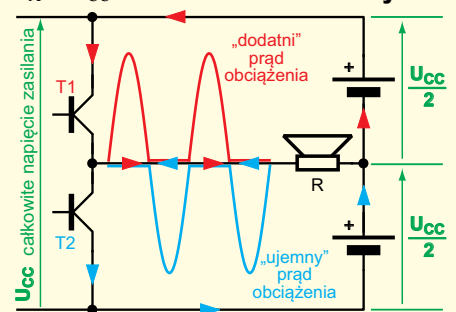
My chcemy obliczyć średnią moc pobieraną ze źródła zasilania. Napięcie zasilania jest niezmiennie i szukamy informacji o średniej wartości prądu pobieranego przez wzmacniacz. Jeszcze raz zwróć uwagę, że w idealizowanym wzmacniaczu zasilanym napięciem symetrycznym o wartości $\pm 0,5 * U_{CC}$, czyli $\pm U_{CC}/2$, napięcie szczytowe, czyli amplituda sinusoidy wyjściowej wyniesie właśnie $U_{CC}/2$. A prąd?

Wartość szczytowa prądu będzie oczywiście równa:

$$I_A = U_A / R = (U_{CC}/2) / R$$

$$I_A = U_{CC} / 2R$$

Rys. 9

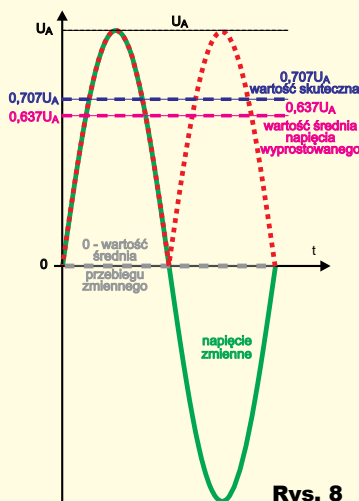


Rys. 7

Czyli amplituda przebiegu sinusoidalnego ma wartość $0,5 * U_{CC}$. Wiadomo (rysunek 8), że przebieg o amplitudzie U_A ma wartość skuteczną pierwiastek z dwóch razy mniejszą od amplitudy ($U_{Sk} = U_{RMS} = U_A/\sqrt{2} \sim 0,707 * U_A$).

Na wyjściu idealizowanego wzmacniacza klasy B mamy więc przebieg sinusoidalny o wartości skutecznej $U_A/\sqrt{2}$, doprowadzony do obciążenia o rezystancji R . W analizowanym przypadku amplituda U_A jest równa połowie całkowitego napięcia zasilania: $U_A = U_{CC}/2$, więc $U_{Sk} = U_{CC}/2\sqrt{2}$. Moc dostarczona do tego obciążenia wyniesie więc $P_O = (U_{Sk})^2 / R = (U_{CC}/2\sqrt{2})^2 / R = (U_{CC}^2 / 8) / R$ $P_O = U_{CC}^2 / 8R$

To było bardzo łatwe. Niewiele trudniejsze jest obliczenie mocy zasilania.



Rys. 8

- dioda LED
- udźwig aż do 2.2kg
- teleskop do 80cm
- wymiary po złożeniu tylko 20cm

12,50zł

Chwytnik magnetyczny HPUT1



sklep.avt.pl handlowy@avt.pl tel.: 22 257 84 50

Nam potrzebna jest średnia wartość prądu w danym półokresie. Jak wiadomo (rysunek 8), dla połówki sinusoidy wartość średnia przebiegu (dotyczy zarówno napięcia, jak i prądu) jest równa $2/\pi$ (czyli około 0,637) wartości amplitudy. Dla napięcia średnia wynosi $2U_A/\pi$, czyli około $0,637 \cdot U_A$ i dla prądu analogicznie $2I_A/\pi$, co pozwala obliczyć prąd średni za pół okresu:

$$I_{sr} = 2I_A/\pi = 2(U_{CC}/2R)/\pi = (2U_A/2R)/\pi = (U_{CC}/R)/\pi$$

Obliczenia dotyczące drugiego, „ujemnego” półokresu będą identyczne. To oznacza, że taki sam będzie średni prąd podczas obu półokresów przebiegu i w sumie podczas całego czasu pracy w pełni wystereowanego wzmacniacza.

Teraz z zależności $P = U \cdot I_{sr}$ możemy obliczyć moc średnią pobieraną w czasie połówki sinusoidy, tylko znów zgodnie z rysunkiem 8 musimy wziąć do obliczeń połowę całkowitego napięcia zasilania ($U_{CC}/2$):

$$P_Z = (U_{CC}/2) \cdot (U_{CC}/\pi R)$$

$$P_Z = U_{CC}^2 / 2\pi R$$

W czasie drugiej połówki sinusoidy będzie tak samo, czyli właśnie obliczyliśmy średnią moc pobieraną przez wzmacniacz ze źródła zasilania.

Wcześniej obliczyliśmy moc wyjściową $P_O = U_{CC}^2 / 8R$

co pozwala obliczyć sprawność przy maksymalnymysterowaniu:

$$\eta = P_O / P_Z$$

$$\eta = (U_{CC}^2 / 8R) / (U_{CC}^2 / 2\pi R)$$

skracamy zarówno U_{CC}^2 , jak i R:

$$\eta = 2\pi / 8 = \pi / 4$$

$$\eta = 0,785398$$

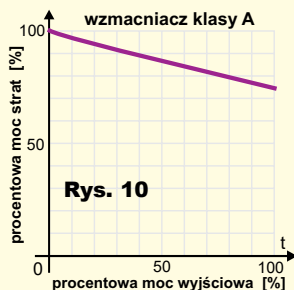
$$\eta = 78,5398\%$$

I oto „własnoręcznie” obliczyliśmy maksymalną sprawność energetyczną idealnego wzmacniacza audio klasy B!

Ale wróćmy do głównego tematu. Nie zapominajmy o głównym celu naszych rozważań: chcemy dobrać radiator do wzmacniacza mocy i do obliczeń potrzebna jest wartość mocy strat.

Z tego, co powiedzieliśmy o wzmacniaczach klasy A, wynika, że moc strat jest w nich największa w spoczynku i maleje do 75% przy pełnymysterowaniu, czyli przy największej możliwej mocy wyjściowej, co można z grubsza zilustrować jak na rysunku 10.

W wzmacniaczu klasy B jest zupełnie inaczej. W spoczynku teoretycznie moc strat



P_D jest równa zero. A przy maksymalnej mocy moc strat P_D wynosi około 21,5% mocy P_Z pobieranej z zasilacza (bo sprawność wynosi 78,5%). Możemy też łatwo wyliczyć, że moc strat P_D to około 27% mocy wyjściowej P_O .

Pięknie, tylko są to wartości teoretyczne, dotyczące pełnego, idealnegoysterowania sygnałem sinusoidalnym.

Po pierwsze nie uwzględniliśmy faktu, że w realnym wzmacniaczu tranzystory mają niezerowe napięcie nasycenia, czyli amplituda napięcia na obciążeniu będzie znacząco niższa niż połowa całkowitego napięcia zasilania.

Po drugie, klasyczne wzmacniacze audio nigdy nie pracują w czystej klasie B, tylko w klasie AB. To znaczy, że przez tranzystory wyjściowe płynie jakiś prąd spoczynkowy. Prąd ten może być mniejszy lub większy. Najczęściej prąd spoczynkowy jest ponad 10 razy mniejszy niż maksymalny prąd obciążenia przy pełnymysterowaniu. Tak czy inaczej dwa wspomniane czynniki zmniejszają sprawność rzeczywistego wzmacniacza mocy poniżej teoretycznej granicy 78,5%.

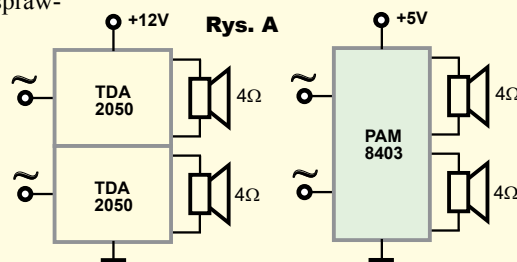
Jest i trzecia sprawa, która w praktyce okazuje się bardzo ważna. Otóż dotychczas mówiliśmy o pełnymysterowaniu i o brakuysterowania wzmacniacza klasy B. Ustaliliśmy, że w spoczynku prąd wyjściowy nie płynie i moc strat P_D jest równa zero, a przy pełnymysterowaniu sprawność wynosi 78,5%. A przy pośrednich wartościachysterowania?

Mamy tylko dwa punkty na wykresie i na razie nie wiemy, jaka jest zależność mocy strat od mocy wyjściowej przebiegu użytkownego. Kilka możliwości pokazanych jest na rysunku 11. Tymi wszystkimi ważnymi zagadnieniami zajmiemy się w następnym odcinku. A na razie zadanie *Policz260*, też związane z mocą.

Zadanie *Policz266*

W układzie, w którym mamy do dyspozycji napięcia +12V oraz +5V, potrzebny jest dwukanałowy wzmacniacz audio. Mamy pod ręką i możemy wykorzystać albo dwie 32-watowe kostki TDA2050 zasilane napięciem 12V, albo zasilany napięciem 5V moduł

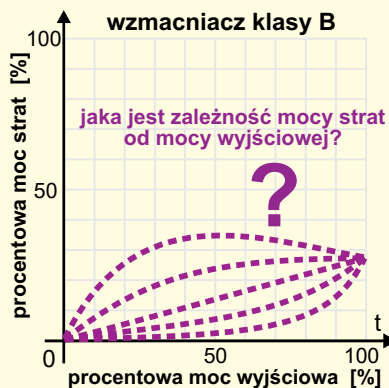
ze wzmacniaczem stereo PAM8403 według rysunku A.



Obciążeniem będą dwa głośniki 4-omowe. W ramach zadania *Policz265* należy porównać:

- jaką moc dostarczy do głośnika jeden i drugi wzmacniacz?

Nagrodami będą kity AVT lub książki, a najaktywniejsi uczestnicy są okresowo nagradzani bezpłatnymi prenumeratami EdW lub innego wybranego czasopisma AVT. Wszystkie rozwiązania nadsyłane w terminie 60 dni od ukazania się tego numeru EdW powinny mieć dopisek *Policz265* (na kopercie, a w tytule e-maila dodatkowo nazwisko, np.: *Policz265.Jankowski*). Z uwagi na specyfikę zadania bardzo proszę o podawanie swojego wieku oraz miejsca nauki czy pracy. W e-mailach podawajcie też od razu swój adres pocztowy. **Zapraszamy do rozwiązania tego zadania zarówno doświadczonych, jak i początkujących elektroników**, którzy nie potrafią przeanalizować wszystkich subtelności układu. Można też jeszcze nadsyłać rozwiązania zadania *Policz264* z poprzedniego miesiąca.



Rys. 11

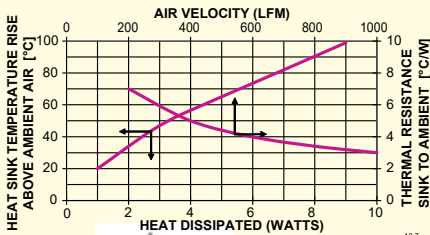
Rozwiązanie zadania *Policz260*

W EdW 10/2017 przedstawione było zadanie *Policz260*, które brzmiało: *Budujemy urządzenie zawierające stabilizator LM317, w którym maksymalnie może się wydzieląć 7,5 wata mocy strat. Chcielibyśmy wykorzystać mały radiator o parametrach jak na rysunku B. W ramach zadania *Policz260* należy oszacować:*

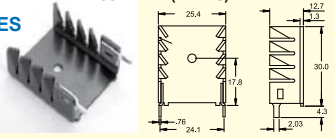
- czy może on pracować bez wentylatora, czy też wentylator byłby konieczny?

Zadanie, a właściwie obliczenia były łatwe dla wszystkich, którzy śledzą systematycznie cykl o chłodzeniu elektroniki.

Do rozważań trzeba wziąć część wykresu, dotyczącą chłodzenia naturalnego. Jak pokazuje rysunek C, przy mocy 7,5 wata należy się spodziewać, że radiator będzie



265 SERIES



Rys. B

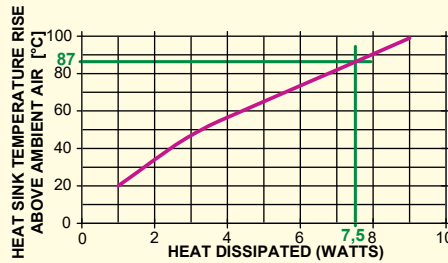
miał temperaturę o 87 stopni wyższą niż temperatura otoczenia. Jest to tylko spadek temperatury na rezystancji R_{thra} : $\Delta T = P * R_{thra}$. A my do obliczeń potrzebujemy informacji dotyczącej temperatury złącza, a nie temperatury radiatora, jak pokazuje to rysunek D.

Do obliczeń należy jeszcze wziąć rezystancje termiczne R_{thcr} oraz R_{thjc} . Dla obudowy TO-220 ze smarem możemy przyjąć $R_{thcr} = 0,5°C/W$. Wiemy już z wcześniejszych rozważań i zadań, że stabilizator LM317 ma stosunkowo dużą rezystancję R_{thjc} . Możemy przyjąć $R_{thjc} = 4,5°C/W$, co wraz z R_{thcr} da okragłe $5°C/W$. Przepływ mocy 7,5 wata spowoduje na takiej rezystancji wystąpienie różnicy temperatur $37,5°C/W$.

W sumie różnica temperatur między złączem a otoczeniem, czyli ΔT_{ja} , wyniesie więc około 125 stopni.

Wiemy, że kostka LM317 ma wbudowane zabezpieczenie termiczne. Jeżeli przyjmiemy, że próg działania tego zabezpieczenia wynosi $+150°C$, to okaże się, iż maksymalna temperatura otoczenia może wynosić $+25°C$, jak pokazuje to rysunek E. Przy wyższej temperaturze otoczenia nastąpi zadziałanie zabezpieczenia termicznego. Przyjęcie maksymalnej spodziewanej temperatury otoczenia równej $+25°C$ byłoby niezbyt rozsądne, bo latem temperatura powietrza czasem przekracza $+30°C$.

Założyliśmy tu maksymalną temperaturę złącza $+150°C$, przy której zadziała zabezpieczenie termiczne. Może się zdarzyć, że w niektórych kostkach próg ten będzie nieco wyższy niż $+150°C$. Ale w większości kata-



Rys. C

logów znajdziemy wskazówki, żeby nie przekraczać temperatury złącza $+125°C$. A to jak pokazuje rysunek F, dawałoby zupełnie nierealną maksymalną dopuszczalną temperaturę otoczenia $0°C$.

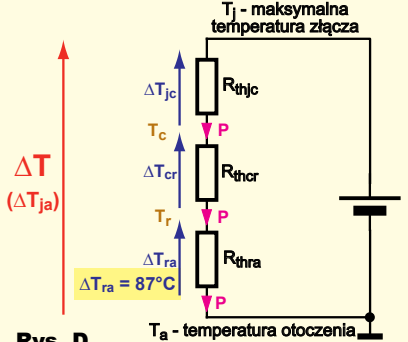
Z przedstawionych rozważań wynika, że przy naturalnym chłodzeniu taki mały blaszany radiator nie może bezpiecznie rozproszyć mocy 7,5 wata traconej w stabilizatorze LM317.

Nawiązując do rozważań niektórych Kolegów, ten mały radiator mógłby z powodzeniem wystarczyć do rozproszenia mocy strat innego elementu niż LM317. Na przykład gdyby elementem czynnym był jakiś tranzystor o rezystancji termicznej R_{thjc} równej $1,5°C/W$ i maksymalnej temperaturze złącza $+150°C$, to sytuacja wyglądałaby jak na rysunku G.

Maksymalna temperatura otoczenia mogłaby wynosić $+48°C$, co jest jak najbardziej realistyczne.

A jeśli elementem czynnym byłby MOSFET o rezystancji termicznej powiedzmy $1°C/W$ i o maksymalnej temperaturze złącza $+175°C$, wtedy jak pokazuje rysunek H, sytuacja byłaby jeszcze lepsza i maksymalna temperatura otoczenia mogłaby sięgać $+76°C$, czyli więcej niż zakładamy nawet w najbardziej pesymistycznych rozważaniach.

Przy mocy strat 7,5W ten sam radia-



Rys. D

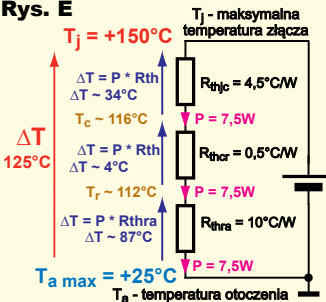
torok nie zda egzaminu przy współpracy z elementami o dużej „wrodzonej” rezystancji fabrycznej R_{thjc} , a dobrze sprawdzi się w przypadku elementów o małej wartości R_{thjc} .

Większość nadesłanych odpowiedzi była prawidłowa, jednak były odpowiedzi, gdzie nie uwzględniono wartości R_{thjc} i R_{thcr} , co dało zbyt optymistyczne wnioski. Nagrody-upominki za zadanie Policz260 otrzymują:

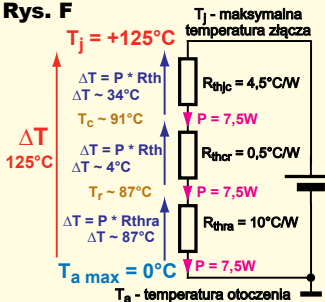
Jakub Nowak – Dawid,
Sebastian Jarmosiewicz – Motwica,
Roman L. – Warszawa.

Wszystkich uczestników dopisuję do listy kandydatów na bezpłatne prenumeraty.

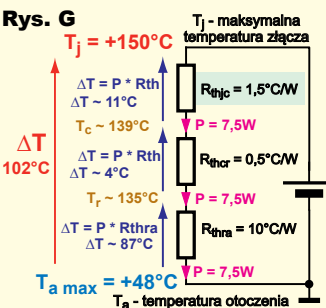
Rys. E



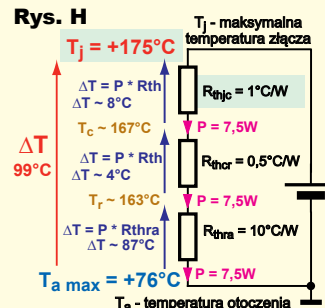
Rys. F



Rys. G



Rys. H



R E K L A M A

AVT 3143 Nakręcany minutnik

Urządzenie do odliczania czasu sygnalizujące sygnałem akustycznym upływ nastawionej wcześniej liczby minut. Wykorzystywane może być m.in. w gospodarstwie domowym, np. w kuchni.



Znajdź nas na



Regulator mocy z wyświetlaczem LCD

Do czego to służy?

Prezentowany układ to regulator mocy urządzeń zasilanych z sieci 230V AC. Prototyp posłużył jako regulator mocy do lutownicy, ale układ może współpracować także z innymi typami grzałek o mocy do 300W. Ponadto dobrze współpracuje z klasycznymi żarówkami oraz z żarówkami LED i świetłówkami kompaktowymi z funkcją „dimmable”. Pomyślnie przeszedł testy także z silnikiem komutatorowym, silnikiem klatkowym oraz transformatorem. Jednak moc takich odbiorników, innych niż rezystancyjne, nie powinna przekraczać 100W. Pewnym ograniczeniem układu jest moc minimalna, która nie powinna być mniejsza niż 10W.

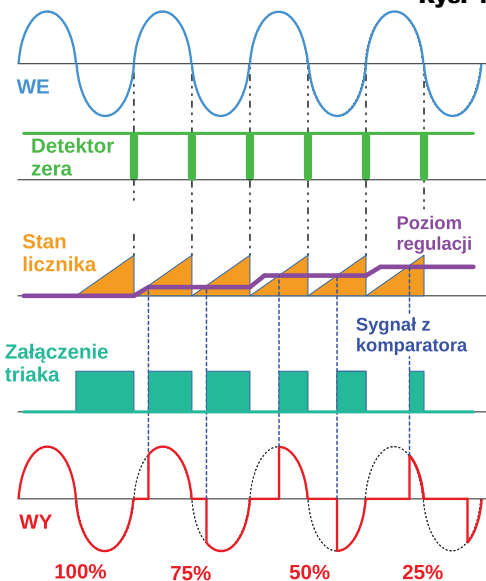
Podstawowe parametry:

- zakres regulacji 0...100%, ze skokiem 1%,
- maksymalne obciążenie 300W,
- zasilanie 230V AC,
- regulacja metodą fazową,
- stopień regulacji prezentowany na wyświetlaczu LCD,
- zapamiętywanie ostatniej nastawy,
- dostosowany do obudowy Z50.

Uwaga! W urządzeniu występują napięcia niebezpieczne dla człowieka.

Jak to działa?

Urządzenie realizuje metodę regulacji fazowej. Dostępne w naszych domach zasilanie z sieci energetycznej 230V jest napięciem przemiennym o kształcie



Rys. 1

sinusoidalnym. Regulacja fazowa polega na tym, że pozwala ustawić jaką część każdego półokresu przebiegu napięcia zostanie doprowadzona do odbiornika, a jaka część zostanie zablokowana. Im większa część przebiegu zostanie doprowadzona do odbiornika, tym większa będzie średnia moc, którą odbiornik zużyje. W dolnej części rysunku 1 przedstawiony jest przebieg napięcia wyjściowego dla przykładowych poziomów regulacji.

Taki sposób działania ma dwie ważne zalety – pozwala uzyskać praktycznie dowolną nastawę w zakresie od 0 do 100% oraz działa w sposób ciągły – to znaczy, że cały proces zamyka się w trakcie jednego półokresu napięcia sieci zasilającej. Dzięki tym cechom nie występuje efekt tzw. migotania żarówki, który ma miejsce przy regulacji metodą grupową. Natomiast wadą takiego rozwiązania jest fakt, że układ generuje zakłócenia elektromagnetyczne. Przy niewielkich mocach obciążenia poziom zakłóceń jest niewielki i nie wpłynie na pracę innych urządzeń.

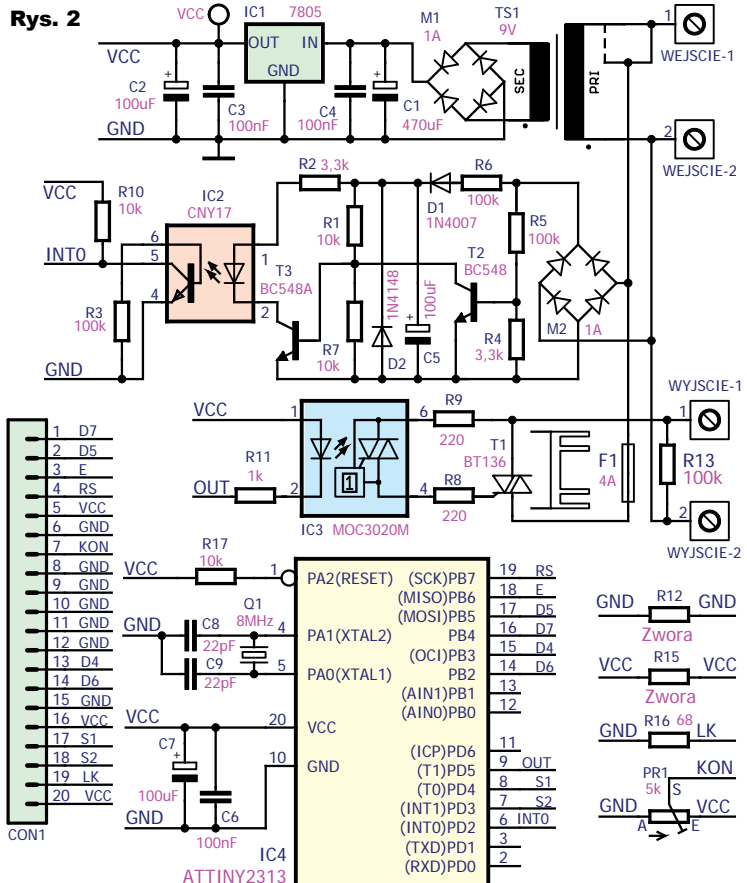
Urządzenie zostało zaprojektowane tak, aby pasowało do obudowy typu Z50, dlatego konstruk-

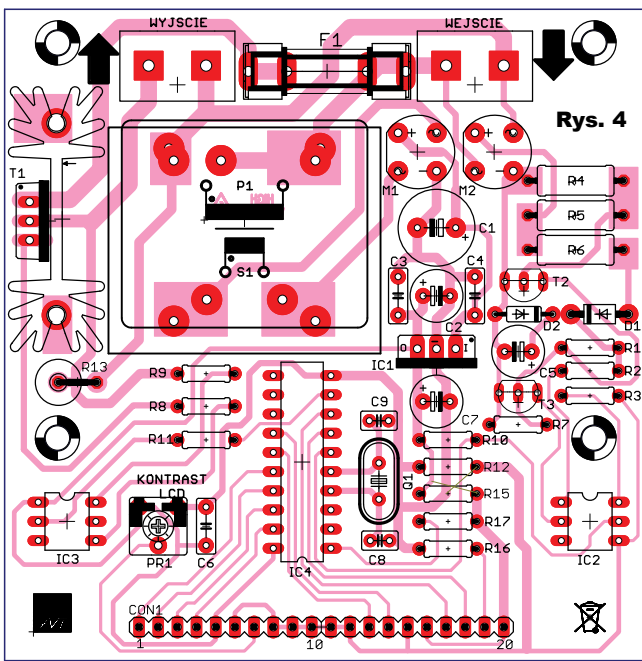
cja została podzielona na dwie płytki: płytkę główną, która zawiera część sterującą i wykonawczą, oraz płytkę interfejsu użytkownika, która zawiera wyświetlacz i przyciski. Dodatkowo powstała jeszcze płytka panelu frontowego, która ma zapewnić dopracowany i estetyczny wygląd gotowego urządzenia.

Schemat płytki głównej przedstawia rysunek 2, można go podzielić na kilka bloków, pierwszy z nich to blok zasilania ze stabilizatorem IC1, który dostarcza stabilizowane napięcia 5V.

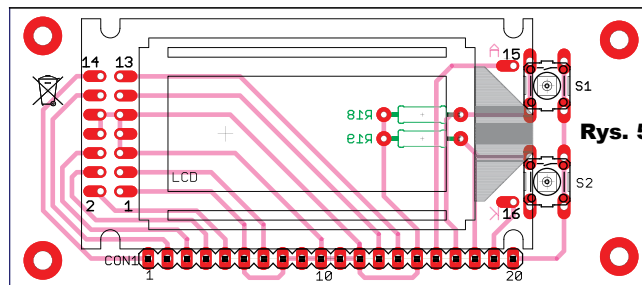


Rys. 2





Rys. 4

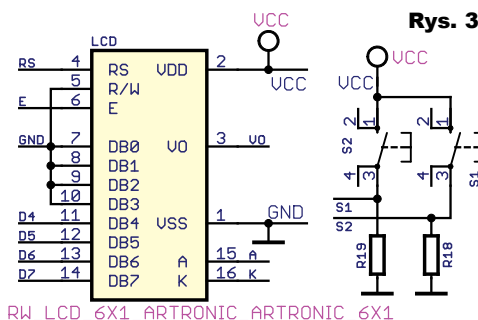


Rys. 5

Kolejny blok to detektor zera w napięciu sieci. Zbudowany jest z elementów od mostka M2 do transoptora IC2. Regulacja fazowa wymaga dostarczenia precyzyjnej informacji o momencie przejścia napięcia sieci przez zero. Na wyjściu transoptora, na wyprowadzeniu 5, występuje przebieg prostokątny z dominującym stanem wysokim i bardzo wąskimi szpilkami stanu niskiego, zsynchronizowanymi z zerem napięcia sieci.

Trzeci blok to stopień mocy. Elementem wykonawczym jest triak T1, powinien to być element o małym prądzie maksymalnym, odpowiedni jest np. BT136. Zastosowanie „mocniejszego” triaka spowoduje podniesienie warto-

1	D7
2	D5
3	E
4	RS
5	UCC
6	GND
7	U0
8	GND
9	GND
10	GND
11	GND
12	GND
13	D4
14	D6
15	GND
16	A
17	S1
18	S2
19	K
20	UCC



Rys. 3

ści mocy minimalnej nawet do 50...80W i uniemożliwi pracę regulatora z niektórymi „mniejszymi” odbiornikami. Natomiast nie zwiększy się w ten sposób mocy maksymalnej, choćby z powodu zbyt małego radiatora. Sterowanie bramką triaka odbywa się za pośrednictwem optotriaka IC3 typu MOC3020. Optotriak można zastąpić modelem z oznaczeniem kończącym się na 21...23, natomiast nie może to być seria zakończona oznaczeniem 40...41 – jest to seria przeznaczona do załączania triaka tylko w zerze przebiegu napięcia sieci.

Czwarty blok to część sterująca. Zawiera mikrokontroler i kilka elementów potrzebnych do jego prawidłowej pracy. Ale najważniejszego elementu nie widać na schemacie – jest to program sterujący zawarty w pamięci mikrokontrolera. Ostatnim blokiem urządzenia jest blok interfejsu użytkownika, którego schemat przedstawia rysunek 3. Zawiera tylko wyświetlacz, który wyświetla aktualną nastawę, oraz przyciski umożliwiające zmianę nastawy. Zaprojektowany jest tak, aby połączyć go z płytką główną pod kątem prostym – tworzy w ten sposób panel sterujący urządzeniem.

Program sterujący. Sygnał z detektora przejścia przez zero, doprowadzony jest do mikrokontrolera, do wejścia, które pełni funkcję sprzętowego przerwania INT0. W chwili wystąpienia zera w napięciu sieci uruchamiana jest procedura obsługi przerwania, a w niej, przede wszystkim, zerowany jest stan wbudowanego w mikrokontroler licznika TIMER1. Od tego momentu licznik odmierza czas, jaki upłynął od wystąpienia zera, a na tej podstawie możemy określić aktualny punkt na przebiegu napięcia sieci.

Porównując stan licznika z nastawą poziomu regula-

cji, można wyznaczyć moment, w którym należy załączyć triak. To porównywanie realizowane jest sprzętowo, poprzez jeden z komparatorów licznika TIMER1. Gdy stan licznika zrówna się ze stanem komparatora wtedy uruchamiane jest kolejne przerwanie, a jego procedura powoduje załączenie triaka. Kolejne etapy całego procesu przedstawione są na rysunku 1.

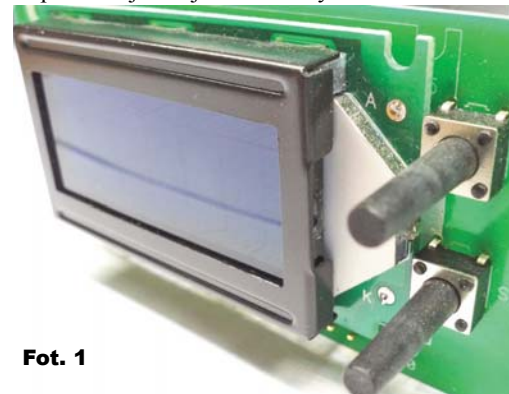
Pozostałe zadania są to przede wszystkim reagowanie na sygnały z interfejsu użytkownika, czyli na wciśnięcie któregoś z przycisków. Zmieniana jest wtedy nastawa regulacji w zakresie od 0 do 100%, następnie nastawa jest odpowiednio przeliczana i wpisywana do komparatora licznika TIMER1. Na koniec nowa wartość nastawy zostaje wysłana do wyświetlacza.

Po każdej zmianie nastawy, po ok. 3 sekundach od ostatniego wciśnięcia przycisku, nowa wartość zapamiętywana jest w pamięci nieulotnej eeprom mikrokontrolera. Dzięki temu, po wyłączeniu urządzenia i późniejszym ponownym włączeniu, automatycznie przywracana jest ostatnio ustawiona wartość nastawy.

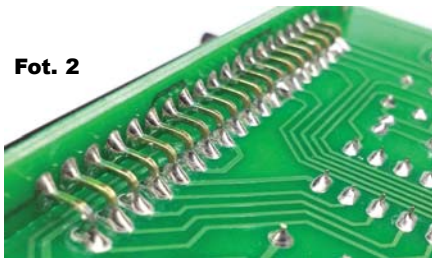
Montaż i uruchomienie

Schematy płytek urządzenia widoczne są na rysunkach 4 i 5. Płytki przygotowane są do elementów przewlekanych, należy je montować zgodnie z ogólnymi zasadami. Na płytce głównej trzeba zwrócić uwagę na polaryzację elementów półprzewodnikowych, zwłaszcza mostków prostowniczych („plusy” powinny być zwrócone w stronę zacisku wejściowego) elementów optoelektronicznych IC2 i IC3 (kropki na układach powinny być zwrócone do środka płytki) oraz mikrokontrolera (kropka na układzie powinna być zwrócona do krawędzi płytki, odwrotnie niż IC2 i IC3). W miejsce rezystorów R12 i R15, które pełnią funkcję zwór, należy wlotować rezystory zerowe – oznaczone jednym czarnym paskiem lub zworki z cienkiego drutu. Na razie nie trzeba lutować szpilek goldpin w miejscu złączającego płytki – CON1.

Na płycie interfejsu użytkownika w pierwszej kolejności należy wlotować



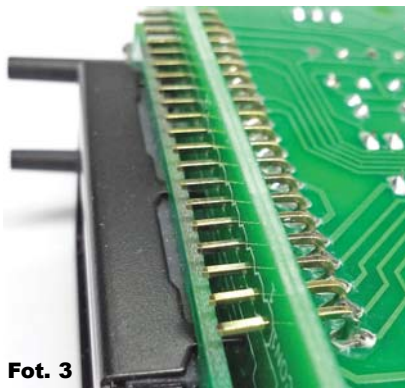
Fot. 1



Fot. 2

dwa rezystory, ale po stronie lutowania, ponieważ po stronie elementów będzie umieszczony tam wyświetlacz. Wyświetlacz trzeba przylutować do płytki poprzez dwurzędową listwę goldpin z lewej strony oraz dwoma odcinkami cienkiego drutu, w punktach K i A z prawej strony – obrazuje to **fotografia 1**.

Po zmontowaniu płytek należy je dokładnie sprawdzić pod kątem poprawności montażu. W urządzeniu występuje napięcie sieci energetycznej 230V i każdy



Fot. 3

błąd w montażu może spowodować poważne uszkodzenia oraz narazić użytkownika na niebezpieczeństwo. Dlatego montaż i uruchomienie urządzenia trzeba przeprowadzić pod nadzorem osoby wykwalifikowanej.

Pierwsze podłączenie urządzenia należy wykonać, zanim połączymy ze sobą płytki. Płytkę główną należy umieścić w podstawie obudowy i przykręcić krótkimi wkrętami do odpowiednich kołków. Następnie wyjąć mikrokontroler z podstawki, dołączyć przewód zasilający do zacisku wejściowego i podłączyć do zasilania. Teraz wolto-mierzem należy skontrolować napięcie pomiędzy 10. i 20. wyprowadzeniem w podstawie mikrokontrolera, powinno wynosić 5V. Dopiero po tym sprawdzeniu można odłączyć zasilanie i przystąpić do kolejnych czynności.

Czas połączyć ze sobą płytki. W tym celu trzeba wlutować listwę szpilek goldpin w miejsce złącza CON1 na płytce interfejsu. Szpilki powinny być wsunięte dłuższą stroną, do połowy długości, od strony lutowania i zagięte w górę. Następnie należy usunąć ze szpilek tworzywo, które utrzymywało je w szeregu i dopiero wtedy wsunąć końce w miejsce złącza CON1 na płytce głównej od strony lutowania i przylutować. Efekt końcowy widać na **fotografiach 2 i 3**. Dopiero teraz można umieścić obie płytki w obudowie, zamocować płytkę panelu frontowego z **rysunku 6** i przykręcić płytkę główną w obudowie. Całość powinna wyglądać jak na **fotografii 4**.

Po dołączeniu zasilania urządzenie wymaga tylko wyregulowania kontra-



Fot. 4

stu wyświetlacza potencjometrem PR1. Gdy na wyświetlaczu ukaże się wartość w procentach, to urządzenie jest gotowe do pracy. Odbiornik, którego moc będzie regulowana, należy dołączyć do zacisków oznaczonych „wyjście”. **Fotografia 5** pokazuje przykład zastosowania jako regulator mocy lutownicy oporowej na 230V.

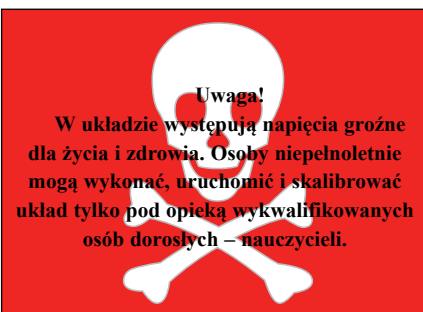
KS

Rys. 6

MASKA
W KOLORZE CZARNYM

REGULATOR MOCY

Fot. 5



Uwaga!

W układzie występują napięcia groźne dla życia i zdrowia. Osoby niepełnoletnie mogą wykonać, uruchomić i skalibrować układ tylko pod opieką wykwalifikowanych osób dorosłych – nauczycieli.

Wykaz elementów

R1, R7, R10, R17, R18, R19	10kΩ
R2	3,3kΩ
R3	100kΩ
R5, R6, R13	100kΩ/2W
R4	3,3kΩ/2W
R8, R9	220Ω
R11	1kΩ
R12, R15	0Ω lub zwora
R16	68Ω
PR1	potencjometr 5kΩ
C1	470uF/16V
C2, C5, C7	100uF/16V
C3, C4, C6	100nF
C8, C9	22pF
D1	1N4007
D2	1N4148
T1	BT136
T2, T3	BC548
M1, M2	mostek prostowniczy 1A/1000V
IC1	7805
IC2	CNY17
IC3	MOC 3020...3023
IC4	ATTINY2313 zaprogramowany
Q1	rezonator 8MHz
F1	bezpiecznik + 4A
CON1	listwa goldpin kątowa 1x20
S1, S2	mikroswitch
LCD	lcd 1 x 6
TS1	transformator 230V/9V 3...4W
WEJŚCIE, WYJŚCIE	DG365-7,5/2
Radiator	SK104
Obudowa	Z50

Komplet podzespołów z płytką jest dostępny w Sklepie AVT jako zestaw AVT3218



Gasnąca lampka LED

Do czego to służy?

Pewnie zastanawiacie się, po co komu taki wynalazek?

Wbrew pozorom, jest to bardzo praktyczny gadżet. Dlatego też zdecydowałem się na tę krótką publikację. Głównymi zaletami jest to, że do jej wykonania potrzebne jest jedynie kilka elementów, a w stanie spoczynku nie pobiera z baterii prądu. Lampki przeznaczone są do umieszczenia w zależności od potrzeb w takich miejscach, gdzie nie opłaca się wykonywać instalacji elektrycznej 230V, a oświetlenie potrzebne jest jedynie na czas bezpiecznego przejścia: schody, ciemne przejścia, drewnia, itp.

Jak to działa?

Działanie jest analogiczne do czasowych automatów schodowych. Jak pokazuje **rysunek 1**, zasada działania lampki jest banalnie prosta. Po naciśnięciu przycisku P1 ładuje się kondensator C1 i włączany jest T1. Stała czasowa R1C1 rozładowania tego kondensatora określa, po jakim czasie napięcie spadnie do wartości około 2V i tranzystor przestanie przewodzić. Z wartościami jak na schemacie czas ten wynosi nieco ponad minutę. Po rozładowaniu kondensatora z baterii nie jest pobierany praktycznie żaden prąd (w naj-

gorszym wypadku maksymalnie 0,5uA przy 25V zasilania – według katalogu).

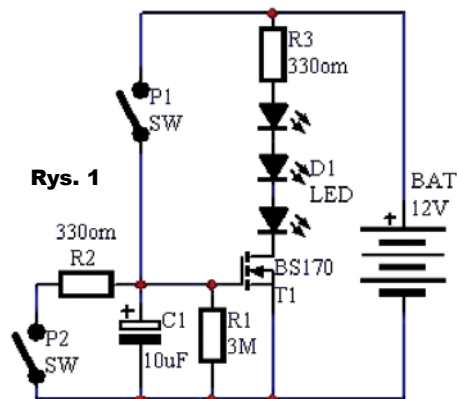
Można też w razie potrzeby dodać drugi przycisk P2 do wyłączania lampki w dowolnym momencie.

Do realizacji układu wykorzystałem prosty układ z rozładowaniem kondensatora oraz opisane wcześniej właściwości tranzystora MOSFET typu BS170. **Rysunek 2** pokazuje zmierzoną krzywą rozładowania RC obwodu 470uF || 100kΩ naładowanego do napięcia 12V – oczywiście wygląda ona dokładnie tak, jak pokazują to wykładnicze krzywe w podręcznikach.

Istotny jest czas, w którym napięcie osiągnie wartość około 2V, co jest napięciem progowym MOSFET-a.

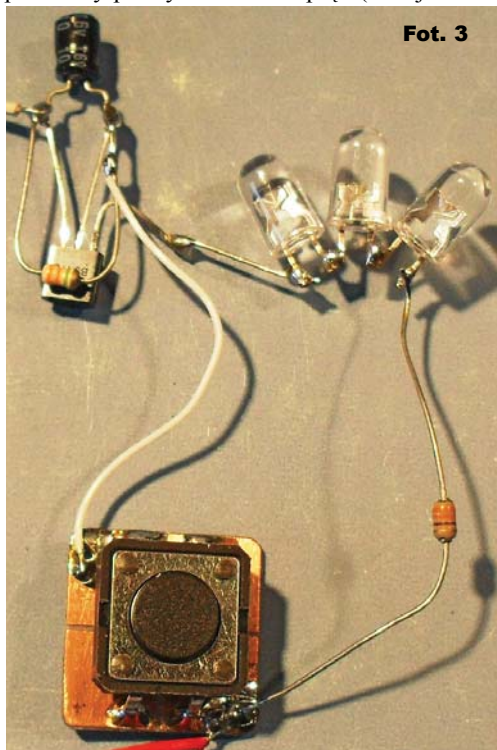
Montaż i uruchomienie

Zastosowałem diodę LED SMD 5050, która zawiera w jednej obudowie trzy struktury, które połączone są szeregowo. Dioda świeci bardzo jasno, a pobór prądu z baterii to zaledwie 3mA. Przez zmianę wartości R1 lub C1 (spis elementów) można wpływać na czas, przez jaki lampka ma świecić. Zastosowane baterie 12V, typu V23A, wystarczają na bardzo długo.

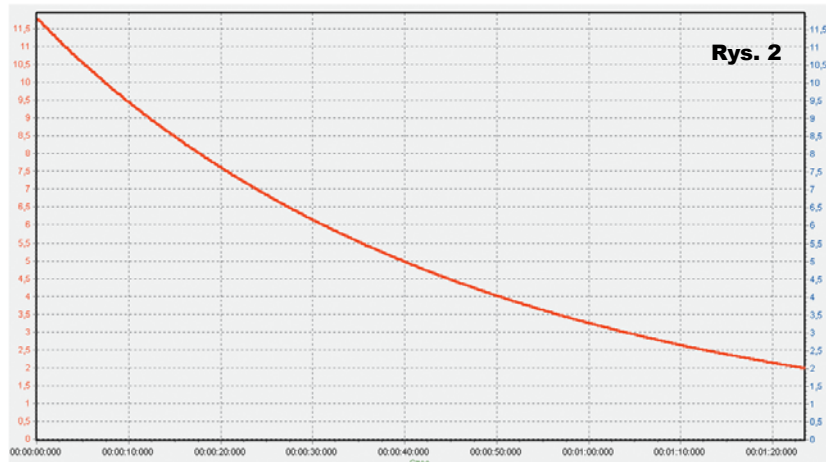


Rys. 1

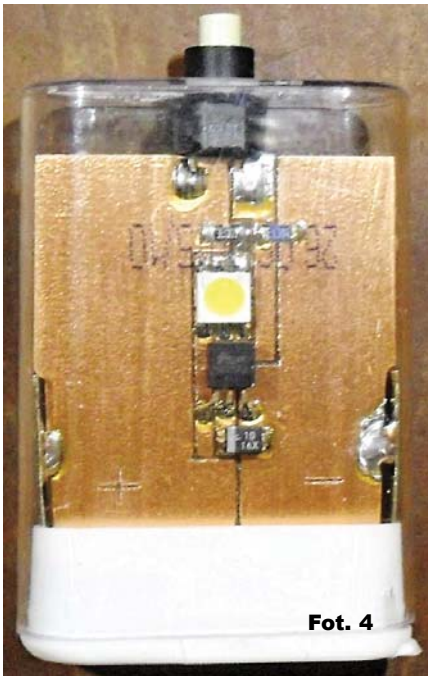
Technika wykonania jest dowolna, można cały układ polutować w „pajaku” jak na **fotografii 3**, potem zabudować do dowolnej obudowy. Można też wykonać małą płytkę drukowaną, zmontować na niej układ, a całość umieścić np. w pudełku po drażetkach, opakowaniu po wiertłach albo po szczoteczce do zębów (**fotografie 4 i 5**). Tu przydaje się własna inicjatywa.



Fot. 3



Rys. 2

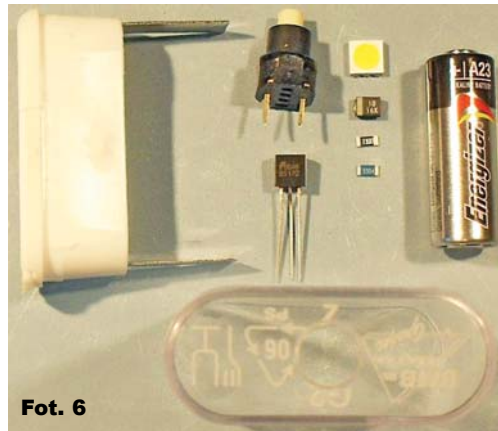


Fot. 4

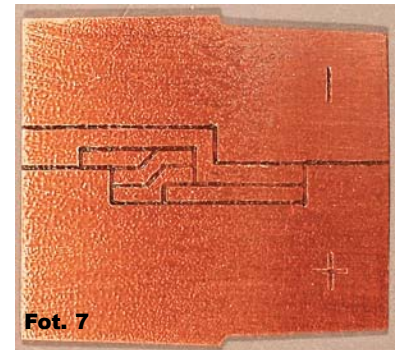
Można oczywiście zastosować fabryczną obudowę np. Z24Ap. Taką lampkę z zastosowanymi trzema pojedynczymi diodami LED znajdziemy na fotografii tytułowej. Przygotowana jest do zamocowania na samoprzylepnej taśmie a duży przycisk można nacisnąć np. łokciem, jeżeli mamy zajęte ręce. Potrzebne części i wykonana przeze mnie płytka do wersji „drażetkowej”, pokazane są na **fotografiach 6 i 7**. Można też wyposażyć lampkę w klosz pochodzący z przepalanej „żarówki” ledowej. Zdjęcie osiągniętego efektu znajdziecie w materiałach dodatkowych, a bonusy pokazują jeszcze inne ciekawe, jak sądzę, pomysły



Fot. 5



Fot. 6



Fot. 7



Fot. 8

nie pojedynczych diod LED i elementów THT. Wtedy nie musimy też wykonywać płytki drukowanej ©. Szczególną ostrożność należy zachować przy lutowaniu tranzystora MOSFET, który łatwo można uszkodzić ładunkiem statycznym przeniesionym np. z ubrania. Ja zawsze lutuję takie elementy, mając na palcu „magiczny” pierścień (fotografia 8), połączony przewodem z uziemionym zaciskiem w lutownicy. Można również zakupić w tym celu specjalne antystatyczne bransolety, opaski i maty, dostępne m.in. w sklepie AVT. Ale mój sposób jest również skuteczny. Dodatkowe pomocne zdjęcia różnych etapów wykonania można znaleźć w materiałach dodatkowych na Elportalu.

Uruchomienie układu nie wymaga użycia żadnych przyrządów i jeżeli nie było żadnego błędu, lampka powinna od razu działać. Baterię podłączamy do zacisków, oczywiście na samym końcu montażu, zwracając uwagę na biegunowość. Pewnej wprawy wymaga przylutowanie diody LED SMD, mniej doświadczonym Czytelnikom polecam zastosowa-

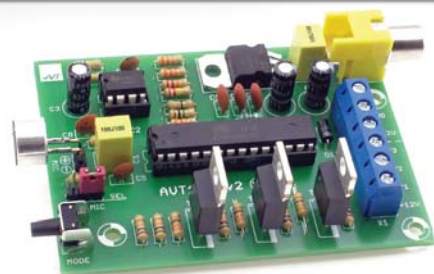


Piotr Świerczek
sp9egm@wp.pl

R E K L A M A

AVT 1853 Iluminofonia LED RGB

Kluczem do udanej imprezy jest nie tylko dobra muzyka, ale także dobre oświetlenie. Układ sterownika LED RGB współpracuje z taśmami LED lub modułami RGB. Konstrukcja spełni oczekiwania nawet najbardziej wybrednych imprezowiczów.



Latarka do... zębów?

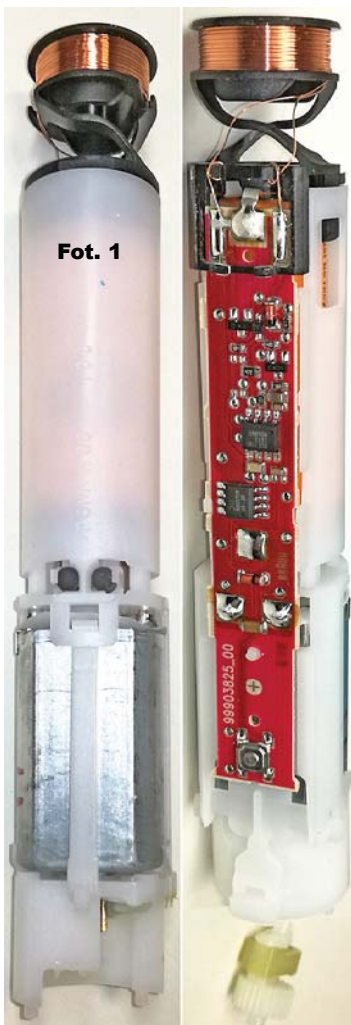
Jako rozwiązanie zadania 250 Szkoły Konstruktorów wykonałem latarkę ze szczoteczki do zębów ładowanej indukcyjnie. Latarka może służyć jako awaryjne oświetlenie w przypadku zaniku napięcia w domu. Dzięki stacji dokującej, jest zawsze na wiadomym miejscu i gotowa do pracy. W przypadku awarii umieszczenie jej w konkretnym miejscu, na przykład obok wejścia do piwnicy z bezpiecznikami, uchroni nas od niebezpieczeństw porażenia czy złamania nogi. Wiadomo, że większość z nas ma telefony komórkowe z funk-

cją latarki, ale dla osób starszych, zwłaszcza mieszkających na wsi, nie jest to oczywiste, a w dodatku ryzyko zaniku napięcia jest tam dość wysokie.

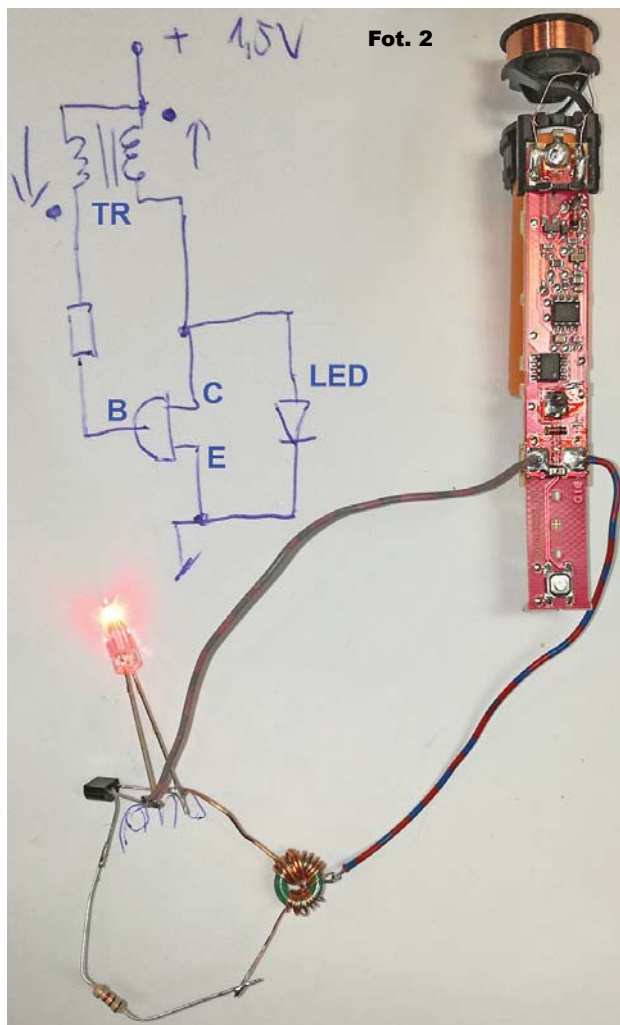
Opis układu

Ciekawiło mnie, co znajduje się w środku wspomnianej szczoteczki i jak można wykorzystać ładowanie indukcyjne, które notabene jest pomału osuwane na łamach EdW.

Bałem się, że układ jest zalany masą izolacyjną, ale otworzenie szczotki okazało się proste. Na szczęście większość popularnych modeli jest przedstawiona na YouTube w filmach instruktażowych dotyczących wymiany akumulatora.



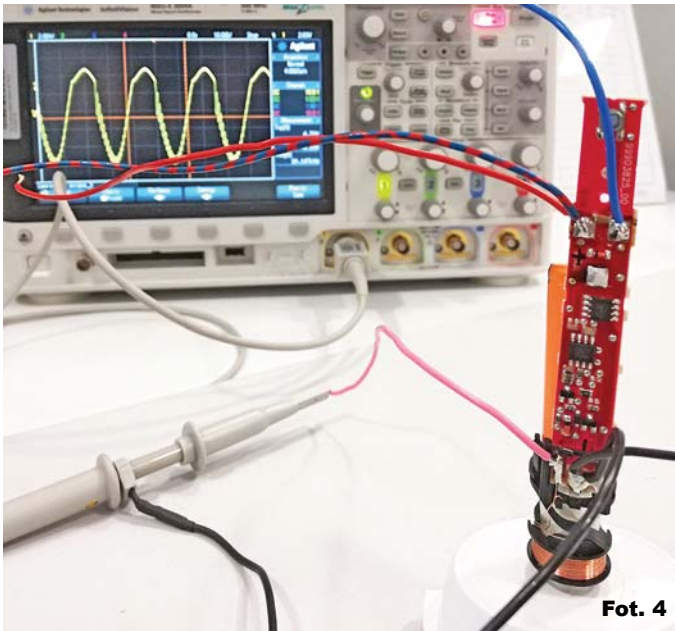
Fot. 1



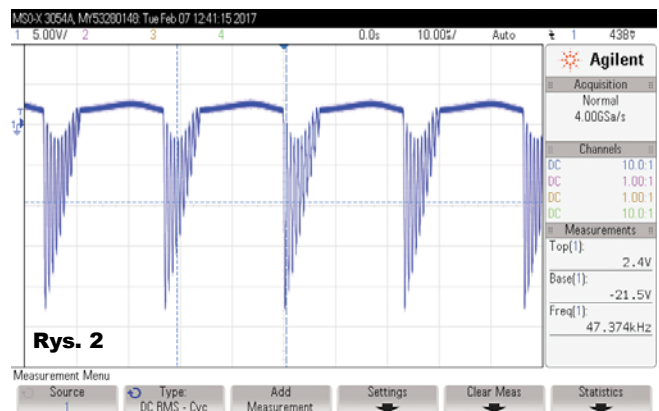
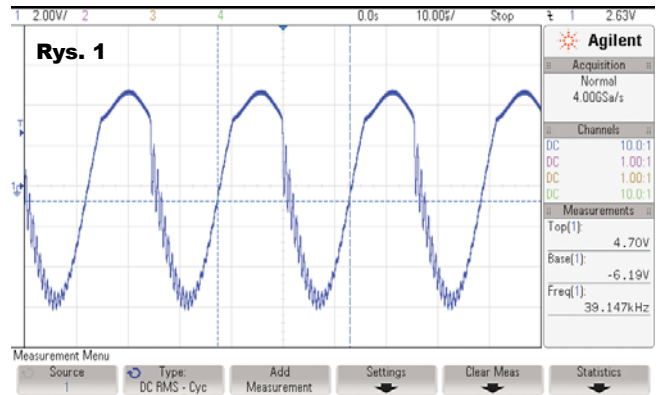
Fot. 2



Fot. 3



Fot. 4



Fot. 5

do rozwiązań modelarskich. Wspomniana cewka, stanowi wraz z podstawką nietypowy transformator.

Podczas przeróbki szczoteczki na latarkę napotkałem trudności związane z niskim napięciem, wynoszącym maksymalnie na naładowanym akumulatoru 1,37V. Jest to za mała wartość do zasilenia białej diody LED. Kolejnym etapem pracy była więc (udana) próba budowy słynnej przetwornicy z transformatora na koraliku, jednego tranzystora i rezystora oraz diody LED – **fotografia 2**. Układ pracuje poprawnie. Jednak docelowo zaimplementowałem układ z lampki solarnej (**fotografia 3**), szeroko omawianej niedawno w EdW. Całość sprawuje się świetnie i śmiało mogę polecić taką przeróbkę.

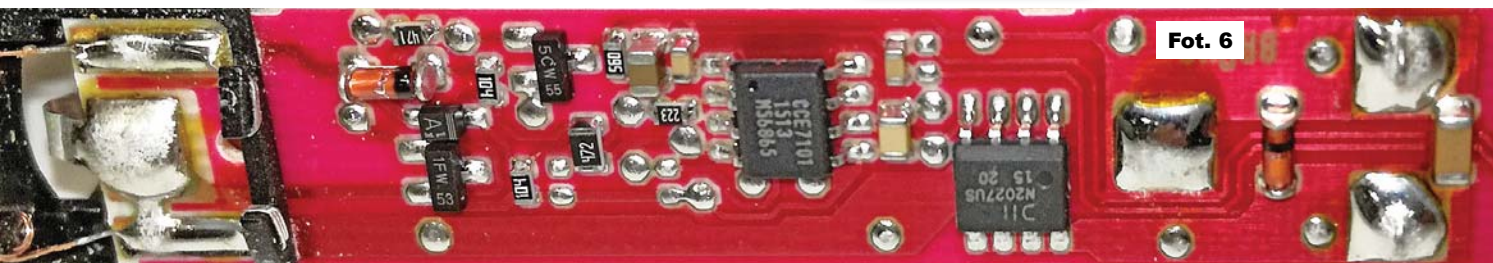
A teraz obrazki dla bardziej dociekliwych. **Fotografia 4** pokazuje układ podczas testów. **Rysunek 1** prezentuje przebieg napięcia indukowanego na cewce, podłączonej do

układu bez akumulatora. **Rysunek 2** to analogiczny przebieg z dołączonym akumulatorem (podczas ładowania). **Fotografia 5** pokazuje pomiar indukcyjności cewki – ponad 120mH. **Fotografia 6** przedstawia płytkę z MOSFET-em DMN2027USS (Diodes) oraz kostką (?) CCE7101 1513 M56865.

Mam nadzieję, że zdjęcia i zrzuty z oscyloskopu przydadzą się osobom, które na zasadzie reverse engineering chciałyby spróbować sił z ładowarkami powietrznymi, coraz popularniejszymi rozwiązaniami dla urządzeń przenośnych. Część ilustracji zmieściła się w artykule, a pozostałe dostępne są w Elportalu wśród materiałów dodatkowych do tego numeru EdW.



Dawid Placha
plachadawid@gmail.com



Fot. 6



Energooszczędny termometr diodowy

Termometry z wykorzystaniem magistrali 1-Wire to temat niezwykle popularny. Do wielu pomysłów realizacji dorzucam się i ja, proponując proste i energooszczędne pokazywanie wyników pomiarowych. Prezentowany układ zawiera praktycznie tylko popularne i tanie elementy elektroniczne. Jest prosty w budowie i działaniu.

Gotowe urządzenie widoczne jest na **fotografii tytułowej**, z której można się domyślić, w jaki sposób działa wyświetlacz. Schemat mojego układu przedstawiony jest na **rysunku 1**. Nie zawiera on żadnych niespodzianek. Podłączenie elementów jest typowe. Sercem układu jest mikrokontroler ATmega88, do którego dołączone są tak naprawdę tylko diody, czujnik DS18B20,

a także kilka niezbędnych kondensatorów i łącze KANDA dla programatora. Do procesora taktowanego wewnętrzną częstotliwością 1MHz został napisany kod w języku C z wykorzystaniem biblioteki Mirosława Kardasia [1Wire]. Ze względu na prawa autorskie tylko część programu dostępna jest w Elportalu w materiałach dodatkowych do tego numeru EdW.

W przedstawionej konstrukcji postanowiłem użyć energooszczędnych diod, które świecą bardzo jasno już przy prądzie 1mA. Wartości rezystorów ograniczających dobrałem do zastosowanych diod.

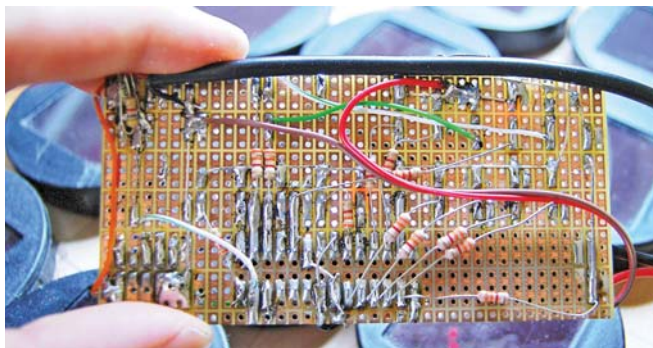
W mojej realizacji zdecydowałem się, by termometr współpracował z panelami słonecznymi, z akumulatorem i nie pobierał zasilania z sieci, ze względu na oszczędność prądu.

Rysunek 2 ukazuje mój sposób podłączenia zewnętrznych u k ł a d ó w zasilających. Panele słoneczne ładują akumulator

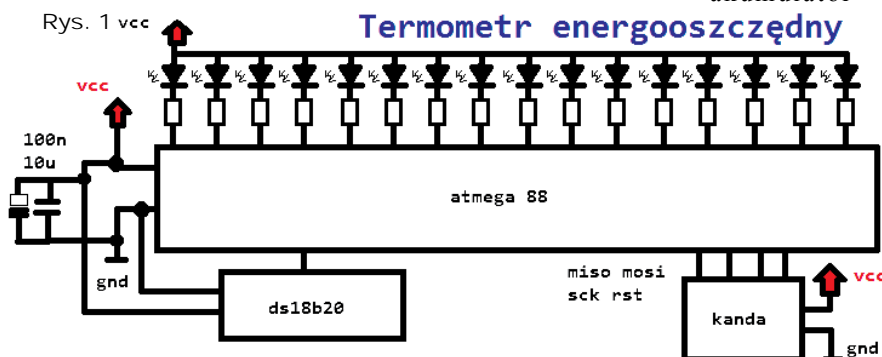
Li-Pol podłączony przez popularną ładowarkę opartą na układzie tp4056. Z ogniw natomiast przetwornica step-up podnosi napięcie do 5V i zasila termometr.

Układ zmontowałem na płytce uniwersalnej. Układ połączeń procesor – diody może być dowolny, ale oczywiście najlepiej zastosować kolejność, która odpowiada wycinkowi kodu dostępnemu w formie zdjęcia w Elportalu. Zmontowany układ należy zaprogramować poprzez, np. jak ja to uczyniłem, AVRStudio4, ze względu na bezpłatną obsługę ATmegi88. Pomiary wykazały, że podczas pracy układ pobierał prąd 10mA. Natomiast panel słoneczny osiągał do około 100mA prądu przy pełnym nasłonecznieniu, co powinno wystarczająco doładowywać akumulator.

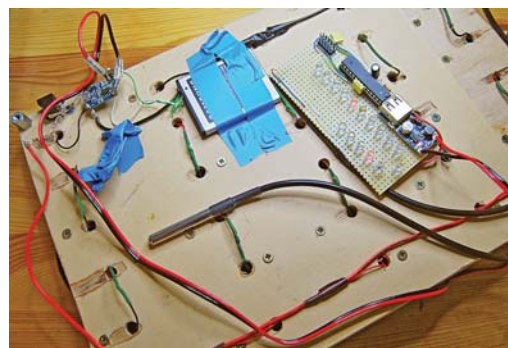
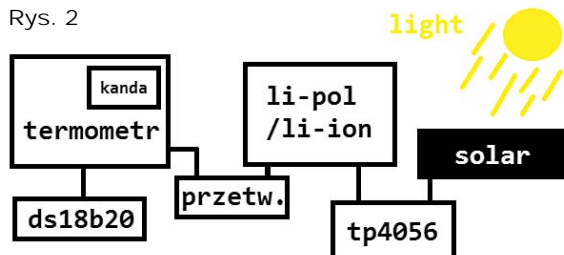
Artur Bereit
bereitartur5@gmail.com



Rys. 1 vcc

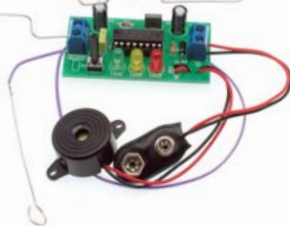


Rys. 2



AVT 723 Uniwersalna gra zręcznościowa

Zadaniem gry jest jak najszybsze "wyjście z labiryntu" bez drgnięcia ręki. Zwycięza ten kto bez błędów pierwszy pokona całą trasę. Dla zwiększenia atrakcyjności układ dopuszcza dwie "wpadki". Stopień trudności można dowolnie regulować przez zmianę kształtu i długości trasy.



Wybrane parametry:

- doskonała rozrywka - indywidualna i grupowa
- cel - bezbłędne i jak najszybsze przejście labiryntu
- doskonale ćwiczenie zręczności i 'pewnej' ręki
- sygnalizacja błędów - diody LED i dźwięk
- zasilanie 4,5...15 VDC (bateria 9V)
- wymiary płytki: 36x57 mm

AVT 1661 Elektroniczna kostka do gry

Układ jest elektronicznym odpowiednikiem tradycyjnej kostki do gry. Przyciśnięcie przycisku powoduje rozpoczęcie generowania wyniku – jest on prezentowany na panelu z diodami świecącymi.



Wybrane parametry:

- jako wskaźnik liczby oczek diody LED
- inicjowanie "losowania" - mikroswitch
- łatwy montaż i uruchomienie
- zasilanie 6...12 VDC
- wymiary płytek: 50x50 mm

AVT 788 Lampko-klask - lampka reagująca na klaśnięcie

Układ reaguje na pojedyncze klaśnięcie w dłonie. Każde kolejne klaśnięcie włącza lub wyłącza białą diodę LED o podwyższonej jasności. Płytką drukowaną została tak zaprojektowana, by zmieściła się na koszyku baterii. Może posłużyć jako bezprzewodowa lampka nocna.



Wybrane parametry:

- źródło światła: biała dioda LED
- pobór prądu w czasie czuwania: <200uA
- pobór prądu w czasie świecenia: max. 15mA
- zasilanie: 4,5 VDC (3x bateria AA)
- wymiary płytki: 35x52 mm

AVT 5554 Snake

Gra „Snake” jest nieskomplikowanym modułem zbudowanym z użyciem mikrokontrolera, wyświetlacza graficznego oraz 5 przycisków pełniących rolę interfejsu użytkownika. Za ich pomocą sterujemy wężem poruszającym się na wyświetlaczu. Naszym celem jest zbieranie jedzenia, od którego – co naturalne – wąż rośnie. Trzeba jednocześnie uważać, aby nie zjeść przy tym własnego ogona...

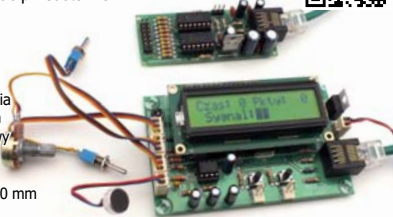


Wybrane parametry:

- interfejs użytkownika:
 - 5 przycisków
 - wyświetlacz 84x48 pixeli
- zasilanie 3...5 VDC / maks. 100 mA
- wymiary płytki: 45x82 mm

AVT 5310 Miernik natężenia okłasków

Applause Meter to rodzaj miernika poziomu dźwięku służącego do pomiaru natężenia okłasków. Często można go zobaczyć podczas turniejów. Proponujemy wykonanie takiego urządzenia. Moduł sprawdzi się podczas konkursów szkolnych lub przedstawień.



Wybrane parametry:

- dwustopniowa regulacja wzmocnienia
- napęd wskazówki: serwomechanizm
- wbudowany mikrofon pojemnościowy
- czas trwania pomiaru 10 lub 20 sek
- zasilanie 7...12 VDC
- wymiary płytek: 100x60 mm i 76x30 mm

AVT 1651 Gra - Kto pierwszy ten lepszy

Ten nieskomplikowany układ, zbudowany z układów cyfrowych - co ważne - bez mikrokontrolera na niewielkiej płytce drukowanej, przyda się np. do organizowania teleturniejów. Grać może do czterech osób, a odpowiednia dioda świecąca sygnalizuje, która z nich w danej rozgrywce była najszybsza.



Wybrane parametry:

- maksymalnie 4 graczy
- sygnalizacja zwycięzcy diodą LED
- dodatkowy przycisk RESET
- łatwy montaż i uruchomienie
- zasilanie 5 VDC
- wymiary płytki: 76x76 mm

AVT 789 Syrena dla pojazdów uprzywilejowanych

Łatwy w wykonaniu moduł o nieograniczonym zastosowaniu, generujący trzy rodzaje syren pojazdów uprzywilejowanych oraz doskonale imitujący strzał z karabinu maszynowego.

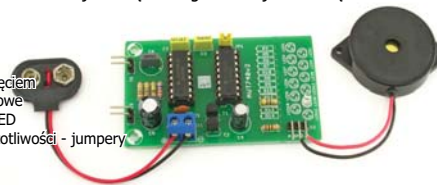


Wybrane parametry:

- 4 różne efekty dźwiękowe
- wybór dźwięku zworką
- źródło dźwięku: przetwornik piezo
- regulacja częstotliwości generowanego dźwięku
- zasilanie 3 VDC (2x AA)
- wymiary płytki: 26x37 mm

AVT 740 Dotykowy generator dźwięku syreny

Układ jest niezwykłym generatorem, którego częstotliwość jest płynnie przestrajana w górę i w dół za pomocą sensorów dotykowych. Duża szybkość przestrajania oraz możliwość przełączenia zakresu częstotliwości za pomocą jumperka, pozwala wytwarzać setki rozmaitych niesamowitych sekwencji dźwięków i gwarantuje świetną zabawę.



Wybrane parametry:

- generator przestrajany napięciem
- sterowanie – sensory dotykowe
- sygnalizacja pracy - dioda LED
- przełączanie zakresów częstotliwości - jumpery
- zasilanie: 3...12 VDC
- wymiary płytki: 43x74 mm

AVT 732 Whisper - łowca szepcót

Moduł jest podsłuchem przewodowym. Idealnie nadaje się również do rozmaitych eksperymentów związanych ze wzmocnianiem bardzo cichych dźwięków. Może być pomocny osobom z lekkim niedosłuchem, doskonale sprawdzi się do podsłuchiwania przyrody ale również monitorowania snu dzieci.



Wybrane parametry:

- wbudowany mikrofon
- odsłuch na słuchawkach (wbudowane gniazdo 3,5mm)
- regulacja wzmocnienia
- zasilanie 3...6 VDC (4x bateria AAA)
- wymiary płytki: 46x56 mm

AVT 5540 Radiodbiornik dla każdego

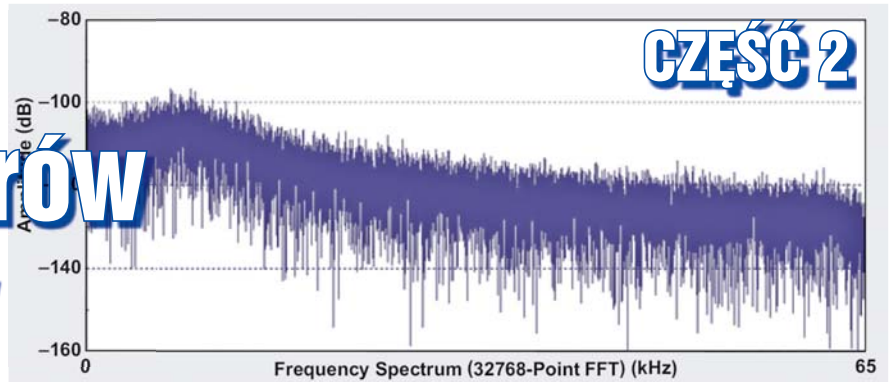
Moduł odbiornika wykonano z użyciem popularnych i łatwych w montażu elementów, więc jego wykonanie nie sprawi trudności nawet początkującym. Łatwy w użytkowaniu dzięki interfejsowi użytkownika zbudowanego z wyświetlacza LCD oraz impulsatorów. Wbudowany wzmacniacz o mocy 2x2W.



Wybrane parametry:

- nie wymaga strojenia
- współpracuje ze słuchawkami lub z zestawem głośników
- odbiór stacji radiowych w zakresie 87.5...108 MHz
- odbiór i wyświetlanie informacji RDS
- pamięć 8 stacji radiowych
- zasilanie: 7...15 VDC/0.3A
- niewielkie wymiary: 140x38x45 mm

Szumy stabilizatorów i zasilaczy



W pierwszej części artykułu omówione zostały parametry szumowe popularnych stabilizatorów LM317, 78xx i wspomniane było o dużych szumach własnych dawnych opracowań stabilizatorów LDO.

W popularnych stabilizatorach liniowych omówionych spodziewana wartość szumów jest rzędu 1 miliwolta lub trochę mniej, ale istnieją nowoczesne stabilizatory, których szumy własne są dziesiątki i setki razy mniejsze!

Skrajnym przykładem jest półamperowy stabilizator Linear Technology o oznaczeniu LT3045. Jak pokazuje rysunek 9, szumy własne w pasmie 10Hz...100kHz wynoszą 0,8uV. Tak, **0,8 mikrowolta**, a nie miliwolta, co daje wartość międzyszczytowa do 5,3 mikrowolta, czyli 0,0053mV.

Zwróć uwagę, że podana wartość współczynnika tłumienia tętnień PSRR, równa 76dB, dotyczy tętnień wejściowych o wysokiej częstotliwości 1MHz (np. z nowoczesnego zasilacza impulsowego). Natomiast typowa wartość tłumienia tętnień o częstotliwości sieci energetycznej 100Hz (120Hz) to imponujące 117dB, czyli ponad 700 tysięcy razy. Gdyby na wejściu występował przebieg z tętnieniami 3Vpp 100Hz, to na wyjściu zostałyby one stłumione do około 4 mikrowoltów. Rysunek 10 pokazuje zależność współ-

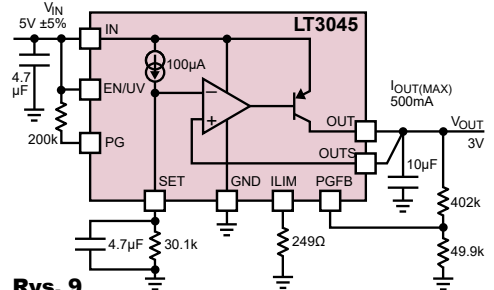
czynnika PSRR od częstotliwości dla dwóch stabilizatorów. Kostka LT3045 do częstotliwości ponad 1MHz zapewnia tłumienie tętnień ponad 75dB, czyli ponad 5600 razy. Dla kostek rodziny 78xx poszczególne firmy podają w tym zakresie znacząco inne dane. Na rysunku 10 pokazana jest charakterystyka z katalogu National Semiconductor (obecnie TI). Widać, że 24-woltowa wersja już przy częstotliwości 0,1MHz ma tłumienie poniżej 40dB, czyli mniej niż 100 razy. Różnica jest ogromna!

Wśród parametrów szumowych tego rodzaju stabilizatorów dość często podaje się też napięciową gęstość szumów w funkcji częstotliwości (w nanowoltach na pierwiastek z herca). Znając gęstość szumów i pasmo częstotliwości, w jakim będzie pracował układ, można wyliczyć wartość napięcia szumów w takich warunkach. Rysunek 11 pokazuje taki wykres dla stabilizatora LT3045.

LINEAR TECHNOLOGY LT3045

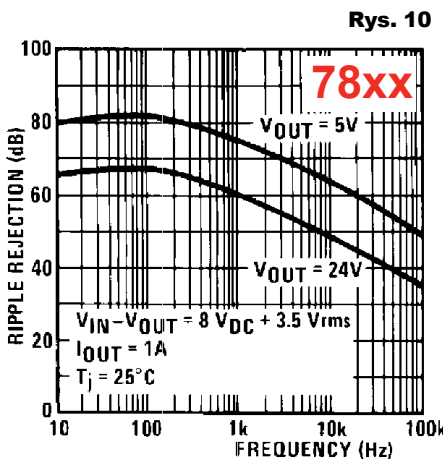
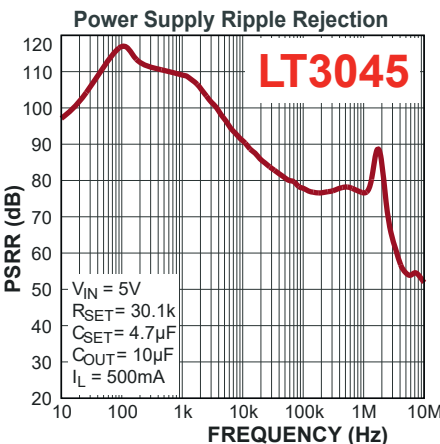
20V, 500mA, Ultralow Noise, Ultrahigh PSRR Linear Regulator
 Ultralow RMS Noise: 0.8µV_{RMS} (10Hz to 100kHz)
 Ultralow Spot Noise: 2nV/√Hz at 10kHz
 Ultrahigh PSRR: 76dB at 1MHz
 Output Current: 500mA
 Wide Input Voltage Range: 1.8V to 20V
 Single Capacitor Improves Noise and PSRR

TYPICAL APPLICATION

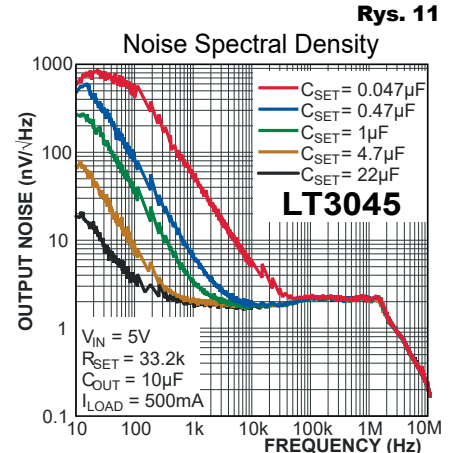


Rys. 9

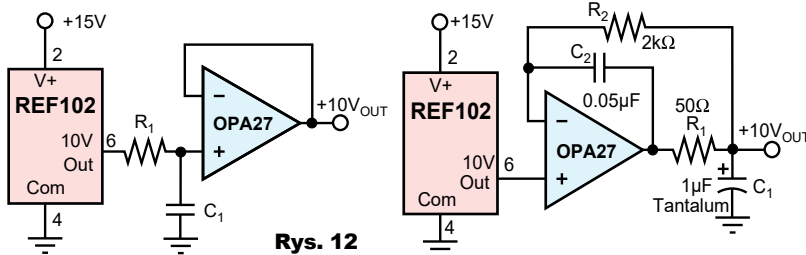
potrzebne jest też jako napięcie wzorcowe (odniesienia) do przetworników ADC o wysokiej rozdzielczości. Przykładowo przetwornik 24-bitowy w idealnym przypadku rozróżnia 2²⁴, czyli 16 777 216 poziomów. Jednemu bitowi odpowiada 0,0000000596 wartości napięcia odniesienia, czyli 0,00000596%, co daje 0,0596ppm (parts per million – części



Rys. 10



Rys. 11



Rys. 12

na milion). Jeżeli 24-bitowy przetwornik ADC byłby idealny, to aby nie stracić dokładności, międzyszczytowa amplituda szumów napięcia odniesienia powinna być mniejsza niż wartość 1LSB, czyli w tym przypadku 0,0596ppm. Gdy przykładowo napięcie odniesienia wynosi 2,5V, jednemu bitowi (1LSB) odpowiada napięcie 0,000000149V, czyli około 0,15 mikrowolta = 150nV. Czyli amplituda szumów powinna być mniejsza niż 150 nanowoltów. Chodzi o amplitudę międzyszczytową – wymagana wartość napięcia skutecznego szumów powinna w tym przypadku wynosić mniej niż 0,023 mikrowolta czyli 23nV. Tak dobrych źródeł napięcia odniesienia... nie ma.

Tu trzeba nadmienić, że szумы o wyższych częstotliwościach można odfiltrować w prosty sposób za pomocą kondensatorów. Ale nie można się w ten sposób pozbyć szumów o małej częstotliwości rzędu 1Hz i mniej, bo wymagałoby to kondensatorów o niepraktycznie dużej pojemności. I właśnie w wielu zastosowaniach przetworników analogowo-cyfrowych kluczowe znaczenie mają składowe szumów o częstotliwościach 0,01Hz...10Hz.

A jeśli chodzi o składowe szumów o najniższych częstotliwościach rzędu ułamków herca, to przechodzą one w kolejny parametr decydujący o precyzji: chodzi o drypt i stabilność długoczasową. W tym zakresie o szumach decydują tylko właściwości użytych elementów półprzewodnikowych. W niektórych zastosowaniach nadal rozsądną alternatywą dla scalonych układów źródeł napięć odniesienia są specjalne (tzw. *buried* – zagrzebane) diody Zenera. W przypadku bardzo popularnych scalonych źródeł napięcia odniesienia typu *bandgap* szумы mogą być stosunkowo duże i wtedy w obwodzie napięcia odniesienia bywają stosowane dodatkowe filtry. Mogą to być filtry pasywne RC. **Rysunek 12** pochodzi z materiałów Texas Instruments (sbva002) i pokazuje dwa przykłady filtracji napięcia odniesienia.

Jeżeli chodzi o niskoszumne źródła napięcia odniesienia, to skrajnym przykładem może być najwyższej klasy układ

Linear Technology LTC6655. Jak widać z **rysunku 13**, międzyszczytowa amplituda szumów to tylko 0,25ppm (0,000025%), co dla wersji 2,5-woltowej daje 625nVpp (0,625uVpp). Inne scalone źródła napięcia odniesienia mają szумы znacznie większe. W szczególności nie będziemy się zagłębiać, w każdym razie poszczególne układy źródeł napięcia odniesienia mają zdecydowanie różne parametry szumowe.

Jeśli ktoś chce zrealizować naprawdę dobry wielobitowy przetwornik analogowo-cyfrowy (lub cyfrowo-analogowy), oprócz wszelkich kwestii „cyfrowych”, powinien też wziąć pod uwagę szумы, zarówno szумы zasilania, jak i szумы napięcia odniesienia.

Inne problemy

Na koniec jeszcze jedna sprawa. Otóż jednym ze źródeł problemu są zakłócenia występujące na wejściach i wyjściach zasilaczy sieciowych, zwłaszcza impulsowych.

W dzisiejszej praktyce problem szumów i zakłóceń często dotyczy właśnie przetwornic impulsowych, które z zasady mają na wyjściu bardzo dużo „śmieci”.

W tanich i prostych przetwornicach trudno zmniejszyć poziom generowanych zakłóceń. Dodanie pomocniczego liniowego stabilizatora 78xx czy LM317 na wyjściu takiego zasilacza impulsowego (według wcześniejszego rysunku 2) poprawia sytuację, ale tylko częściowo. Walka z zakłóceniami i szumami w zasilaczach impulsowych jest bardzo trudna, bo w grę wchodzi nie tylko współczynnik PSRR, ale i przenikanie zakłóceń na wyjście innymi drogami. Zwykle najlepszym rozwiązaniem jest zasilanie układów najbardziej niskoszumnych za pomocą klasycznych zasilaczy liniowych albo baterii czy akumulatorów.

W niektórych układach, gdzie niezbędne są małe szумы, stosuje się zasilanie bateryjne. Według powszechnej opinii *baterie i akumulatory nie szumią, bo nie ma w nich elektroniki*. Rzeczywiście,

LINEAR
TECHNOLOGY

NOW PART OF

ANALOG
DEVICES

LTC6655

0.25ppm Noise, Low Drift Precision References

Low Noise: 0.25ppm_{p-p} (0.1Hz to 10Hz)
625nV_{p-p} for the LTC6655-2.5

Low Drift: 2ppm/°C Max

High Accuracy: ±0.025% Max

No Humidity Sensitivity (LS8 Package)

Thermal Hysteresis (LS8): 30ppm (–40°C to 85°C)

Long-Term Drift (LS8): 20ppm/√kHr

100% Tested at –40°C, 25°C and 125°C

Load Regulation: <10ppm/mA

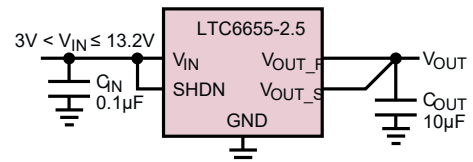
Sinks and Sources Current: ±5mA

Low Dropout: 500mV

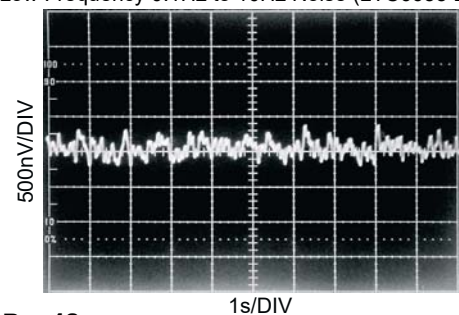
Maximum Supply Voltage: 13.2V

Low Power Shutdown: <20µA Max

Available Output Voltages: 1.25V, 2.048V, 2.5V, 3V, 3.3V, 4.096V, 5V



Low Frequency 0.1Hz to 10Hz Noise (LTC6655-2.5)



Rys.13

szумы baterii, a zwłaszcza akumulatorów są znikome. Ale też nie są to elementy bezszumne. Po pierwsze, najprościej biorąc, szumi ich rezystancja wewnętrzna. Wprawdzie jest ona mała i na schemacie zastępczym połączona jest z reaktancją pojemnościową o znikomej wartości, jednak już prosty model zastępczy wskazuje, że jakieś niewielkie szумы występują także na zaciskach baterii i akumulatorów. Nieco więcej szczegółów można znaleźć w Internecie, choćby na stronie <https://tf.nist.gov/general/pdf/1133.pdf> w skrócie: <https://goo.gl/Zjz14P>.

Przedstawiony materiał absolutnie nie wyczerpuje tematu szumów, precyzji i dokładności. Sygnalizuje jedynie niektóre zagadnienia z tym związane oraz uczula na problem zakłóceń i szumów.

(red)

Taśmy LED

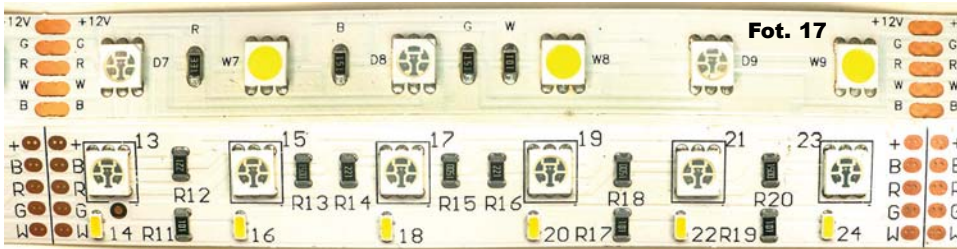
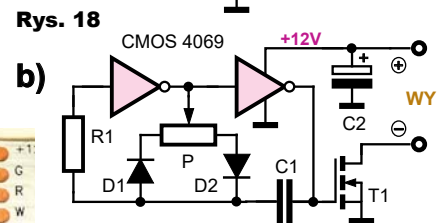
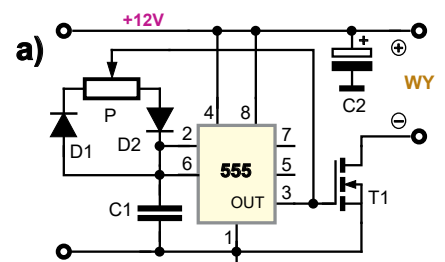
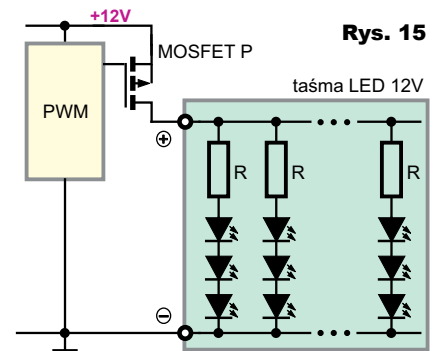
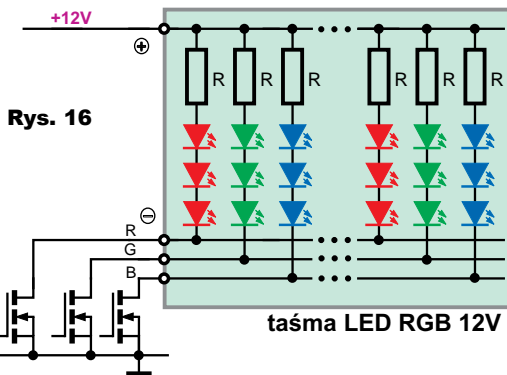
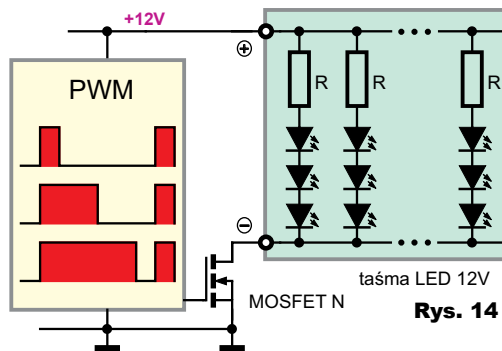
Wskazówki praktyczne

W pierwszej części artykułu omówiliśmy budowę i specyfikę sterowania różnych odmian taśm LED.

Sterowanie taśm 12V

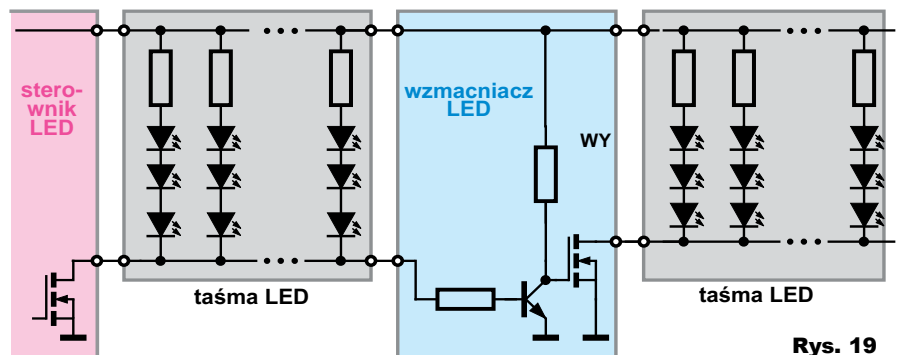
Choć zmiana jasności taśm LED przez zmianę wartości napięcia zasilania ma istotne zalety, według powszechnych zaleceń, **regulacja jasności powinna być realizowana w sposób impulsowy**. Stabilizowane napięcie zasilające powinno wynosić 12V, a zmieniać należy wypełnienie impulsów o częstotliwości 100...1000Hz, na przykład według idei z **rysunku 14**. Za małą częstotliwość, poniżej 100Hz, spowoduje widoczne, irytujące i męczące oczy migotanie diod, natomiast za duża, powyżej 1kHz może niepotrzebnie powodować zwiększenie zakłóceń impulsowych emitowanych przez taką instalację.

Zmiana wypełnienia impulsów oznacza, że regulujemy **średnią** wartość napięcia i prądu. A w tych momentach, gdy dioda pracuje, niezależnie



od wypełnienia impulsów jej prąd pracy jest jednakowy w całym zakresie regulacji, a to oznacza, że nie zmienia się barwa światła, co jest korzystne. Przy sterowaniu impulsowym zakres liniowej regulacji jest szerszy niż przy sterowaniu „napięciowym” (pozostaje tylko kwestia logarytmicznej czułości oka).

Układ sterowania „od dołu” według rysunku 14 jest powszechnie stosowany, ale w zasadzie pojedynczymi taśmami równie dobrze można stero-



wać „od góry” według **rysunku 15**. Najczęściej jednak wykorzystuje się sterowanie „od dołu” według rysunku 14 właśnie z wykorzystaniem tranzystorów MOSFET N,

a nie tranzystorów NPN, co też byłoby możliwe. Sterowanie „od dołu” jest koniecznością w przypadku taśm RGB i RGBW, gdzie elektrodą wspólną jest zawsze dodatnia linia zasilania – **rysunek 16** i **fotografia 17**.

Ściemniacze diod LED są więc generatorami o zmiennym wypełnieniu z MOSFET-em wyjściowym. Proste przykłady masz na **rysunku 18**. Istnieją też tak zwane *wzmocniacze LED*. Są stosowane tam, gdzie istniejący sterownik ma za małą wydajność prądową i dołączenie dłuższych taśm LED wymaga dodatkowego układu sterującego według **rysunku 19**.

W handlu, oprócz prostych ściemniaczy, dostęp-



Fot. 20

nych jest mnóstwo skomplikowanych sterowników, także zdalnie sterowanych podczerwienią lub częściej radiowo. Coraz częściej spotykane są sterowniki zdalnie sterowane, także przez Wi-Fi i Internet – przykład na **fotografii 20** (miLight).

Jak widać, podstawowe zasady sterowania taśmami LED są więc bardzo proste, jednak w praktyce nawet doświadczeni elektrycy i elektronicy są często zaskakiwani zachowaniem taśm LED. Omówimy to w trzeciej części artykułu.

Piotr Górecki

R E K L A M A

Moduły do projektów ARDUINO

- Bazowe kompatybilne z Arduino i Shield'y
- Audio, Video
- Czujniki
- Dekodery
- Generatory
- GPS, GSM, WiFi, Bluetooth
- Interfejsy, konwertery
- Klawiatury Joysticki Piloty
- Moduły z przekaźnikami
- Radiowe
- Na pasmo 433MHz, 868MHz, 2.4GHz
- Pomiar czasu
- Pomiar częstotliwości
- Pomiar temperatury
- Przetwornice
- RFID
- Robotyka, platformy jezdne, obudowy
- Serwomechanizmy
- Sterowniki, regulatory
- Wyświetlacz
- Wzmocniacze
- Zasilanie

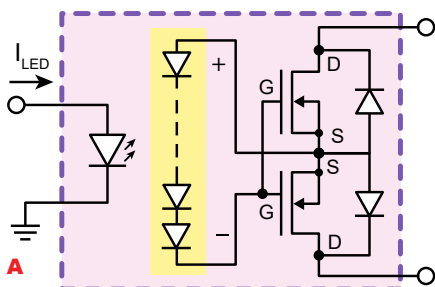
Zapraszamy się do zapoznania z ofertą sklepu: sklep.avt.pl/category/moduly



sklep.avt.pl handlowy@avt.pl tel.: 22 257 84 50

Jak to działa

Na **rysunku A** przedstawiony jest nieskomplikowany układ elektroniczny.



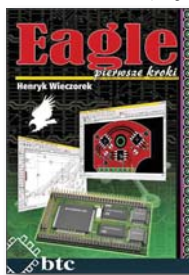
Jak zwykle zadanie konkursowe polega na rozszyfrowaniu

Jak działa i do czego służy taki układ?

Odpowiedzi, koniecznie oznaczone dopiskiem **Jak3**, należy nadsyłać w terminie 45 dni od ukazania się tego numeru EdW. Nagrodami w konkursie będą 3 cenne książki wydawnictwa BTC: Henryk Wiczorek, **EAGLE – pierwsze kroki**.

Rozwiązanie zadania z EdW 11/2017

W numerze 11/2017 przedstawiony był, pokazany na **rysunku B**, nieskomplikowany układ z kostką NE555.



Omawiany układ ma wytwarzać silne, ale bardzo krótkie impulsy prądu i te impulsy mają pomóc w oczyszczeniu powierzchni płyt z nieprzewodzącego osadu/nalotu.

Zasiarczenie płyt, najogólniej biorąc, polega na wytworzeniu na ich powierzchni nieprzewodzących kryształów siarczynu ołowiu ($PbSO_4$). Podczas użytkowania powstawanie siarczynu ołowiu jest normalnym procesem (dwutlenek ołowiu z płyt w reakcji z kwasem siarkowym z elektrolitu tworzy siarczyn ołowiu). Problemem jest to, że przy całkowitym rozładowaniu akumulatora i pozostawieniu go w takim stanie na dłuższy czas siarczyn ołowiu zamienia się w postać krystaliczną, która nie przewodzi prądu.

Powstanie na powierzchni płyt nieprzewodzącej warstwy krystalicznej jest niekorzystne, bowiem zmniejsza pojemność, a usunięcie nieprzewodzących kryształów jest trudne, czasem niemożliwe. Stosuje się różne sposoby technologiczne, by zmniejszyć ryzyko zasiarczenia. Jednak generalnie akumulatory kwasowe nie powinny być rozładowywane „do zera”

i absolutnie nie powinny być pozostawiane w takim stanie. Problem dotyczy też długiego pozostawiania w stanie poważnego rozładowania lub na granicy rozładowania, gdy elektrolit to prawie sama woda – to także sprzyja zasiarczeniu. Akumulatory kwasowe powinny pozostawać (i być przechowywane) w stanie naładowania. Niestety „od zawsze” zdarzały się i zdarzają sytuacje, gdy akumulator kwasowy pozostaje przez długi czas całkowicie rozładowany lub jest na granicy rozładowania, a to powoduje mniejsze czy większe zasiarczenie.

Dawniej stosowano różne, dość skomplikowane próby odsiarczenia, czyli usuwania nieprzewodzących kryształów z powierzchni płyt. Zwykle wiązało się to z wymianą elektrolitu i zmianą jego parametrów, ale takie zabiegi dawały albo niewielki, albo krótkotrwały efekt. Opis przykładowej recepty, podobno autorstwa Stefana Sękowskiego, można znaleźć na stronie

http://solis.pl/index.php/elektronika/informacje_dodatkowe/jak_uratowac_zuzyty_akumulator

w skrócie: <https://goo.gl/VE3s11>.

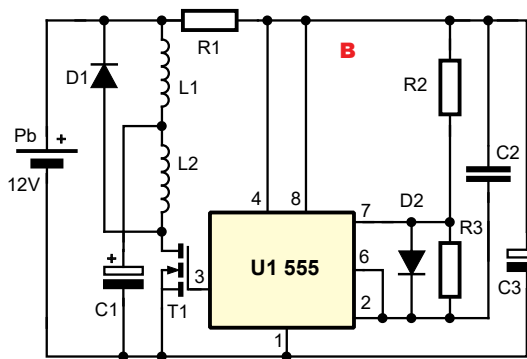
Od wielu lat mówi się, że *skutecznym sposobem odsiarczenia płyt jest przepuszczanie przez akumulator krótkich, bardzo silnych impulsów prądowych*.

W Internecie jest mnóstwo informacji na ten temat, ale niestety, przytłaczająca większość to powtarzanie stereotypowych sformułowań i opinii bez głębszego zrozumienia zagadnienia. Bardzo trudno jest znaleźć rzetelne, sprawdzone informacje. Przy obecnych cenach i dostępności akumulatorów można się też zastanawiać, na ile opłacalne i sensowne są takie zabiegi.

Niemniej w wielu przypadkach warto spróbować, choćby tylko z ciekawości. I wtedy w grę wchodzi między innymi wykorzystanie koncepcji z rysunku B.

Aby lepiej zrozumieć działanie układu, warto przerysować schemat do nieco uproszczonej postaci z **rysunku C**. Mamy tu trzy bloki: akumulator, filtr LC oraz przetwornicę impulsową.

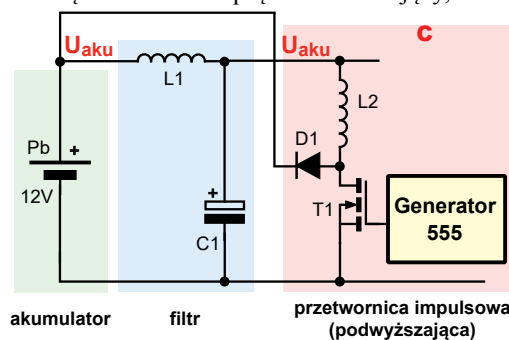
Prąd i energia do zasilania przetwornicy impulsowej jest oczywiście pobierana z akumulatora. Ale co ważne, pomocniczym źródłem zasilania przetwornicy jest kondensator C1. Obecność gromadzącej energię filtru L1, C1 powoduje, że prąd pobierany z akumulatora nie ma charakteru impulsowego. Na schemacie nie są podane wartości elementów, jednak można przypuszczać, że indukcyjność L1 jest stosunkowo duża. Cewka L1 ma małą rezystancję i dużą reakcję. Mała rezystancja powoduje, że dla napięcia stałego cewka L1 jest zwarcieniem i napięcie po obu jej stronach jest jednakowe (U_{aku}). Dla przebiegów zmiennych i impulsowych cewka ma dużą reakcję, więc dla takich przebiegów oddziela akumulator od pomocniczego kondensatora C1. Dzięki filtrowi L1, C1 z akumulatora pobierany jest stały, niezmienny prąd obciążenia. Jest to prąd rozładowujący,

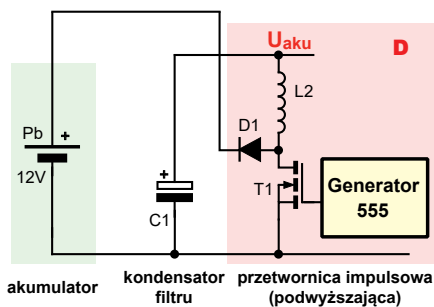


Jest to...

...tak zwany *desulfator*, czyli układ, który ma zapobiegać zasiarczeniu płyt akumulatorów kwasowo-ołowiowych i takie zasiarczenie usuwać. Tego rodzaju układy bywają stosowane do przedłużenia żywotności, a także przy próbach reanimowania, czyli przywrócenia do życia uszkodzonych akumulatorów kwasowo-ołowiowych.

Uwaga! Układ nie był testowany w redakcji EdW ani w pracowni AVT – jest to tylko schemat wzorowany na podobnych schematach, których wiele można znaleźć w Internecie.





„ujemny”. Co ważne, dzięki obecności separującej cewki L1 można stwierdzić, że dla przebiegów zmiennych przetwornica jest oddzielona od akumulatora, jak pokazuje to **rysunek D**.

Układ z cewką L2, diodą D1 i tranzystorem-kluczem T1 wygląda jak klasyczna przetwornica podwyższająca. I tak jest w rzeczywistości, tylko nietypowe jest obciążenie tej przetwornicy. Ale rysunek D wyjaśnia, że w chwilach, gdy cewka L2 wytwarza impuls napięcia, prąd płynie przez diodę D1 i ładuje akumulator.

Działanie układu jest więc dość dziwne. Z jednej strony rozładowuje on akumulator jakimś niewielkim prądem stałym, ale większość pobranej energii wraca z powrotem do akumulatora w postaci impulsów ładujących. Wraca nie cała energia, tylko większość z uwagi na niedoskonałą sprawność desulfatora.

Wcześniej stwierdziliśmy, że *skutecznym sposobem odsiarczania płyt jest przepuszczanie przez akumulator krótkich, bardzo silnych impulsów prądowych*. A to znaczy, że parametry układu z rysunku B należy tak dobrać, żeby akumulator był ładowany krótkimi, silnymi impulsami prądowymi.

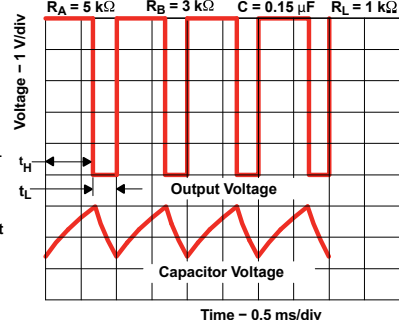
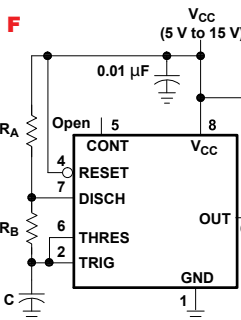
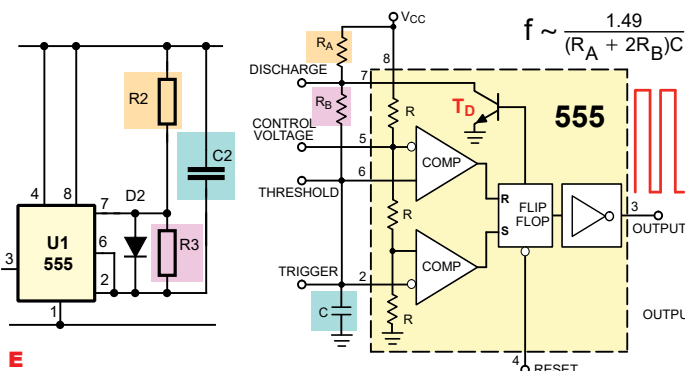
W tym miejscu warto przyjrzeć się układowi 555. Z prawej strony **rysunku E** mamy fragment schematu z rysunku B. Natomiast z prawej strony mamy pochodzący z karty katalogowej kanoniczny układ generatora na kostce 555. Różnice są niewielkie. Porównanie to wskazuje, że w układzie z rysunku B kostka 555 ma być generatorem. Dołączenie kondensatora C2 do plusa zasilania, a nie do masy, nie ma żadnego wpływu na pracę generatora. Istotna natomiast jest obecność diody D2.

Działanie klasycznego generatora można przeanalizować w oparciu o prawą część rysunku E. Gdy tranzystor rozładowujący T_D jest zatkany, na wyjściu 3 panuje stan wysoki. Kondensator C ładuje się wtedy prądem płynącym przez szeregowo połączone rezystory R_A i R_B. Płynnie narasta napięcie na kondensatorze C i napięcie to doprowadzone jest do wejść dwóch komparatorów przez nóżki 2 i 6. Gdy napięcie na kondensatorze C wzrośnie i przekroczy wartość

2/3 napięcia zasilania (takie napięcie występuje na nóżce 5 połączonej z dzielnikiem składającym się z trzech jednakowych rezystorów), zostanie przełączony „górny” komparator. Spowoduje to zmianę stanu przerzutnika flip-flop: na wyjściu pojawi się stan niski i zostanie otwarty tranzystor rozładowujący T_D. Otwarcie T_D, czyli w sumie wewnętrzne zwarcie nóżki 7 do masy spowoduje, że kondensator C, naładowany wcześniej do 2/3 napięcia zasilania zacznie się rozładowywać. Napięcie na kondensatorze C zacznie się zmniejszać i gdy zmniejszy się do wartości 1/3 napięcia zasilania, zostanie przełączony „dolny” komparator, który z kolei przełączy przerzutnik flip-flop. Spowoduje to pojawienie się na wyjściu 3 stanu wysokiego i zatkanie tranzystora rozładowującego T_D. Kondensator C znów zacznie się ładować prądem płynącym od plusa zasilania przez szeregowo połączone rezystory R_A i R_B. Cykl będzie się powtarzał.

Zwróć uwagę, że ładowanie odbywa się tu prądem mniejszym, bo płynącym przez sumę rezystancji R_A + R_B. Natomiast prąd rozładowania jest większy, określony tylko przez rezystancję R_B. Obecność rezystora R_A nie ma znaczenia podczas rozładowania. A więc czas rozładowywania będzie krótszy niż czas ładowania. A to znaczy, że w klasycznym generatorze tego rodzaju czas trwania stanu niskiego na wyjściu 3 jest krótszy niż czas trwania stanu wysokiego, jak ilustruje to pochodzący z karty katalogowej **rysunek F**. Co najważniejsze w tych rozważaniach, wypełnienie przebiegu na wyjściu 3 w takim generatorze nie może być mniejsze niż 50%.

Na rysunku B mamy dodatkowo diodę, włączoną równolegle z rezystorem R3 (R_B na rysunkach E, F). Obecność tej diody znacząco zmienia sytuację. Mianowicie dioda niejako zwiera rezystor R3 (R_B) w tej fazie pracy, gdy napięcie na nóżkach 2, 6 narasta. A więc przy obecności D2 o prądzie ładowania i o czasie trwania stanu dodatniego na wyjściu 3 decyduje tylko rezystor R2 (R_B). Natomiast gdy przewodzi tranzystor rozładowujący T_D, dioda D2 jest spolaryzowana zaporowo



i podobnie jak w klasycznym układzie bez diody, o czasie rozładowania decyduje tylko R3 (R_B). Oznacza to, że na rysunku B mamy generator przebiegu prostokątnego, którego wypełnienie można dowolnie regulować, praktycznie od zera do 100%. O wypełnieniu decyduje stosunek rezystorów R2/R3.

Ponieważ na rysunku B nie ma wartości elementów, nie wiadomo, jakie miałyby być wypełnienie impulsów sterujących bramkę MOSFET-a T1. Ale biorąc pod uwagę przeznaczenie układu, w tym kwestię krótkich i silnych impulsów, można się domyślać, że albo wypełnienie będzie bliskie zero, albo bliskie 100%. W obu wypadkach nasuwają się pytania i wątpliwości, ale wykracza to poza ramy zadania *Jak to działa?*.

Byś może dlatego tym razem rozwiązań było mniej niż zwykle. Zadanie było trudne, niemniej wszystkie nadesłane odpowiedzi były poprawne.

Nagrody – cenne książki wydawnictwa WKiŁ: Grażyna Jastrzębska, *Energia ze źródeł odnawialnych i jej wykorzystanie*.

Michał Adamczyk – Wesola
Jakub Kluczewski – Bierkowo
Kazimierz Walasek – Wrocław

Wszyscy uczestnicy zostają dopisani do listy kandydatów na bezpłatne prenumeraty.



Stały konkurs: Czego tu brak?

Zadanie Brak1803

Na rysunku pokazany jest schemat pewnego prostego, popularnego układu (obwodu) elektronicznego.

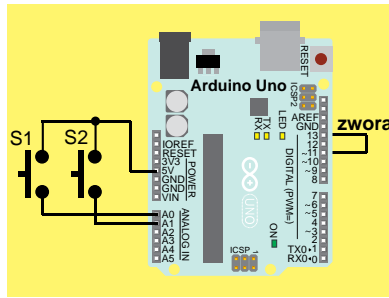
Zadaniem uczestników jest określenie, czego brakuje na schemacie.

Najmilej widziane są krótkie odpowiedzi tekstowe zawarte bezpośrednio w e-mailu (unikajcie obszernych opracowań w postaci kilku spakowanych dokumentów), ale mogą też mieć postać kompletnego schematu, narysowanego dowolnym sposobem (np. skan lub fotografia odręcznego rysunku z kartki).

E-maile z odpowiedziami należy nadsyłać w ciągu miesiąca od ukazania się numeru na adres: konkursy@elportal.pl.

W tytule e-maila należy podać nazwę konkursu, numer zadania i własne nazwisko, np. *BRAK1710Kowalski*.

Wśród autorów prawidłowych odpowiedzi rozlosowane zostaną 3 kity AVT-1731.



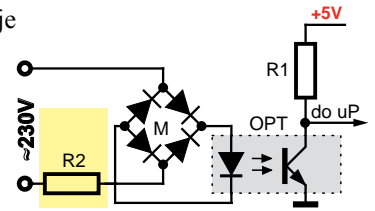
Rozwiązanie zadania Brak1712

Rysunek pokazuje czego brakowało na schemacie z EdW 12/2017.

W detektorze przejścia napięcia sieci przez zero brakowało rezystora ograniczającego prąd.

Upominki w postaci kitu AVT-2857, otrzymują:

Gabriel Pytel – Łódź,
Michał Majda – Wola Zabierzowska,
Jerzy Jurasz – Wróblewice.



W najbliższych numerach EdW planujemy

EdW 4/2018

Maglev czyli lewitacja magnetyczna

Nie jest to wcale prima aprilisowy żart! Nieskomplikowane od strony elektronicznej, a bardzo widowiskowe urządzenia maglev są dowodem, że różne obiekty mogą stabilnie lewitować w powietrzu, nie będąc ani podparte, ani zawieszane.



EdW 5/2018

Dwa barometry

W ostatnich latach upowszechniły się najróżniejsze czujniki. Między innymi precyzyjne i niedroge czujniki ciśnienia atmosferycznego. Prezentowany projekt demonstruje, że ich wskazania można zobrazować w najróżniejszy, także bardzo nietypowy i oryginalny sposób.



EdW 6/2018

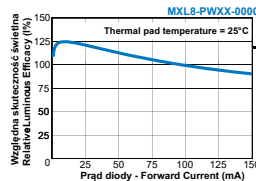
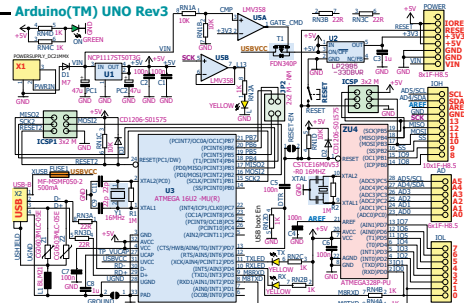
Moja własna karta audio USB

Przekonaj się, że dzięki kostce CM119 budowa komputerowej karty dźwiękowej wcale nie jest tak trudna, jak mogłoby się wydawać. Przy okazji poznaj pokrewne układy scalone C-Media, przydatne także w mniej tradycyjnych zastosowaniach.



W kolejce na publikację czekają m.in.:

Sprzętowe tajemnice Arduino
Kurs Arduino byłby niepełny bez gruntownego zapoznania się ze sprzętem.



Taśmy LED
Najczęstsze problemy
Ważne informacje dla każdego, wykorzystującego taśmy LED.

Alarm do pieca centralnego ogrzewania.
Postawiony problem można rozwiązać na wiele różnych sposobów. Ale dlaczego nie miałby to być sposób jak najbardziej energooszczędny?



Rozproszony system automatyki domowej
Projekt, który powstał na podstawie pracy dyplomowej pokazuje, jak można wyposażać mieszkanie czy dom w elementy automatyki.

**UWAGA!
Zostań
autorem krzyżówki!**

Zachęcamy do nadsyłania krzyżówek – prostych a także bardziej skomplikowanych (edw@elportal.pl).

Mogą to być krzyżówki panoramiczne, jolki, kwadraty magiczne, łamigłówki, szarady i inne. Nagrodą dla Autora będzie 6...18 miesięcy bezpłatnej prenumeraty EdW (dla dotychczasowych prenumeratorów możliwość zamiany na inne pisma AVT).

ARYTMOLAMIGŁÓWKA

Wiedząc, że takim samym liczbom odpowiadają takie same litery, należy wpisać w rzędy diagramu słowa związane z elektroniką (jedno z nich jest już wpisane), a następnie przenieść litery zgodnie z numeracją do dolnej kratownicy by odczytać rozwiązanie zadania.

1	2	3											
4	5	6	7										
8	9	10	11	12									
13	14	15	16	6	17								
18	14	2	19	20	21	6							
G	Ł	O	Ś	N	I	K							
1	11	1	22	8	17	20	17						
4	5	16	2	23	20	21	3	17					
7	13	8	10	24	25	11	20	21	11				
23	15	19	23	21	11	16	4	17	3	10			
13	2	16	11	20	3	26	2	1	11	16	8		
3	10	24	7	16	2	16	4	21	23	2	19	27	
1	21	6	8	2	6	2	20	16	8	2	4	11	8

ROZWIĄZANIE:

	23	15	9	17	26	20	2	19	27				
17	4	16	11	8	20	17	16	2	8	17			


Autorem łamigłówki jest **Wojciech Gieroń** z Lisiej Góry. **Autor w nagrodę otrzymuje 12-miesięczną prenumeratę EdW.**


UWAGA! Przesyłając rozwiązanie krzyżówki, nie zapominajcie o podaniu w e-mailu pełnych danych adresowych.

Ich brak uniemożliwia wysłanie, a więc także przyznanie Czytelnikowi nagrody/upominku.

AVT stosuje system rabatów dla wszystkich wiernych Czytelników EdW, dokonujących zakupów w sieci handlowej AVT drogą sprzedaży wysyłkowej. Naklejenie na kartonik zamówienia trzech kuponów wyciętych z trzech kolejnych najnowszych wydań EdW uprawnia do: **10% zniżki** na zakup kitów AVT, TSM, Vellemana, **10% zniżki** na książki w ramach Księgarni Wysyłkowej AVT. **Już zakup na sumę 120 zł pozwala zaoszczędzić kwotę równą cenie jednego numeru EdW.**

Uwaga! Podane zniżki dotyczą wyłącznie zamówień osób prywatnych.


**Kupon
rabatowy
EdW
3/2018**


**Kupon
rabatowy
EdW
3/2018**


**Kupon
rabatowy
EdW
3/2018**

Rozwiązanie krzyżówki z EdW 12/2017 to:

1,41 dioda.

Upominki w postaci kitów AVT otrzymują:

Leszek Mirjuk – Otwock, **Edward Ciechanowicz** – Biała Podlaska, **Bartosz Dzieciol** – Kobyłka.

Listy i e-maile z propozycjami krzyżówek oraz listy z ich rozwiązaniami powinny być opatrzone dopiskiem „Krzyżówka” lub znacznikiem # oraz numerem tego wydania EdW. Wraz z propozycją nowej krzyżówki należy przysłać oświadczenie, że krzyżówka jest oryginalnym dziełem podpisanego i że nie była nigdzie publikowana. Redakcja nie ingeruje w treść merytoryczną (precyzyjnie sformułowań) haseł krzyżówki.

Rozwiązania z tego numeru (tylko hasło) należy nadsyłać w ciągu 45 dni od ukazania się tego numeru EdW.

EdW 3/2018 – lista osób nagrodzonych:

Michał AdamczykWesoła bronekp.....	Jerzy JuraszWróblowice	Gabriel Pytel.....Łódź
Edward Ciechanowicz.... Biała Podlaska	Krzysztof Kasprzak..... Lublin	Radosław SmalecZabrze
Bartosz DzięciolKobyłka	Jakub Kluczewski Wrocław	Michał Stach..... Kamionka Mała
Mateusz Dziuba Wejherowo	Mariusz Koc..... Kalisz	Krzysztof Smoliński.....Poznań
Marian Gabrowski..... Polkowice	Roman L. Warszawa	Jakub ZagórnyKraśnik
Piotr Gajdosz Grybów	Michał Majda..... Wola Zabierzowska	Jarosław Tarnawa Zasole
Wojciech Gieroń..... Lisia Góra	Leszek Mirjuk Otwock	Daniel Turbasa..... Kraków
Sebastian Jarmosiewicz..... Motwica	Tomasz Nowacki Zambrów	Kazimierz Walasek Wrocław
Rafał Jasiński Plock	Jakub Nowak Dawidy	Maciej Zieliński Kraków
	Dawid Placha Rdzawa	

Uwaga! Jeśli do końca marca poczta nie dostarczy osobie z powyższej listy przesyłki z nagrodą, prosimy zgłosić ten fakt redakcji (22 783 00 20, ewa.dudzik@elportal.pl)

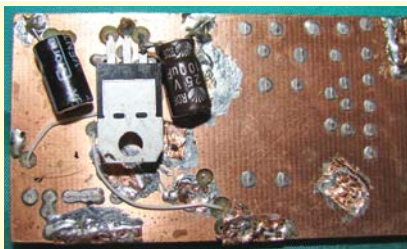
Przysyłając rozwiązanie dowolnego konkursu, NIE ZAPOMINAJCIE o podaniu w e-mailu pełnych danych adresowych. Ich brak uniemożliwia wysłanie, a więc także przyznanie Czytelnikowi nagrody/upominku.

Zajrzyj do interesujących materiałów

„Świat Radio” 3/18

Projekty do programu ARISS

W ŚR 3/18 prezentowane są kolejne projekty zrealizowane w ramach przygotowań do łączności ARISS: przedwzmacniacz oraz sequencer skonstruowane przez SP7ROH i SQ7IQA oraz konwerter do tunera satelitarnego 2,4GHz zgłoszony przez SQ4AVS w ramach projektu PUK 2017.



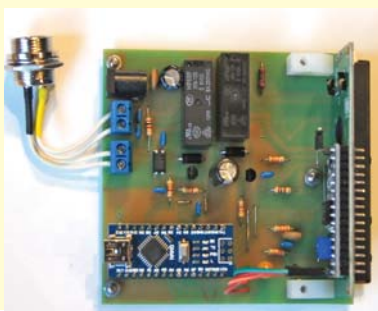
Odbiorniki nasłuchowe SSB/CW

Wśród początkujących krótkofalowców (nasłuchowców) nie słabnie zapotrzebowanie na tanie odbiorniki na pasma amatorskie HF. Szczególnym zainteresowaniem cieszą się opisy wykonania prostych układów na dostępnych podzespołach. Prezentujemy cztery modele odbiorników, jakie w ostatnim czasie zostały skonstruowane przez krótkofalowców polskich.



Sterownik rotora anteny obrotowej

Unowocześniona wersja sterownika do popularnych rotorów antenowych typu RAU i RAK to kolejny przykład zastosowania ulubionego przez hobbystów modułu Arduino. Zaprojektowany i wykonany przez SP5GNI sterownik obsługuje jeden rotor (obrót anteny w płaszczyźnie poziomej). Urządzenie jest wyposażone w dwulinijkowy wyświetlacz LCD.



R E K L A M A

Errare Humanum Est

EdW 1/2018

Str. 29, kol. 1, wiersz 8 – jest:
Rezystor R1 rozładuje..., powinno być: Rezystor R2 rozładuje...

EdW 2/2018

Str. 58, kol. 2, wiersz. 5 od dołu – jest: ...odeczone z termometru, powinno być: ...odeczone z potencjometru.

Upominek w postaci kitu AVT otrzymuje

Marian Gabrowski – Polkowice.

Materiały z EdW 11/2017, które uznaliście za najbardziej interesujące:

- Zegar/termometr matrycowy
- Girlanda LED
- Wykorzystanie modułów przetwornic

Upominki w postaci kitów AVT otrzymują:

Mariusz Koc – Kalisz,
Tomasz Nowacki – Zambrów,
Jakub Zagórny – Kraśnik.

AVT 740 Dotykowy generator dźwięku syreny

Układ jest niezwykłym generatorem, którego częstotliwość jest płynnie przestrajana w górę i w dół za pomocą sensorów dotykowych. Odpowiednie dobranie parametrów układu powoduje, że moduł szczególnie dobrze nadaje się do naśladowania syren policyjnych. Duża szybkość przestrajania oraz możliwość przełączenia zakresu częstotliwości, pozwala wytwarzać setki rozmaitych niesamowitych sekwencji dźwięków.



Znajdź nas na 



Jesteś zainteresowany zamieszczeniem ogłoszenia w rubryce Market? Skontaktuj się:

reklama@elportal.pl
tel. 22 257 84 64



OBUDOWY DO URZĄDZEŃ AUDIO
tel. +48 501 400 411, mail: info@modushop.pl

- aluminiowy panel - szeroki wybór wielkości - personalizacja -



Galaxy GX 388 159,- Galaxy GX 148 99,- Economica 39,-

www.modushop.pl





HPS140MK2 650zł

Podręczny oscyloskop kieszonkowy, następca popularnego modelu HPS140i.
Próbkowanie 40MS/s, pasmo 10MHz.
Na tle poprzednika wyróżnia się nie tylko nowoczesnym wyglądem i czytelnym wyświetlaczem ale przede wszystkim możliwością rozszerzenia funkcjonalności za pomocą opcjonalnych akcesoriów.

sklep.avt.pl handlowy@avt.pl tel.: 22 257 84 50

KEY PRODUCENT AUTOMATYKI GRZEWCZEJ
11-200 Bartoszyce ul. Bohaterów Warszawy 67 pwkey@onet.pl
tel. (89)7635050 fax (89)7635051

TANIE REGULATORY

DO KOTŁÓW WĘGLOWYCH I NA DREWNO
z wbudowanym termostatem pokojowym zapewniającym komfort i oszczędność



REGULATORY DO KOTŁÓW Z PODAJNIKIEM

REGULATORY POGODOWE

- Prosta obsługa, bogate możliwości programowania
- Możliwość dopasowania do każdego kotła i rodzaju paliwa
- Wysoka jakość
- Gwarancja 24 miesiące

www.pwkey.pl



Turbiny wiatrowe, baterie słoneczne, regulatory ładowania, wentylatory solarne, lampy solarne LED
www.gtb-solaris.pl
tel: 22 864 25 36, 606 292 727, e-mail: info@gtb-solaris.pl
ul. Przytyk 6/31, 01-962 Warszawa



OBWODY DRUKOWANE
Produkcja, Projektowanie, Montaż

Płytki jednostronne	Serie dowolne	Dokumentacja technologiczna	Montaż elektroniki
Płytki dwustronne	Prototypy	Dokumentacja konstrukcyjna	Ilości modelowe produkcyjne
Płytki na podłożu aluminium	Maksymalny wymiar płytek 1w 630 mm	Płyty czolowe FR4	Krótkie terminy
Aktywny kalkulator prototypów na stronie internetowej	Pokrycie Sn lub SnPb inne na życzenie	Trawione szablony SMD	Wykonania super ekspresowe
	Maski, opisy montażowe w różnych kolorach		

Zakład produkcyjny:
05-260 Marki ul. Duża 1
tel. 22 781 63 95 22 761 95 80
fax. 22 781 63 95 w 23
www.elmax.waw.pl
elmax@elmax.waw.pl




Obwody drukowane
OBWODY JEDNOSTRONNE I DWUSTRONNE Z METALIZACJĄ OTWORÓW
PROJEKTY I DOKUMENTACJE
OBWODÓW DRUKOWANYCH
SZABLONY SMD

KRÓTKIE TERMINY
wykonania superexpresowe
SERIE DOWOLNE - również prototypy

biuro: 02-743 Warszawa, ul. J.S.Bacha 22 tel.: 0-601-24-81-44
Zakład Produkcyjny: tel./fax: (0-22) 843-17-68, 847-48-29
05-200 Wołomin, Legionów 115 e-mail: biuro@elpinpcb.com.pl

Latarka robocza EWL5

- 2 źródła światła: LED COB 3W 120lm + LED 40lm
- łamana głowica 90°
- długość 26cm
- antypoślizgowe gumowe wykończenie
- zasilanie: 4 baterie AAA (brak w zestawie)

cena 32zł



sklep.avt.pl handlowy@avt.pl tel.: 22 257 84 50

Zestaw narzędzi 1PK-943B, niezbędny dla każdego!

Całość zapakowana w poręczne etui zapinane na zamek błyskawiczny

W zestawie:

- uchwyt z grzechotką
- szczypce proste wydłużone 160mm
- klucz nastawny 6 cali
- elastyczna przedłużka 190mm
- magnetyczny adaptor-bitów 60mm
- bity: PH, PZ, '-', imbusy, Torx, kwadrat
- klucze nasadowe calowe i metryczne
- adaptor nasadek
- wiertła: 1.5, 2.5, 3, 4, 5mm
- etui 217x165x55mm

155,50zł

sklep.avt.pl handlowy@avt.pl tel.: 22 257 84 50

Gałki z serii RN-110

Gałki na oś 6mm.

Występują w dwóch wersjach: ze znacznikiem oraz znacznikiem i dodatkowym czerwonym wskaźnikiem.

Powierzchnia otworu osi: metalowa gładka.

Mocowanie: mechanizm zaciskowy z wkrętem maskowanym zaślepką.

GAŁ RN-110F

fi16x18mm

4.40zł



GAŁ RN-110FH

fi16x19mm

4.90zł



GAŁ RN-110E

fi22x19mm

4.40zł



GAŁ RN-110EH

fi22x20mm

5zł



GAŁ RN-110D

fi19x20mm

3.80zł



GAŁ RN-110DH

fi19x21mm

4.5zł



GAŁ RN-110C

fi37x21mm

5.5zł



GAŁ RN-110CH

fi37x22mm

5.5zł



sklep.avt.pl handlowy@avt.pl tel.: 22 257 84 50

Miniaturowa stacja lutownicza



50zł
ZD-927

- wymiary stacji: 6 x 5 x 9.5cm
- moc 8W
- zakres regulacji temperatury od 100°C do 450°C
- lutownica z grotem precyzyjnym (stożek 0.5mm)

sklep.avt.pl handlowy@avt.pl tel.: 22 257 84 50

W zestawie 38 elementów:

- 13 kluczy nasadowych od 4 - 14 mm
- 18 kluczy (nasadek) trzpieniowych:
 - imbus: 3; 4; 5; 6; 8mm
 - Torx: T10; T15; T20; T25; T30; T40
 - krzyżowe: PH1; PH2; PZ1; PZ2
 - płaskie: 4mm; 5.5mm; 6.5 mm
- grzechotka 145 mm
- uchwyt prosty
- pokrętło z przetyczką 115mm
- przegub Cardana
- przedłużacz uchylny
- przedłużacz prosty 55mm
- adapter z 3/8" na 1/4"
- metalowa skrzynka 245x120x35mm

Profesjonalny zestaw kluczy trzpieniowych i nasadowych 1/4" z grzechotką i przedłużkami



SK-23801M 172,20zł

sklep.avt.pl handlowy@avt.pl tel.: 22 257 84 50

Indukcyjna stacja lutownicza WEP-900H

Ciepło grotu otrzymywane w wyniku procesu indukcyjnego, dzięki temu szybko jest osiągana zadana temperatura na grotcie, a po zakończeniu grzania grot szybko się ochładza.



- grzałka 90W
- regulacja temperatury 100°C - 480°C
- obudowa i kolba ESD SAFE
- wymiary: 155x98x135mm
- możliwość ustawienia hasła

432zł

- W zestawie:
- stacja lutownicza
 - uchwyt/podstawa na kolbę
 - gąbka do czyszczenia grotu
 - kolba lutownicza z grottem

sklep.avt.pl handlowy@avt.pl tel.: 22 257 84 50

Mini płytki stykowe



cena za 1szt
7zł

- 170 pól stykowych
- wymiary 35x47mm
- możliwość łączenia płytek
- dwustronna taśma mocująca do podłoża

SD17NW ZIEŁONA
SD17NW ŻÓŁTA
SD17NW CZERWONA
SD17NW BIAŁA
SD17NW NIEBIESKA
SD17NW CZARNA

sklep.avt.pl handlowy@avt.pl tel.: 22 257 84 50

Przecinarka do styropianu i pianek polistyrenowych

Doskonały zestaw narzędzi do wycinania dowolnych wzorów, projektów modelarskich, dekoracji, indywidualnych wypełnień piankowych do opakowań, itp.

VTFC2, cena 95zł



sklep.avt.pl handlowy@avt.pl tel.: 22 257 84 50

Miniaturowa, przenośna lutownica USB ZD20U

Możliwe sposoby zasilania:

- z portu USB w laptopie, komputerze itp
- z ładowarki sieciowej USB (adapter USB do gniazdka)
- w samochodzie: z adaptera USB do gniazda zapalniczki
- w terenie: z zasilacza awaryjnego PowerBank

**Idealna
do pracy
w terenie!**



**ZD-20U
18.50zł**

- zasilanie 5V USB (natężenie min. 1A)
- moc 8W
- czas nagrzewania grotu 15s

sklep.avt.pl handlowy@avt.pl tel.: 22 257 84 50

Podstawowy zestaw lutowniczy ZD-920B



**LUT0015
33.50zł**

sklep.avt.pl handlowy@avt.pl tel.: 22 257 84 50

Wysokiej jakości, ręczna opalarka na gorące powietrze z dwustopniową regulacją mocy i nadmuchu

Moc wyjściowa:

- poziom 1: 750W
- poziom 2: 1500W

Temperatura:

- poziom 1: 300°C
- poziom 2: 500°C

Przepływ powietrza:

- poziom 1: 280 L/min
- poziom 2: 480 L/min

Pro'sKit

W zestawie 4 dysze i skrobak.
Całość zapakowana w poręczną walizkę

**SS-611B
122zł**



sklep.avt.pl handlowy@avt.pl tel.: 22 257 84 50

Zasilacze 12V stałoprądowe do LED



ADM1212 / **1A** / 14,10zł



ADM2412 / **2A** / 19,30zł



ADM3612 / **3A** / 23,90zł



ADM3612 / **5A** / 28,20zł

sklep.avt.pl handlowy@avt.pl tel.: 22 257 84 50



KS-170405

Mikrokontrolery AVR i ARM. Sterowanie wyświetlaczami LCD

Jeśli dostrzeżesz konieczność opracowywania lepszych interfejsów graficznych dla swoich projektów, chcesz pełnym garściami korzystać z możliwości oferowanych przez nowoczesne mikrokontrolery oraz wyświetlacze lub po prostu interesuje Cię ten temat i pragniesz poszerzyć swoje umiejętności projektowania atrakcyjnych i użytecznych rozwiązań, sięgnij po tę książkę. Wprowadzi Cię ona w świat kontrolerów LCD i nauczy techniki programistycznych, dzięki którym będziesz w stanie zaprojektować, opracować i zoptymalizować GUI w taki sposób, aby chciało się go używać i aby było to czystą przyjemnością!

Tomasz Francuz, stron 496, cena 89 zł

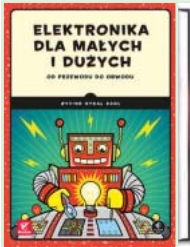


KS-160500

Elektronika. Od praktyki do teorii. Wydanie II

Książka „Elektronika. Od praktyki do teorii” to nietuzinkowy podręcznik podstaw elektroniki. Nauka następuje głównie dzięki eksperymentom, a podstawy teoretyczne są przedstawiane nieco później. Świat elektroniki staje się fantastycznym polem do własnych badań, prób i odkryć. Książka ma tę samodzielną drogę Czytelnikowi wskazać i ułatwić. Zawiera przy tym mnóstwo cennych wskazówek, począwszy od listy niezbędnych przyrządów, na instrukcjach rozwiązywania problemów skończywszy. Autor zachęca do prawdziwego eksperymentowania, w którym ewentualne pomyki będą istotnym elementem procesu nauki!

Charles Platt, stron 392, cena 69 zł

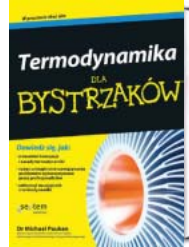


KS-170500

Elektronika dla małych i dużych

Większość dzieciaków interesuje się fascynującym światem urządzeń elektrycznych. Obserwują, jak działa żarówka, zdalnie sterowany samochód czy odtwarzacz muzyki. Kiedy mały odkrywca postanawia sprawdzić, co one kryją w środku, często ofiarą tej nieopohamowanej ciekawości pada jakiś gadżet. Warto pokazać najmłodszym, że elektronika najwięcej radości (i wiedzy!) daje wtedy, gdy zbuduje się coś własnymi rękoma! W książce przedstawiono spory zbiór praktycznych projektów do samodzielnego wykonania, które uzupełniono wyjaśnieniami zagadnień teoretycznych.

Oyvind Nydal Dahl, stron 292, cena 39 zł



KS-170903

Termodynamika dla bystrzaków

Dzięki temu praktycznemu przewodnikowi zabłyśniesz na zajęciach z termodynamiki za sprawą łatwych do zrozumienia i opisanych przystępnym językiem wyjaśnień sposobów wykorzystywania energii w takich urządzeniach, jak samochody, samoloty, klimatyzatory i elektrorownie.

W książce znajdziesz: zasady termodynamiki, istotne parametry i ich relacje, informację o substancjach stałych, cieczach i gazach, zasady współdziałania pracy i ciepła, cykle zasilające procesy termodynamiczne, związki i reakcje chemiczne, pionierów termodynamiki, rzeczywiste zastosowania zasad i koncepcji termodynamicznych

Michael Pauken, stron 352, cena 49 zł

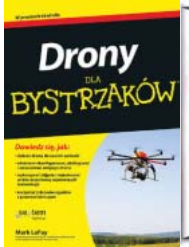


KS-171212

Zrób to sam. Generowanie ruchu, światła i dźwięku za pomocą Arduino i Raspberry Pi

Książka jest przewodnikiem po elektronice dla początkujących. Szczegółowe i praktyczne instrukcje przeprowadzą Cię krok po kroku przez wiele projektów, dzięki czemu mimochodem przyswoisz sobie podstawy elektroniki i równocześnie odkryjesz potencjał płytek Arduino i Raspberry Pi. Szybko nauczysz się sterować pracą diod LED, różnych silników, cewek, urządzeń zasilanych prądem przemiennym, grzejników, urządzeń chłodniczych, wyświetlaczy i generatorów dźwięku. Dowiesz się, jak możesz monitorować pracę tych urządzeń i kierować nimi poprzez internet.

Simon Monk, stron 312, cena 49 zł



KS-161102

Drony dla bystrzaków

Dowiedz się, jak: dobierać drony do swoich potrzeb; właściwie konfigurować, obsługiwać i serwisować swojego drona; wykonywać zdjęcia i rejestrować materiały wideo za pomocą najnowszych technologii a także jak korzystać z dronów zgodnie z prawem lotniczym. Ten zabawny i przystępny poradnik zawiera całą wiedzę niezbędną do rozpoczęcia przygody z dronami. Znajdziesz w nim profesjonalne porady dotyczące zakupu drona dostosowanego do swoich potrzeb, sposobów korzystania z bezałogowych statków powietrznych i zachowywania się zgodnie z etykietą przyjętą przez społeczność użytkowników dronów, a także wielu innych spraw

Mark LaFay, stron 256, cena 40 zł

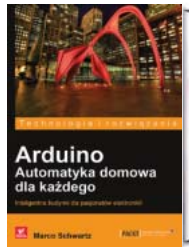


KS-160100

Zabawa w programowanie. Język C dla nastolatków

W książce znajdziesz wszystko, czego potrzebuje początkujący programista — od wskazówek w kwestii instalacji odpowiednich programów, przez omówienie niezbędnych komend, stałych i zmiennych, aż do tablic i funkcji. Dowiesz się, jak działają pętle i operatory, do czego służą struktury i unie, jak działają kompilator, preprocesor i linker. Wreszcie zrozumiesz, o co chodzi z operacjami wejścia i wyjścia, a także nauczysz się zapewnian bezawaryjne działanie Twojego programu. I nagle okaże się, że odkrywasz całkiem nowy, fascynujący świat, a porozumienie z Twoim komputerem jest w gruncie rzeczy dziecinnie proste! Sprawdź to!

Michał Wiszniewski, stron 104, cena 25 zł

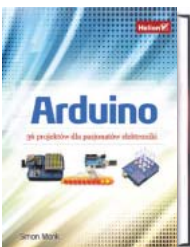


KS-150802

Arduino. Automatyka domowa dla każdego

Z tej książki nauczysz się między innymi: łączyć czujniki z płytką Arduino; budować energooszczędne czujniki ruchu z użyciem technologii XBee; sprzągać przełączniki z płytką Arduino w celu sterowania urządzeniami elektrycznymi; korzystać z łączności Wi-Fi do sterowania oświetleniem domowym; wysyłać dane o zużyciu energii elektrycznej do chmury obliczeniowej. Jeśli chcesz konstruować własne systemy automatyki domowej przy użyciu platformy Arduino, ta książka jest dla Ciebie. Aby zrozumieć opisane w niej projekty, musisz już mieć pewną wiedzę na temat Arduino i znać zasady programowania w takich językach, jak C i C++.

Marco Schwartz, stron 112, cena 33 zł



KS-151002

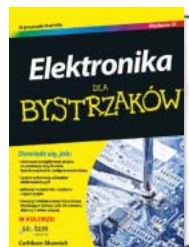
Arduino. 36 projektów dla pasjonatów elektroniki

Arduino to płytka, która zmieniła świat elektroniki. Dzięki niej ten magiczny świat stał się otworem przed wszystkimi chętnymi. Jeżeli marzysz o zbudowaniu własnego układu elektronicznego, realizującego ciekawe zadania, trafiles na doskonałą książkę. Znajdziesz w niej szczegółowe omówienie 36 niezwykłych projektów!

Dzięki tej książce: przygotujesz swoje środowisko pracy; zbudujesz własny laserowy alarm; opracujesz licznik Geigera.

Spełnij swoje marzenia o własnym układzie elektronicznym!

Simon Monk, stron 376, cena 67 zł

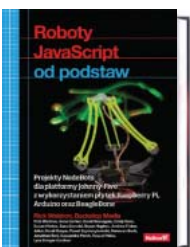


KS-161103

Elektronika dla bystrzaków. Wydanie III

Ta bogato ilustrowana kolorowymi schematami i zdjęciami książka zawiera szczegółowe instrukcje, jak przeprowadzać eksperymenty objaśniające zasadę działania różnych elementów elektronicznych, porady na temat sposobu posługiwania się podstawowymi narzędziami oraz ciekawe projekty, które można wykonać w pół godziny. Rozdział po rozdziale będziesz czuć przypływ energii, dzięki której przetworzysz teorię w praktykę!

Cathleen Shamieh, stron 368, cena 49 zł

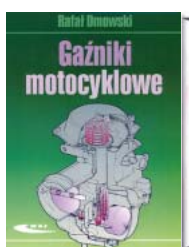


KS-160400

Roboty JavaScript od podstaw. Projekty NodeBots dla platformy Johnny-Five z wykorzystaniem płytek Raspberry Pi, Arduino oraz BeagleBone

Powszechnie JavaScript uważa się za doskonałe narzędzie do programowania aplikacji internetowych. To jednak nie wszystko: ten język świetnie nadaje się również do programowania robotów. Umożliwia zaprojektowanie zachowania robota, określenie, w jaki sposób będzie on zmierzał do celu, reagował na sygnały z otoczenia, oczekiwał na instrukcje czy omijał przeszkody. Bardzo wygodnym narzędziem do programowania robotów jest platforma Johnny-Five. Umożliwia ona programowanie robotów zbudowanych z płytek Arduino, Raspberry Pi i BeagleBone.

Rick Waldron, stron 256, cena 59 zł



KS-170904

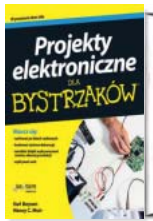
Gazniki motocyklowe

W książce opisano budowę, obsługę i użytkowanie gazników układów doświetlenia w współpracujących i klasycznych motocyklach. Podano informacje o gaznikach motocyklowych z uwzględnieniem ich budowy, opisu działania poszczególnych elementów i układów oraz czynności obsługowo-naprawczych możliwych do wykonania przez indywidualnego użytkownika motocykla. Zamieszczono liczne fotografie i rysunki, ułatwiające zrozumienie zasad działania zespołów i układów, a także regulację, demontaż, naprawę i montaż.

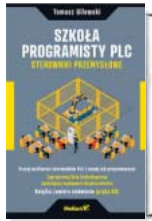
Rafał Dmowski, stron 104, cena 45 zł



kod KS-150302
Drony
Wprowadzenie,
Ty Audronisi
stron 122, cena 33 zł

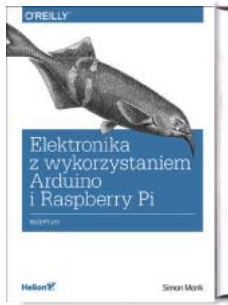


kod KS-150500
Projekty elektroniczne
dla bystrzaków
Earl Boysen, Nancy C. Muir
stron 376, cena 50 zł



kod KS-170201
Szkoła programisty
PLC. Sterowniki
Przemysłowe,
Tomasz Gilewski
stron 264, cena 59 zł

Polecamy



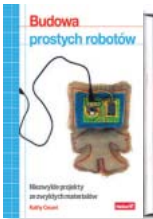
KS-180100

Simon Monk
stron 424
cena 77zł

Elektronika z wykorzystaniem Arduino i Rapsberry Pi. Receptury

Praktyczne przepisy rozwiązań elektronicznych z wykorzystaniem Arduino i Rasperry Pi

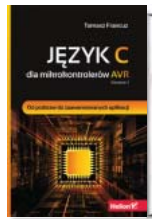
Niniejsza książka jest solidnym punktem wyjścia dla elektroników amatorów i osób pragnących zająć się tą dziedziną profesjonalnie. Szczególnie ważnym elementem książki jest omówienie podstaw elektroniki oraz zagadnień stosowania mikrokontrolerów, tworzenia zasilaczy różnego rodzaju, używania odpowiednich tranzystorów do różnych celów, stosowania analogowych i cyfrowych układów scalonych, a także budowania projektów i prototypów oraz posługiwania się sprzętem badawczym. Książkę zorganizowano w ten sposób, aby ułatwić szybkie odnalezienie potrzebnego zagadnienia i skorzystanie z odpowiedniego przepisu.



kod KS-160000
Budowa prostych
robotów,
Kathy Ceceri
stron 208, cena 40 zł



kod KS-170402
Podstawy konstrukcji
maszyn,
Praca zbiorowa
stron 612, cena 73 zł



kod KS-150100
Język C dla
mikrokontrolerów AVR,
Tomasz Francuz
stron 712, cena 99 zł

Ostatnio dodane pozycje w ofercie AVT

KS-160300	Półprzewodniki. Nowoczesne rozwiązania w układach scalonych	89,00
KS-160400	Roboty JavaScript od podstaw. Projekty...	59,00
KS-160401	Instalacje elektryczne	49,00
KS-160500	Elektronika. Od praktyki do teorii. Wydanie II	69,00
KS-160501	Nauka robotyki z językiem Python	49,00
KS-160700	Domowe laboratorium naukowe. Zrób to sam	44,00
KS-160701	Budowa robotów dla początkujących. Wydanie III	69,00
KS-161100	Apokalipsa zombie nadchodzi! Obróń swoją bazę za pomocą prostych obwodów, Arduino i Rasperry Pi	49,00
KS-161101	Proste projekty dla młodych majsterkowiczów	30,00
KS-161102	Drony dla bystrzaków	40,00
KS-161103	Elektronika dla bystrzaków. Wydanie III	49,00
KS-170004	Łączenie tworzyw sztucznych metodami spawania, zgrzewania, klejenia i laminowania	41,00
KS-170005	Obsługa i programowanie obrabiarek CNC Podręcznik operatora	64,00
KS-170006	Praktyczne podstawy eksploatacji obrabiarek CNC z wykorzystaniem komputerowego systemu wszkoleniowego MTS	54,00
KS-170007	Nawigacja satelitarna w praktyce	57,00
KS-170008	Automatyka przemysłowa w praktyce	58,00
KS-170012	Drony-teoria i praktyka	39,00
KS-170200	English 4 IT. Praktyczny kurs języka angielskiego dla specjalistów IT i nie tylko	39,00
KS-170201	Szkoła programisty PLC. Sterowniki Przemysłowe	59,00
KS-170400	Montaż i konserwacja instalacji elektrycznych	75,00
KS-170401	Naprawa podzespołów i zespołów pojazdów samochodowych	75,00
KS-170402	Podstawy konstrukcji maszyn	73,00
KS-170403	Podwozia i nadwozia pojazdów samochodowych. Podstawy budowy, diagnozowania i naprawy	79,00
KS-170404	Silniki pojazdów samochodowych. Podstawy budowy, diagnozowania i naprawy	75,00
KS-170405	Mikrokontrolery AVR i ARM. Sterowanie wyświetlaczami LCD	89,00
KS-170500	Elektronika dla małych i dużych. Od przewodu do obwodu	39,00
KS-170600	Internet rzeczy. Budowa sieci z wykorzystaniem technologii webowych i Rasperry Pi	67,00
KS-170601	Instalacje elektryczne do zasilania urządzeń elektrycznych, których funkcjonowanie jest niezbędne w czasie pożaru. Zagadnienia wybrane Vademecum elektro.info	45,00
KS-170900	LEGO. Księga przygód. Wydanie II. Kosmiczne podróże, piraci, smoki i jeszcze więcej!	39,00
KS-170901	Drony. Tajniki zdjęć i filmów lotniczych	59,00
KS-170902	Lego Mindstorms EV3. Programowanie robotów	69,00
KS-170903	Termodynamika dla bystrzaków	49,00
KS-171212	Zrób to sam. Generowanie ruchu, światła i dźwięku za pomocą Arduino i Rasperry Pi	49,00
KS-180100	Elektronika z wykorzystaniem Arduino i Rapsberry Pi. Receptury	77,00

Książki można zamówić w sklepie internetowym AVT lub wypełnić zamówienie (formularz na stronie 81) i wysłać na adres:
AVT – Księgarnia Wysyłkowa, ul. Leszczyńska 11, 03-197 Warszawa
tel. (+48) 22 257 84 50 do 53, faks (+48) 22 257 84 55
e-mail: handlowy@avt.pl



kod KS-170200
English 4 IT. Praktyczny kurs języka angielskiego dla specjalistów IT i nie tylko,
stron 288, cena 39 zł



kod KS-161101
Proste projekty dla młodych majsterkowiczów,
Autorzy: Redaktorzy magazynu Make
stron 150, cena 30 zł



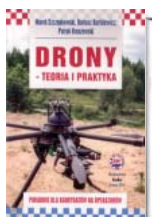
kod KS-140600
Szalony Geniusz,
Brad Graham,
Kathy McGowan
stron 384, cena 59 zł



kod KS-150800
Fascynujący świat robotów
John Baichtal
stron 384, cena 67 zł



kod KS-141001
Programowanie układów AVR dla praktyków
Elliot Williams
stron 432, cena 89 zł



kod KS-170012
Drony - teoria i praktyka,
praca zbiorowa
stron 165, cena 39 zł



kod KS-160700
Domowe laboratorium naukowe. Zrób to sam,
Windle Oskay, Raymond Barrett
stron 344, cena 44 zł



kod KS-110207
Egzamin kwalifikacyjny elektryka w pytaniach i odpowiedziach,
Władysław Orlik
stron 456, cena 78 zł



kod KS-140888
Świat druku 3D. Przewodnik,
Anna Kazunas France
stron 224, cena 49 zł

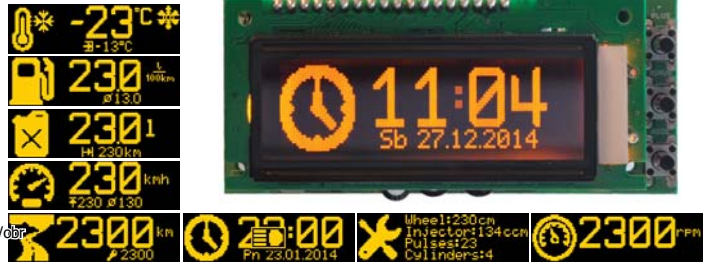
AVT 5495 Uniwersalny komputer samochodowy

Moduł uniwersalnego komputera samochodowego przeznaczony dla osób, które chcą podwyższyć funkcjonalność swojego samochodu.

- dostępne funkcje i zakresy pomiarowe:
 - temperatura zewnętrzna i wewnętrzna: -55°C...+99°C
 - prędkość pojazdu: 0...255 km/godz
 - prędkość obrotowa silnika: 0...9999 obr./min
 - chwilowe zużycie paliwa: 0...99,9 l/100 km
 - średnie zużycie paliwa: 0...25,5 l/100 km
 - paliwo dostępne w baku: 0...99,9 l
 - przejechany dystans: 0...9999 km
 - dostępny dystans: 0...999 km
 - liczba zapłonów: 0...9999
- napięcie zasilające: 8...15 VDC
- wymiary płytki: 89x39 mm



- zakresy regulacji parametrów konfiguracyjnych:
 - stała wtryskiwacza: 1...999 ml/min
 - stała przetwornika drogi: 1...99 imp./obr
 - obwód opony: 1...255cm
 - liczba cylindrów: 2...8



AVT 990 Automatyczny włącznik świateł

Układ służy do samoczynnego włączania świateł chwilę po uruchomieniu silnika oraz ich natychmiastowego wyłączenia po zgaszeniu silnika.



Wybrane parametry:

- automatyczne włączanie oświetlenia pojazdu po uruchomieniu silnika
- automatyczne wyłączenie świateł po wyłączeniu zapłonu
- dwa niezależne przełączniki sterujące
- obciążalność styków każdego przełącznika 10 A
- zabezpieczenie przed odwrotnym podłączeniem zasilania
- prosty montaż, niewielka liczba elementów
- wymiary płytki: 60x40 mm



AVT 1850 Zmierzchowy przełącznik świateł

Układ wzbogaca samochód o funkcję automatycznego przełączania świateł dziennych na światła mijania. Moduł umożliwia programowanie przez użytkownika ustawień dostosowanych do własnych potrzeb czyniąc układ niezwykle funkcjonalnym.



Wybrane parametry:

- w pełni konfigurowalny sposób zasilania oraz działania układu
- możliwość podtrzymania świateł tzw. "follow me home"
- proste programowanie przez użytkownika indywidualnych ustawień
- zasilanie z akumulatora lub „po stacyjce"
- pobór prądu w czasie czuwania: 5 mA
- odporność na oświetlenie przez reflektory innego samochodu
- przeznaczony do instalacji ze światłami do jazdy dziennej
- wymiary płytki: 57x42 mm



AVT 1877 Automatyczny włącznik zasilania do instalacji samochodowej

Urządzenie steruje zasilaniem w sposób najbardziej optymalny, wykrywa moment rozruchu oraz wyłączenia silnika odpowiednio sterując wyjściem przełącznika. Parametry czasu i napięcia w prosty sposób ustawiane są przez użytkownika.



Wybrane parametry:

- ustawiane napięcie załączenia oraz wyłączenia
- ustawiany czas opóźnienia załączenia oraz wyłączenia
- prosty sposób ustawiania (nie wymaga multimetru)
- element wykonawczy przełącznik (lub opcjonalnie tranzystor mosfet)
- max prąd wyjściowy 20 A (8 A wersja tranzystorowa)
- napięcie zasilania 10...15 VDC (7...20 VDC wersja tranzystorowa)
- pobór prądu: ok. 1mA w stanie wyłączenia, w stanie aktywnym 80mA
- wymiary 57x42 mm, dedykowana obudowa Z68.



AVT 1599 Softstart do żarówek samochodowych

Urządzenie, które w momencie włączania oświetlenia dołącza do żarówek dodatkową, szeregową rezystancję. Ogranicza to prąd włókna do bezpiecznej wartości. Dopiero po upływie pewnego czasu, podczas którego żarnik jest wstępnie rozgrzany, następuje jego pełne zasilanie.

Wybrane parametry:

- opóźnione, pełne zasilanie żarówek samochodowych
- przeznaczony do samochodów osobowych
- prąd wstępnie rozgrzewający żarnik ograniczony do 5 A
- czas rozgrzewania (opóźnienia pełnego zasilania) około 5 sekund
- możliwość zastosowania jednego lub dwóch softstartów w samochodzie
- wymiary płytki: 57x42 mm



AVT 3095 Komputer samochodowy z wyświetlaczem LCD

Moduł służy do monitorowania stanu pojazdu. Podstawowym jego zastosowaniem jest podawanie parametrów związanych ze spalaniem paliwa, przebyтым dystansem czy temperaturą wewnętrzną i zewnętrzną oraz datą i godziną. Dzięki możliwości wyboru kolorystyki wyświetlacza LCD łatwo i estetycznie można zintegrować moduł z deską rozdzielczą pojazdu.

Wybrane parametry:

- pomiar dwóch temperatur (zewnętrzna i wewnętrzna) z rozdzielczością 0,1°C
- obsługa całości 2 przyciskami lub opcjonalnie enkodernem
- zegar, data (miesiąc wyświetlany słownie lub cyfrowo),
- pomiar spalania benzyny lub gazu LPG; chwilowe, średnie, w trasie,
- prędkość chwilowa, średnia i maksymalna,
- pomiar przyspieszeń (do wartości ustawionej z zakresu 60...200, oraz 1 mili czyli 413m)
- licznik kilometrów, licznik do okresowego przeglądu (tzw inspekcja),
- ostrzeżenie o niewłączonych światłach oraz o gołodziezi,
- wymiary płytki: 102x47 mm



AVT 3095/1
Wyświetlacz BlackLine White (białe znaki)



AVT 3095/2
Wyświetlacz BlackLine Green (zielone znaki)



AVT 3095/3
Wyświetlacz BlackLine Amber (bursztynowe znaki)



AVT 3095/4
Wyświetlacz BlackLine Blue (niebieskie znaki)

AVT 528 Inteligentny sterownik oświetlenia wnętrza

Sterownik lampki oświetlenia kabiny pojazdu powoduje przedłużenie czasu jej świecenia po zamknięciu drzwi. Pozwala to na wygodne włożenie kluczyka do stacyjki i zapięcie pasów. Lampka świeci się ok. pół minuty a następnie zostaje stopniowo, płynnie wygaszona. Jeżeli jednak w międzyczasie zostanie włączony zapłon, oświetlenie wnętrza wyłączone jest natychmiast.



Wybrane parametry:

- czas świecenia (do wygaszania) po zamknięciu drzwi - 30 sek.
- czas świecenia dla nie zamkniętych drzwi - 150 sek.
- czas świecenia (do wygaszania) po wyłączeniu zapłonu - 30 sek.
- płynne załączanie i wyłączanie oświetlenia
- dioda kontrolna LED
- wymiary płytki: 64x42 mm



AVT 1484 Wskaźnik temperatury silnika

Dzisiejsze pojazdy w zdecydowanej większości wyposażone są w tradycyjny, wskazówkowy wskaźnik temperatury silnika. Zastąpienie go modułem elektronicznym może sprawić wiele satysfakcji fanom „czterech kółek”.



Wybrane parametry:

- dwa alternatywne sposoby wyświetlania temperatury:
 - słupek lub punkt
- prezentacja temperatury na 10 diodach LED
- możliwość zastosowania diod LED w dowolnym kolorze
- termistorowy czujnik temperatury
- zasilanie: 12 VDC
- wymiary płytki: Ø 43 mm





sklep.avt.pl

Produkty z oferty i wyroby AVT można nabyć na kilka sposobów:

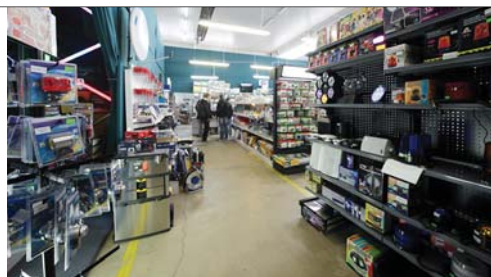
W sklepie internetowym sklep.avt.pl
Zamówienia o wartości powyżej 100zł
złożone w sklepie internetowym premiujemy bonusem!
Aktualna lista bonusów dostępna jest na stronie głównej sklepu.

W sklepie firmowym AVT:
Warszawa - Żerań
ul. Leszczynowa 11



U dystrybutorów:
Lista dystrybutorów dostępna jest na stronie sklepu:
sklep.avt.pl/webpage/dystrybutorzy.html

Wypełniając poniższy formularz zamówienia
Formularz można wyłać faksem nr: 22 257 84 55,
lub pocztą tradycyjną na adres:
AVT-Korporacja Sp. z o. o.
Dział Handlowy
03-197 Warszawa
ul. Leszczynowa 11



prześlij faksem: 22 257 84 55
lub pocztą na adres:

AVT-Korporacja Sp. z o.o.
Dział Handlowy

03-197 Warszawa, ul. Leszczynowa 11

Miejsce na
kupon
rabatowy
EdW 1/2018

Miejsce na
kupon
rabatowy
EdW 2/2018

Miejsce na
kupon
rabatowy
EdW 3/2018

Tu wklej kupony z ostatnich 3 numerów EdW
a uzyskasz **zniżkę 10%** dla stałych czytelników.
(szczegóły na stronie 73)
Prenumeratory nie muszą wklejać kuponów,
wystarczy, że podadzą nr prenumeraty!

ZAMÓWIENIE na artykuły z oferty AVT

Kity

Oznaczenia wersji kitów:

- [A] płytką drukowaną PCB
- [UK] zaprogramowany układ
- [A+] płytką PCB i zaprogramowany układ
- [B] płytką PCB (lub płytki), UK (jeśli występuje) i komplet elementów elektronicznych wymienionych w dokumentacji zestawu.
- [C] zestaw zmontowany

Numer kitu AVT	A	A+	B	C	UK

Inne artykuły z oferty AVT

Kod – Nazwa	Ilość

Nadawca:

imię i nazwisko

mój numer prenumeraty

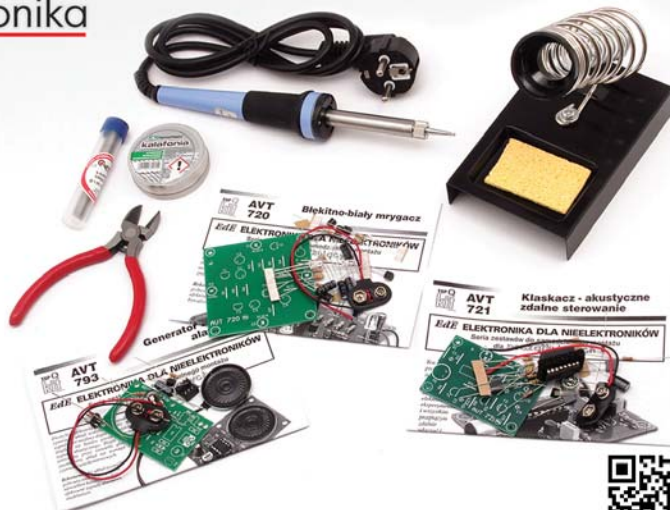
Adres:

wysyłka pobraniowa kurierem: 19zł

AVT PSE Pakiet Startowy Elektronika

Elementy pakietu:

- Wysokiej jakości lutownica elektryczna kolbowa 25W/230V
- Podstawka pod lutownicę z gąbką czyszczącą
- Kalafonia aktywna ułatwiająca lutowanie
- Cyna 1mm we fiolce
- Szczypce tnące boczne ze stali narzędziowej nierdzewnej
- 3 zestawy do samodzielnego montażu:
 - AVT720 Błękitno-biały mrygacz
 - AVT721 Klaskacz - akustyczne zdalne sterowanie
 - AVT793 Generator dźwięków alarmowych
- Pudełko zbiorcze



kod handlowy: AVTPSE
80 zł



EDW A09 KPL Elektroniczny zestaw edukacyjny dla początkujących

Zapraszamy wszystkich do fascynującej przygody, jaką jest zgłębianie tajników elektroniki! Dla tych, którzy chcieliby rozpocząć kurs, przygotowaliśmy kompletny zestaw, w praktycznym pudełku.

Komplet zawiera:

- Pudełko narzędziowe
- Zestaw EdW09, zawierający płytę stykową oraz wszystkie elementy, jakie będą potrzebne do wykonania kilkunastu projektów z PKE.
- AVT3072 C - Zasilacz płytek stykowych
- Zasilacz impulsowy 12V/1.4A
- Płyta CD z kompletem materiałów (wszystkie części kursu)



kod handlowy:
EDWA09KPL
80 zł



Prenumerujesz więcej niż jedno czasopismo wydawnictwa AVT?

Witaj w Klubie AVT!

Możesz co miesiąc otrzymywać **darmowe numery archiwalne** według formuły **n-1** (gdzie **n** to liczba opłaconych prenumerat).

Szczegóły na www.avt.pl/klub

Jeszcze nie prenumerujesz? Złóż zamówienie na prenumeratę:

- www.avt.pl/prenumerata
- prenumerata@avt.pl

AVT 2728 Wzmacniacz mikrofonowy

Uniwersalny wzmacniacz mikrofonowy może bezpośrednio współpracować z popularnymi mikrofonami elektretowymi. Dzięki zastosowaniu elementów o bardzo dobrych parametrach moduł nadaje się również do współpracy z dobrej jakości mikrofonami dynamicznymi.



Wybrane parametry:

- zniekształcenia nieliniowe: <0,09% (przy wzmacnieniu maksymalnym)
- pasmo przenoszenia: >25 kHz
- możliwość regulowania wzmacnienia w zakresie 0,9...100x
- regulacja wzmacnienia skokowa i płynna
- obwód zasilania mikrofonu elektretowego
- napięcie zasilania: 7...24 VDC
- wymiary płytki: 30x45 mm



AVT 1721 Miniaturowy wzmacniacz mikrofonowy

Uniwersalny wzmacniacz mikrofonowy przeznaczony do współpracy z popularnymi, 2-końcówkowymi mikrofonami elektretowymi. Zastosowanie elementów SMD powoduje, że układ zamontowany na miniaturowych wymiarach płytce może zostać wbudowany w praktycznie każde urządzenie audio lub stać się częścią dowolnej nowej konstrukcji.



Wybrane parametry:

- wzmacnienie regulowane w zakresie 1x...80x
- obwód zasilania mikrofonu elektretowego
- płynna regulacja wzmacnienia
- napięcie zasilania: 6...16 VDC
- pobór prądu: 5 mA
- wymiary płytki: 24x10 mm



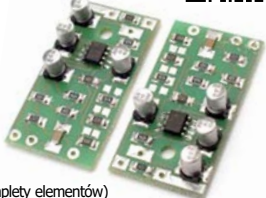
AVT 2392 Wzmacniacz mikrofonowy SMD

Jest to uniwersalny przedwzmacniacz wszelkich sygnałów zmiennych o częstotliwościach 20Hz...100kHz. Przede wszystkim będzie stosowany jako przedwzmacniacz mikrofonowy. Układ może współpracować z mikrofonami dynamicznymi, elektretowymi (pojemnościowymi) 2 i 3-końcówkowymi



Wybrane parametry:

- niskie szumy
- wzmacnienie regulowane w zakresie 1x...80x
- rezystancja wejściowa: $\geq 1 \text{ M}\Omega$
- rezystancja wyjściowa: $\leq 1 \Omega$
- zasilanie pojedynczym napięciem 6...15 VDC
- pobór prądu poniżej 5 mA
- wymiary płytki: 37x21 mm x 2 szt. (w zestawie dwa komplety elementów)



AVT 1033 Przedwzmacniacz mikrofonowy

Prosty, stereofoniczny przedwzmacniacz mikrofonowy o bardzo dobrych parametrach akustycznych, przeznaczony do współpracy z mikrofonami dynamicznymi. Zastosowane potencjometry w torze wejściowym pozwalają na ustawienie poziomu napięcia wejściowego z mikrofonu, a tym samym zmniejszenie ryzyka przesterowania. Szeroki zakres napięć zasilających pozwala na równie szeroki zakres zastosowań.



Wybrane parametry:

- pasmo przenoszenia: 70 Hz...12 kHz
- zawartość harmonicznych poniżej 0,1%
- napięcie zasilania: 9...18 VDC
- pobór prądu: 15...25 mA
- wymiary płytki: 60x60 mm



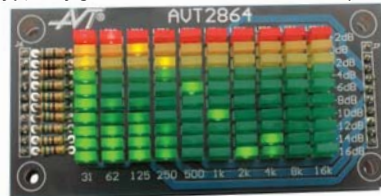
AVT 2864 Analizator widma

Układ służący do analizowania i wyświetlania widma sygnału audio, bardzo często stosowany i spotykany w torach audio sprzętu audiofilskiego. W zestawie zastosowano, oprócz części analogowej, cyfrową obróbkę sygnału, czyli DSP. Pracą analizatora steruje mikrokontroler, a zawarty w jego strukturze przetwornik A/C służy odpowiedniej zamianie sygnału analogowego na cyfrowy a ten kierowany jest do wyświetlacza zrealizowanego z szeregow diod LED. Filtry, dzięki którym zapalają się odpowiednie 'słupki' zrealizowano na drodze programowej. Uprościło to znacznie konstrukcję analizatora sprawiając, że o jego zbudowanie i uruchomienie może pokusić się nawet średnio zaawansowany elektronik.



Wybrane parametry:

- monofoniczny analizator widma - 1 kanał
- widmo sygnału przedstawione na 10-słupkowym wyświetlaczu złożonym z wielokolorowych diod LED (100 szt.)
- częstotliwości środkowe filtrów: 31 Hz, 62 Hz, 125 Hz, 250 Hz, 500 Hz, 1 kHz, 2 kHz, 4 kHz, 8 kHz i 16 kHz
- amplitudy sygnału powodujące zaświecenie kolejnych diod w słupku: od -16 dB do +2 dB z krokiem 2 dB
- praca w trybie wyświetlania słupków lub punktów
- tryb pokazywania maksymalnej wartości amplitudy
- zasilanie: 12 VDC
- wymiary płytek: płytka LED 96x45 mm, płytka główna 96x45 mm



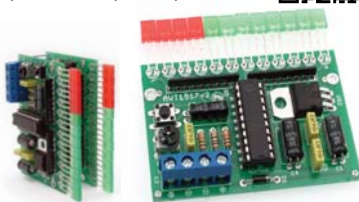
AVT 1517 Wskaźnik nie tylkoysterowania

Moduł może służyć jako wskaźnik poziomu sygnału np. audio czyniąc go wskaźnikiemysterowania. Może służyć również jako monitor napięcia stając się miernikiem napięć stałych. Płytkadrukowana została zaprojektowana w taki sposób, aby możliwe było bezproblemowe połączenie dwóch takich wskaźników.



Wybrane parametry:

- praca jako wskaźnikysterowania lub wskaźnik siły sygnału
- możliwy pomiar napięcia stałego
- 8 funkcji, wybieranych i konfigurowanych przez użytkownika
- napięcie zasilania: 5...20 VDC
- wymiary płytki: 45x63 mm



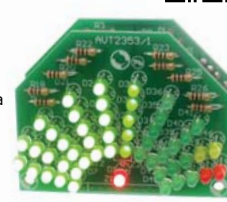
AVT 2353 Pseudoanalogowy wskaźnikysterowania

Układ jest efektywnym pseudoanalogowym wskaźnikiemysterowania, gdzie diody LED udają wychylającą się wskazówkę. Układ ten posiada dwa tryby wyświetlania poziomu sygnału: tryb wskaźnkowy oraz tryb wyświetlania w postaci rozkładającego się wachlarza.



Wybrane parametry:

- 2 tryby wyświetlania poziomu sygnału - tryb wskaźnkowy oraz tryb wyświetlania w postaci rozkładającego się wachlarza
- pomiar wartości szczytowej sygnału wejściowego
- stała czasowa narastania wynosi ok. 5 ms
- stała czasowa opadania ok. 250 ms
- napięcie zasilania: 12 VDC
- wymiary płytki: 60x46 mm - 2szt.



AVT 2375 Wskaźnikysterowania 2x5 LED

Układ prostego, stereofonicznego wskaźnikaysterowania o charakterystyce logarytmicznej. Układ zbudowany został w oparciu o układ scalony KA2281. Każdy kanał zawiera 5 diod w trzech kolorach. Wskaźnik idealnie sprawdzi się jako moduł do wbudowania w posiadany lub nowo konstruowany sprzęt audio.



Wybrane parametry:

- 3 kolory diod LED
- 2x5-punktowe pole odczytowe
- wskazania dla -13, -8, -3, -0, +3 dB
- napięcie zasilania: 9...15 VDC
- wymiary płytki: 37x43 mm



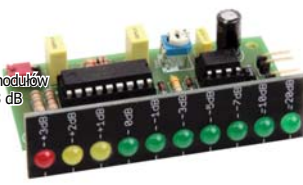
AVT 1650 10-diodowy wskaźnik poziomu sygnału audio

Moduł znajdzie zastosowanie głównie we wszelkiego rodzaju przedwzmacniaczach i mikserach. Umożliwia szybką kontrolę wzmacnienia toru wzmacniacza, co pozwala wyeliminować jego przesterowanie. Układ jest prosty w wykonaniu a modułowa konstrukcja ułatwia montaż w urządzeniach z wieloma torami sygnałowymi.



Wybrane parametry:

- 10-punktowe pole odczytowe
- układ 1-kanałowy z możliwością zestawienia wielu modułów
- wskazania dla -20, -10, -7, -5, -3, -1, 0, +1, +2 i +3 dB
- możliwość pracy w trybie liniowym lub punktowym
- płynna regulacja wzmacnienia
- zasilanie 9 VDC
- wymiary płytek: 67x32 mm i 67x16 mm (skala)



POLECAMY!

Precyzyjna wiertarka VTHD09 z wałkiem giętkim i zestawem akcesoriów



velleman® 



Zestaw 190 - elementowy!



- prędkość obrotowa regulowana: 10000-32000 obr/min.
- maksymalna średnica tarczy ścierniej: 35mm
- maksymalna średnica wiertła: 3.2mm
- pobór mocy: 135W
- zasilanie: 230 VAC
- całość zapakowana w organizer o wymiarach: 36x26x6.5cm

VTHD09 179.80zł



sklep.avt.pl

AVT-Korporacja Sp. z o.o., 03-197 Warszawa, ul. Leszczynowa 11
Dział Handlowy tel.: (22) 257 84 50 e-mail: handlowy@avt.pl

eWydanie dla: Michał Sas-Uhrynowski (msas)

Wydanie elektroniczne przeznaczone wyłącznie do użytku własnego bez prawa do rozpowszechniania.