AUDIOFILSKI POTENCJOMETR PVC

www.ep.com.pl

ELEKTRONIKA PRAKTYCZNA

CD +FTP

● Międzynarodowy magazyn elektroników konstruktorów ● CZERWIEC ● 06/2011

Niezbędnik elektronika na CD

TYLKO DLA PRENUMERATORÓW

Konkurs ● Microchip

TEMAT NUMERU

Programatory stacjonarne

Stacja pogodowa ● Moduł przekaźników sterowany przez Bluetooth ● Timer fotograficzny do lustrzanki ● Minimoduł z Atmega8 ● Symetyzator sygnału audio ● AVTduino – manipulator dla Arduino ● Rubidowy, korygowany GPS-em wzorzec częstotliwości i czasu

inspirujące użyteczne projekty

Multimetr U1252B ● VibroDAQ ● LPC177x/8x Wyposażenie i kompatybilność ● AFE: sposób na sensory ● i.MX28 – Nowa skala intergracji: ARM926 + DC/DC w jednej obudowie ● Cyfrowe rejestratory dźwięku ISD151xx

podzespoły sprzęt

MCUISP – Alternatywny Flash loader dla mikrokontrolerów STM32

tutoriale

- Projektowanie płytek za pomocą Altium Designer
- Technologia GSM w elektronice
- Kurs Arduino
- Cyfrowe zwrotnice głośnikowe
- Kurs programowania mikrokontrolerów PIC
- Wprowadzenie do Linuxa Embedded

KURSY

Wyszukaj ze 100 000 artykułów

www.elfaelektronika.pl - łatwo zamawiasz - szybko otrzymujesz

- Elektromechanika
- Automatyka
- Przewody i kable
- Elementy aktywne i pasywne
- Złącza
- Przyrządy pomiarowe
- Narzędzia i wiele innych

ELFA

EP ISSN 1230-3526 Indeks 357677

cena: 16,00 zł (w tym 8% VAT)

PRICE: 8 EUR Nakład 29000 egz.

Rewolucyjny MSP430FR

Odnoszę wrażenie, że niegdyś mikrokontrolery i mikroprocesory były bardzo do siebie podobne. Następnie wiodące firmy położyły nacisk głównie na rozwój mikroprocesorów wytwarzając je głównie z myślą o komputerach PC. W tamtym momencie rozwój mikrokontrolerów poszedł tak jakby inną drogą, a dziś oba rodzaje aplikacji z tymi układami rządzą się odmiennymi prawami. Z jednej strony mamy więc oprogramowanie dla tak zwanych „pecetów", które pochłania ogromne zasoby komputera i apetyt którego stale rośnie, a z drugiej niewielkie, energooszczędne, niewymagające aplikacje dla urządzeń embedded i to nie tylko tych przenośnych. Kolejny krok w stronę ewolucji układów z mikrokontrolerami uczyniła ostatnio firma Texas Instruments. Jest to wydarzenie na tyle ważne dla rynku elektroniki, że warto o nim poinformować w ten szczególny sposób.

W maju br. firma TI ogłosiła wprowadzenie na rynek pierwszego mikrokontrolera z wbudowaną pamięcią FRAM. Początkowo myślałem – żadna rewelacja, przecież ten rodzaj pamięci jest znany już od blisko 20 lat. A jeśli FRAM zastąpił pamięć Flash, to nowy produkt zyskał co najwyżej na szybkości programowania, być może obniżono wartość napięcia zasilającego oraz pobór energii, ale konkurencja na tym rynku trwa od lat i co rusz jakaś firma prezentuje nowy wyrób. Jednak po rozmowie z przedstawicielem firmy zrozumiałem błąd swojego rozumowania.

Firma TI nie tylko zastąpiła pamięcią FRAM tradycyjnie stosowaną pamięć FLASH, ale również **wszystkie rodzaje pamięci mikrokontrolera**. A więc i pamięć operacyjną, i EEPROM przechowujący dane. Programista na etapie tworzenia programu może sobie dowolnie podzielić pamięć na partycje i używać tak, jak jest mu wygodnie! Ze swojej praktyki konstruktora pamiętam, że podstawowymi kryteriami doboru mikrokontrolera do aplikacji były obok wymagań odnośnie do szybkości, pojemność pamięci RAM i pamięci FLASH. Wiele razy musiałem zmieniać model mikrokontrolera a nierzadko i producenta tylko dlatego, że np. zabrakło pamięci na stos lub zmienne. Nawet jeśli pozostawał wolny pewien obszar FLASH, to nie mogłem przeznaczyć go na zmienne dynamiczne. A teraz, dzięki zastosowaniu pamięci FRAM stało się to możliwe.

Aktualnie firma TI wprowadziła na rynek „sondażową" wersję mikrokontrolerów MSP430FR57x. Literki FR oznaczają, że jest w nich użyta pamięć FRAM. Nowe układy są wyposażone w stosunkowo niewielką pamięć, bo tylko 16 kB, ale wbrew pozorom to całkiem sporo jak dla wielu aplikacji embedded typu płytki sterujące do sprzętu gospodarstwa domowego, układy do akwizycji danych i wielu innych, których używamy na co dzień. Podczas prezentacji nowego mikrokontrolera oczywiście ktoś poddał w wątpliwość zasadność produkcji układu z tak niewielką pamięcią, ale przedstawiciel TI stwierdził, że na dziś dzień firma jest w stanie wytwarzać układy w konkurencyjnych cenach z pamięciami o wielkości do 256 kB, a w przyszłości (ponieważ prace rozwojowe nadal trwają) większą.

Oprócz możliwości partycjonowania, nowy MSP430FR57x pobiera 250 razy mniej energii, a czas zapisu do danych jest rzędu nanosekund. Firma gwarantuje liczbę zapisów przekraczającą 10^{14}, podczas gdy dla pamięci FLASH wynosi ona 10^4, a dla EEPROM 10^5. Łatwo zauważyć, że FRAM pracujący jako pamięć operacyjna będzie zużywał się, ale czas do uszkodzenia wyniesie co najmniej 8 lat. Czas ten można wydłużyć stosując pewne, dobrze znane triki w programie.

Moim zdaniem szykuje się kolejna rewolucja na miarę tej spowodowanej przez ARM. Tylko czekać, kiedy kolejne firmy wyprodukują kolejne podobne układy scalone. A i sama firma TI ma ogromny potencjał i olbrzymie portfolio różnych układów (przypomnijmy tylko OMAP stosowane w urządzeniach przenośnych, Stellaris z rdzeniem ARM, procesory DSP, dobrze znane MSP430), więc nawet tylko dzięki jej wyrobom może dokonać się przełom w sposobach programowania i wyboru mikrokontrolerów do aplikacji. Nowemu mikrokontrolerowi poświęcimy artykuł w najbliższej EP, gdy tylko dotrze do nas zamówiony, „gorący" zestaw ewaluacyjny.

Jacek Bogusz

Miesięcznik „**Elektronika Praktyczna**"
(12 numerów w roku) jest wydawany
przez AVT-Korporacja Sp. z o.o. we współpracy
z wieloma redakcjami zagranicznymi.

Wydawca:
AVT-Korporacja Sp. z o.o.
03-197 Warszawa, ul. Leszczynowa 11
tel.: +48 22 257 84 99, faks: +48 22 257 84 00

Adres redakcji:
03-197 Warszawa, ul. Leszczynowa 11
tel.: +48 22 257 84 49, +48 22 257 84 60
tel.: +48 22 257 84 65, +48 22 257 84 48
faks: +48 22 257 84 67
e-mail: redakcja@ep.com.pl
www.ep.com.pl

Redaktor Naczelny:
Wiesław Marciniak

**Redaktor Programowy,
Przewodniczący Rady Programowej:**
Piotr Zbysiński

**Zastępca Redaktora Naczelnego,
Redaktor Prowadzący:**
Jacek Bogusz, tel. +48 22 257 84 49

Redaktor Działu Projektów:
Piotr Witczak, tel. +48 22 257 84 61

Redaktor Działu Podzespołów i Sprzętu:
Jerzy Pasierbiński

Szef Pracowni Konstrukcyjnej:
Grzegorz Becker, tel. +48 22 257 84 58

Dyrektor Działu Marketingu i Reklamy:
Katarzyna Wiśniewska, tel. +48 22 257 84 65

Product Menager:
Justyna Warpas, tel. +48 22 257 84 62

Marketing i Reklama:
Katarzyna Gugała, tel. +48 22 257 84 64
Bożena Krzykawska, tel. +48 22 257 84 42
Grzegorz Krzykawski, tel. +48 22 257 84 60
Andrzej Tumański, tel. +48 22 257 84 63
e-mail: reklama@ep.com.pl

Sekretarz Redakcji:
Grzegorz Krzykawski, tel. +48 22 257 84 60

DTP:
Dariusz Welik, tel. +48 22 257 84 48

Projekt graficzny okładki:
Jakub Tarnowski

Redaktor strony internetowej:
Marek Dzwonnik

Stali Współpracownicy:
Arkadiusz Antoniak, Rafał Baranowski, Marcin Chruściel, Jarosław Doliński, Andrzej Gawryluk, Krzysztof Górski, Tomasz Jabłoński, Krzysztof Paprocki, Krzysztof Pławsiuk, Sławomir Skrzyński, Jerzy Szczesiul, Ryszard Szymaniak, Marcin Wiązania, Tomasz Włostowski, Robert Wołgajew

Uwaga! Kontakt z wymienionymi osobami jest możliwy via e-mail, według schematu: imię.nazwisko@ep.com.pl

Prenumerata:
tel.: +48 22 257 84 22, faks: +48 22 257 84 00
e-mail: prenumerata@avt.pl
Sklep: tel. +48 22 257 84 66

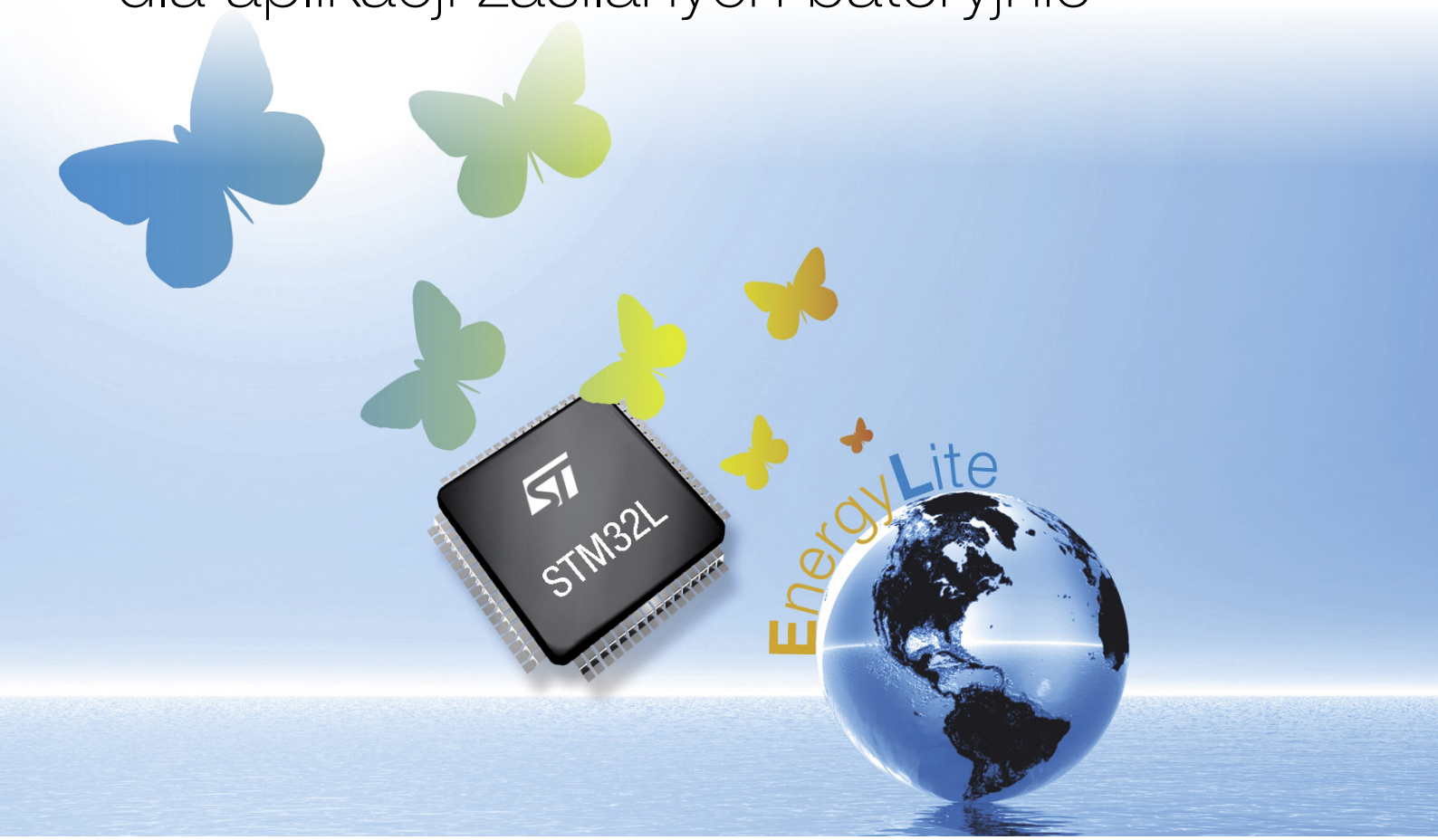
Wydawnictwo
AVT-Korporacja Sp. z o.o.
należy do **Izby Wydawców Prasy**

**Copyright AVT-Korporacja Sp. z o.o.
03-197 Warszawa, ul. Leszczynowa 11**

Projekty publikowane w „Elektronice Praktycznej" mogą być wykorzystywane wyłącznie do własnych potrzeb. Korzystanie z tych projektów do innych celów, zwłaszcza do działalności zarobkowej, wymaga zgody redakcji „Elektroniki Praktycznej". Przedruk oraz umieszczanie na stronach internetowych całości lub fragmentów publikacji zamieszczanych w „Elektronice Praktycznej" jest dozwolone wyłącznie po uzyskaniu zgody redakcji. Redakcja nie odpowiada za treść reklam i ogłoszeń zamieszczanych w „Elektronice Praktycznej".

EnergyLite

STM32L STM8L
energooszczędne mikrokontrolery dla aplikacji zasilanych bateryjnie



Energooszczędne mikrokontrolery ST w skrócie:

- wydajne rdzenie (Cortex-M3 w STM32L, STM8 w STM8L)
- nowoczesna architektura
- zaawansowane mechanizmy przyspieszania wykonywania programu
- duża wydajność obliczeniowa
- zaawansowane systemy przerwań
- bogate wyposażenie w peryferia
- wysokie maksymalne częstotliwości taktowania:
  - STM32L do 32 MHz
  - STM8L do 16 MHz
- niewielki pobór mocy:
  - STM32L od 230 µA/MHz
  - STM8L od 150 µA/MHz

www.st.com

MASTERS

Dystrybutor:
MASTERS Sp. z o.o., ul. Objazdowa 5b, 83-010 Straszyn k. Gdańska
tel. +48 58 691 06 91, masters@masters.com.pl, www.masters.com.pl

strona 42

Audiofilski układ regulacji siły głosu

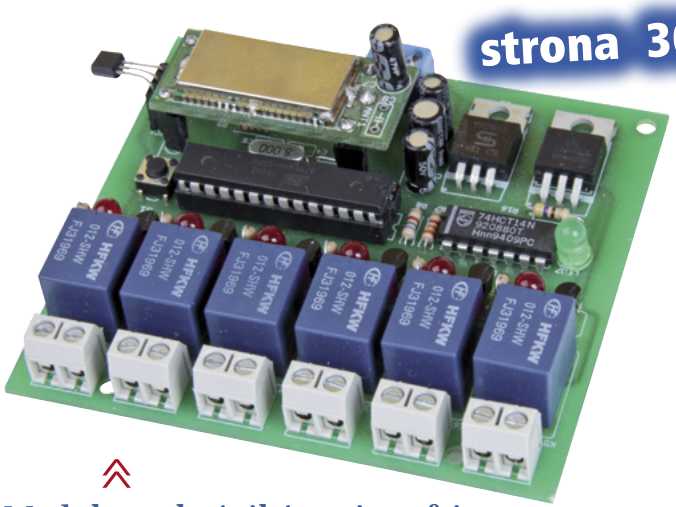
Wśród miłośników słuchania muzyki z najlepszą jakością istnieje grupa zwolenników konstrukcji minimalistycznych, w których liczbę elementów składowych redukuje się do źródła, wzmacniacza mocy i zestawów głośnikowych. Wychodząc naprzeciw ich potrzebom publikujemy projekt wysokiej jakości regulatora PVC (Passive Volume Control) o konstrukcji opartej o przekaźniki i odpowiednio dobrane rezystory.

strona 35

Timer do lustrzanek ze zdalnym wyzwalaniem

Fabryczny osprzęt do lustrzanek cyfrowych jest zwykle bardzo drogi. Mając troszkę wiedzy na temat fotografowania oraz smykałkę do elektroniki można samodzielnie zbudować wiele ciekawych urządzeń rozszerzających możliwości funkcjonalne aparatu. Prezentujemy timer, który umożliwia naświetlenie zdjęć przez zadany czas, a oprócz tego pozwala na zdalne wyzwalanie aparatu np. podczas fotografowania zwierząt.

strona 30



Moduł przekaźników z interfejsem Bluetooth

Moduł jest wyposażony w 6 przekaźników, które są sterowane za pomocą interfejsu Bluetooth. Sygnał załączający może być przesyłany prze komputer PC lub telefon komórkowy. Przykłady zastosowania takiego tandemu można mnożyć – od zwykłego załączania urządzeń do otwierania zamków przez uprawnione osoby.

strona 23

Stacja pogodowa z funkcją prognozy

Projekty stacji pogodowych cieszą się niesłabnącym zainteresowaniem Czytelników. Zapewne dlatego, że po pierwsze można coś zrobić dla domu, a po drugie – pochwalić się unikatowymi funkcjami przed znajomymi. Prezentujemy projekt takiej stacji z funkcją prognozowania pogody.

ELEKTRONIKA PRAKTYCZNA

Nr 6 (222)
Czerwiec 2011

Projekty

Stacja pogodowa z prognozowaniem pogody 23
Blue Relay. Moduł przekaźników sterowany przez Bluetooth 30
Timer fotograficzny do lustrzanki 35
Pasywny regulator siły głosu PVC 42

Miniprojekty

Minimoduł z ATmega8 49
Symetryzator sygnału audio 50
AVTduino JOY – manipulator dla Arduino 52

Projekty Czytelników

Rubidowy, korygowany GPS-em wzorzec częstotliwości i czasu 53

Wybór konstruktora

Programatory stacjonarne TEMAT NUMERU 114
Zestawy ewaluacyjne dla µC z rdzeniami Cortex-M (2) 130

Sprzęt

Multimetr U1252B Agilent Technologies 64
Hameg HMO1024. Oscyloskop edukacyjny 101
VibroSPY. Przemysłowe podglądanie 122

Podzespoły

LPC177x/8x. Wyposażenie i kompatybilność 74
AFE: sposób na sensory 76
i.MX28. Nowa skala intergracji: ARM926 + DC/DC w jednej obudowie 79
Cyfrowe rejestratory dźwięku ISD151xx (2) 92
STM32W. Cortex-M3 i zaawansowany tor RF 109
Energooszczędne MSP430. Nowa seria 0xx 126

Kursy

Kurs programowania mikrokontrolerów PIC (3). Interfejs użytkownika 58
Projektowanie płytek za pomocą Altium Designer Summer 09 (7) 68
Technologia GSM w elektronice (12). Tryby oszczędzania energii 82
Kurs Arduino (3). Obsługa modułu LCD 86
Cyfrowe zwrotnice głośnikowe (2). Problem synchronizacji zegara systemowego 89
Wprowadzenie do Linuksa embedded (3). Obsługa portów GPIO 96

Prezentacje

Programatory firmy Wellon TEMAT NUMERU 105
Małogabarytowe kondensatory mocy do zastosowań w motoryzacji 106
Telekomunikacja czwartej generacji dzięki układom Texas Instruments 112
Produkcja robi to inaczej TEMAT NUMERU 118
PIC32. Korzyści płynące ze stosowania 32-bitowych mikrokontrolerów 134

Automatyka i Mechatronika Praktyczna

Pozycjonowanie napędu SINAMICS (2) 149
Gotowe moduły liniowe. Aplikacje związane z paletyzacją i przenoszeniem produktów 153

Konkursy

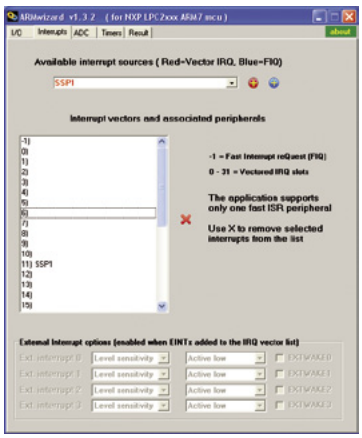
Microchip 136

Od wydawcy 4
Nie przeocz. Niezbędnik elektronika 8
Nie przeocz. Podzespoły 10
Nie przeocz. Koktajl niusów 16
Sprzężenie zwrotne. Poczta 18
Sprzężenie zwrotne. Ankieta 20
Prenumerata 21
Kramik i rynek 137
Info 143
Biblioteka 146
Księgarnia wysyłkowa 157
Oferta 159
Zapowiedź następnego numeru 164

REKLAMA

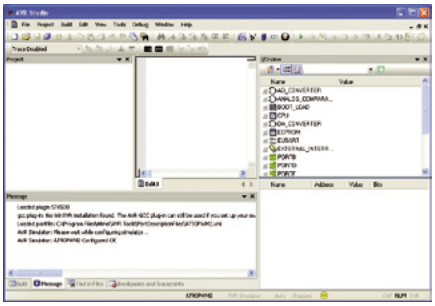
niezbędnik
elektronika

TYLKO DLA PRENUMERATORÓW



ARMwizard v1.3.2

ARMwizard jest aplikacją pomagającą edytować rejestry, przerwania, timery mikrokontrolerów ARM7 rodziny LPC2xxx firmy Philips (NXP). Jest przeznaczony głównie do stosowania ze środowiskiem KEIL, ale mimo tego utworzony kod można z łatwością zmodyfikować dla innych kompilatorów.

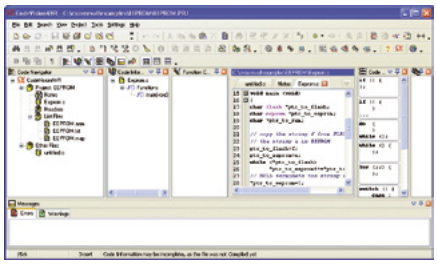


AVR Studio 4.18_build 684

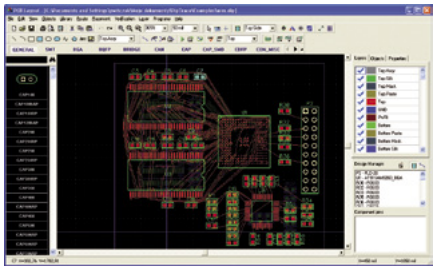
AVR Studio jest zintegrowanym środowiskiem programistycznym dla mikrokontrolerów Atmel AVR. Program działa pod Win XP, Vista i zawiera komplet narzędzi do zarządzania projektem m.in. edytor plików źródłowych, symulator, programator.

CodeVisionAVR Evaluation V2.05.1b

CodeVisionAVR jest kompilatorem C dla mikrokontrolerów AVR firmy Atmel. Aplikacja działa pod kontrolą Windows XP, 7 i Vista. CodeVisionAVR zawiera

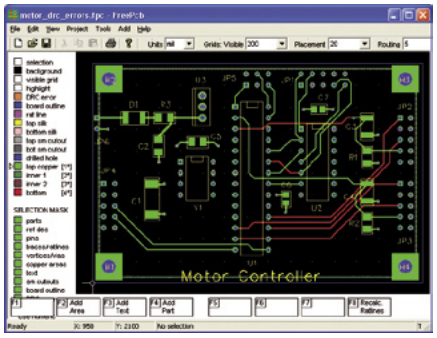


interfejs programatora, dzięki któremu w szybki i łatwy sposób jest możliwe załadowanie do mikrokontrolera wcześniej napisanego programu. Interfejs ten współpracuje m.in. z STK500, STK600, AVRISP.



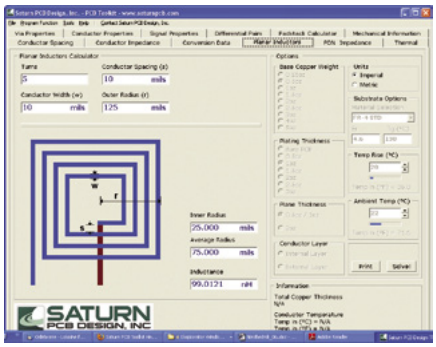
DIPtrace

DIPtrace jest programem służącym do projektowania płytek drukowanych. Zawiera edytor schematów, edytor PCB, edytor bibliotek, autorouter, pełny pakiet bibliotek elementów. Ograniczeniem w wersji demo jest czas użytkowania do 30 dni i użycie maksymalnie 300 pinów w edytorze PCB.



FreePCB1.2

FreePCB to darmowy program dla projektantów PCB. Program umożliwia projektowanie 16-warstwowych płytek PCB o wymiarach 150 cm×150 cm, wybór jednostek miary, pokrycie płytki miedzią, import i eksport listy połączeń, generowanie plików w formacie Gerber.



SaturnPCB 5.32

Saturn PCB Toolkit jest niezbędnym narzędziem wspomagającym projektowanie PCB. Pozwala na szybkie obliczenie wielu parametrów obwodów drukowanych m.in. szerokości ścieżek zależnie od natężenia prądu, odległości między nimi, ich indukcyjności.

StellarisWare 7243

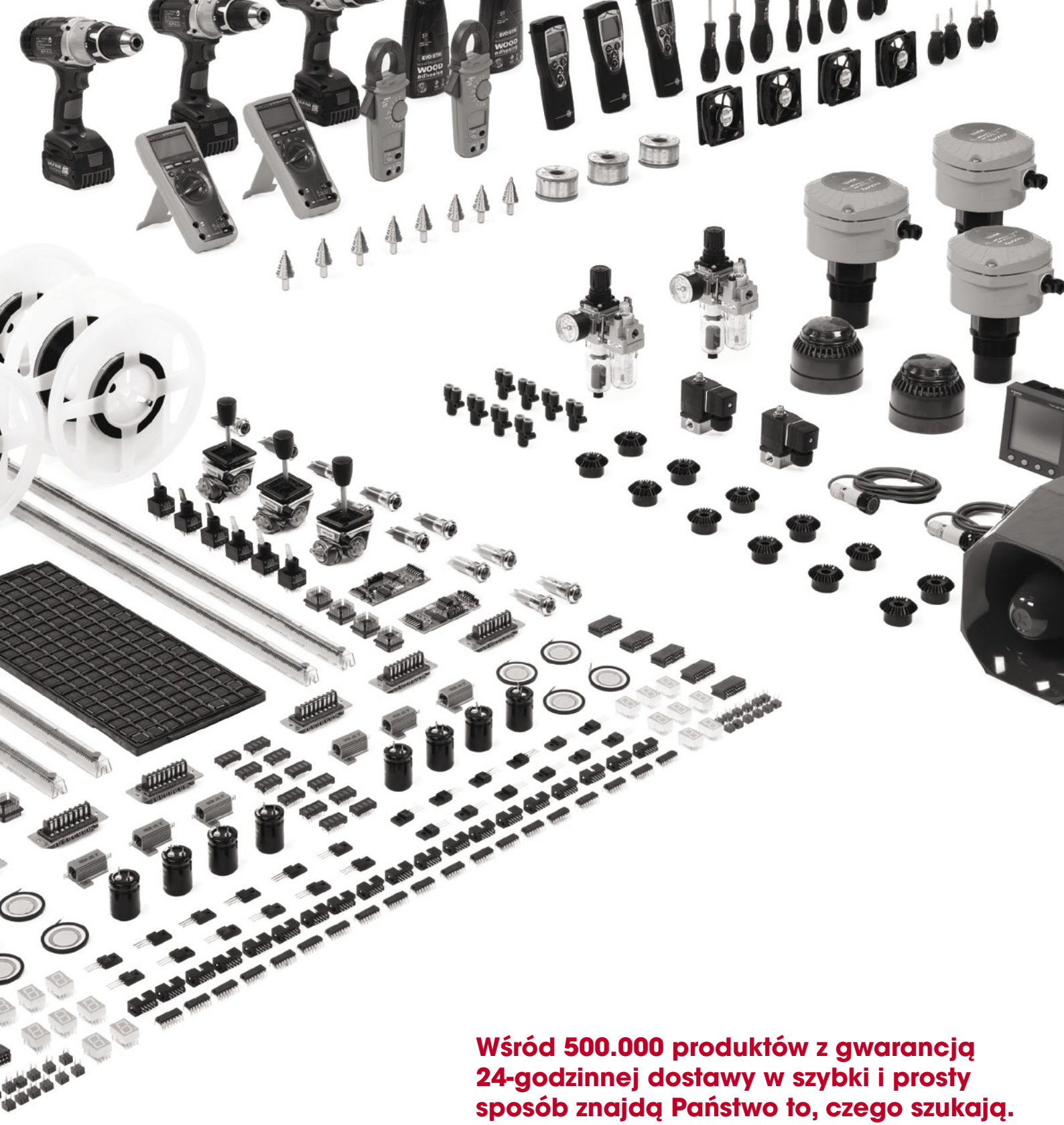
Oprogramowanie StelarisWare jest pakietem narzędzi upraszczającym i przyśpieszającym opracowywanie aplikacji z mikrokontrolerami Stelaris.

STLink

ST-LINK jest debuggerem i programatorem dla mikrokontrolerów z rodzin STM8 i STM32.
Współpracuje m.in. z ST Visual Develop (STVD) oraz ST Visual Programmer (STVP).

TI Audio Guide.pdf

Przewodnik po układach audio firmy Texas Instruments. Zaprezentowane w przewodniku układy należą m.in. do grup wzmacniaczy w klasach AB, D, G, przetworników audio, procesorów DSP, interfejsów i przełączników źródeł.



Wśród 500.000 produktów z gwarancją 24-godzinnej dostawy w szybki i prosty sposób znajdą Państwo to, czego szukają.

www.rspoland.com

nowe podzespoły

Z kilkuset nowości wybraliśmy te, których nie wolno przeoczyć
Bieżące nowości można śledzić na www.elektronikaB2B.pl

Kontrolery PFC do zasilaczy AC-DC o mocy wyjściowej od 300 W do 8 kW

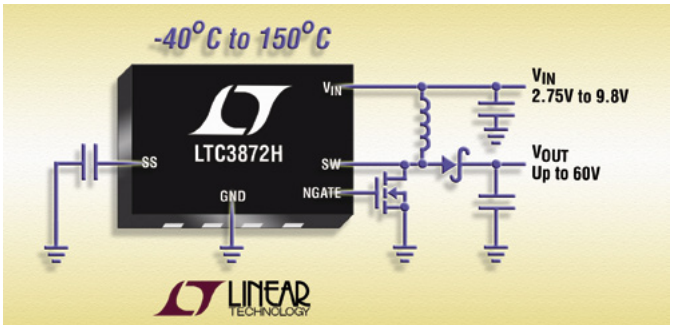
Kontrolery PFC z nowej serii µPFC typu IR115x są przeznaczone do korekcji współczynnika mocy w zasilaczach AC-DC o mocy wyjściowej od 300 W do 8 kW. Wykorzystują technologię OOC (*One Cycle Control*) zapewniającą duży współczynnik PF, małą zawartość harmonicznych i bardzo dobrą regulację napięcia na szynie DC przy minimalnej liczbie elementów współpracujących i prostej aplikacji. Są produkowane w obudowach SOIC-8. Zostały zaprojektowane do pracy w konwerterach Boost PFC pracujących w trybie przewodzenia ciągłego, akceptujących napięcie wejściowe z zakresu od 85 do 264 V_{AC}. Zawierają zabezpieczenie nadnapięciowe, podnapięciowe, zabezpieczenie przed rozwarciem pętli, ogranicznik prądowy i programowalny układ miękkiego startu. Różnią się zakresem napięcia zasilania, prądem wyjściowym, częstotliwością pracy i wbudowanymi obwodami zabezpieczeń. Inicjowany przez użytkownika tryb uśpienia zapewnia kompatybilność z wymogami norm Energy Star, Green Power i Blue Angel odnośnie do poboru mocy w trybie standby.

http://www.irf.com

	V_{CC} [V]	Prąd wyjściowy [A]	Częstotliwość pracy [kHz]	Zabezpieczenie nadnapięciowe	Zabezpieczenie Brown-out
IR1152S	14...17	±0,75	66	Podwójne	tak
IR1153S	14...17	±0,75	22,2	Tak	tak
IR1155S	12...20	±1,5	48...200	Tak	nie

Wysokonapięciowy kontroler DC-DC Boost o zakresie temperatury pracy złącza –40...+150°C

Kontroler przetwornicy DC-DC LTC3872 jest obecnie dostępny w wersji H-grade o rozszerzonym dopuszczalnym zakresie temperatury pracy złącza. Wynosi ona od –40°C do +150°C. Jest to kontroler typu Boost (podwyższający napięcie wejściowe) o stałej częstotliwości pracy (550 kHz), zasilany napięciem z zakresu 2,5...9,8 V. Pozwala na generowanie napięć wyjściowych do 60 V i nie wymaga szeregowego rezystora pomiarowego R_{SENSE}. Wymaga zewnętrznego tranzystora MOSFET z kanałem typu „n" oraz kilku podzespołów biernych. Zawiera źródło napięcia referencyjnego o dokładności ±1,5% utrzymywanej w całym dopuszczalnym zakresie temperatury pracy.



LTC3872 ma tryb pracy *pulse-skipping* aktywowany w zakresie małego obciążenia, który pozwala na zredukowanie poboru prądu spoczynkowego do 250 µA. Dzięki temu jest możliwe wydłużenie czasu pracy urządzeń bateryjnych przebywających przez dłuższy czas w trybie standby. Zawiera układ miękkiego startu, regulowany ogranicznik prądu wyjściowego i wejściowe zabezpieczenie podnapięciowe. Jest produkowany w obudowach DFN-8 i ThinSOT-8.

http://www.linear.com

Prosty w aplikacji sterownik diod LED o parametrach wyjściowych 30 V/1,5 A

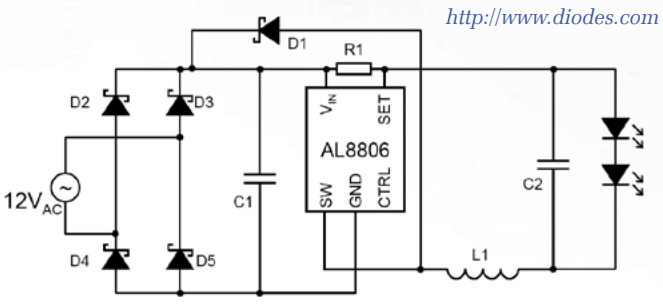
AL8806 to tani i prosty w implementacji sterownik wysokoprądowych diod LED umożliwiający ich zasilanie napięciem do 30 V i prądem o natężeniu do 1,5 A. Dzięki temu może zasilać grupy 8 połączonych szeregowo diod LED o wysokiej jasności. Układ powinien być zasilony ze źródła o napięciu od 6 do 30 V. W najprostszej aplikacji wymaga dołączenia jedynie 4 elementów zewnętrznych. Natężenie prądu wyjściowego jest regulowane sygnałem PWM.

AL8806 zapewnia największą sprawność spośród oferowanych przez firmę Diodes sterowników sięgającą nawet 98%. Jest produkowany w obudowie MSOP-8EP o małej rezystancji termicznej, umożliwiającej pracę z pełną wydajnością w szerokim zakresie temperatury otoczenia.

Pozostałe cechy:

- zabezpieczenie przed rozwarciem wyjścia,
- częstotliwość taktowania do 1 MHz,
- dokładność regulacji prądu wyjściowego lepsza niż 5%,
- zabezpieczenie ESD do 2,5 kV HBM,
- dopuszczalny zakres temperatury pracy złącza od –40°C do +125°C.

http://www.diodes.com



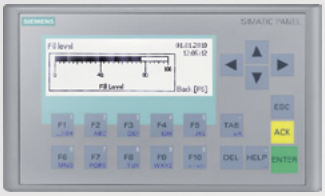
Zestawy startowe SIMATIC S7-1200

Zestaw SIMATIC S7-1200

CPU 1212C AC/DC/RLY,
Symulator wejść,
SIMATIC STEP 7 Basic,
Kabel ethernet,
Dokumentacja na CD,

Nr zam. 6ES7212-1BD31-4YB0

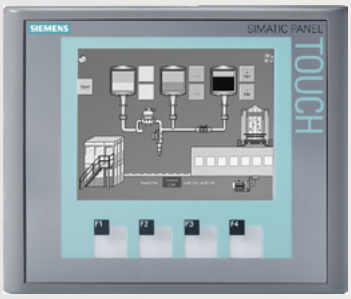
Dostępny już za 1200 zł*



Zestaw SIMATIC S7-1200 + KP 300 Basic

CPU 1212C AC/DC/RLY,
Symulator wejść,
Panel Kp300 Basic mono PN,
SIMATIC STEP 7 Basic,
Kabel ethernet,
Dokumentacja na CD,

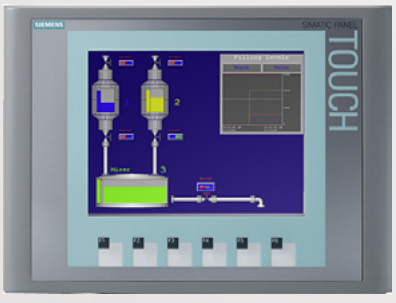
Nr zam. 6AV6651-7HA01-3AA1



Zestaw SIMATIC S7-1200 + KTP 400 Basic

CPU 1212C AC/DC/RLY,
Symulator wejść,
Panel KTP400 Basic mono PN,
SIMATIC STEP 7 Basic,
Kabel ethernet,
Dokumentacja na CD,

Nr zam. 6AV6651-7AA01-3AA1



Zestaw SIMATIC S7-1200 + KTP 600 Basic

CPU 1212C AC/DC/RLY,
Symulator wejść,
Panel KTP600 Basic color PN,
SIMATIC STEP 7 Basic,
Kabel ethernet,
Dokumentacja na CD,

Nr zam. 6AV6651-7DA01-3AA1

Zestawy startowe S7-1200

Zestaw startowy jest doskonałym narzędziem, za pomocą którego można zapoznać się ze wszystkimi funkcjami sterownika SIMATIC S7-1200 w oparciu o oprogramowanie Step7 Basic.
Produkt do kupienia u autoryzowanych dystrybutorów: **www.siemens.pl/simatic/dystrybutorzy**

* Sugerowana cena detaliczna netto.

www.siemens.pl/S7-1200

SIEMENS

Prenumeruj!

za darmo lub półdarmo

Jeśli jeszcze nie prenumerujesz Elektroniki Praktycznej, spróbuj za darmo! My damy Ci bezpłatną prenumeratę próbną od lipca 2011 do września 2011, Ty udokumentuj swoje zainteresowanie wpłatą kwoty 144,00 zł na kolejne 9 numerów EP (październik 2011 – czerwiec 2012). Będzie to coś w rodzaju zwrotnej kaucji. Jeśli nie uda nam się przekonać Cię do prenumeraty i zrezygnujesz z niej przed 16.09.2011 r. – otrzymasz zwrot całej swojej wpłaty.

BEZPŁATNA PRENUMERATA PRÓBNA	PRENUMERATA 9-MIESIĘCZNA (VAT 8%)
od lipca 2011 r. do września 2011 r.	od października 2011 r. do czerwca 2012 r.
3×0,00 zł=0,00 zł	9×16,00 zł=144,00 zł

Jeśli już prenumerujesz EP, nie zapomnij przedłużyć prenumeraty! Rozpoczynając drugi rok nieprzerwanej prenumeraty EP nabywasz prawa do zniżki.
W przypadku prenumeraty rocznej jest to zniżka w wysokości ceny 2 numerów. Rozpoczęcie trzeciego roku prenumeraty oznacza prawo do zniżki o wartości 3 numerów, zaś po 3 latach nieprzerwanej prenumeraty masz możliwość zaprenumerowania EP w cenie obniżonej o wartość 4 numerów. Jeszcze więcej zyskasz, decydując się na prenumeratę 2-letnią – nie musisz mieć żadnego stażu Prenumeratora, by otrzymać ją w cenie obniżonej o wartość aż 8 numerów! Więcej – po 3 latach nieprzerwanej prenumeraty upust na cenie prenumeraty 2-letniej równy jest wartości 10 numerów, a po 5 latach zniżka osiąga wartość 12 numerów, tj. **50%**

CENY PRENUMERATY (VAT 8%, standardowa cena prenumeararty rocznej – 170,50 za rok)				
	okres dotychczasowej nieprzerwanej prenumeraty			
	rok	2 lata	3 lata lub 4 lata	5 i więcej lat
rocznej	160,00 zł (2 numery gratis)	144,00 zł (3 numery gratis)	128,00 zł (4 numery gratis)	
2-letniej	256,00 zł (8 numerów gratis)		224,00 zł (10 numerów gratis)	192,00 zł (12 numerów gratis)

PAMIĘTAJ ! TYLKO PRENUMERATORZY:*

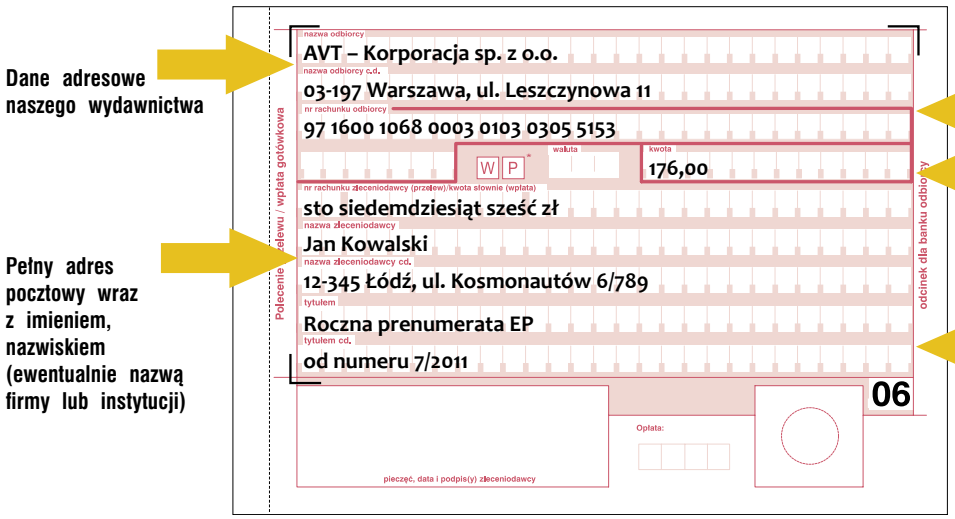
- otrzymują 80% zniżki na równoległą prenumeratę e-wydań (zamówienia na *www.avt.pl/eprenumerata*)
- otrzymują co miesiąc „Niezbędnik Elektronika" na CD
- mają bezpłatny dostęp do specjalnego serwisu EP na stronie *www.ep.com.pl/archiwum* (dla pozostałych Czytelników – dostęp za mikropłatnościami SMS-ami)
- mogą otrzymywać co miesiąc bezpłatny numer archiwalny EP (zamawiając dowolne z dostępnych jeszcze wydań sprzed czerwca 2010 r. – otrzymasz je wraz z prenumeratą; zamówienie możesz wysłać mailem na nasz adres prenumerata@avt.pl)
- zostają członkami Klubu AVT-elektronika i otrzymują wiele przywilejów oraz rabatów (patrz *www.avt.pl/klub*)
- mają prawo do upustów w sklepie *www.sklep.avt.pl*

*) nie dotyczy prenumerat zamówionych u pośredników (RUCH, Poczta Polska i in.); nie dotyczy bezpłatnych prenumerat próbnych.

CENY PRENUMERATY WERSJI ELEKTRONICZNEJ EP (VAT 23%)			
	6-miesięcznej	12-miesięcznej	24-miesięcznej
standardowe	6×6,50 zł = 39,00 zł	12×5,70 zł = 68,40 zł	24×5,17 zł = 124,80 zł
dla Prenumeratorów wersji papierowej	80% zniżki = 7,80 zł	80% zniżki = 13,65 zł	80% zniżki = 24,95 zł

Prenumeratę zamawiamy:

Najprościej → **dokonując wpłaty**



Najłatwiej → **wypełniając formularz w Internecie**
(na stronie *www.ep.com.pl*)
– tu można zapłacić kartą

Najwygodniej → wysyłając na numer 663 889 884 SMS-a o treści PREN
– oddzwonimy i przyjmiemy zamówienie (koszt SMS-a wg Twojej taryfy)

lub → **zamawiając** za pomocą telefonu, e-maila, faksu lub listu.

**Dział Prenumeraty Wydawnictwa AVT, ul. Leszczynowa 11, 03-197 Warszawa,
tel.: 22 257 84 22, faks: 22 257 84 00, e-mail: prenumerata@avt.pl**

Stacja pogodowa z prognozowaniem pogody

AVT 5294

Współczesne, miniaturowe czujniki wielkości nieelektrycznych umożliwiają łatwe zbudowanie urządzeń, które na podstawie zmierzonych wielkości, przetwarzanych za pomocą mikrokontrolerów, są w stanie podejmować różne akcje. Może to być na przykład dostosowanie temperatury i wilgotności powietrza do wymagań magazynowanej żywności lub – jak w naszym projekcie – do prognozowania pogody.

Rekomendacje: *stacja z prognozowaniem pogody przyda się w każdym domu; może być użytecznym gadżetem, który np. uchroni domowników przed zmoknięciem w czasie letniego spaceru.*

Przykładem czujnika wielkości nieelektrycznych wykonanego w technologii MEMS jest czujnik ciśnienia atmosferycznego z wyjściem cyfrowym typu MPL115A2 produkowany przez firmę Freescale. Zanim jednak opiszę metodę jego aplikacji, warto poświęcić kilka słów zagadnieniom teoretycznym.

Zagadnienia wstępne

Ciśnienie atmosferyczne jest stosunkiem wektora siły będącego wypadkową wszystkich sił działających na powierzchnię Ziemi do wielkości pola tej powierzchni. Wynika z tego, że ciśnienie atmosferyczne w górach jest niższe, a na nizinach wyższe, ponieważ słup powietrza ma w tych rejonach różną wysokość. Zależność ta ma w przybliżeniu charakter wykładniczy, co pokazano na **rysunku 1**.

Wartość ciśnienia atmosferycznego dla standardowych warunków pogodowych dla wysokości *h* (n.p.m.) możemy wyznaczyć z uproszczonej zależności opisanej wzorem:

$$p_h = p_0 \cdot e^{\frac{-h}{7990}}$$

gdzie:

$\mathbf{p_h}$ – ciśnienie na wysokości h n.p.m. dla standardowych warunków pogodowych

$\mathbf{p_0}$ – ciśnienie ma poziomie morza w standaryzowanych warunkach pogodowych równe 1013,25 hPa

h – wysokość n.p.m.

Ciśnienie atmosferyczne ulega ciągłym zmianom, które są zależne od zmian pogody. Śledząc zmiany ciśnienia na danej stałej wysokości n.p.m., możemy z pewnym prawdopodobieństwem prognozować pogodę. Na przykład na umiarkowanych szerokościach geograficznych powolne obniżanie się ciśnienia oznacza zbliżanie się niżu, czyli pogorszenia się pogody (wystąpienie chmur, opadów, ochłodzenia w lecie, ocieplenia w zimie), zaś wzrost ciśnienia łączy się z poprawą pogody (ustąpieniem mgły czy zachmurzenia, osłabieniem wiatru itp.). Ciśnienie wyższe o kilka hPa od ciśnienia p_h dla danego rejonu zwiastuje utrwalenie się ładnej pogody, jednak z drugiej strony, szybki wzrost ciśnienia może także zwiastować burzę. Co ciekawe, na podstawie gradientu zmian ciśnienia można dość dokładnie określić spodziewaną siłę wiatru. Pomimo tych, wydawałoby się nieskomplikowanych zależności, przewidywanie zmian pogody jest procesem trudnym (a zwłaszcza przygotowywanie prognoz długookresowych) i obarczo-

AVT-5294 w ofercie AVT:

AVT-5294A – płytka drukowana

Podstawowe informacje:

- Pomiar temperatury i ciśnienia z funkcją prognozowania pogody
- Napięcie zasilania: 5...9 V_{DC}
- Maksymalny prąd obciążenia (z podświetleniem/bez podświetlenia): 30 mA/9 mA
- Zakres wysokości dla funkcji przewidywania pogody: 0...990 m n.p.m.
- Zakres pomiarowy wbudowanego termometru: 0...99°C
- Zakres pomiarowy barometru: 500...1150 hPa
- Ustawienia ważniejszych fusebitów:
  CKSEL3...0: 0010
  SUT1...0: 10
  CKDIV8: 1
  CKOUT: 1
  WDTON: 1
  SPIEN: 0
  BOOTRST: 1
  BODLEVEL2...0: 111

Dodatkowe materiały na CD/FTP:

ftp://ep.com.pl, user: 10925, pass: 87thc181

- wzory płytek PCB
- karty katalogowe i noty aplikacyjne elementów oznaczonych w Wykazie elementów kolorem czerwonym

Projekty pokrewne na CD/FTP:

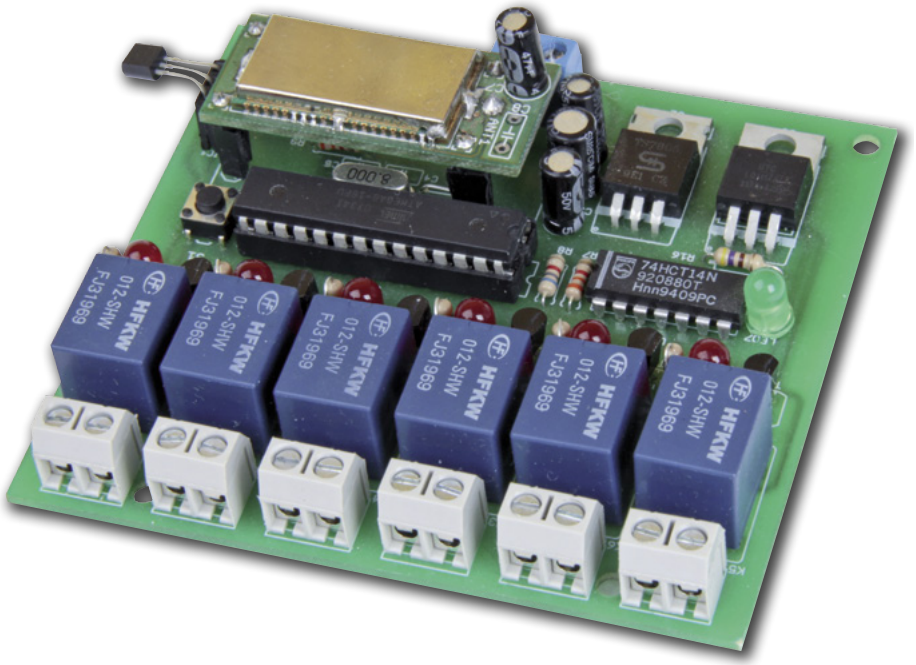
(wymienione artykuły są w całości dostępne na CD)

AVT-5141	Internetowa stacja pogodowa Yahoo (EP 7–8/2008)
AVT-961	Domowa stacja pogodowa (EP 12/2006)
AVT-2787	PC-Termometr – termometr internetowy (EdW 5/2006)
AVT-5060	Domowa stacja meteo ze zdalnym pomiarem temperatury (EP 4–5/2002)

Blue Relay

AVT 5295

Moduł przekaźników sterowany przez Bluetooth



Standard łączności bezprzewodowej Bluetooth jest jednym z najczęściej używanych w różnych urządzeniach powszechnego użytku. W odpowiednie interfejsy radiowe są wyposażane komputery PC, telefony komórkowe, smartphone'y i inne. Dla urządzeń przeznaczonych do tzw. inteligentnego domu najciekawsze wydaje się zwłaszcza połączenie siły oprogramowania komputera PC z urządzeniami wykonawczymi sterowanymi bezprzewodowo, co uwalnia użytkownika od konieczności doprowadzenia kabli połączeniowych.

***Rekomendacje:** moduły można stosować do załączania/wyłączania domowych odbiorników energii elektrycznej w miejscach, w których doprowadzenie kabli jest trudne lub wręcz niemożliwe.*

Zaprojektowane urządzenie to moduł wyposażony w 6 przekaźników sterowanych bezprzewodowo za pomocą Bluetooth z każdego. Jego oprogramowanie jest podzielone na dwie części, z których pierwsza jest przeznaczona do komputera PC pracującego pod kontrolą systemu Windows i wyposażonego w interfejs Bluetooth, a druga dla mikrokontrolera sterującego pracą modułu.

Urządzenie jest łatwe w budowie, więc nawet początkujący elektronicy nie powinni mieć trudności z jego wykonaniem. Ma ono funkcję niezależnego sterowania 6 przekaźnikami wykonawczymi. Dla każdego z nich można ustawić indywidualną zwłokę czasową, by go włączyć lub wyłączyć o określonym czasie. Kolejną funkcją jest jednoczesne włączanie lub wyłączanie wszystkich przekaźników. Aby uniknąć przypadkowego załączania przekaźników, program ma proste zabezpieczenie: w momencie zerwania połączenia pomiędzy komputerem a modułem, np. przy wyłączeniu komputera lub za pomocą przycisku Rozłącz, urządzenie wyłączy wszystkie przekaźniki.

Sercem modułu jest mikrokontroler ATmega8. Rozważałem zastosowanie tańszego ATTiny2313, lecz jego pamięć Flash mogłaby nie pomieścić programu sterującego. Mikrokontroler jest taktowany sygnałem o częstotliwości 8 MHz generowanym z użyciem zewnętrznego rezonatora kwarcowego. Jego użycie jest niezbędne, ponieważ do połączenia z modułem BTM jest używany szeregowy, asynchroniczny interfejs UART wymagający do prawidłowej pracy stabilnej częstotliwości zegarowej.

Zasilanie cewek przekaźników wykonawczych jest załączane za pomocą tranzystorów BC547. Stanem aktywnym (załączającym przekaźnik) jest poziom wysoki napięcia na wyprowadzeniu portu mikrokontrolera. Dodatkowo, wykorzystując komunikację dwukierunkową, moduł wyposażono w sensor temperatury. Dzięki temu do komputera PC jest przesyłania informacja o temperaturze w pomieszczeniu, w którym pracuje moduł. Funkcjonalności tej można użyć na przykład do budowy regulatora temperatury sterowanego za pomocą komputera PC.

AVT-5295 w ofercie AVT:
AVT-5295A – płytka drukowana
AVT-5295B – płytka drukowana + elementy

Podstawowe informacje:
- Sterowanie z komputera PC za pomocą interfejsu Bluetooth
- Oprogramowanie wykonane w Bascom AVR oraz MS Visual Studio 2005
- Mikrokontroler ATmega8
- Moduł Bluetooth BTM-222. Zasięg około 10...15 m.
- Zasilanie 12 V_{DC}, pobór prądu około 800 mA

Dodatkowe materiały na CD/FTP:
ftp://ep.com.pl, user: 10925, pass: 87thc181
- wzory płytek PCB
- karty katalogowe i noty aplikacyjne elementów oznaczonych w Wykazie elementów kolorem czerwonym

Projekty pokrewne na CD/FTP:
(wymienione artykuły są w całości dostępne na CD)

AVT-5250	Karta przekaźników z interfejsem Ethernet (EP 8/2010)
AVT-2938	Blue Supply (EdW 4/2010)
AVT-5200	Uniwersalny sterownik ethernetowy (EP 9/2009)
AVT-5157	Przekaźnik internetowy (EP 11/2008)
AVT-2859	Internetowy sterownik urządzeń (EdW 3/2008)
AVT-966	Karta przekaźników sterowana przez internet (EP 2/2007)
AVT-956	Ethernetowy sterownik (EP 11/2006)
AVT-925	Karta przekaźników na USB (EP 4/2006)

Budowa

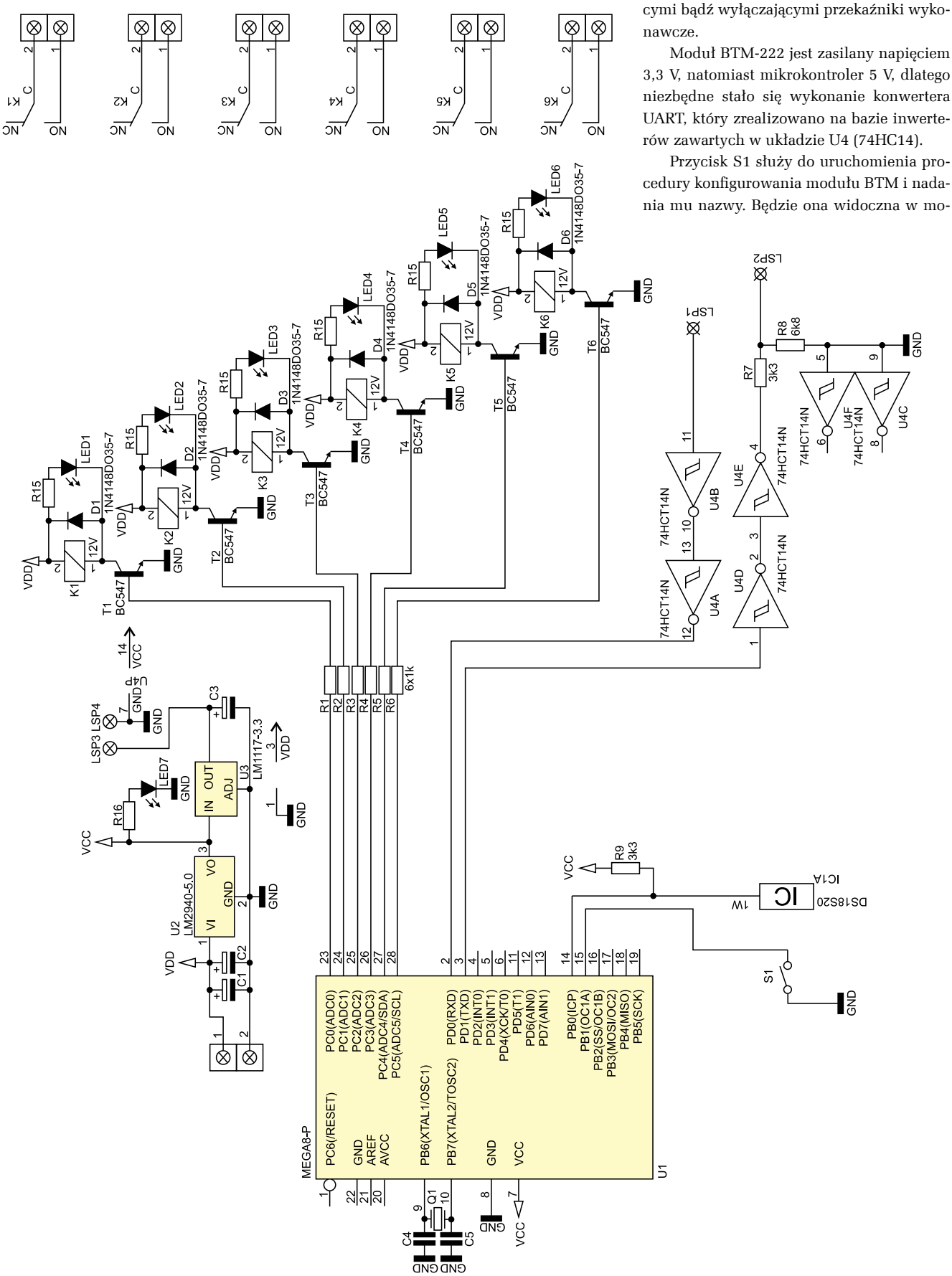
Schemat ideowy modułu pokazano na **rysunku 1**. Zasilacz składa się z połączonych kaskadowo stabilizatorów IC2 i IC4.

IC2 (LM2940T-5) służy do zasilania mikrokontrolera oraz reszty układów napięciem 5 V. Drugi stabilizator to LM1117-3.3 stabilizujący napięcie przeznaczone do zasilania modułu Bluetooth BTM-222.

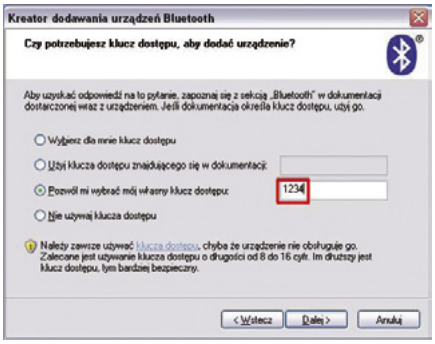
BTM-222 to jeden z najpopularniejszych w Polsce modułów. Na jego popularność wpływa zapewne łatwość obsługi programowej i zaimplementowanie przez producenta kompletnego stosu komunikacyjnego Bluetooth. Mikrokontroler jest dołączony do modułu za pomocą UART. Polecenia są przesyłane przez komputer PC, odbierane i interpretowane przez mikrokontroler, który następnie odpowiednio steruje tranzystorami włączającymi bądź wyłączającymi przekaźniki wykonawcze.

Moduł BTM-222 jest zasilany napięciem 3,3 V, natomiast mikrokontroler 5 V, dlatego niezbędne stało się wykonanie konwertera UART, który zrealizowano na bazie inwerterów zawartych w układzie U4 (74HC14).

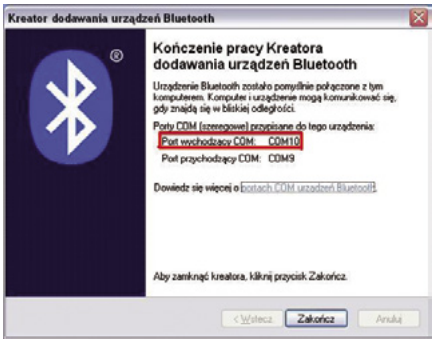
Przycisk S1 służy do uruchomienia procedury konfigurowania modułu BTM i nadania mu nazwy. Będzie ona widoczna w mo-



Rysunek 1. Schemat ideowy modułu przekaźników z interfejsem Bluetooth



Rysunek 6. Wpisanie klucza dostępu



Rysunek 7. Wirtualne porty COM po sparowaniu modułu

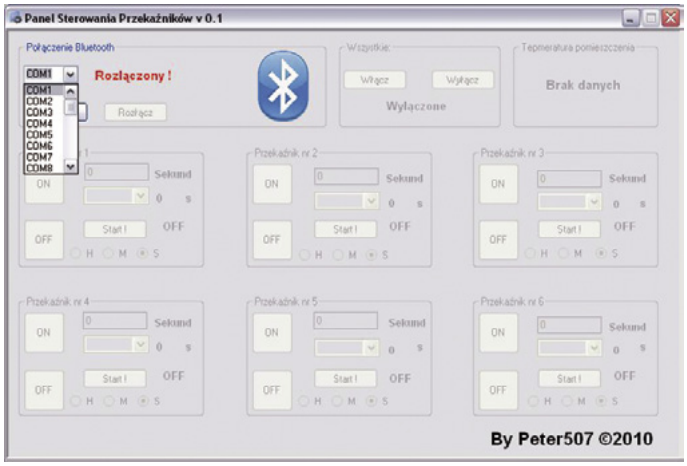
Kolejnym etapem jest sparowanie modułu przekaźników z komputerem PC, do czego jest niezbędne uruchomienie funkcji jego wyszukiwania. W tym celu wchodzimy w menu *Start –> Panel Sterowania –> Urządzenia Bluetooth*. Wybieramy opcję „Dodaj" (**rysunek 3**), następnie zaznaczamy, że nasze urządzenie jest gotowe do odnalezienia (**rysunek 4**) i klikamy „Dalej". Czekamy, aż komputer odnajdzie moduł Bluetooth, co pokazano na **rysunku 5**. Zaznaczamy go, klikamy „Dalej" i oznaczamy opcję z wyborem własnego klucza dostępu do urządzenia. Tu wpisujemy „1234" (**rysunek 6**).

Gdy proces łączenia urządzeń w parę zakończy się, wyświetlą się dwa nowe, wirtualne porty COM (**rysunek 7**). Program sterujący używa tylko portu wychodzącego, którego numer należy wpisać w jego parametrach.

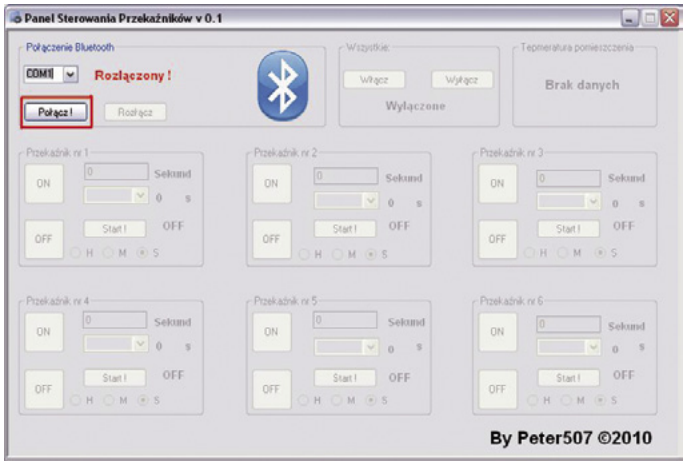
Aplikacja sterująca

Przed uruchomieniem programu *Relay Controller* należy zainstalować pakiet firmy *Microsoft* o nazwie *.NET Framework 2.0*. Po uruchomieniu programu wybieramy z listy numer portu COM, który wyświetlił się nam wcześniej (**rysunek 8**) i klikamy przycisk „Połącz" (**rysunek 9**). Jeśli pojawi się komunikat „Połączony", to oznacza, że komputer nawiązał komunikację z modułem.

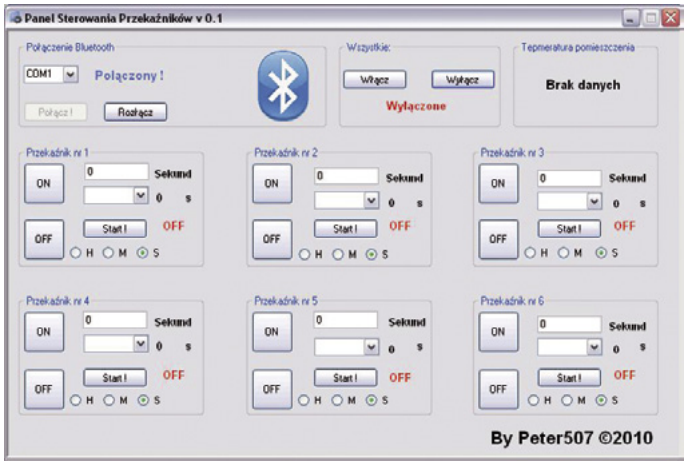
Okno główne programu przedstawia **rysunek 10**. Rozwijana lista w lewym, górnym rogu okna programu służy do wyboru wirtualnego portu COM, przez który moduł łączy się z komputerem PC. Przycisk służący do połączenia umieszczono niżej. Obok niego jest umieszczony przycisk służący do zerwania połączenia. Odpowiednie funkcje wywoływane po naciśnięciu przycisków otwierają lub zamykają wybrany wcześniej z listy port COM oraz powodują zmianę komunikatu „Rozłączony" na „Połączony". W grupie „Temperatura pomieszczenia" jest wynik pomiaru temperatury przesłany przez urządzenie. Jego odświeżanie następuje co 1 sekundę.



Rysunek 8. Wybór portu COM po uruchomieniu panelu kontrolnego



Rysunek 9. Przycisk łączący z modułem przekaźników



Rysunek 10. Okno główne wirtualnego panelu kontrolnego

Każda grupa o nazwie przekaźnik zawiera niezbędne elementy do niezależnego sterowania każdym z 6 przekaźników. Znajdują się w niej przyciski ON/OFF oraz START, który włącza odmierzanie czasu. W górnym polu należy wpisać czas zwłoki, a następnie wybierać, czy przekaźnik ma zostać włączony, czy wyłączony. Na końcu zaznaczamy, czy wpisany czas został określony w godzinach, minutach czy sekundach. Następnie klikamy na przycisk START. Na ekranie wyświetli się czas pozostały do uruchomienia zaprogramowanej akcji. Te same funkcje możemy zdefiniować dla każdego z dostępnych przekaźników.

Piotr Rosenbaum
peter507@o2.pl

Timer fotograficzny do lustrzanki

AVT 5296

Fabryczny osprzęt do lustrzanek cyfrowych jest zwykle bardzo drogi. Mając troszkę wiedzy na temat fotografowania oraz smykałkę do elektroniki, można samodzielnie zbudować wiele ciekawych urządzeń rozszerzających możliwości funkcjonalne aparatu. Prezentujemy timer, który umożliwia naświetlenie zdjęć przez zadany czas, a oprócz tego pozwala na zdalne wyzwalanie aparatu np. podczas fotografowania zwierząt.

Rekomendacje: *użyteczny gadżet do aparatu fotograficznego, który przyda się wszystkim posiadaczom lustrzanek Pentax i Canon.*

Od kiedy stałem się właścicielem lustrzanki cyfrowej, zacząłem się zastanawiać nad możliwością zbudowania dodatkowych urządzeń usprawniających fotografowanie. Pierwszym bardzo użytecznym dodatkiem do mojego systemu stał się pilot na podczerwień. Możliwość zdalnego wyzwolenia migawki powoduje, że na zdjęciach pojawia się częściej sam fotograf, a zdjęcia wykonywane bez dotykania aparatu umieszczonego na statywie są bardziej ostre, co jest szczególnie przydatne w trybie makrofotografii.

W miarę zdobywania doświadczeń wielu entuzjastów fotografii szuka sposobów na powiększenie możliwości swoich aparatów. Te najbardziej oczywiste, czyli zakup nowych elementów (obiektywów, lamp błyskowych, statywów itp.), często pociągają za sobą spore wydatki. Oprócz tego można wzbogacać swój warsztat fotografik,a stosując użyteczne, ale tanie, wykonane własnoręcznie akcesoria dodatkowe. Jednym z takich przykładów jest prezentowany timer.

Cyfrowe lustrzanki firm Pentax i Canon są wyposażane w identyczne gniazdo do podłączenia „wężyka" spustowego. Niegdyś wężyk spustowy był kawałkiem giętkiej linki w pancerzu. Aktualnie, w aparatach wspomnianych producentów, „wężyk" spełnia podobną funkcję, ale działa na zasadzie zwierania obwodów sterujących wbudowanych w aparat – **rysunek 1**. Sterowanie aparatem polega na zwieraniu styków wtyku do wspólnego przewodu masy. Jeżeli jest ustawiony tryb wyzwalania B (*Bulb*), to zwarcie na stałe przez przełącznik styku sterowania migawką powoduje podniesienie lustra i otwarcie migawki na tak długo, jak jest zwarty. We wszystkich innych trybach pracy aparatu zwarcie styku sterowania migawką powoduje jednorazowe wyzwolenie migawki na czas określony przez program aparatu. Zwarcie styku sterowania ostrzeniem powoduje uruchomienie automatycznego wyostrzenia (*Auto Focus*).

Gniazdo w aparacie jest przystosowane do standardowego stereofonicznego wtyku mikro jack, więc wykonanie takiego elektronicznego „wężyka" jest łatwe i tanie. Po kilku próbach ze sterowaniem aparatem postanowiłem ten prosty układ trochę rozbudować. Pierwsza myśl, która przyszła mi do głowy, to sterowanie w trybie „B" przez zadany czas. Czas otwarcia migawki ustala fotograf i w przypadku zwykłego zwierania styków przez przełącznik musi się on posłużyć stoperem, tak aby po określonym czasie rozewrzeć styki i zakończyć naświetlanie. Żeby ułatwić sobie zadanie, można zwieranie i rozwieranie styków zlecić programowanemu czasomierzowi. Ustawiamy żądany czas, starujemy odliczanie, a resztę wykona czasomierz. Można by też wykonywać serię ujęć co pewien czas, na przykład co pół godziny.

Do odliczania ustawionego czasu najlepiej zaprząc sterownik mikroprocesorowy z wyświetlaczem i jakimś elementem ma-

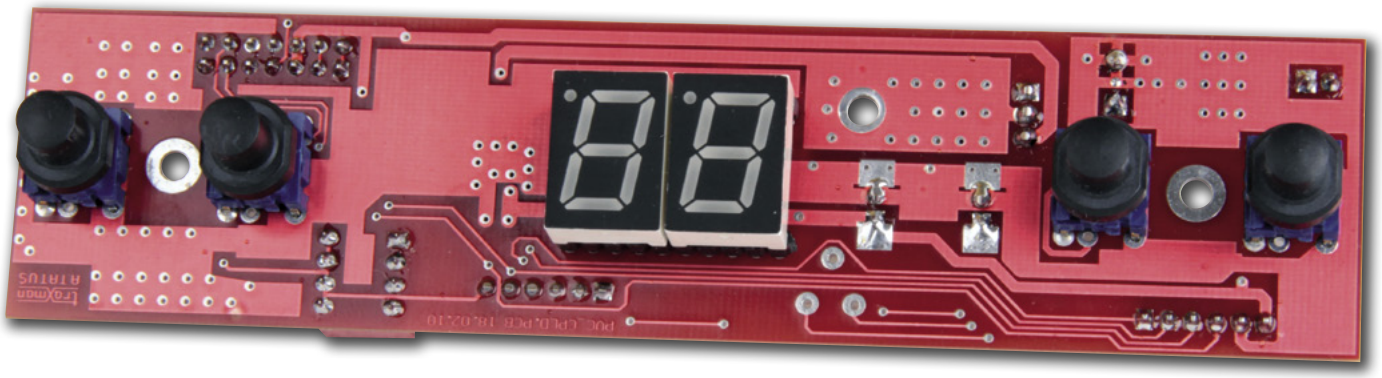
AVT-5296 w ofercie AVT:
AVT-5296A – płytka drukowana

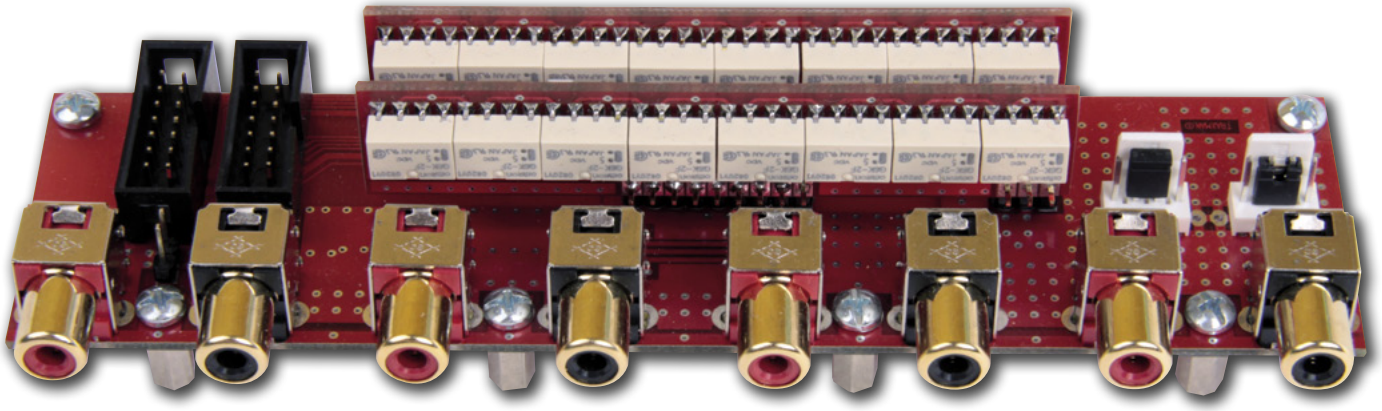
Podstawowe informacje:
- Sterowanie lustrzankami Canon i Pentax.
- Pełni funkcję timera oraz układu zdalnego wyzwalania
- Sterowanie za pomocą fal radiowych lub nadajnika podczerwieni
- Zasilanie timera i nadajnika z 4 akumulatorków AA
- Zasięg praktyczny sterowania radiowego ok. 30 m

Dodatkowe materiały na CD/FTP:
ftp://ep.com.pl, user: 10925, pass: 87thc181
- wzory płytek PCB
- karty katalogowe i noty aplikacyjne elementów oznaczonych w Wykazie elementów kolorem czerwonym

Projekty pokrewne na CD/FTP:
(wymienione artykuły są w całości dostępne na CD)

AVT-5221	DSLR shutter – Elektroniczny wężyk do aparatu fotograficznego (EP 2/2010)
AVT-5202	Pilot do zdalnego sterowania lustrzankami cyfrowymi (EP 9/2009)
AVT-427	Uniwersalny regulator temperatury dla fotografików (EP 6/1998)
AVT-346	Fotograficzna lampa ciemniowa (EP 11/1997)
---	Timer z licznikiem naświetleń (EP 4/2002)
---	Prosty minutnik fotograficzny (EP 2/1997)





Pasywny regulator siły głosu PVC

AVT 5297

Wśród miłośników słuchania muzyki z najwyższą jakością istnieje grupa zwolenników konstrukcji minimalistycznych, w których liczba elementów jest ograniczona do absolutnego minimum, a tor audio składa się tylko ze źródła, regulatora głośności, wzmacniacza mocy i zestawów głośnikowych. Niniejszy projekt to opis propozycji wykonania regulatora siły głosu przeznaczonego do takiego toru audio.

Rekomendacje: *audiofile, konstruktorzy wzmacniaczy wysokiej jakości.*

Układ regulacji siły głosu w opisanym we wstępie torze nie może opierać się na rozwiązaniach półprzewodnikowych, jak np. popularne układy z serii PGAxxxx. Stąd na rynku pojawiła się oferta układów określanych skrótem PVC (*Passive Volume Control*) wypierających klasyczne potencjometry mechaniczne, których największą wadą jest problem ze współbieżnością kanałów stereofonicznych szczególnie przy położeniach bliskich najmniejszej głośności oraz stopniowe zużywanie się ścieżki oporowej.

Układy PVC, w zależności od wykonania, oparte są na potencjometrze, w którym ślizgacz i ścieżka oporowa zostały zastąpione przełącznikiem wielopozycyjnym wysokiej jakości z odpowiednio dobraną drabinką rezystorową. Schemat najprostszego układu PVC zamieszczono na **rysunku 1**. Wielopozycyjny przełącznik obrotowy ma najczęściej od 24 do 36 pozycji, co pokrywa cały konieczny zakres regulacji głośności, przy komfortowym dla słuchacza kroku regulacji. Poszczególne pozycje oraz właściwą charakterystykę regulacji układu zapewniają rezystory precyzyjne. Ich wartości są nietypowe (spoza typoszeregu), szczególnie gdy układ ma logarytmiczną charakterystykę regulacji. Niestety, wadą takiego rozwiązania są: jego wysoka cena, ograniczona dostępność

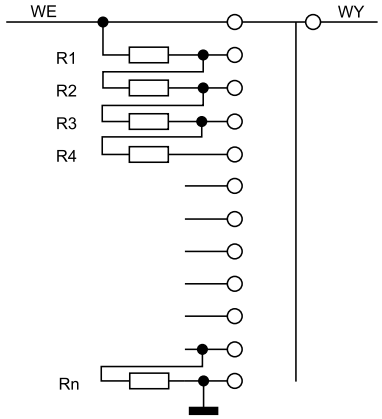
AVT-5297 w ofercie AVT:
AVT-5297A – płytka drukowana

Podstawowe informacje:
- Rezystancja wejściowa 25,5 kΩ
- Zasilanie 9...12 V_{AC}
- Sterowanie przez układ CPLD Xilinx XC9572XL
- 64 poziomy głośności
- Logarytmiczna charakterystyka regulacji
- Wejście/wyjście stereofoniczne
- Dwa wyświetlacze 7-segmentowe wyświetlające numer poziomu głośności
- Wyposażony w selektor 4 źródeł sygnału stereofonicznego

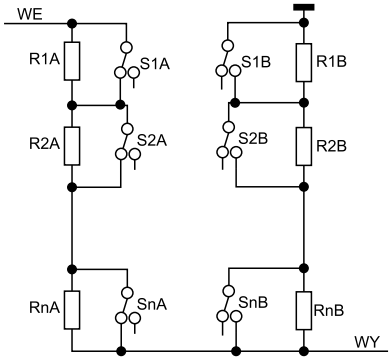
Dodatkowe materiały na CD/FTP:
ftp://ep.com.pl, user: 10925, pass: 87thc181
- wzory płytek PCB
- karty katalogowe i noty aplikacyjne elementów oznaczonych w Wykazie elementów kolorem czerwonym

Projekty pokrewne na CD/FTP:
(wymienione artykuły są w całości dostępne na CD)

AVT-5283	Potencjometr audio (EP 3/2011)
AVT-5267	Lampowy potencjometr siły głosu (EP 12/2010)
AVT-5237	Potencjometr z impulsatorem (EP 6/2010)
AVT-5206	Cyfrowy potencjometr audio (EP 10/2009)
AVT-5185	Volumer – Elektroniczny potencjometr audio (EP 5/2009)
AVT-945	Audiofilski potencjometr i regulator balansu (EP 8/2006)
AVT-5027	Audiofilski potencjometr elektroniczny (EP 9/2001)



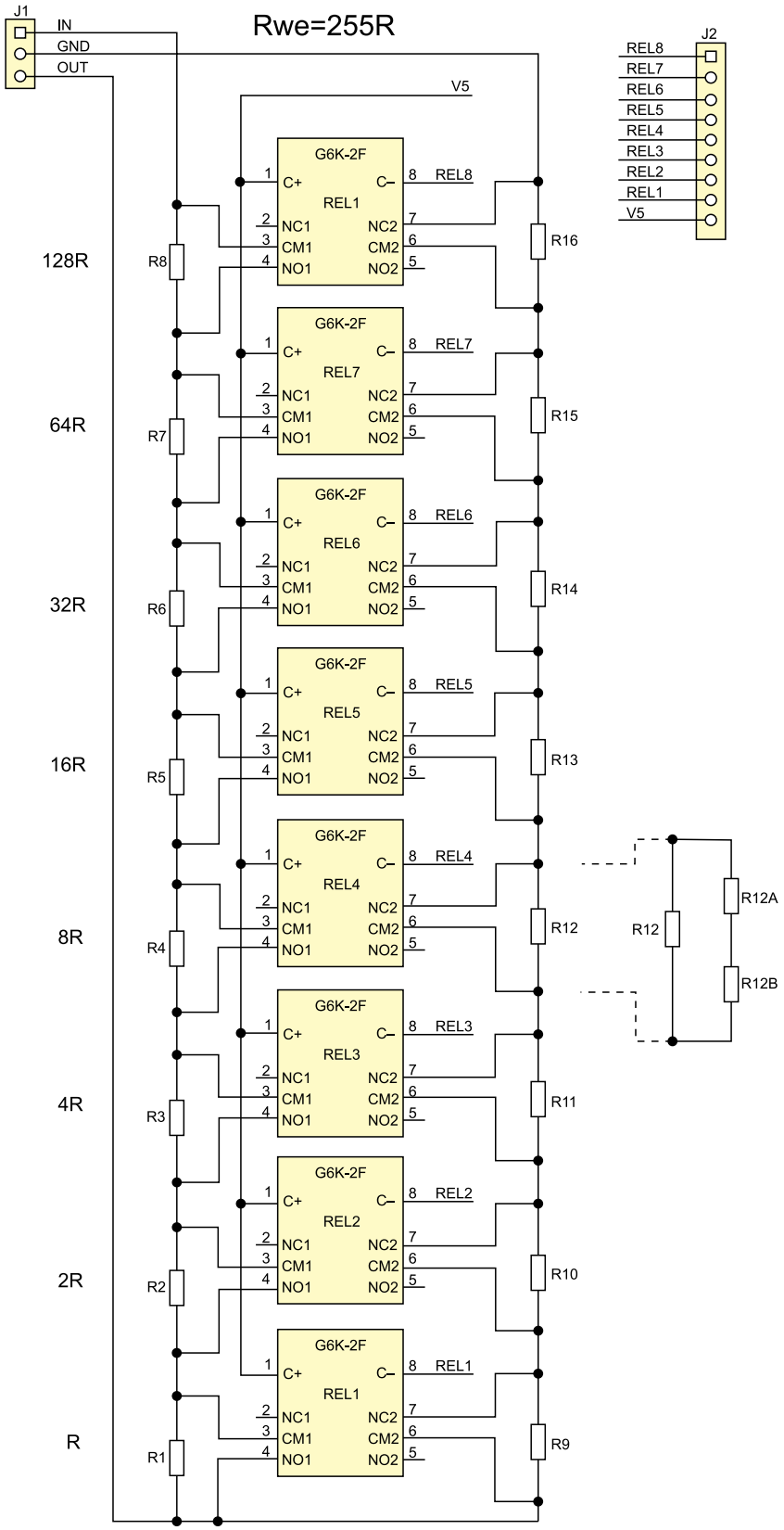
Rysunek 1. Typowy układ pasywnej regulacji głośności PVC



Rysunek 2. Zmodyfikowany układ pasywnej regulacji głośności PVC

przełącznika wielopozycyjnego oraz kłopoty z zakupem rezystorów precyzyjnych o niestandardowych rezystancjach.

Po modyfikacji układu z rysunku 1 z prostego potencjometru na układ drabinki R...n×R można zmniejszyć wymaganą liczbę rezystorów oraz stosować standardowe wartości rezystancji z szeregu. Schemat zmodyfikowanego układu PVC przedstawiono na **rysunku 2**. Układ ma jednak tę wadę, że każda pozycja regulacji wymaga podwójnego styku NO+NC. Jest to rekompensowane mniejszą całkowitą liczbą potrzebnych styków potrzebnych do realizacji regulacji oraz zmniejszoną liczbą rezystorów dzielnika. Wprowadzając dalsze modyfikacje w postaci zmiany sterowania mechanicznego na elektroniczne i zastąpienia styków przełącznika obrotowego przekaźnikami, można drabinkę uprościć do postaci R-2R...n×R, co zostało zastosowane w modelu urządzenia. Schemat takiej drabinki przedstawia **rysunek 3**. Sumaryczna wartość rezystancji widzianej od wejścia to 255×R (R1). Na rysunku nie przedstawiono wartości rezystorów ze względu na możliwość stosowania wartości najbardziej odpowiadających wymaganej wartości potencjometru. W modelu zastosowano R1=R9=100 Ω, R2=R10=200 Ω, ..., R8=R16=12,8 kΩ, co daje sumaryczną wartość rezystancji 25,5 kΩ. Taki potencjometr nadaje się zarówno do współpracy ze wzmacniaczami lampowymi, jak i tranzystorowymi. Oczywiście nic nie stoi na przeszkodzie, aby dobrać rezystancje w taki sposób, aby osiągnąć sumaryczną rezystancję wynoszącą 10...100 kΩ. Aby ułatwić dobranie rezystorów, każdy z nich został na płytce zaprojektowany jako połączenie szeregowo-równoległe trzech elementów, jak przedstawiono na przykładzie R12. Ułatwia to dokładny dobór rezystancji elementów dzielnika, co jest szczególnie istotne w układzie regulacji stereofonicznej.



Rysunek 3. Schemat ideowy układu pasywnej regulacji głośności PVC_G6K

W charakterze przełącznika zastosowano nowoczesne, miniaturowe przekaźniki G6K-2F przeznaczone do montażu SMD, zasilane napięciem 5 V, co owocuje zmniejszeniem wymiarów drabinki oraz redukcją poboru

Montaż

Montaż układu nie wymaga specjalnego opisu. Należy tylko zwrócić uwagę na polaryzację wbudowanych w przyciski diod LED. Połączenie modułów sterownika i selektora wykonane jest z taśmy przewodów 14×1,25 mm o odpowiedniej długości z zaciśniętymi trzema wtykami żeńskimi IDC14, tak aby uzyskać połączenie 1-1...14-14.

Zmontowany prawidłowo układ wymaga zaprogramowania. W tym celu odłączamy płytkę selektora, do złącza J3-JTAG podłączamy programator i przy wykorzystaniu programu *Impact*, po włączeniu zasilania, programujemy układ plikiem *PVC.JED*. Wyłączamy zasilanie, odłączamy programator, podłączamy płytkę selektora i po włączeniu zasilania sprawdzamy funkcjonowanie układu.

Adam Tatuś
atatus@op.pl

REKLAMA

AVT5094
Bezprzewodowy regulator temperatury

Urządzenie mogące zastąpić przestarzałe regulatory temperatury z wyjściami przekaźnikowymi. Pomiar temperatury odbywa się z rozdzielczością 0,1°C, a temperatura zadana może być ustawiana z dokładnością 1°C w zakresie od 0 do 99°C.

- pomiar temperatury z rozdzielczością 0,1°C
- niezależne nadzorowanie temperatury dla dnia oraz dla nocy
- pory dnia są definiowane przez użytkownika
- temperatura zadana może być ustawiana z dokładnością 1°C w zakresie od 0°C do 99°C
- możliwość pracy ręcznej
- bezprzewodowa transmisja radiowa (zasięg ok. 300m)
- sterowanie obciążeniem o mocy do 3,5kW

www.sklep.avt.pl

Listing 1. Komponent CNT_VOL64B

```
-------------------------------------------------
-- licznik binarny 0-63 dwukierunkowy z resetem
-- ograniczeniem zliczania <0:63>
-- i wpisem wartosci poczatkowej
-- Adam Tatus traxman - PCV.prj regulacja glosnosci
-------------------------------------------------
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity CNT_VOL64B is
  port(
    Reset:   in std_logic;  -- reset licznika
    Clock:   in std_logic;  -- wejscie sygnalu zegarowego
    Dir:     in std_logic;  -- sygnal kierunku zliczania
    PSetVal: in std_logic_vector(5 downto 0);  -- wartosc wpisu poczatkowego
    PSet:    in std_logic;  -- ustawianie wartosci poczatkowej
    CkOut:   out std_logic_vector(5 downto 0) );  -- wyjscie licznika
end CNT_VOL64B;

architecture Behavioral of CNT_VOL64B is
  signal temp: std_logic_vector(5 downto 0);
  begin
  process(Clock, Reset)
  begin
    if Reset='1' then temp <= „000000";  -- reset licznika „0"
    elsif ( Clock'event and Clock='1') then
      if PSet='1' then temp <= PSetval;  -- licznika z PSetVal
      elsif (PSet='0' and Dir='0') then  -- Dir=0, zliczanie w gore
        if temp="111111" then  -- ograniczenie zliczania do 63
        temp<="111111"; else temp <= temp + 1;
      end if;
      elsif (PSet='0' and Dir='1') then  -- zliczanie w dol
        if temp="000000" then  -- ograniczanie zliczania do 0
          temp<="000000"; else temp <= temp - 1;
        end if;
      end if;
    end if;
  end process;
  CkOut <= temp;  -- przepisanie zmiennej temp na wyjscie licznika
end Behavioral;
```

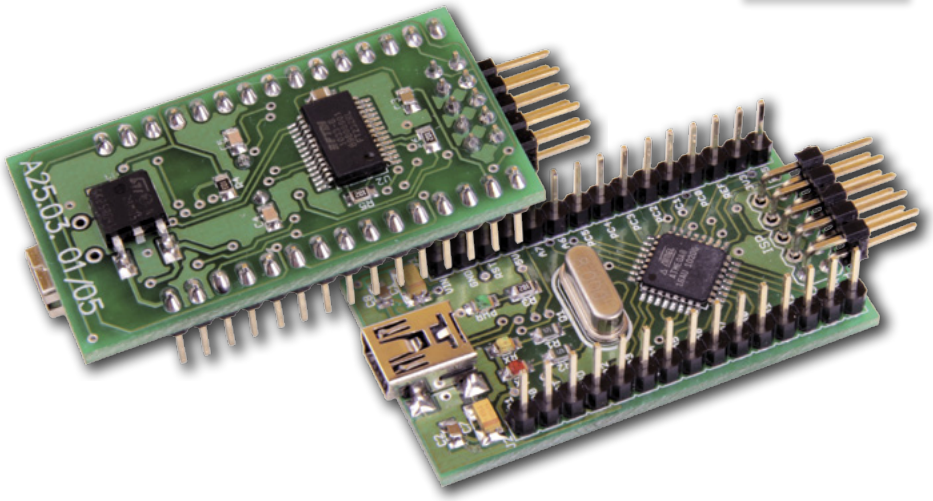
Listing 2. Komponent REL_LUT

```
-------------------------------------------------
-- dekoder przekaznikow LED 0-63
-- Adam Tatus traxman - PCV.prj
-------------------------------------------------
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity REL_LUT is
  port (
   REL_IN: in std_logic_vector (5 downto 0);    -- poziom glosnosci VOL do
zdekodowania
   REL_OUT: out std_logic_vector (7 downto 0)); -- sterowanie przekaznikami wg
LUT
end REL_LUT;

architecture arch of REL_LUT is
  begin
    with REL_IN select
    REL_OUT <= „00000000" when „000000",  -- „ min"
      „00000001" when „000001", „00000010" when „000010",
      „00000011" when „000011", „00000100" when „000100",
      „00000101" when „000101", „00000110" when „000110",
      „00000111" when „000111", „00001000" when „001000",
      „00001001" when „001001", „00001010" when „001010",
      „00001011" when „001011", „00001100" when „001100",
      „00001101" when „001101", „00001110" when „001110",
      „00001111" when „001111",  -- „do 15 co 1"
      „00010000" when „010000", „00010010" when „010001",
      „00010100" when „010010", „00010110" when „010011",
      „00011000" when „010100", „00011010" when „010101",
      „00011100" when „010110", „00011110" when „010111",
      „00100000" when „011000", „00100010" when „011001",
      „00100100" when „011010", „00100110" when „011011",
      „00101000" when „011100", „00101010" when „011101",
      „00101100" when „011110",
      „00101110" when „011111", -- „do 31 co 2"
      „00110000" when „100000", „00110100" when „100001",
      „00111000" when „100010", „00111100" when „100011",
      „01000000" when „100100", „01000100" when „100101",
      „01001000" when „100110", „01001100" when „100111",
      „01010000" when „101000", „01010100" when „101001",
      „01011000" when „101010", „01011100" when „101011",
      „01100000" when „101100", „01100100" when „101101",
      „01101000" when „101110",
      „01101100" when „101111",  -- „do 47 co 4"
      „01110100" when „110000", „01111100" when „110001",
      „10000100" when „110010", „10001100" when „110011",
      „10010100" when „110100", „10011100" when „110101",
      „10100100" when „110110", „10101100" when „110111",
      „10110100" when „111000", „10111100" when „111001",
      „11000100" when „111010", -- „do 58 co 8"
      „11001100" when „111011", -- „co 9"
      „11010101" when „111100", -- „co 10"
      „11011111" when „111101", -- „co 16"
      „11101111" when „111110", -- „co 16"
      „11111111" when „111111",  -- „max"
      „00000000" when others;    -- „min"
end arch;
```

Minimoduł z ATmega8

AVT 1622

Prezentowany układ to niewielki moduł z mikrokontrolerem ATmega8. Takiego miniaturowego modułu można użyć jako uniwersalnego sterownika w dowolnym, konstruowanym przez siebie urządzeniu. Dodatkowo, moduł został wyposażony w konwerter UART/USB typu FT232R, umożliwiający komunikację mikrokontrolera z komputerem PC.



Schemat elektryczny modułu pokazano na **rysunku 1**. Wszystkie wyprowadzenia mikrokontrolera zostały dołączone do złącz umieszczonych na krawędziach płytki drukowanej. Minimoduł może być zasilany z doprowadzenia 1 (VIN) złącza J1. Napięcie wejściowe jest podawane na stabilizator U3 (78M05) dostarczający napięcie +5 V zasilające cały moduł. Napięcie to jest dostępne na nóżce 4 złącza J2 i może posłużyć do zasilania układów zewnętrznych. Moduł może być także zasilany również z USB. Dioda świecąca PWR sygnalizuje załączenie zasilania modułu.

Do mikrokontrolera dołączono kwarc Q1, którego częstotliwość można dobrać do potrzeb aplikacji. Programowanie układu odbywa się za pomocą typowego, 10-wyprowadzeniowego złącza kompatybilnego z STK200, dzięki czemu do programowania można użyć dowolnego programatora dla AVR.

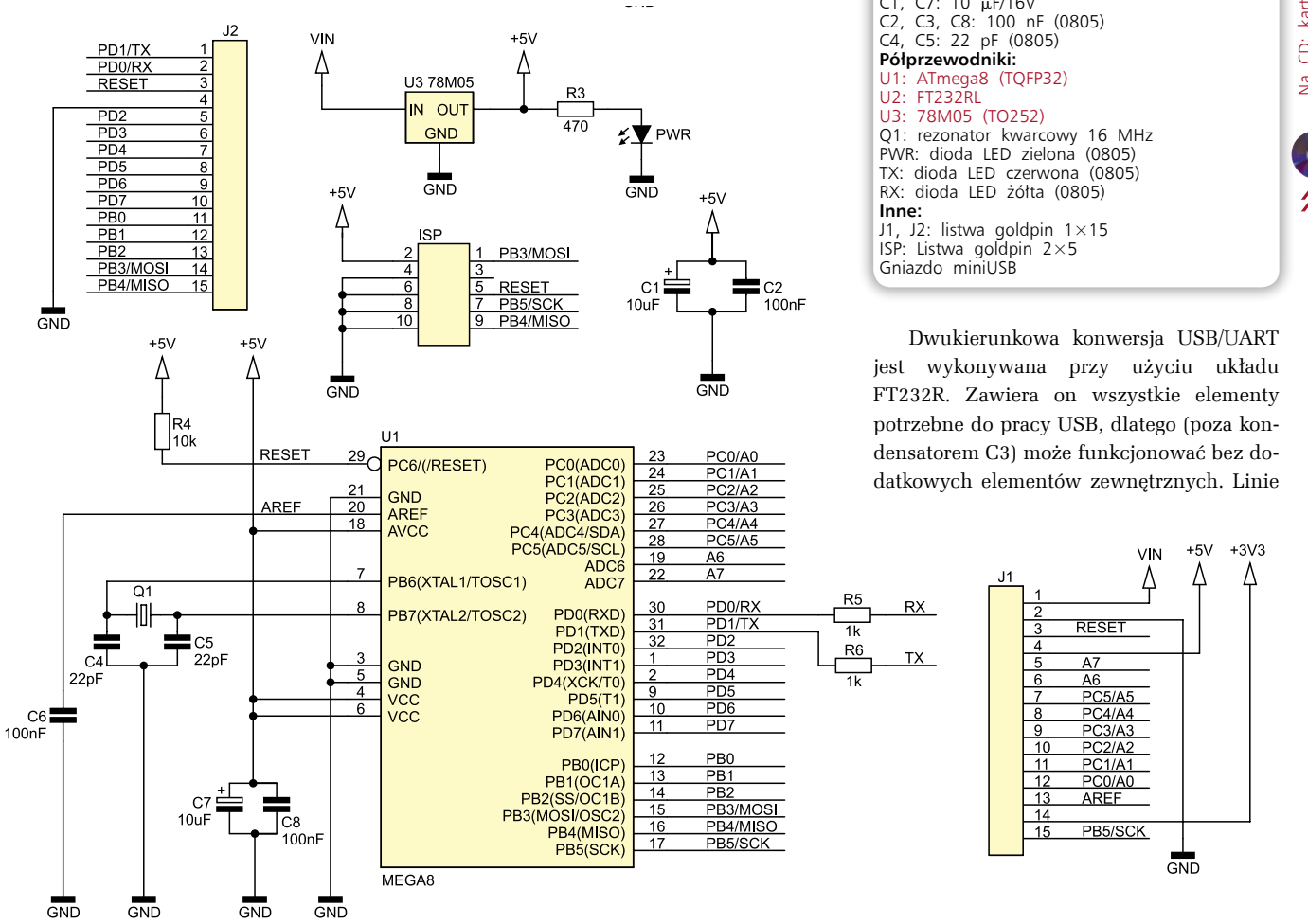
AVT-1622 w ofercie AVT:
AVT-1622A – płytka drukowana
AVT-1622B – płytka drukowana + elementy

Dodatkowe materiały na CD/FTP:
ftp://ep.com.pl, user: 10925, pass: 87thc181
- wzory płytek PCB
- karty katalogowe i noty aplikacyjne elementów oznaczonych w Wykazie elementów kolorem czerwonym

Wykaz elementów

Rezystory:
R1...R3: 470 Ω (0805)
R4: 10 kΩ (0805)
R5, R6: 1 kΩ (0805)
Kondensatory:
C1, C7: 10 µF/16V
C2, C3, C8: 100 nF (0805)
C4, C5: 22 pF (0805)
Półprzewodniki:
U1: ATmega8 (TQFP32)
U2: FT232RL
U3: 78M05 (TO252)
Q1: rezonator kwarcowy 16 MHz
PWR: dioda LED zielona (0805)
TX: dioda LED czerwona (0805)
RX: dioda LED żółta (0805)
Inne:
J1, J2: listwa goldpin 1×15
ISP: Listwa goldpin 2×5
Gniazdo miniUSB

Na CD: karty katalogowe i noty aplikacyjne elementów oznaczonych w wykazie elementów kolorem czerwonym



Rysunek 1. Schemat ideowy minimodułu z ATmega8

Dwukierunkowa konwersja USB/UART jest wykonywana przy użyciu układu FT232R. Zawiera on wszystkie elementy potrzebne do pracy USB, dlatego (poza kondensatorem C3) może funkcjonować bez dodatkowych elementów zewnętrznych. Linie

zasilany napięciem z szerokiego zakresu. Jest on zasilany napięciem symetrycznym, które można pobrać bezpośrednio z zasilacza końcówki mocy. Różne wzmacniacze maja różne napięcia zasilania i dlatego należy odpowiednio dobrać rezystory R1 i R2. Wartości rezystorów dla przykładowych napięć zasilania zamieszczono w **tabeli 1**. Warto zwrócić uwagę, że w torach sygnałowych nie zastosowano żadnych kondensatorów, dzięki czemu pasmo przenoszenia układu zaczyna się już od 0 Hz i nie ma wpływu na sumaryczne pasmo przenoszenia wzmacniacza.

Schemat montażowy przedstawiono na **rysunku 2**. Płytkę wykonano na laminacie jednostronnym. Układ jest zbudowany tylko z kilku elementów przewlekanych. Po zmontowaniu symetryzator jest gotowy do pracy, ale zalecane jest zmierzenie, najlepiej za pomocą oscyloskopu, czy oba sygnały wyjściowe maja tę samą amplitudę. Jeśli nie, to należy odpowiednio dobrać wartości rezystorów R5 i R6. Symetryzator dołączamy do wzmacniacza zgodnie z **rysunkiem 3**.

Układ współpracuje z tylko tymi wzmacniaczami, które nie są zbudowane jako mostkowe. Najczęściej w takich wzmacniaczach masa sygnału wejściowego i wyjściowego są ze sobą połączone galwanicznie. Impedancja głośnika powinna być dwukrotnie większa, niż zalecana dla pojedynczego kanału. Jeżeli wzmacniacz współpracuje z dwoma kolumnami 4 Ω, to po zmianie konfiguracji na mostkową zastosowany głośnik powinien mieć impedancję 8 Ω lub większą i być podłączony pomiędzy wyjścia kanałów lewego i prawego z pominięciem masy.

AW

AVTduino JOY – manipulator dla Arduino

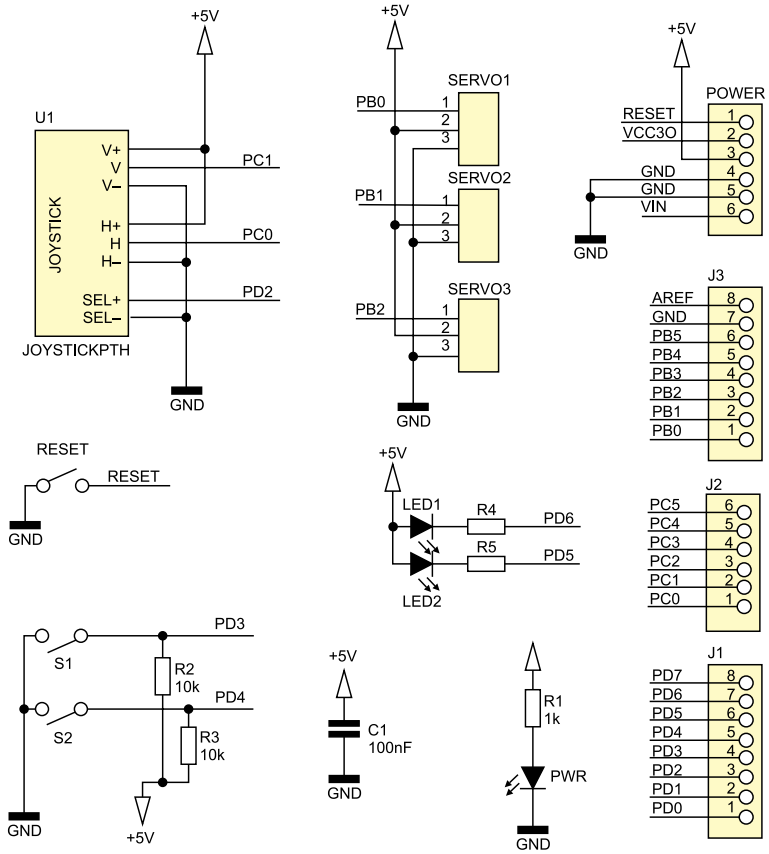
AVT 1618

Moduł jest nakładką na płytkę bazową projektu AVTduino (AVT5272). Został wyposażony został w joystick, dwa przyciski oraz złącza umożliwiające dołączenie trzech serwomechanizmów. Dzięki takiej konfiguracji jest możliwe zbudowanie prostego manipulatora sterowanego z platformy AVTduino.

AVTduino kompatybilne z ARDUINO

Schemat ideowy modułu pokazano na **rysunku 1**, natomiast montażowy na **rysunku 2**. W roli joysticka U1 zastosowano gotowy, zintegrowany, liniowy, dwuosiowy joystick, którego wyprowadzenia dołączone zostały do portów PC0 (ADC0) i PC1 (ADC1). Przycisk zintegrowany z joystickiem został doprowadzony bezpośrednio do portu PD2, natomiast sygnały z dwóch dodatkowych przycisków S1 i S2 do portów PD3 i PD4. Poziomem aktywnym dla wszystkich przycisków jest logiczne „0". Złącza oznaczone jako SERVO1...SERVO3 umożliwiają bezpośrednie dołączenie do płytki serwomechanizmów modelarskich. Impulsy sterujące ich pracą powinny pojawić się na portach PB0... PB2. Diody LED1 i LED2 mogą pełnić rolę sygnalizatorów, a sterowane są z portów PD5 i PD6. Dioda LED PWR informuje o obecności napięcia zasilania płytki manipulatora.

EB

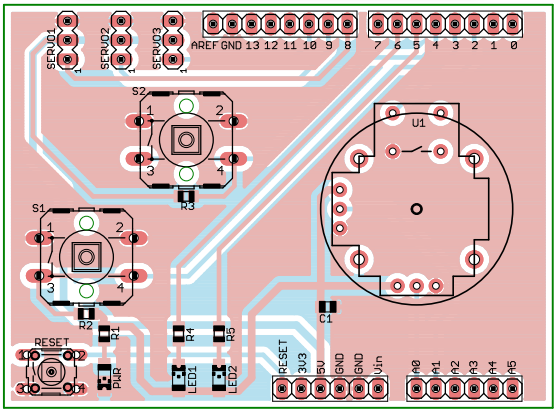


Rysunek 1. Schemat ideowy modułu joysticka do Avtduino

AVT-1618 w ofercie AVT:
AVT-1618A – płytka drukowana
AVT-1618B – płytka drukowana + elementy

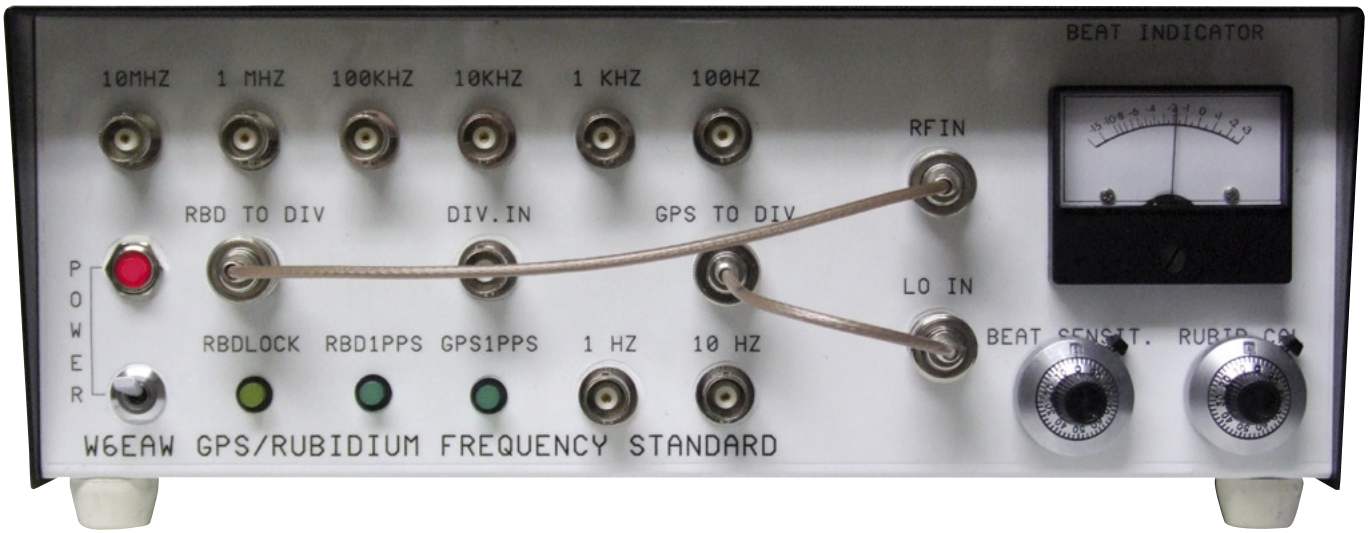
Dodatkowe materiały na CD/FTP:
ftp://ep.com.pl, user: 10925, pass: 87thc181
- wzory płytek PCB
- karty katalogowe i noty aplikacyjne elementów oznaczonych w Wykazie elementów kolorem czerwonym

Wykaz elementów
R1, R4, R5: 1 kΩ (SMD 0805)
R2, R3: 10 kΩ (SMD 0805)
C1: 100 nF (SMD 0805)
U1: potencjometr-joystick JV1603N-B10K
PWR, LED1, LED2: dioda LED (1206)
S1, S2: przycisk mikroswitch 12 mm×12 mm
RESET: przycisk mikroswitch 1 mm
J1...J3, POWER: listwa goldpin



Rysunek 2. Schemat montażowy modułu joysticka do Avtduino

Dział „Projekty Czytelników” zawiera opisy projektów nadesłanych do redakcji EP przez Czytelników. Redakcja nie bierze odpowiedzialności za prawidłowe działanie opisywanych układów, gdyż nie testujemy ich laboratoryjnie, chociaż sprawdzamy poprawność konstrukcji.
Prosimy o nadsyłanie własnych projektów z modelami (do zwrotu). Do artykułu należy dołączyć podpisane oświadczenie, że artykuł jest własnym opracowaniem autora i nie był dotychczas nigdzie publikowany. Honorarium za publikację w tym dziale wynosi 250,– zł (brutto) za 1 stronę w EP. Przysyłanych tekstów nie zwracamy. Redakcja zastrzega sobie prawo do dokonywania skrótów.



Rubidowy, korygowany GPS-em wzorzec częstotliwości i czasu

Projekt 194

Od zarania radiokomunikacji istniał problem generowania sygnału o stabilnej częstotliwości i jej pomiaru. Wielki skok, jaki dokonał się w technologii półprzewodnikowej, wzrost skali integracji i postęp w wielu dziedzinach nauki dają niemal idealne narzędzia rozwiązujące wspomniane wcześniej problemy.

Urządzenia używane w amatorskiej radiokomunikacji są wyposażone w syntezery DDS zapewniające dużą dokładność i stabilność częstotliwości, pokrycie bardzo szerokiego pasma częstotliwości, małą zawartość szumów fazowych itp. W dalszym ciągu jednak ich główny element – zegar kwarcowy „ożywiający” DDS – podlega powolnemu starzeniu się płytki kwarcowej i innych komponentów, co zmusza do okresowej kalibracji częstotliwości za pomocą odpowiednio dokładnego wzorca.

Stacje radiowe (jak np. WWV) nadające sygnały o częstotliwościach 5, 10 i 15 MHz wystarczają do zgrubnej kalibracji urządzeń, chociaż ich użycie wiąże się nieraz z koniecznością przezwyciężenia problemów z propagacją, zakłóceniami sygnału radiowego itp. Stara metoda „dudnieniowa” stwarza kłopoty przy kalibracji podczas emisji SSB, ponieważ druga wstęga jest wycinana i przy dużej stromości filtru kwarcowego słyszalność bardzo małej częstotliwości jest praktycznie niemożliwa.

Najlepsze obecnie wzorce atomowe – wodorowe i cezowe są na ogół poza zasięgiem radioamatora. Ale radioamator może mieć do nich dostęp za pośrednictwem systemu satelitów GPS. Wspomnę tu o artykule autorstwa SQ4AVS i SP2IQW zamieszczonym w „Świecie Radio” w styczniu 2009, w którym opisano budowę wzorca częstotliwości i czasu z użyciem odbiornika GPS.

Zamieszczony w miesięczniku „QEX” artykuł Boba KE6F pt. *Atomic Frequency reference for your shack*, sprowokował mnie do budowy własnego wzorca z użyciem łatwo-dostępnego na portalu *www.e-bay.com* generatora rubidowego. Podczas konstrukcji zdecydowałem się dodać także wzorzec z odbiornikiem GPS *Thunderbolt Precision GPS Standard*. Obydwa wzorce są oferowane najczęściej przez chińskich sprzedawców na aukcjach z opcją „kup teraz”. Płaci się za pomocą serwisu PayPal, który ubezpiecza kupującego. Sprzedawcy są bardzo solidni, dostarczają wszystkie potrzebne informacje.

Podczas wyszukiwania ofert wystarczy w oknie wyszukiwarki na stronie *www.ebay.com* wpisać *EFRATOM 10 MHz rubidium frequency standard*, a pojawi się kilkadziesiąt ofert w cenach od 50 do 100 dolarów. Wzorzec typu FRS-C jest tańszy, bo wymaga specjalnego wtyku, który trudno kupić. Model LPRO-101 jest nieco droższy, ale łatwiejszy w montażu (**fotografia 1**). Przypuszczalnie te urządzenia pracowały w telefonii komórkowej w Chinach przez około 5 lat i po wymianie na nowsze modele są sprzedawane z tzw. demobilu po bardzo niskiej cenie (w porównaniu do ceny nowego zegara rubidowego, która wynosi około 950 dolarów).

Czytelnik zdecydowany na nieco większy wydatek może nabyć na e-bay *Thunder-*

Kurs programowania mikrokontrolerów PIC (3)

Interfejs użytkownika

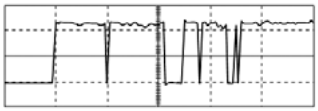
Interfejs użytkownika jest przeznaczony do komunikacji pomiędzy osobą obsługującą urządzenie a obsługiwanym urządzeniem (w tym przypadku sterownikiem mikroprocesorowym). Interfejs musi dawać użytkownikowi możliwość wprowadzania parametrów i poleceń. Najprostszym urządzeniem wejściowym jest klawiatura. Naciskając klawisze uruchamiany realizację zadania, wprowadzamy parametry np. godzinę i datę, dokonujemy wyboru funkcji z menu itp. Sterownik używa do sygnalizowania swojego stanu różnych elementów: diod LED, wyświetlaczy, miniaturowych głośników itp. W tym odcinku zajmiemy się się głównie obsługą komponentów menu użytkownika.

Dodatkowe materiały na CD/FTP:
ftp://ep.com.pl, user: 10925, pass: 87thc181
- poprzednie części kursu

Klawiatura

Klawisze ze stykami zwiernymi są jednymi z podstawowych elementów sterowników. Wydawałoby się, że nie ma nic prostszego niż wykrycie zwartego styku. Ale jak to zwykle bywa, diabeł tkwi w szczegółach. W czasie zwierania kontaktów na skutek zjawiska sprężystości odbijają się one od siebie, co jest powodem powstawania krótkotrwałych przerw w obwodzie, w czasie których obwód elektryczny jest zamykany i otwierany. Zależnie od rodzaju przycisku to zjawisko może trwać nawet do 20 ms od pierwszego zwarcia. Jeżeli nasz mikrokontroler jest taktowany sygnałem o częstotliwości 4 MHz, to rdzeń będzie taktowany z częstotliwością 4 MHz/4=1 MHz. Rozkaz testowania styków wykona się w czasie 1/1 MHz= 1 ms. Jak widać mikrokontroler może testować stan wejścia, do którego jest dołączony styk szybciej niż – nawet kilkakrotnie w czasie trwania drgań. Może to powodować (i najczęściej powoduje) błędną interpretację stanu przycisku.

Najczęściej spotykanym sposobem radzenia sobie z tym problemem jest programowa eliminacja zakłóceń. Załóżmy, że program ma określić czy styk jest zwarty, czy nie. Po wykryciu zmiany stanów na którejś z linii klawiatury odmierzamy opóźnienie ok. 20...30 ms i po tym czasie ponownie odczytujemy stan klawiatury. Mamy wtedy pewność, że ten stan jest stabilny.



Rysunek 1. Przebiegi na linii wejściowej po naciśnięciu przycisku

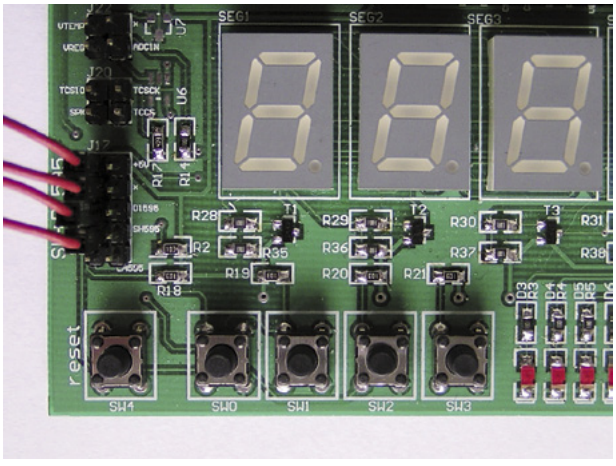
Zwykle styki klawiatury zwierają linie portów mikrokontrolera do masy i jednocześnie te linie są połączone do plusa zasilania przez rezystory podciągające. Jeżeli styk jest rozwarty, to na wejściowej linii portu jest stan wysoki wymuszony przez rezystor. Po zwarciu styku na linii jest stan niski. Taki układ z 4 stykami jest zabudowany w module ewaluacyjnym (**rysunek 2**). Wyprowadzenia SW0...SW3 są połączone z pinami złącza J17 i trzeba je połączyć przewodami z liniami portu mikrokontrolera. Na fotografii 3 pokazano umiejscowienie na płytce styków SW1...SW3 i złącza J17 (z wetkniętymi kablami).

Obsługę klawiatury można wykonywać na kilka sposobów. Procedura czytania może zwracać kod przyciśniętego klawisza po jednokrotnym naciśnięciu i puszczeniu klawisza. Taki sposób jest wygodny w wielu zastosowaniach, ale czasami trzeba aby kod przyciśniętego klawisza był powtarzany co zadany czas tak długo, jak styk jest zwarty (na przykład przy ustawianiu zegara). Na początek zajmijmy się pierwszym przypadkiem.

Obsługę klawiatury można podzielić na kilka etapów:

1. Czekanie na zwolnienie przycisku. Właściwa procedura czytania stanu styków może rozpocząć się tylko wtedy, gdy żaden z klawiszy nie jest wciśnięty.
2. Czekanie na wciśnięcie klawisza.
3. Po stwierdzeniu, że jeden z przycisków jest wciśnięty zostaje uruchomione odliczanie opóźnienia eliminującego możliwość odczytania stanu styków podczas ich drgań.
4. Po odliczeniu czasu ponownie jest sprawdzany stan linii portów dołączonych do styków. Jeżeli żaden styk nie jest zwarty, to program wraca do punktu 2 lub kończy działanie z informacją, że żaden ze styków nie został wciśnięty. Jeżeli styk jest zwarty, to określamy jego kod (numer).
5. Opcjonalnie, czekamy na zwolnienie klawisza.

Do prób z klawiaturą połączymy SW0... SW3 z liniami RA0...RA3 portu PORTA mikrokontrolera. Poprawność działania procedury odczytu klawiatury będziemy sygnalizować za pomocą zaświecenia lub zgaszenia diod LED LD0...LD3 połączonych z liniami RB0...RB3 portu PORTB. Do odliczania opóźnień zostanie wykorzystana znana już procedura delay wykorzystująca przerwanie od przepełnienia licznika Timer0. Linie RA0...



Fotografia 2. Klawiatura modułu ewaluacyjnego

Listing 1. Procedura odczytania stanów klawiszy SW0...SW3

```
unsigned char GetKey(void){
unsigned char key;
  while((PORTA&0x0f)!=0x0f);       //czekaj na puszczenie klawisza
  while((PORTA&0x0f)==0x0f);       //czekaj na przyciśnięcie klawisza
  delay(25);                       //odczekanie drgań styków
  key=~PORTA;                      //aktywne jedynka
  key=key&0x0f;                    //tylko 4 młodsze bity
  while((PORTA&0x0f)!=0x0f);       //czekaj na puszczenie klawisza
  return(key);
}
```

Listing 2. Testowanie działania procedury GetKey

```
unsigned char  kl;

  while(1){
  kl=GetKey();    //odczytaj stan klawiatury
  if(kl==1)       //klawisz SW0
  RB0=~RB0;
  if(kl==2)       //klawisz SW2
  RB1=~RB1;
  if(kl==4)       //klawisz SW3
  RB2=~RB2;
  if(kl==8)       //klawisz SW4
  RB3=~RB3;
}
```

RA3 muszą być skonfigurowane jako wejściowe, a linie RB0...RB3 jako wyjściowe (sterowanie diodami LED).

Na **listingu 1** zamieszczono procedurę GetKey działająca według algorytmu opisanego wyżej w punktach. Po wykryciu zwarcia styków i odczekaniu 25 ms do zmiennej key jest wpisywana zanegowana wartość odczytana z portu PORTA. Zrobiona tak dla wygody użytkownika, ponieważ stanem aktywnym jest poziom niski, a jest wygodniej posługiwać się na przykład dla wciśniętego SW1 kodem 0b00000010 niż nie 11111101. Funkcja zwraca następujące kody wciśniętych klawiszy: SW1=0x01, SW2=0x02, SW3=0x04, SW4=0x08. Jeśli po wykryciu zwarcia i doliczeniu 25 ms ponowne odczytanie nie wykryje zwarcia styku, procedura zwróci 0x00 i można to traktować jako kod błędu.

Listing 3. Testowanie ciągłego czytania klawiatury

```
unsigned char  kl, port;
  while(1){
  while((PORTA&0x0f)!=0x0f){       //wykonuj pętlę, gdy klawisz wciśnięty
    kl=~PORTA;                     //aktywne jedynka
    kl=kl&0x0f;                    //tylko 4 młodsze bity
    if(kl==1)
    port=port+1;
    if(kl==2)
    port=port-1;
    port=port&0x0f;
    PORTB=port;
    delay(200);                    //opóźnienie 200msek
  }
}
```

Listing 4. Zmodyfikowane testowanie ciągłego czytania klawiatury

```
...
kl=0;
while(1){
  while((PORTA&0x0f)!=0x0f){       //wykonuj pętlę, gdy klawisz wciśnięty
    kl=~PORTA;                     //aktywne jedynka
    kl=kl&0x0f;                    //tylko 4 młodsze bity
    if(kl==1)
    port=port+1;
    if(kl==2)
    port=port-1;
    if(kl==8)                      //kod ESCAPE
    break;
    port=port&0x0f;
    PORTB=port;
    delay(200);
  }
  if(kl==8)
  break;
}
PORTB=0x0f;                        //sygnalizacja zakończenia „regulacji”
while(1);
```

Automatyczne powtarzanie wciśniętego klawisza

Jest wiele sytuacji, w których „impulsowe” przyciskanie klawisza jest uciążliwe. Na przykład, jeżeli chcemy regulować siłę głosy za pomocą potencjometru cyfrowego to zmiana tłumienia o 10 dB z krokiem co 0,5 dB wymagałaby 20-krotnego naciśnięcia klawisza obsługiwanego przez procedurę GetKey. Bardziej naturalne wydaje się naciśnięcie i przytrzymanie przycisku aż poziom sygnału będzie odpowiedni. Do tego celu czytanie klawiatury można zorganizować inaczej. W pętli nieskończonej jest sprawdzany warunek przyciśnięcia klawisza i gdy jest on wciśnięty, to jest identyfikowany jego kod. Następnie zależnie od kodu wciśniętego klawisza program podejmuje jakąś akcję. Pomiędzy kolejnymi odczytami stanu styków upływa czas na wykonanie czynności zależnej od wciśniętego klawisza i dodatkowe opóźnienie. To dodatkowe opóźnienie określa, jak często jest powtarzany kod wciśniętego klawisza.

Na **listingu 3** zamieszczono fragment programu demonstrujący ideę takiego czytania klawiatury. Po każdym naciśnięciu SW1 jest zwiększana zmienna port, a po naciśnięciu SW2 – zmniejszana. Potem ta wartość jest przepisywana do rejestru portu PORTB. Stan czterech młodszych linii PORTB jest wyświetlany przez LED0...LED3. Opóźnienie 200 ms określa, jak często jest czytana klawiatura i modyfikowany stan linii PORTB.

REKLAMA

Multimetr U1252B Agilent Technologies

Praktykujący elektronik może się jakoś obyć bez oscyloskopu, nawet bez generatora też sobie poradzi. Na jego biurku nie może natomiast zabraknąć lutownicy i miernika uniwersalnego. Uruchomienie każdego urządzenia zawsze rozpoczyna się od sprawdzenia napięć zasilających. Niewykonanie tego kroku może skończyć się koniecznością wymiany połowy komponentów elektronicznych już na początku pracy. Często wymagane są pomiary dużo bardziej złożone, a wtedy nie sposób obyć się bez dobrego multimetru.

Multimetr cyfrowy, jak każde urządzenie elektroniczne, podlegał na przestrzeni lat ewolucji. Nie była ona chyba tak dynamiczna, jak na przykład w przypadku oscyloskopów cyfrowych, jednak gdy porównamy pierwsze multimetry cyfrowe z tymi, które są oferowane dzisiaj, to mimowolnie na twarzy pojawia się uśmiech. Zmieniały się podzespoły elektroniczne, zmieniały się materiały, z których wykonywano obudowy, przybywało funkcji pomiarowych, wzrastała dokładność pomiaru. Dynamika rozwoju multimetrów od pewnego czasu już nie jest tak duża. Można wręcz zaryzykować twierdzenie, że chyba wszystko, co było w tym temacie do wymyślenia, zostało już wymyślone. Mimo to, producenci przyrządów pomiarowych co jakiś czas „wypuszczają" nowe modele, w których można doszukać się jakichś nowinek.

Zabawki i przyrządy

Multimetry – kiedyś nazywano je (bardziej po polsku) miernikami uniwersalnymi. Jak wskazuje sama nazwa, są to przyrządy służące do mierzenia kilku różnych wielkości elektrycznych. Cyfrowe metody pomiaru pozwoliły znacznie poszerzyć funkcjonalność tych przyrządów. Na przykład w chwili pojawienia się multimetrów cyfrowych dużo łatwiejszy stał się pomiar częstotliwości i pojemności kondensatorów. Pomiarów takich na ogół nie wykonywały analogowe mierniki uniwersalne.

Jednym z najważniejszych parametrów multimetru jest niepewność pomiarowa, potocznie zwana dokładnością pomiaru. Niestety, parametr ten jest często mylony z rozdzielczością, a to sprawia, że mniej doświadczeni elektronicy mają przekonanie o superdokładności mierników cyfrowych. Jakże bowiem mogłoby być inaczej, skoro wyświetlany wynik jest najczęściej co najmniej 4-cyfrowy. Nic bardziej mylącego. Więcej na ten temat pisaliśmy w artykule *Błąd pomiaru, czyli procenty mieszane z cyframi*, zamieszczonym w EP 12/2008. Zwykle najprostszą metodą weryfikacji jakości przyrządu, bez wnikliwego analizowania jego parametrów technicznych, jest cena. Uzyskanie wysokiej klasy dokładności wymaga zastosowania odpowiednich środków, a te niestety kosztują. Z tego względu mierniki za kilkadziesiąt złotych mogą być traktowane jedynie jako zabawki. Do poważnych pomiarów należy sięgać po sprzęt renomowanych firm. Niewątpliwie jedną z czołowych pozycji wśród producentów multimetrów cyfrowych zajmuje firma Agilent Technologies, której multimetr cyfrowy typu U1252B opisano w artykule.

Dokładność pomiarów

Multimetr U1252B należy do przyrządów wysokiej klasy. Jego wybrane parametry techniczne, w tym dokładność na poszczególnych zakresach, zestawiono w **tabeli 1**.

Cechą charakterystyczną multimetrów cyfrowych jest zmieniająca się dokładność nie tylko w poszczególnych trybach pomiarowych, ale nawet w różnych zakresach tego samego trybu. Praktyka pokazuje, że przy większości pomiarów nie ma potrzeby prowadzenia dokładnego rachunku błędów, aczkolwiek osoba wykonująca pomiary zawsze powinna mieć świadomość przynajmniej tego, jaki jest rząd wielkości popełnianego błędu. Jest to istotne szczególnie wtedy, gdy wyniki pomiarów są umieszczane w dokumentacji jakiegoś urządzenia stanowiąc ważne informacje dla jego użytkowników. Błędy pomiarowe wynikają na pewno z jakości zastosowanych przetworników, ale należy pamiętać również o tym, że dołączenie miernika do badanego obwodu jest zawsze równoznaczne z wprowadzeniem do niego dodatkowego obwodu zmieniającego warunki pracy. Zatem nawet miernik idealnie dokładny nie pokaże prawdziwego wyniku, jeśli w istotnym stopniu te warunki zmieni. Parametrem mającym kluczowe znaczenie jest impedancja wejściowa miernika, która dla zakresów napięciowych powinna być jak największa, a dla prądowych jak najmniejsza. Impedancja wejściowa multimetru U1252B jest na zakresach napięciowych prawie niezmiennie równa ok. 10 MΩ, zaś na prądowych wynosi ona od 0,01 Ω do 100 Ω.

Duża część pomiarów przedstawionych w tabeli 1 jest obwarowana określonymi warunkami, które powinny być spełnione, w celu zachowania gwarantowanej przez producenta dokładności. O kilku z nich będzie mowa dalej, z innymi należy zapoznać się w instrukcji obsługi miernika. Na przykład warto pamiętać, że niepewność pomiaru napięcia stałego dla najmniejszego zakresu może być zwiększona do 0,05%+5, jeśli zostanie użyta funkcja pomiaru względnego.

Dodatkowe informacje:
Przyrząd do testów udostępniła firma AM Technologies z Warszawy. Więcej informacji na stronie internetowej *www.amt.pl*.

Tabela 1. Wybrane parametry techniczne multimetru U1252B

Napięcia stałe		
Zakres	Rozdzielczość	Dokładność
50,000 mV	0,001 mV	0,05%+50
500,00 mV	0,01 mV	0,025%+5
1000,0 mV	0,1 mV	
5,0000 V	0,0001 V	
50,000 V	0,001 V	
500,00 V	0,01 V	0,03%+5
1000,0 V	0,1 V	
Prąd stały		
500,00 µA	0,01 µA	0,05%+5
5000,0 µA	0,1 µA	
50,000 mA	0,001 mA	0,15%+5
440,00 mA	0,01 mA	
5,0000 A	0,0001 A	0,3%+10
10,000 A	0,001 A	0,3%+5
Rezystancja		
500,00 Ω	0,01 Ω	0,05%+10
5,0000 kΩ	0,0001 kΩ	0,05%+5
50,000 kΩ	0,001 kΩ	
500,00 kΩ	0,01 kΩ	
5,0000 MΩ	0,0001 MΩ	0,15%+5
50,000 MΩ	0,001 MΩ	1%+5
500,00 MΩ	0,01 MΩ	3%+10 (<200 MΩ) 8%+10 (>200 MΩ)
500,00 nS	0,01 nS	1%+10
Pojemność		
10,000 nF	0,001 nF	1%+8
100,00 nF	0,01 nF	1%+5
1000,0 nF	0,1 nF	
10,000 µF	0,001 µF	
100,00 µF	0,01 µF	
1000,0 µF	0,1 µF	
10,000 mF	0,001 mF	
100,00 mF	0.01 mF	3%+10
Częstotliwość		
99,999 Hz	0,001 Hz	0,02%+3 (<600 kHz)
999,99 Hz	0,01 Hz	
9,9999 kHz	0,0001 kHz	
99,999 kHz	0,001 kHz	
999,99 kHz	0,01 kHz	
Przebieg prostokątny		
Parametr	Rozdzielczość	Dokładność
Zakres częstotliwości: 0,5, 1,2, 5, 10, 15, 20, 25, 30, 40, 50, 60, 75, 80, 100, 120, 150, 200, 240, 300, 400, 480, 600, 800, 1200, 1600, 2400, 4800 Hz	0,01 Hz	0,005%+2
Współczynnik wypełnienia: 0,39...99,6%	0,390625%	0,4% pełnej skali
Szerokość impulsu: 1/częstotliwość	podzakres/256	0,2 ms+podzakres/256
Amplituda: ustalona od 0 do 2,8 V	0,1 V	0,2 V

Jej włączenie pozwala bowiem eliminować efekt termiczny. Z kolei w trybie pomiaru względnego zastosowanego podczas pomiaru rezystancji na zakresach 500 Ω i 5 kΩ eliminowany jest nie tylko wpływ efektów termicznych, ale również wpływ rezystancji przewodów na wynik pomiaru. Pomiary rezystancji na dwóch najniższych zakresach muszą być przeprowadzane w warunkach zachowania wilgotności względnej na poziomie nie większym niż 60%. Jak zwykle, szczególną ostrożność należy zachowywać podczas pomiarów natężenia prądu. Restrykcje dotyczą ograniczenia czasu pomiaru na

REKLAMA

Projektowanie płytek za pomocą Altium Designer Summer 09 (7)

Obecnie każdy zajmujący się projektowaniem urządzeń elektronicznych korzysta z programów EDA. Jednym z nich jest Altium Designer. Niniejszy cykl artykułów ma na celu nauczenie Czytelników stosowania z tego oprogramowania w zakresie projektowania obwodów drukowanych. W tej części kursu opisano sposób wykonania projektu płytki drukowanej z wykorzystaniem wielu wariantów montażowych i działanie narzędzia do zmiany wyprowadzeń i części (Pin/Part Swapping) pomocnego podczas optymalizacji przebiegu połączeń.

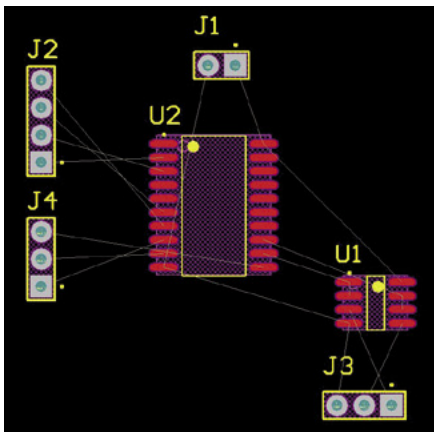
Dodatkowe materiały na CD/FTP:
ftp://ep.com.pl, user: 10925, pass: 87thc181
- poprzednie części kursu

Pin/Part Swapping

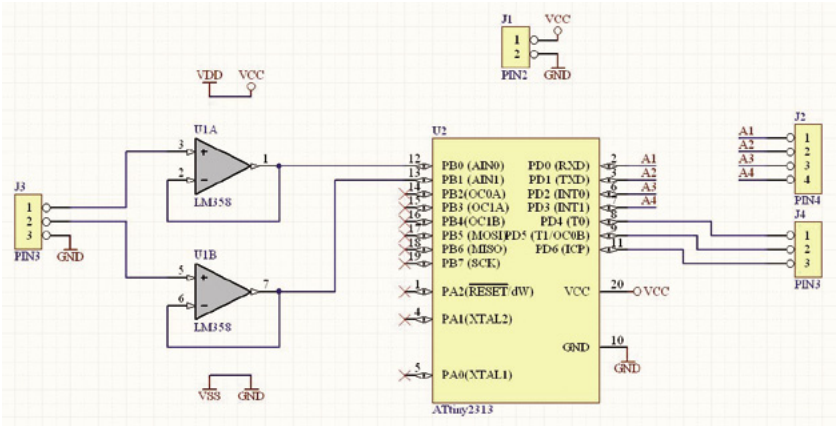
W trakcie tworzenia schematu z użyciem mikrokontrolerów, wzmacniaczy operacyjnych, bramek logicznych czy różnego rodzaju złącza, nieraz zastanawiamy się, których wyprowadzeń użyć, aby uzyskać jak najprostszy przebieg ścieżek na płytce drukowanej. Najczęściej po przeniesieniu danych ze schematu do edytora PCB i rozmieszczeniu elementów zauważamy, że niektóre połączenia należałoby zmienić. Nasuwa się myśl, że trzeba ponownie wrócić do edytora schematu, zmienić odpowiednie połączenia i przenieść modyfikacje do edytora płytki drukowanej. Takie postępowanie nie stwarza większych problemów przy niezbyt skomplikowanych projektach, ale w wypadku bardziej rozbudowanych staje się co najmniej niewygodne.

Altium Designer ma narzędzie pomocne w takich sytuacjach. Służy ono do optymalizacji przebiegu połączeń. Po wykonaniu takiej optymalizacji jest możliwe przeniesienie informacji o zmianach wykonanych w trakcie projektowania płytki drukowanej do edytora schematu.

Przejdźmy teraz do przykładu, który wyjaśni sposób konfiguracji i wykorzystania narzędzia o nazwie Pin/Part Swapping. W tym celu tworzymy nowy projekt PCB i edytujemy schemat zgodnie z **rysunkiem 121**. Na schemacie zastosowano symbol wzmacniacza operacyjnego zawierający ukryte wyprowadzenia zasilające domyślnie połączone z sieciami o nazwach VDD i VSS (odpowiednia biblioteka znajduje się w materiałach dodatkowych). Teraz kompilujemy projekt i przenosimy zmiany do edytora PCB. Następnie rozmieszczamy elementy na przykład tak jak na **rysunku 122**. Jak widać, połączenia wychodzące z elementów J2, J3 i J4 krzyżują się i niemożliwe będzie proste poprowadzenie ścieżek na jednej warstwie elektrycznej. W celu poprawienia tej sytuacji skorzystamy z narzędzia Pin/Part Swapping. Istotne jest, aby jeszcze przed przystąpieniem do konfigurowania odpowiednich parametrów ustalić, które wyprowadzenia i w jakich elementach mogą być zamieniane. W tym przykładzie przyjmujemy, że kolejność połączeń może być zamieniana w złączu J4. Może być także zmieniana kolejność połączeń wyprowadzeń mikrokontrolera U2 dołączonych do złącza J2 oraz mogą być zamieniane wzmacniacze wewnątrz struktury układu wzmacniacza operacyjnego U1.



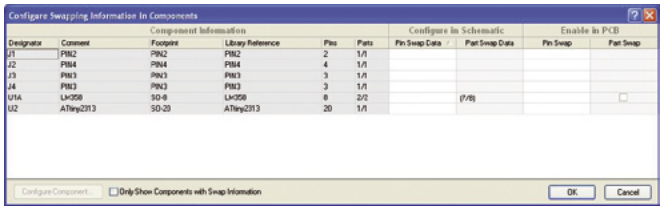
Rysunek 122. Przykładowe rozmieszczenie elementów



Rysunek 121. Schemat obwodu z ćwiczenia

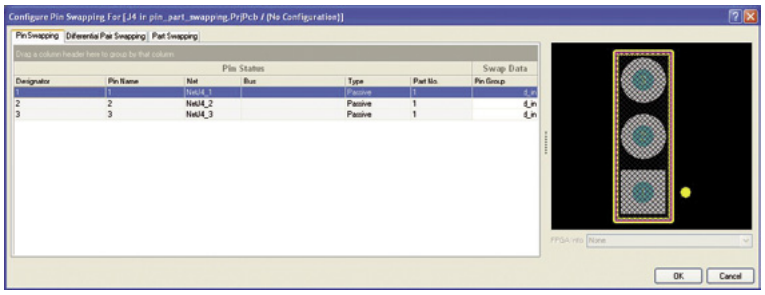
Wyprowadzenie komponentu może być zamienione z innym wyprowadzeniem w danym komponencie, jeśli należą one do tej samej grupy wyprowadzeń. Podobnie jest w przypadku zamiany części, ale dodatkowo należy określić zgodność wyprowadzeń pomiędzy częściami. Ustawień odpowiednich parametrów, czyli przypisania wyprowadzeń oraz części do odpowiednich grup, można dokonać zarówno w edytorze schematu, jak i w edytorze płytki drukowanej.

Zaczniemy od skonfigurowania ustawień dla złącza J4. W tym celu w edytorze PCB wybieramy *Tools –> Pin/Part Swapping –> Configure...*. Otworzy się okno zatytułowane *Configure Swapping Information In Components* (**rysunek 123**). W tym oknie pokazano ogólne informacje na temat wszystkich ele-

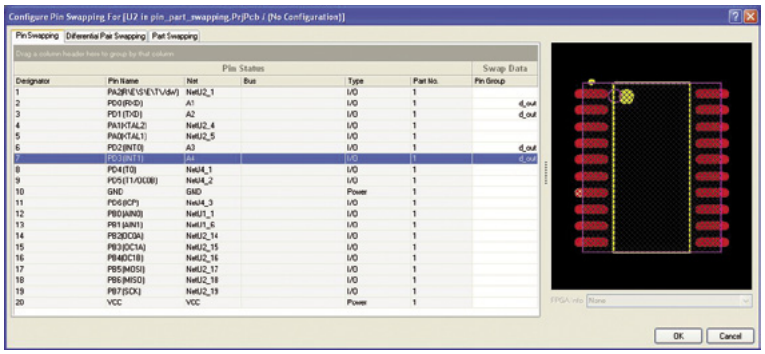


Rysunek 123. Okno *Configure Swapping Information In Components*

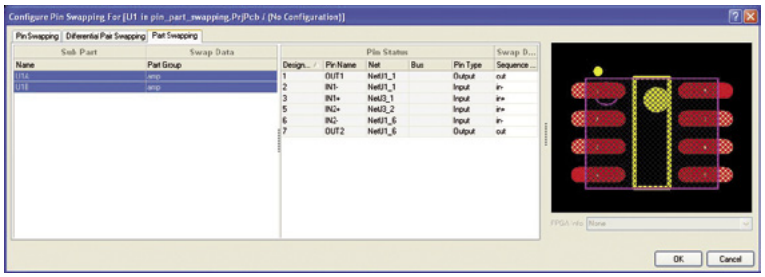
mentów znajdujących się na projektowanej płytce, takie jak liczba wyprowadzeń, liczba części w tym skonfigurowanych do zamiany oraz zezwolenia na zamianę wyprowadzeń i części. Zaznaczamy teraz na liście element J4 i klikamy na przycisk *Configure Component...*. Pojawi się okno zatytułowane *Configure Pin Swapping For...*. Z poziomu edytora schematu można uzyskać do niego dostęp, klikając prawym klawiszem myszy na symbolu złącza J4, a następnie wybierając *Part Actions –> Configure Pin Swapping...*. Teraz w zakładce *Pin Swapping* w kolumnie *Pin Group* nadajemy identyczną nazwę grupy dla każdego wyprowadzenia, np. *d_in*. Można wpisać nazwę dla każdego z wyprowadzeń z osobna lub wpisać nazwę dla pierwszego pinu, a dla każdego następnego kliknąć prawym klawiszem myszy i wybrać *Add To Pin-Swap Group* i wskazać nazwę grupy, którą podaliśmy. Przykładową konfigurację dla złącza J4 przedstawia **rysunek 124**. Zamykamy okno, klikając na przycisk OK. Teraz w podobny sposób nadajemy nazwę wyprowadzeniom numer 2, 3, 6 i 7 mikrokontrolera U2 (**rysunek 125**). Kolejnym krokiem jest skonfigurowanie ustawień dla wzmacniacza operacyjnego U1. Otwieramy okno *Configure Pin Swapping* dla układu U1 i przechodzimy do zakładki *Part Swapping*. W kolumnie *Part Group*, podobnie jak w przypadku wyprowadzeń we wcześniej opisanych elementach, nadajemy tę samą nazwę obydwu częściom, np. amp. Teraz musimy jeszcze określić zgodność wyprowadzeń pomiędzy częściami. Zaznaczamy na liście obydwie części wchodzące w skład wzmacniacza operacyjnego i w kolumnie *Sequence ID* nadajemy te same oznaczenia dla wyprowadzeń pełniących tę samą funkcję, np. dla wyjść wpisujemy out itp. Prawidłowe ustawienia parametrów przedstawia **rysunek 126**. Musimy jeszcze zezwolić na zamianę wyprowadzeń i części. Możliwe jest to jedynie z poziomu edytora PCB. Wystarczy w oknie *Configure Swapping Information In Components* zaznaczyć pola w kolumnach *Pin Swap* oraz *Part Swap* (**rysunek 127**). Dodatkowo widzimy w tym oknie, że wszystkie wyprowadzenia złącza J4 oraz cztery z dwudziestu wyprowadzeń układu U2 mają wymagane dane do zamiany. Można także zaznaczyć odpowiednie elementy umieszczone na płytce drukowanej i uruchomić panel inspektora klawiszem F11 na klawiaturze. Odnajdujemy w nim pola *Enable Pin Swapping* oraz *Enable Part Swapping* i zaznaczamy je (**rysunek 128**). Ostatnią czynnością do wykonania pozostało ustawienie odpowiednich parametrów w opcjach projektu. W tym celu wybieramy *Project –> Project Options* i przechodzimy do zakładki *Options* (**rysunek 129**). W sekcji *Allow Pin-Swapping Using These Methods* są do wyboru dwie opcje: *Adding/Removing Net-Labels* oraz *Changing Schematic Pins*. Jeśli *Adding/Removing Net-Labels* jest jedyną zaznaczoną opcją, to podczas wyboru polecenia zamiany wyprowadzeń czy części będziemy mieli dostęp wyłącznie do wyprowadzeń połączonych poprzez etykiety sieciowe. Opcja *Changing Schematic Pins* zezwala na zamianę wyprowadzeń w symbolu elementu na schemacie. Dzięki tej opcji będziemy mieli możliwość zamiany wyprowadzeń oraz części w komponentach, które na schemacie połączone są innymi metodami niż etykiety sieciowe. Proponuję wykonać ustawienia zgodnie z **rysunkiem 129**.



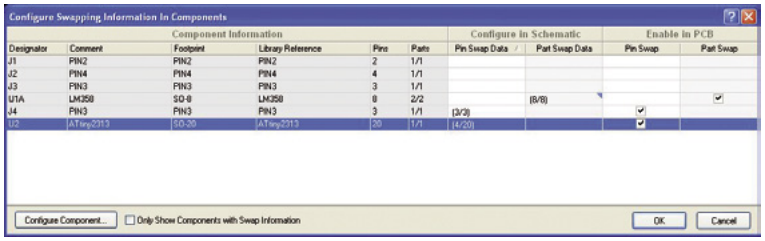
Rysunek 124. Przykład konfiguracji dla złącza J4



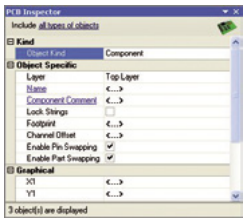
Rysunek 125. Nazwy pozostałych wyprowadzeń mikrokontrolera



Rysunek 126. Ustawienie parametrów wzmacniacza



Rysunek 127. Zezwolenie na zmianę wyprowadzeń i części

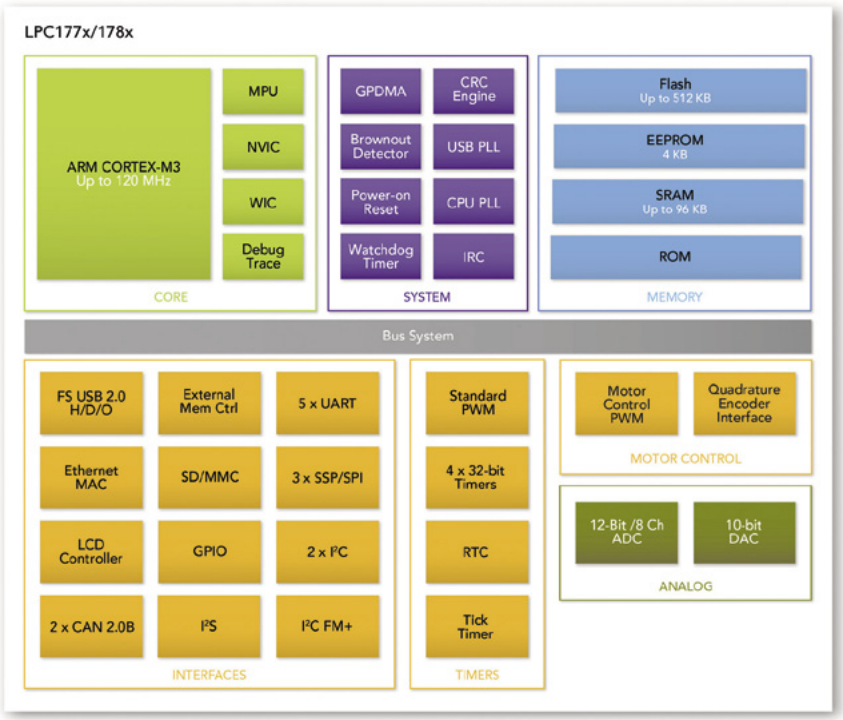


Rysunek 128. Panel inspektora

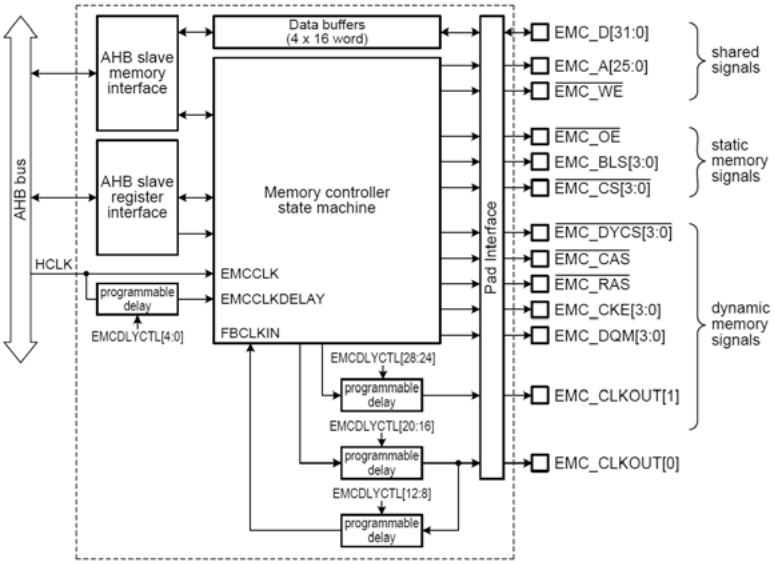
LPC177x/8x
Wyposażenie i kompatybilność

Nowe podrodziny mikrokontrolerów LPC17xx, które prezentujemy w artykule, są doskonałym uzupełnieniem dotychczasowej oferty produkcyjnej firmy NXP: udostępniają konstruktorom szybkość, możliwości i wyposażenie charakterystyczne dla rodziny LPC1700, zapewniając jednocześnie możliwość łatwej migracji projektów, w których stosowano starsze mikrokontrolery LPC238x, LPC245x, LPC246x oraz LPC247x.

Dodatkowe materiały na CD/FTP:
ftp://ep.com.pl, user: 10925, pass: 87thc181

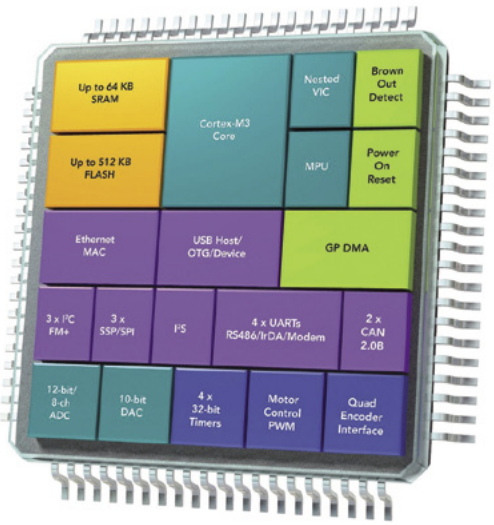


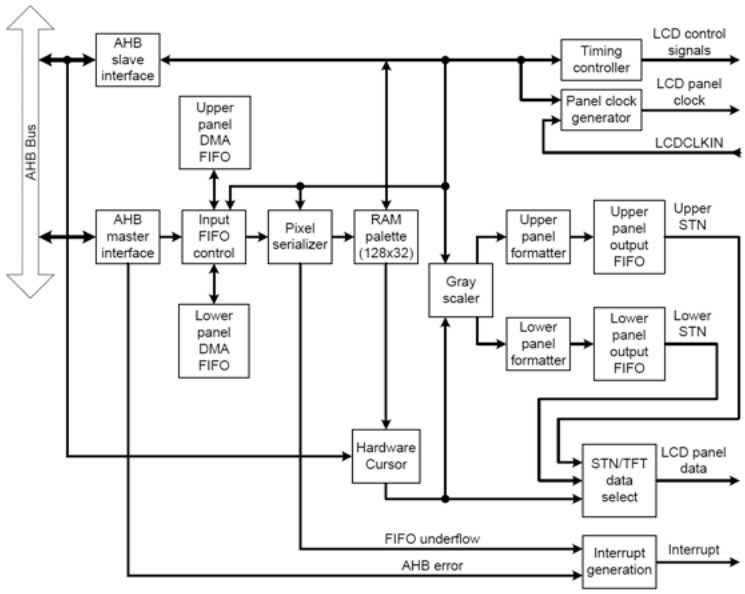
Rysunek 1. Schemat blokowy mikrokontrolerów LPC177x/LPC178x



Rysunek 2. Schemat blokowy kontrolera pamięci zewnętrznych EMC

Nowe w ofercie NXP mikrokontrolery LPC177x oraz LPC178x wyposażono w rdzeń Cortex-M3 taktowany sygnałem zegarowym o częstotliwości do 120 MHz (niektóre źródła podają, że będzie ona wynosić 100 MHz, dokładne dane nie są znane w chwili pisania artykułu), wewnętrzną pamięć Flash o pojemności do 512 kB, EEPROM do 4 kB i SRAM o pojemności do 96 kB. Zastosowana w nich jednostka centralna należy do drugiej generacji Cortex-M3 (rev 2), w jej skład wchodzi m.in. kontroler MPU (*Memory Protection Unit*) z możliwością podziału pamięci na 8 obszarów chronionych niezależnie oraz kontroler WIC (*Wake-up Interrupt Controller*) współpracujący ze standardowym kontrolerem przerwań NVIC. Bloki te nie są dostępne w mikrokontrolerach oferowanych przez głównych rynkowych konkurentów, zastosowano w nich bowiem rdzenie Cortex-M3 pierwszej generacji. Schemat blokowy prezentowanych mikrokontrolerów pokazano na **rysunku 1**.





Rysunek 3. Budowa sprzętowego kontrolera LCD (wbudowany w mikrokontrolery LPC178x)

Tabela 1. Zestawienie podstawowych cech i wyposażenia mikrokontrolerów LPC177x i LPC178x

Typ	Flash [kB]	SRAM [kB]	GPIO	Interfejsy USB	USB device	USB host	USB OTG	CAN	UART	Ethernet
LPC177x – kompatybilne pinowo z LPC238x, LPC245x, LPC246x										
LPC1774FBD144	128	40	109	1	1	0	0	2	4	–
LPC1774FBD208	128	40	165	1	1	0	0	2	5	–
LPC1776FBD208	256	80	165	1	1	1	1	2	5	+
LPC1776FET180	256	80	141	1	1	1	1	2	5	+
LPC1777FBD208	512	96	165	1	1	1	1	2	5	–
LPC1778FBD144	512	96	109	1	1	1	1	2	5	–
LPC1778FBD208	512	96	165	1	1	1	1	2	5	+
LPC1778FET180	512	96	141	1	1	1	1	2	5	+
LPC1778FET208	512	96	165	1	1	1	1	2	5	+
LPC178x – kompatybilne pinowo z LPC247x										
LPC1785FBD208	256	80	165	1	1	1	1	2	5	–
LPC1786FBD208	256	80	165	1	1	1	1	2	5	+
LPC1787FBD208	512	96	165	1	1	1	1	2	5	–
LPC1788FBD144	512	96	109	1	1	1	1	2	5	+
LPC1788FBD208	512	96	165	1	1	1	1	2	5	+
LPC1788FET180	512	96	141	1	1	1	1	2	5	+
LPC1788FET208	512	96	165	1	1	1	1	2	5	+

W **tabeli 1** znajduje się zestawienie podstawowych cech i wyposażenia mikrokontrolerów LPC177x i LPC178x, przy czym w celu zwiększenia jego czytelności pominięto większość parametrów wspólnych dla nowych układów oraz znanych z wcześniejszych wersji LPC1700.

Interesującymi nowościami wprowadzonymi do wyposażenia podrodzin LPC177x i LPC178x są:

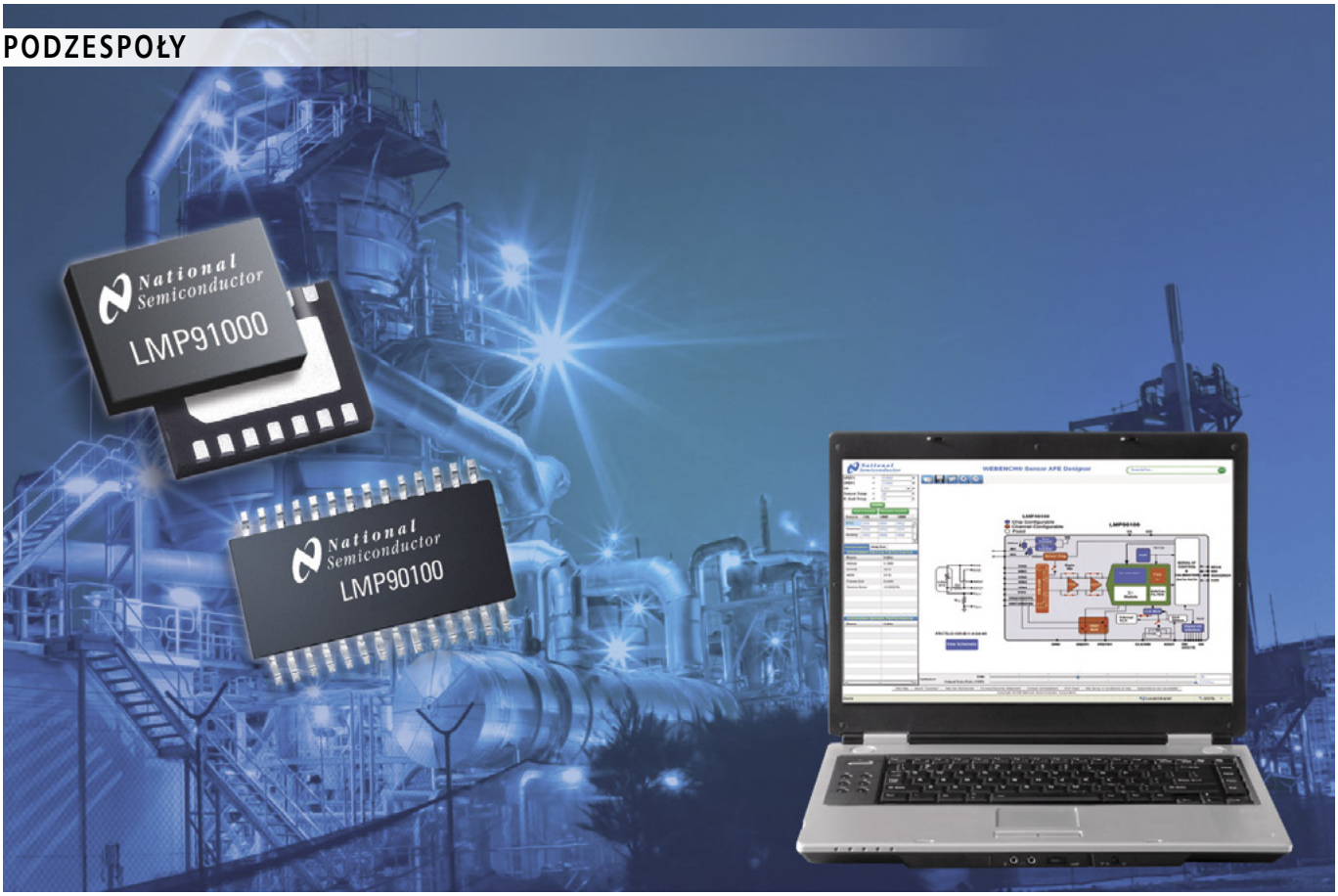
– kontroler pamięci zewnętrznych (**rysunek 2**) umożliwiający dołączenie do mikrokontrolera zewnętrznej pamięci SDRAM, Flash lub SRAM o pojemności do 64 MB o szerokości magistrali danych 8/16/32 bity w różnych konfiguracjach, zależnie od typu mikrokontrolera,
– pamięć EEPROM o pojemności 4 kB, dostęp do zawartości której jest możliwy wyłącznie za pośrednictwem CPU w sposób uniemożliwiający uruchamianie z niej programów i trwałości co najmniej 100000 cykli kasowanie/programowanie (z podziałem na 64 niezależne strony),
– kontroler LCD (obsługujący wyświetlacze monochromatyczne i kolorowe STN i TFT, wyłącznie w LPC178x – **rysunek 3**) umożliwiający wyświetlanie obrazów o wymiarach od 320×200 pikseli do 1024×768 pikseli,
– sprzętowy interfejs kart SD (MCI – *Multimedia Card Interface*), który obsługuje karty SD w trybie danych od 1 do 4 bitów.

Standardowym wyposażeniem wszystkich mikrokontrolerów LPC177x/LPC1778x zapowiadanych przez producenta są interfejsy cyfrowego audio I²S, 6-kanałowe generatory PWM ze sterownikami zoptymalizowanymi do generacji przebiegów zasilających uzwojenia wielofazowych silników, interfejsy QEI (*Quadrature Encoder Interface*) oraz typowa mikrokontrolerowa „drobnica" w postaci szerokiej gamy i dużej liczby interfejsów komunikacyjnych (w tym I²C działający w trybie Fm+ – do 1 MHz), przetworników A/C i C/A, a także sprzętowego generatora CRC (przydatny m.in. podczas transferu danych do i z kart SD), który obsługuje trzy popularne wielomiany zabezpieczające: CRC-CCITT, CRC-16 i CRC-32. Większość prezentowanych mikrokontrolerów wyposażono także w ethernetowy MAC, komunikujący się z zewnętrznym układem PHY za pomocą interfejsu RMII lub MII.

Poza doskonałym wyposażeniem, mikrokontrolery LPC177x i LPC178x charakteryzują się dodatkowym atutem: są fizycznie wymienne z mikrokontrolerami starszej generacji z rodziny LPC2000. Mikrokontrolery LPC177x mogą być stosowane w miejsce mikrokontrolerów LPC238x, LPC245x/6x montowanych w takich samych obudowach. Mikrokontrolery LPC178x mogą równie wygodnie zastąpić układy z serii LPC247x. W ten sposób firma NXP przygotowała dla swoich wiernych konstruktorów „spadochron bezpieczeństwa", pozwalający szybko, bezpiecznie i bez konieczności ponoszenia wielkich nakładów unowocześnić wcześniejsze opracowania. Kolejny dowód na to, że Corteksy górą!

Tomasz Starak

http://sklep.avt.pl



AFE: sposób na sensory

Jednym z trudniejszych zadań w projektowaniu układów pomiarowych są analogowe obwody wejściowe, odpowiadające za „kondycjonowanie" sygnałów z czujników pomiarowych. Szczególnie trudne jest zbudowanie toru wejściowego potrafiącego współpracować z wieloma typami czujników jednocześnie, dzięki czemu można jednocześnie mierzyć wiele parametrów fizycznych monitorowanego procesu.

Dodatkowe informacje...
...są dostępne pod adresem:
www.national.com/sensor_afe

Nowoczesne rozwiązanie tego problemu zaproponowała firma National Semiconductor, wprowadzając do produkcji konfigurowalne układy analogowe AFE (*Analog Front End*) oraz oprogramowanie narzędziowe umożliwiające ich dostosowanie do wymogów aplikacji. Pomysł, który prezentujemy w artykule, jest dość nowy (ostatnie miesiące 2010 roku), dlatego w ofercie producenta znajdują się dwa układy scalone należące do rodziny AFE, ale ich budowa i wynikające z niej możliwości spełniają większość spodziewanych wymogów.

LMP90100: uniwersalny układ wejściowy

Układ LMP90100 pojawił się na rynku jako pierwszy w rodzinie AFE, jego budowę pokazano na **rysunku 1**. Jednym z najważniejszych bloków wewnętrznych tego układu jest 24-bitowy przetwornik A/C (sigma-delta) o częstotliwości próbkowania programowanej w zakresie od 1,7 do niemal 215 próbek na sekundę. Przetwornik wyposażono w system autokalibracji oraz filtr służący do minimalizowania wpływu zakłóceń sieciowych (50 lub 60 Hz) na wyniki pomiarów.

Sygnał podawany na wejście przetwornika A/C może być wzmacniany za pomocą dwustopniowego wzmacniacza o programowanym wzmocnieniu (od 1 do 128 V/V), przy czym wzmacniacz oraz bufor wejściowy mogą pracować w konfiguracji różnicowej lub symetrycznej, w zależności od rodzaju sensora dołączonego do wejść układu. W torze wejściowym producent zastosował 8-kanałowy multiplekser analogowy, który umożliwia wykonywanie pomiarów w 4 kanałach różnicowych lub 7 asymetrycznych.

W przypadku wykorzystywania czujników zasilanych prądowo (np. mostków), konstruktorzy urządzeń pomiarowych bazujących na układzie LMP90100 mają uproszczone zadanie, bowiem wyposażono go w dwa programowane źródła prądowe o wydajności do 1 mA. Dzięki wyposażeniu torów wejściowych w analogowy system diagnostyczny prezentowany układ może samodzielnie wykryć fakt zwarcia wyprowadzeń czujnika lub przerwę w jego obwodzie, co minimalizuje ryzyko wykonania błędnych pomiarów. System ochrony jakości pomiarów wprowadzono także „po stronie" cyfrowej: dane przesyłane z układu LPM90100 za pomocą interfejsu SPI do mikrokontrolera są zabezpieczane sumą kontrolną CRC, co ogranicza ryzyko powstania błędnych odczytów.

Duża liczba kanałów wejściowych pozwala dołączać do układu LPM90100 kilka czujników jednocześnie, dzięki czemu pojedynczy układ AFE spełnia zadanie kompletnego interfejsu mierzącego np. temperaturę, wilgotność i ciśnienie. Żeby ułatwić odczyt danych z kilku czujników, konstruktorzy układu wyposażyli go w programowany sekwencer odczytu, który może pracować w kilku trybach, począwszy od zewnętrznego wyzwalania, aż po samoczynne odczyty kanałów w ustalonej kolejności.

http://avt.pl

Prezentowany układ wyposażono także w 7-bitowy, cyfrowy port I/O (dostępny poprzez SPI), który można wykorzystać do sterowania LED, przekaźników lub kluczy ustalających zakres w dzielniku wejściowym lub w dowolny inny, wygodny w aplikacji sposób. Port cyfrowy można wykorzystać także do odczytu stanu styków klawiatury lub monitorowania dowolnych stanów cyfrowych.

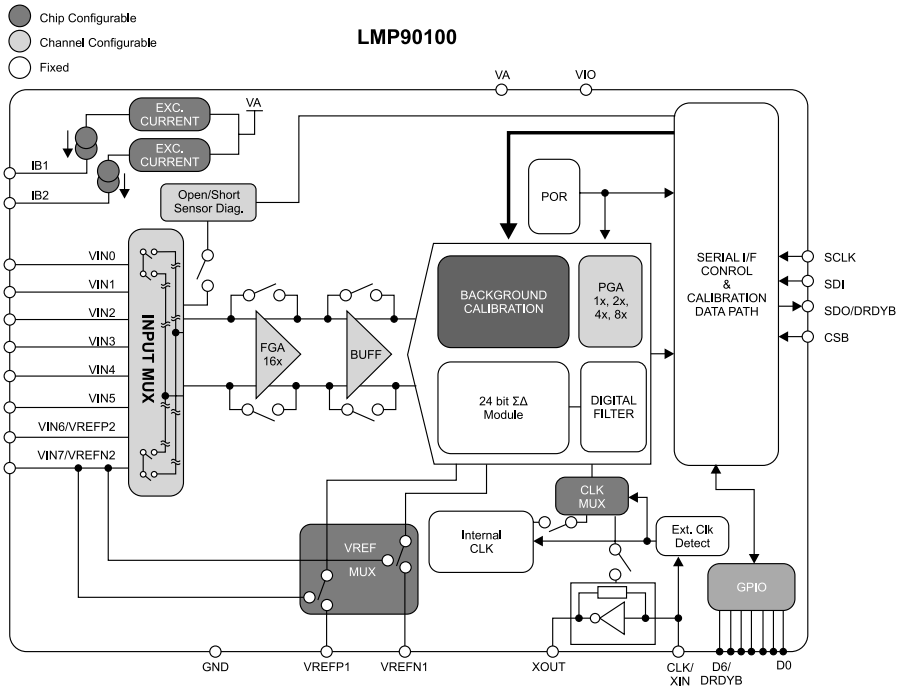
Pomimo niewielkiego poboru mocy, układ LMP90100 wyposażono w system oszczędzania energii, który pozwala go programowo deaktywować (dostępne są tryby: *stand-by* oraz *power-down*).

Prostotę aplikacyjną prezentowanego układu dopełnia fakt dostarczania go w łatwej do montażu obudowie SOIC28, przystosowanej do pracy w zakresie temperatur od –40 do +105°C.

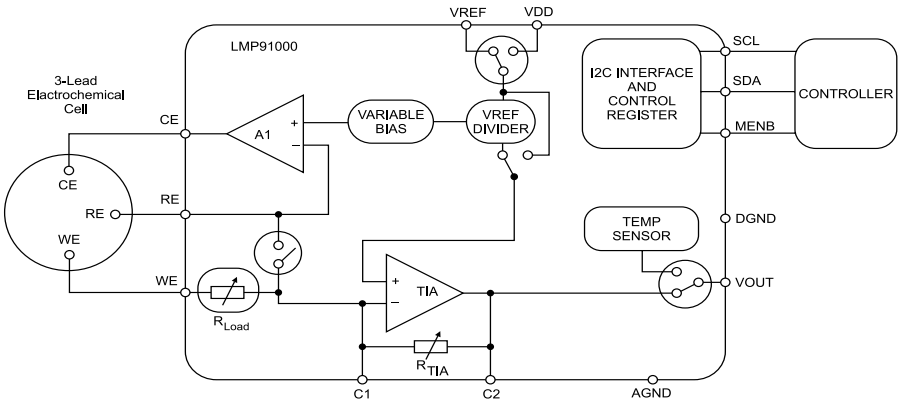
LMP91000: AFE dla chemików

Zupełnie inaczej jest zbudowany drugi z układów z serii AFE – LMP91000 (schemat blokowy pokazano na **rysunku 2**) – przeznaczony do współpracy z sensorami czynników chemicznych, przede wszystkim gazów. Zadaniem tego układu jest wytworzenie analogowego napięcia, proporcjonalnego (w zadanej przez użytkownika skali) do prądu płynącego przez 2- lub 3-przewodowy czujnik dołączony do jego wejścia. Układ wyposażono także w analogowy czujnik temperatury, którego wyjście może być dołączane do wyjścia pomiarowego przemiennie z wyjściem wzmacniacza TIA (*TransImpedance Amplifier*), który odgrywa rolę konwertera prąd-napięcie.

Prezentowany układ wyposażono w interfejs I²C, ale służy on wyłącznie do konfiguracji wzmacniacza TIA, który jest jego mózgiem. Wykorzystanie jego możliwości w systemie cyfrowym wymaga zastosowania zewnętrznego przetwornika A/C.



Rys. 1. Schemat blokowy układu LMP90100 z zaznaczonymi blokami konfigurowalnymi



Rys. 2. Schemat blokowy układu LMP91000

Układy LMP91000 są oferowane w 14-pinowych obudowach QFN, mogą pracować w zakresie temperatur od –40 do +85°C.

REKLAMA

Technologia GSM w elektronice (12)

Tryby oszczędzania energii

W tym artykule opiszemy tryby pracy, które są dostępne w programowalnych modułach Sierra Wireless™ AirPrime Q26 oraz korzyści wynikające z ich zastosowania.

Dodatkowe materiały na CD/FTP:
ftp://ep.com.pl, user: 10925, pass: 87thc181
• poprzednie części kursu

Różnorodność potencjalnych zastosowań, w których mogą być wykorzystane programowalne urządzenia Sierra Wireless, jest dość szeroka. Aby sprostać wymaganiom, jakie stawia się współczesnym układom programowalnym, szczególnie dotyczącemu optymalizacji zużycia energii, w programowalnych modułach Sierry Wireless mamy do dyspozycji kilka różnych trybów pracy. Każdy z nich cechuje się inną charakterystyką pracy, dostępnością interfejsów peryferyjnych oraz oczywiście poborem prądu.

ACTIVE mode

Podstawowym trybem pracy modułu jest tryb ACTIVE mode. W tym trybie wszystkie funkcje modułu są aktywne. Istnieją dwa warianty tego trybu: z włączonym i wyłączonym stosem GSM. Stos GSM to jeden z procesów w systemie operacyjnym modułu. Odpowiedzialny jest za obsługę części radiowej oraz za współpracę z siecią GSM. Wyłączenie stosu GSM pozwala na oszczędność energii, jednak wszystkie serwisy Open AT oraz komendy AT powiązane z obsługą sieci i obsługą SIM nie będą działały.

Aby wyłączyć stos GSM, należy wydać komendę AT+CFUN=0. Po tej komendzie nastąpi reset, moduł uruchomi się bez obsługi stosu GSM.

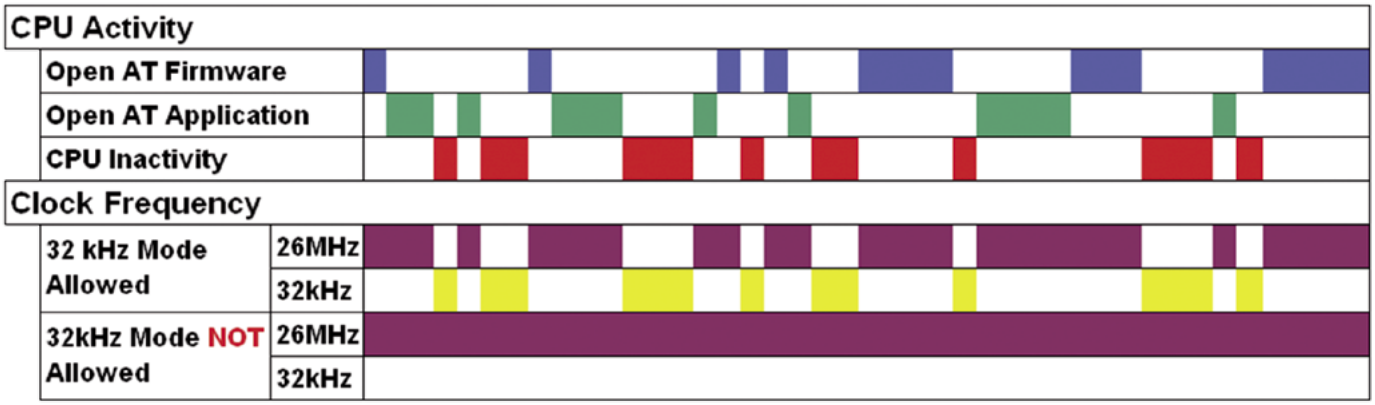
Aby ponownie włączyć stos GSM, wydajemy komendę AT+CFUN=1,1 (reset modułu) lub AT+CFUN=1,0

W trybie ACTIVE MODE procesor modułu jest taktowany domyślnie z prędkością 26 MHz, a jak pisaliśmy w jednym z wcześniejszych odcinków naszego cyklu, maksymalna prędkość taktowania procesora wynosi 104 MHz. Na **listingu 1** przedstawiamy, w jaki sposób kontrolować prędkość taktowania modułu.

Przytoczony przykład tworzy dwie komendy AT, których zadaniem jest przełączenie prędkości taktowania zgodnie ze swoją nazwą, a następnie wywołanie prostej pętli opóźniającej. Po jej wykonaniu wyświetlany jest komunikat OK. Obserwując czas wykonania pętli opóźniającej, możemy porównać moc obliczeniową dla obu częstotliwości taktowania. Według informacji producenta moduł pracujący z częstotliwością 26 MHz dysponuje mocą obliczeniową na poziomie 21 MIPS-ów, natomiast jeśli pracuje z częstotliwością 104 MHz, to moc obliczeniowa równa jest 87 MIPS-ów. Oczywiście wyższa

Listing 1. Kontrola częstotliwości taktowania

```
#include "adl_global.h"
#include "generated.h"

s32 VariHandle ;
//-------------------------------------------------------------------------
void Empty (u32 Data) {

        static u8 a =0;
        a++;
}
//-------------------------------------------------------------------------
void Fun_slow (adl_atCmdPreParser_t * paras) {

    u16 Cnt, Cnt1 ;

    TRACE((2,"Start Normal mode"));
    adl_vsSetClockMode (VariHandle, ADL_VS_MODE_STANDARD); //26MHz

    Cnt = 0xffff;           //opoznienie
    while (Cnt--) {
        Cnt1 = 50 ;
        while (Cnt1--) {
            Empty (Cnt) ;
        }
    }
    adl_atSendStdResponse(ADL_AT_RSP,ADL_STR_OK); //wyswietl OK
}

//-------------------------------------------------------------------------
void Fun_fast(adl_atCmdPreParser_t * paras) {

    u16 Cnt, Cnt1 ;

    TRACE((2, "Start in VariSpeed mode")) ;
    adl_vsSetClockMode (VariHandle, ADL_VS_MODE_BOOST) ; //104MHz

    Cnt = 0xffff ;              ///opoznienie
    while (Cnt--) {
        Cnt1 = 50 ;
        while (Cnt1--) {
            Empty (Cnt ) ;
        }
    }

    adl_atSendStdResponse(ADL_AT_RSP,ADL_STR_OK); //wyswietl OK
}
//-------------------------------------------------------------------------
void main_task ( void )
{
    // TODO Insert your task initialization code here
        VariHandle = adl_vsSubscribe () ;
    TRACE((1,"VariHandle = %d", VariHandle));

    adl_atCmdSubscribe("AT+FAST", Fun_fast, ADL_CMD_TYPE_ACT);
    adl_atCmdSubscribe("AT+SLOW", Fun_slow, ADL_CMD_TYPE_ACT);
}
```



Rysunek 1. Schemat zmiany częstotliwości taktowania w funkcji zmian obciążenia procesora

prędkość taktowania oznacza większe zużycie prądu przez moduł.

SLEEP mode

Kolejnym, bardzo pożytecznym trybem jest tryb SLEEP mode. W trybie tym wewnętrzny procesor jest w trybie obniżonego poboru energii i jest taktowany częstotliwością 32 kHz. Dostęp do interfejsów zewnętrznych jest ograniczony, w związku z czym interfejsy UART, USB, SPI, I²C, GPIO, ADC oraz Buzzer nie są dostępne. Przetwarzanie aplikacji Open AT jest wstrzymane. Wznowienie jej działania następuje w przypadku zaistnienia poniższych zdarzeń:

- pojawienie się alarmu RTC,
- zdarzenie od sprzętowego Timera TCU,
- pojawienie się zewnętrznego przerwania na pinie INT,
- naciśnięcie klawisza z matrycy 5×5,
- zmiana stanu linii DTR portu UART1 na niski.

Po wykonaniu zadań związanych ze zdarzeniem, które było powodem wybudzenia, moduł ponownie przełącza się w tryb SLEEP. W okresie wybudzenia funkcjonalność modułu jest zgodna z trybem ACTIVE mode i wszystkie interfejsy znowu są dostępne. Schemat zmian częstotliwości najlepiej zobrazuje **rysunek 1**.

Wyjście ze stanu obniżonego poboru energii następuje również regularnie, zgodnie z pagingiem sieci GSM (oczywiście o ile stos GSM nie został wyłączony). Wygląda to tak, że co określony czas (zależny od sieci GSM, z reguły co 2 s lub 0,5 s) moduł budzi się na bardzo krótko, żeby posłuchać sieci GSM, a następnie ponownie przełącza się w tryb obniżonego poboru energii. Dzięki temu możliwa jest reakcja modułu na przychodzące połączenie oraz możliwe jest odbieranie SMS-ów.

Wprowadzenie modułu w tryb SLEEP mode uzyskujemy, wydając komendę AT+W32K=1 lub AT+W32K=1,0 (ignorowany jest stan linii DTR). Moduł po wydaniu komendy potrzebuje od 1 s do 15 s czasu na przejście do trybu SLEEP. Trwałe opuszczenie trybu SLEEP mode następuje po wydaniu komendy AT+W32K=0.

Wykorzystanie linii DTR jako jednego z warunków wejścia w tryb SLEEP mode jest podyktowane tym, że w niektórych zastosowaniach moduł pracuje jedynie jako zwykły modem odpowiadający na komendy AT (bez aplikacji Open AT), wysyłane przez system nadrzędny(Host). W takim przypadku, gdy Host ma jakieś komendy do wydania do modułu, blokuje on możliwość przejścia w tryb SLEEP mode, ustawiając linie DTR w stan wysoki. Dzięki temu UART nie zostanie wyłączony w trakcie przesyłania informacji. Po przesłaniu całej informacji Host zmienia stan na linii DTR, pozwalając modułowi ponownie wejść w tryb SLEEP mode. W przypadku wywołania trybu SLEEP mode ignorującego linię DTR, wyjście z trybu SLEEP kontrolowane jest przez aplikację Open AT (np. po doliczeniu timera lub po wystąpieniu zdarzenie GSM).

Tak samo jak w przypadku ACTIVE mode, tu też istnieje możliwość wyboru trybu SLEEP z wyłączoną obsługą stosu GSM. W tym celu wydajemy komendę AT+WBHV=1,2. Opuszczenie trybu SLEEP w takim przypadku realizowane jest przez komendę AT+WBHV=1,0.

Na koniec jedna uwaga dotycząca trybu SLEEP mode – w trybie tym zostaje dezaktywowany wewnętrzny, hardware'owy wachdog systemowy. W warunkach normalnych przerywa on działanie każdej pętli trwającej dłużej niż 5 s, co zabezpiecza nasz system przed wystąpieniem tzw. martwych pętli. Zatem przed uruchomieniem trybu SLEEP mode musimy zadbać o takie warunki aplikacji, aby żadna tego typu sytuacja nie zaistniała.

Alarm mode

Tryb ALARM mode oparty jest na zegarze RTC modułu i polega na wyłączaniu modemu i automatycznym jego włączaniu się w momencie wystąpienia wcześniej zaprogramowanego alarmu. Zasilanie RTC jest realizowane za pomocą osobnego pinu BAT-RTC, dzięki czemu kiedy modem jest wyłączony, pobór energii będzie ograniczony do prądu pobieranego przez pin RTC (2-3uA). W przypadku modemu Fastrack zasilanie RTC zrealizowane jest wewnątrz modemu w postaci baterii pastylkowej, natomiast w przypadku projektów opartych na module Q26, o zasilanie dla RTC musimy zadbać samodzielnie.

Ustawienie wewnętrznego zegara RTC najczęściej realizowane jest komendą AT+CCLK="yy/mm/dd,hh:mm:ss", ale może być też wykonane przez aplikację Open AT, wykorzystując funkcję ADL-a o nazwie *adl_rtcSetTime()*. Jeżeli w układzie zostało zrealizowane podtrzymanie RTC, to zegar będzie pracował niezależnie od tego, czy modem jest włączony, czy nie. System RTC ma dodatkowo możliwość ustawienia do 16 alarmów. Alarm ustawiamy komendą AT+CALA="yy/mm/dd,hh:mm:ss" (na razie nie mam możliwości ustawienia go za pomocą funkcji ADL). Po ustawieniu zegara oraz alarmu można wyłączyć moduł komendą AT+CPOF. Wydając komendę AT+CPOF, należy się upewnić, że pin ON/OFF jest w stanie niskim. Stan wysoki na tym pinie będzie powstrzymywał moduł przed wyłączeniem. W momencie nadejścia czasu alarmu moduł zostaje automatycznie włączony. Przykładowe zastosowanie przedstawionych możliwości zostało wykorzystane w **listingu 2**.

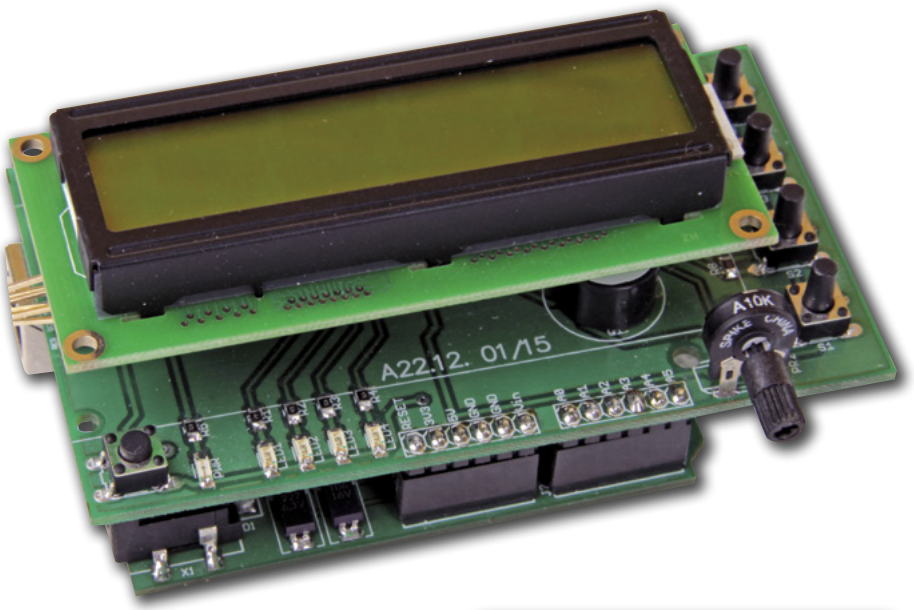
Działanie przedstawionej aplikacji polega na ustawieniu alarmu za pięć minut od odczytanego czasu RTC. Jednocześnie uruchomione zostają dwa timery: jeden wykonujący cyklicznie wyświetlanie tekstu (może to być oczywiście każda inna funkcja), a drugi to timer, który spowoduje wyłączenie się modułu. Po pięciu minutach, czyli kiedy uruchomi się alarm, moduł ponownie się włączy, wykona te same zadania i sam się wyłączy.

W aplikacji zastała również zamaskowana systemowa komenda AT+CCLK. Własna interpretacja tej komendy ma na celu ponowne ustawienie alarmu, w momencie kiedy czas RTC zostanie przedstawiony za pomocą tej komendy. Warto zauważyć, że funkcja *ustaw_alarm()* najpierw kasuje poprzedni

Kurs Arduino (3)

Obsługa modułu LCD

W EP 4/2011 opublikowaliśmy opis modułu LCD z przyciskami (AVT1615) współpracującego z płytką Arduino oraz kompatybilną – AVTduino. W tym artykule pokażemy sposób obsługi programowej tego modułu oraz komponentów interfejsu użytkownika, które on zawiera: przycisków, potencjometru, generatora piezzo, diod LED oraz czujnika temperatury LM35.



Zacznijmy od przykładów praktycznych rozwiązań, których będzie można użyć w samodzielnie opracowywanych aplikacjach. Na **listingu 1** zamieszczono przykładowy program dla modułu AVT1615. Program ten wyświetla komunikat na wyświetlaczu LCD, mierzy napięcie na suwaku potencjometru, odczytuje temperaturę, wyświetla umowny numer naciśniętego przycisku oraz zaświeca umieszczoną koło niego diodę LED. Dodatkowo, naciśnięcie przycisku jest sygnalizowane dźwiękiem brzęczyka piezzo.

W Arduino do obsługi modułów wyświetlaczy z kontrolerem HD44780 jest przeznaczona biblioteka *LiquidCrystal (LCD)* która umożliwia sterowanie modułem wyświetlacza za pomocą interfejsu z 4- lub 8-bitową szyną danych. W **tabeli 1** zamieszczono komendy dostępne w tej bibliotece.

Niektóre z komend wymienionych w tabeli 1 zastosowano w przykładowym programie z listingu 1. Jak można zauważyć, w programie w pierwszej kolejności dołączono bibliotekę obsługi LCD – *LiquidCrystal.h*. Następnie zadeklarowano stałe definiujące wyprowadzenia, do których dołączono diody LED, klawisze i brzęczyk piezzo. Dla lepszej czytelności programu nadano im łatwe do zapamiętania nazwy. Numery wyprowadzeń są zgodnie z opisem na płytkach Arduino Uno i Avtduino LCD. Dalej, za pomocą komendy *LiquidCrystal(rs, enable, d4, d5, d6, d7) zgodnie ze schematem ideowym* przypisano wyprowadzenia, do których został dołączony LCD. W dalszej części programu utworzono zmienne wykorzystywane do obliczenia wartości zmierzonego napięcia z suwaka potencjometru dołączonego do nóżki A0 oraz do obliczenia temperatury zmierzonej przez termometr LM35 dołączony do nóżki A1. Dzięki 8-bajtowej tablicy *byte st[8]={…}* zdefiniowano symbol stopnia, który jest używany jako jednostka temperatury.

W funkcji *setup()* za pomocą komendy *lcd.begin(16, 2)* zdefiniowano rozdzielczość zastosowanego wyświetlacza LCD. Pierwszy parametr określa liczbę znaków w wierszu (kolumn) a drugi liczbę wierszy. Jak łatwo domyślić się, w przykładzie zastosowano wyświetlacz 2×16 znaków. Do utworzenia znaku stopnia służy komenda *lcd.createChar(0, st), której argumentami są numer znaku (kod) oraz bajty definicji (w naszym wypadku jest to tablica st[])*. Komenda *analogReference(DEFAULT)* ustala napięcie odniesienia dla wewnętrznego przetwornika A/C mikrokontrolera na napięcie zasilające (w tym wypadku 5 V). Przetwornik mikrokontrolera będzie mierzył sygnały analogowe

Dodatkowe materiały na CD/FTP:
ftp://ep.com.pl, user: 10925, pass: 87thc181
- poprzednie części kursu

Tabela 1. Komendy obsługi wyświetlacza LCD

LiquidCrystal()	Definiuje piny do których został dołączony LCD
begin()	Definiuje rozdzielczość zastosowanego LCD
clear()	Czyści ekran LCD
home()	Ustawia kursor na początku ekranu LCD
setCursor()	Ustawia kursor w zadanym miejscu LCD
write()	Zapisuje znak do LCD
print()	Zapisuje znak lub znaki do LCD
cursor()	Włącza kursor
noCursor()	Wyłącza kursor
blink()	Włącza miganie kursora
noBlink()	Wyłącza miganie kursor
display()	Włącza ekran LCD
noDisplay()	Wyłącza ekran LCD
scrollDisplayLeft()	Przesuwa zawartość LCD w lewo
scrollDisplayRight()	Przesuwa zawartość LCD w prawo
autoscroll()	Automatyczne przesuwanie zawartości na LCD
noAutoscroll()	Wyłączenie automatycznego przesuwania zawartości na LCD
leftToRight()	Ustawia kierunek zapisu tekstu od prawej do lewej
rightToLeft()	Ustawia kierunek zapisu tekstu od lewej do prawej
createChar()	Umożliwia zdefiniowanie własnego znaku

Listing 1. Przykładowy program dla modułu AVT11615

```
/*
Przykład programu do obsługi modułu AVT1615 z:
- wyświetlaczem LCD 2x16 znaków
- 4 diodami LED
- 4 przyciskami
- brzęczykiem piezzo
- czujnikiem temperatury LM35
*/


#include <LiquidCrystal.h>  //biblioteka obsługi LCD

const int Led1 =  13;       //przypisanie aliasów do pinów portów
const int Led2 =  12;
const int Led3 =  11;
const int Led4 =  10;
const int SW1 =  3;
const int SW2 =  2;
const int SW3 =  1;
const int SW4 =  0;
const int Buzzer =  A5;

LiquidCrystal lcd(8, 9, 4, 5, 6, 7);  //konfigurowanie linii do których został
dołączony LCD

int wart_pot;          //zmienna na wartość zmierzona z potencjometru
int wart_czuj;         //zmienna na wartość zmierzonej z czujnika temperatury
float wart_nap;        //zmienna na wartość zmierzonego napięcia
float temperatura;     //zmienna na wartość zmierzonej temperatury

byte st[8] = {         //tablica znaku stopnia dla wyświetlacza LCD
  B00100,
  B01010,
  B00100,
  B00000,
  B00000,
  B00000,
  B00000,
};

void setup() {             //funkcja inicjalizacji
  lcd.begin(16, 2);        //konfigurowanie rozdzielczości LCD
  lcd.createChar(0, st);  //funkcja utworzenia własnego znaku z tablicy st
o kodzie 0
  analogReference(DEFAULT); //konfigurowanie napięcia odniesienia
                            //dla przetwornika A/C - domyślnie 5V
  pinMode(Led1, OUTPUT);    //konfigurowanie I/O, do których są
                            //dołączone diody LED
  pinMode(Led2, OUTPUT);
  pinMode(Led3, OUTPUT);
  pinMode(Led4, OUTPUT);
  pinMode(Buzzer, OUTPUT);  //konfigurowanie I/O, do której jest
                            //dołączony brzęczyk piezzo
  pinMode(SW1, INPUT);      //konfigurowanie I/O, do których są
                            //dołączone przyciski
  pinMode(SW2, INPUT);
  pinMode(SW3, INPUT);
  pinMode(SW4, INPUT);
  digitalWrite(SW1, HIGH);  //dołączenie wewnętrznych rezystorów
                            //zasilających
  digitalWrite(SW2, HIGH);
  digitalWrite(SW3, HIGH);
  digitalWrite(SW4, HIGH);
  digitalWrite(Led1, HIGH);    //wyłączenie diod LED
  digitalWrite(Led2, HIGH);
  digitalWrite(Led3, HIGH);
  digitalWrite(Led4, HIGH);
  digitalWrite(Buzzer, HIGH);  //wyłączenie brzęczyka piezzo
}

void loop() {              //pętla główna programu
  lcd.clear();             //czyszczenie LCD
  lcd.setCursor(4, 0);     //ustawienie kursora w 5 kolumnie
                           //pierwszego wiersza
  lcd.print(„Arduino”);    //wyświetlenie na LCD napisu Arduino
//pomiar napięcia z potencjometru i dodanie wyniku do wart_pot
  for (int i = 0; i < 20; i++) {    //pętla wykonywana 20 razy
    wart_pot = wart_pot + analogRead(A0);
  }
//obliczenie średniej arytmetycznej z 20 pomiarów
  wart_pot = wart_pot / 20;
//przeliczenie odczytanej wartości na napięcie
  wart_nap=(5.0*wart_pot)/1024.0;
//ustawienie kursora w pierwszej pozycji drugiego wiersza LCD
  lcd.setCursor(0, 1);
  lcd.print(„U=”);         //wyświetlenie napisu U=
  lcd.print(wart_nap);     //wyświetlenie napięcia
  lcd.print(„V”);          //wyświetlenie znaku V

  for (int i = 0; i < 20; i++) {    //pętla wykonywana 20
//pomiar napięcia z czujnika temperatury i dodawanie wart_czuj
    wart_czuj = wart_czuj + analogRead(A1);
  }
//obliczenie średniej arytmetycznej z 20 pomiarów
  wart_czuj = wart_czuj / 20;
//przeliczenie wartości na stopnie Celsjusza
temperatura=(5.0*wart_czuj*100)/1024.0;
//ustawienie kursora na pozycji 9 drugiego wiersza LCD  lcd.setCursor(9, 1);
  lcd.print(„T=”);             //wyświetlenie napisu T=
  lcd.print((long)temperatura);
//wyświetlenie wartości temperatury zaokrąglonej do pełnych stopni
  lcd.write(0);                //wyświetlenie znaku stopnia
```

o napięciu od 0 V do 5 V z rozdzielczością 10-bitow. Komendy *pinMode()* definiują sposób pracy portów I/O mikrokontrolera. Linie, które sterują diodami LED oraz generatorem piezzo skonfigurowano jako wyjścia, a linie, do których dołączono przyciski jako wejścia. Za pomocą funkcji *digitalWrite()* dołączono rezystory zasilające pull-up, które polaryzują wejścia. Przyciśnięcie przycisku zmienia poziom dołączonego wejścia na niski.

W funkcji *loop()* znajduje się program główny. W pierwszej kolejności za pomocą komendy *lcd.clear()* jest czyszczony ekran LCD. Następnie, za pomocą komendy *lcd.setCursor(4, 0)* kursor jest ustawiany w pierwszym wierszu i 5 kolumnie LCD (współrzędne numerowane są od 0). Komenda *lcd.print(„Arduino”)* wyświetla od ustawionej pozycji kursora komunikat *Arduino*. Jako parametr funkcji *lcd.print()* można użyć stałych w cudzysłowie lub zmiennych typu: char, byte, int, long, string. Funkcja ta ma również drugi opcjonalny parametr, który umożliwia formatowanie wyświetlanych liczb. Dozwolone są następujące ich formaty: DEC (dziesiętny), BIN (binarny), OCT (ósemkowy) i HEX (szesnastkowy). Następnie, w programie przykładowym za pomocą komendy *analogRead(A0)* jest odczytywane napięcie zmierzone przez przetwornik A/C na wejściu *A0. Jak pamiętamy, jest to napięcie z suwaka potencjometru*. Dla uniknięcia błędów pomiar jest wykonywany 20 razy w pętli *FOR*, jego wynik jest sumowany w zmiennej *wart_pot*, a następnie jest obliczana średnia arytmetyczna z wyników pomiarów. Za pomocą wyrażenia *wart_nap=(5.0*wart_pot)/1024.0* wartość obliczonej średniej jest zamieniana na napięcie. Stała „5.0” to napięcie odniesienia dla przetwornika, natomiast „1024” to jego rozdzielczość. Tak przeliczona wartość zmiennej zostaje wyświetlona w drugiej linii wyświetlacza LCD.

Podobnie w dalszej części programu jest wykonywany pomiar temperatury za pomocą pomiaru napięcia na wyjściu czujnika LM35. Zmiana temperatury o 1ºC powoduje wzrost napięcia na wyjściu czujnika temperatury o 10 mV. Łatwo wywnioskować, że zmiana temperatury o 10ºC będzie odpowiadała zmianie napięcia wyjściowego czujnika o 100 mV. Temperatura jest odczytywana za pomocą funkcji *analogRead(A1)* mierzącej równoważne jej napięcie z czujnika LM35 na wejściu *A1*. Również w tym wypadku jest wykonywane 20 pomiarów, z których jest obliczana średnia arytmetyczna. Następnie za pomocą wyrażenia *temperatura=(5.0*wart_czuj*100)/1024.0* liczba odczytana z rejestru przetwornika A/C jest zamieniana na temperaturę. Wartość temperatury jest wyświetlana na wyświetlaczu w pozycji wskazywanej przez komendę *lcd.setCursor(9, 1)*. Komenda *lcd.write(0)* powoduje wyświetlenie po wartości temperatury symbolu stopnia.

Cyfrowe rejestratory dźwięku ISD151xx (2)

Układy z rodziny ISD151xx to zaawansowane, cyfrowe układy służące do rejestracji dźwięku. Mogą być elastycznie konfigurowane, a wbudowane kompresory/dekompresory pozwalają na zapisanie nawet 30 minut komunikatów słownych w najbardziej pojemnym układzie ISD15132. Producent nie zapomniał też o dźwięku lepszej jakości wyposażając układy w interfejs I²S i możliwość operowania na 16-bitowych nieskompresowanych danych audio.

Dodatkowe materiały na CD/FTP:
ftp://ep.com.pl, user: 10925, pass: 87thc181
• pierwsza część artykułu

Układ zegara systemowego

Zadaniem układu zegara systemowego jest generowanie sygnału Master Clock MCLK taktującego wszystkie bloki cyfrowego toru danych audio z pięciu źródeł:

- Wbudowanego generatora kwarcowego z dołączanym do wejść XTALIN i XTALOUT zewnętrznym oscylatorem kwarcowym.
- Zewnętrznego generatora kwarcowego (sygnał podawany na XTALIN).
- Wewnętrznego generatora RC z rezystorem podłączanym do XTALIN.
- Wbudowanego generatora (bez elementów zewnętrznych).
- Sygnału SCK z portu I²S.

Częstotliwość MCLK jest generowana przez układ PLL jest wyliczana według wzoru: Fmclk=Fosc×N/2M, w którym:

- Fmclk – częstotliwość MCLK,
- Fosc – częstotliwość wejściowa układu PLL
- M, N współczynniki układu PLL

Współczynnik M może przyjmować wartości 1,2, lub 3, a współczynnik N wartości 8, 16 lub 32. Częstotliwość próbkowania jest wyliczana z zależności: Fs=Fmclk/512.

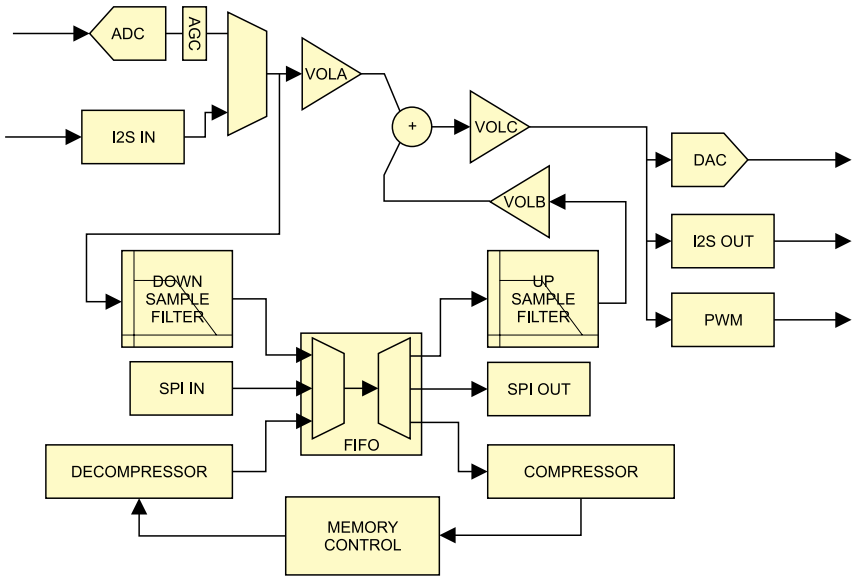
Dla standardowej częstotliwości próbkowania Fs=32 kHz Fmclk=16,385 MHz, dla Fs=44,1 kHz Fmclk=22,5792 kHz i dla fs=48 kHz Fmclk=24,576 MHz.

Po wybraniu częstotliwości próbkowania w układzie downsamplera można ją dzielić na subczęstotliwości próbkowania (SubFs). Pokazano je na **rysunku 9**.

Organizacja pamięci

Wbudowana pamięć Flash na pojemność (zależnie od typu układu) od 2 do 64 Mb. Przestrzeń adresowa zaczyna się od adresu 0. Komórki pamięci są adresowane liniowo za pomocą 24-bitowego licznika adresowego. Jednocześnie pamięć jest podzielona na sektory o pojemności 4 kB. Sektor jest najmniejszą jednostką, która można wyczyścić.

Układy ISD151xx mają wbudowane mechanizmy zarządzania zawartością pamięci pozwalające na efektywne zapisywania i odtwarzanie danych audio. W czasie pracy układu przestrzeń adresowa jest dzielona na 2 części przez zawartość rejestru RMP (*Recording Memory Pointer*) – **rysunek 10**.



Rysunek 8 Droga sygnałów cyfrowych układu ISD151xx

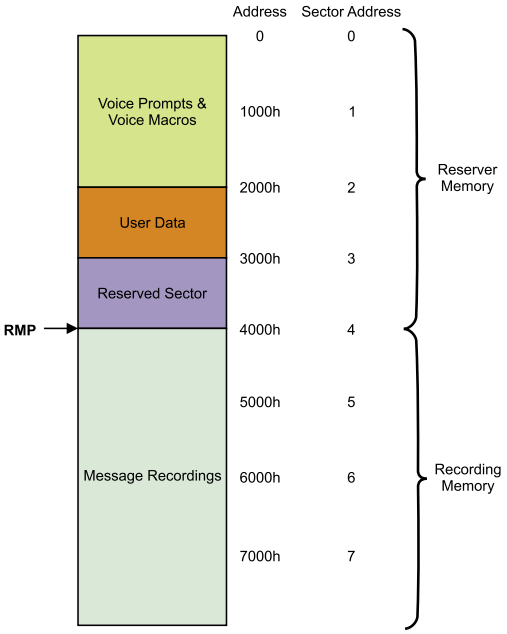
RMP zawiera numer ostatniego sektora pamięci zarezerwowanej. W pamięci zarezerwowanej są zapisane wcześniej dane: dane użytkownika, przygotowane wcześniej i nagrane fragmenty materiału dźwiękowego (*Voice Prompt*), makra i zarezerwowane sektory. Można programowo włączyć protekcję zapisu danych w wektorach poniżej zawartości RMP za wyjątkiem zarezerwowanych sektorów. W strukturze z **rysunku 10** RMP ma wartość 4, czyli adres podziału na pamięć zarezerwowaną i pamięć przeznaczona do nagrywania jest równy szesnastkowo 4000.

Wszystkie sektory powyżej RMP aż do końca pamięci mogą być wykorzystane do nagrywania komunikatów dźwiękowych (Message Recordings).

Bardzo ciekawą cechą ISD151xx jest możliwość nagrywania i odtwarzania Voice Prompt. Są to wcześniej nagrane i zapisane w pamięci komputera krótkie frazy, efekty dźwiękowe, fragmenty muzyczne itp. Po przygotowaniu są przegrywane do pamięci układu i mogą być odtwarzane w dowolnej kolejności i dowolną ilość razy – na przykład w pętli. O sposobie odtwarzania decyduje aplikacja użytkownika.

Lokalizacja Voice Prompt w pamięci jest określona przez swój indeks. Adresowanie indeksowane pozwala na modyfikacje zawartości zapisanych w pamięci Voice Prompt bez konieczności zmiany w aplikacji sterującej. Przykładem wykorzystania tej właściwości jest aplikacja głośnomówiącego termometru. Z nagranych wcześniej liczebników można dowolnie składać komunikaty dźwiękowe.

Drugim silnym wsparciem jest możliwość budowania makr. Makra pozwalają na grupowanie odtwarzania wielu Voice Prompt, ale także innych komend układu w jedna strukturę. Odtwarzanie makra jest wywoływane jedną komendą, co znacznie upraszcza aplikację użytkownika. Makra są identyfikowane w pamięci poprzez adresowania indeksowe, tak jak w przypadku Voice Prompt. Dwa pierwsze indeksy są zarezerwowane dla celów specjalnych: indeks 0 dla makra inicjalizacyjnego wykonywanego po włączeniu zasilania *Power On Initialization* (POI) i indeks 1 dla makra



Podział Fs	Częstotliwość Fs (kHz)		
	32	44,1	48
8	4	5,5125	6
6	5,333	7,35	8
5	6,4	8,82	9,6
4	8	11,025	12
2,5	12,8	17,64	19,2
2	16	22,05	24
1	32	44,1	48

Rysunek 9. Dostępne nastawy częstotliwości SubFs

inicjalizacyjnego wykonywanego po programowym włączeniu zasilania komendą SPI - *Power Up* (PU). W makrach inicjalizacyjnych ustawiany jest układ zegara systemowego i najważniejsze rejestry konfiguracyjne.

Zarezerwowane sektory są grupą 4 kB wolnych (skasowanych) sektorów zarezerwowanych dla pustych sektorów (*Empty Sector*) zapisywanych komunikatów. Pusty sektor jest sektorem początkowym każdego zapisywanego komunikatu. Po wykonaniu komendy startu zapisywania menadżer pamięci adresuje początek nagrywania w pustym sektorze, a po jego zapełnieniu (jeżeli nagrywane dane maja więcej niż 4 kB) adresuje kolejny sektor z obszaru powyżej RMP.

Jak już wspomniałem Voice Prompt i makra są identyfikowane w pamięci poprzez indeks. Na początku pierwszego sektora pamięci (od adresu 0) jest umieszczony nagłówek pamięci składający się z 17 bajtów. Po nagłówku użytkownik zapisuje tablice konwersji alokacji kolejnych makr i Voice Prompt.

Nagłówek i początek tablicy konwersji są pokazane na **rysunku 11**.

Bajt 1 i 2 to 16-bitowy rejestr RMP. RMP jest numerem ostatniego sektora pamięci chronionej. W kolejnych 2 bajtach jest zapisana zawartość rejestru PMP (*Protected Memory Pointer*) związanego z mechanizmami protekcji zapisu opisanymi powyżej. Sześciobajtowe pola POI_VM i PU_VM zawierają 24-bitowe, rzeczywiste adresy początku i końca makr *Power On Initialization* (POI) i *Power Up*. Po tych polach zaczynają się kolejne 6-bajtowe pola Voice Prompt lub makr. W każdym z nich są również zapisane 24-bitowe adresy początku i końca.

Voice Prompt i makra są adresowane w czasie zapisywania lub odtwarzania przez nadanie indeksu. Układ sterowania ISD151xx w czasie odtwarzania na przykład Voice Prompt o indeksie IDX wylicza adres początkowy w tablicy według zależności ADD=6×IDX+5. Dla IDX=2 mamy adres 17.

W czasie wykonywania komendy odtwarzania Voice Prompt lub makra na podstawie znanego indeksu układ sterowania wylicza adres początkowy w pierwszym sektorze pamięci Flash, a następnie pobiera 3 bajty do rejestru adresu początku i 3 kolejne bajty do rejestru adresu końca. Każdy sektor pamięci w którym jest zapisana wiadomość lub część dłuższej wiadomości ma swój nagłówek (**rysunek 14**).

Ilość sektorów, w których jest wyzerowany BOM określa ilość nagranych wiadomości

REKLAMA

Wprowadzenie do Linuksa embedded (3)

Obsługa portów GPIO

Dodatkowe materiały na CD/FTP:
ftp://ep.com.pl, user: 10925, pass: 87thc181
• wszystkie poprzednie części kursu

Porty GPIO są jednym z najprostszych układów peryferyjnych, od obsługi których zwykle rozpoczynamy tworzenie programów zapoznając się z danym typem mikrokontrolera. W wypadku aplikacji pracującej pod RTOS'em lub bez systemu operacyjnego, obsługa portów GPIO sprowadza się do wpisania do rejestrów portu kilku wartości określających jego konfigurację, czyli kierunku wejście lub wyjście, załączenie lub wyłączenie rezystorów podciągających itp. W wypadku systemu Linux jest inaczej, ponieważ programista ma do czynienia z obsługą urządzeń, a porty GPIO umożliwiając ich sterowanie nie są urządzeniem w ogólnym rozumieniu tego słowa.

Po skonfigurowaniu portu, najczęściej wpisując lub odczytując dane z pojedynczego rejestru, mamy możliwość zmiany stanu linii wyjściowej lub odczytu stanu linii wejściowej. W podobny sposób są zbudowane porty mikrokontrolerów rodziny AT91, do której należy m.in.AT91RM9200. Mamy w nim możliwość sterowania linia GPIO za pomocą kilkunastu rejestrów. W wypadku Linuksa sytuacja komplikuje się, ponieważ bezpośredni dostęp do portów wejścia wyjścia z poziomu programu użytkownika nie jest możliwy, a komunikacja pomiędzy urządzeniami a oprogramowaniem musi odbywać się za pomocą sterowników. Istnieje możliwość obejścia tego za pomocą sterownika /dev/mem, ale nie jest to rozwiązanie zalecane. Brak bezpośredniego dostępu do rejestrów oraz całej fizycznej przestrzeni adresowej jest podyktowany względami bezpieczeństwa. Nietrudno sobie wyobrazić aplikację, która w wyniku zapisania błędnych danych powoduje awarię całego systemu. W Linuksie nie istnieje jeden generyczny mechanizm dostępu do portów GPIO z przestrzeni użytkownika, o tym dlaczego tak jest napiszemy w dalszej części tekstu. W części praktycznej natomiast pokażemy w miarę bezpieczny i uniwersalny sposób na dostęp do portów GPIO w Linuksie z programu użytkownika, na przykładzie płyty prototypowej BF210 za pomocą jednego z mechanizmów – SYSFS GPIO, który nie jest jednak całkowicie zgodny z filozofią sterowników urządzeń.

Sposoby dostępu do portów GPIO w systemie Linux

W Linuksie długo nie istniał standardowy, bezpośredni sterownik linii GPIO, co ma swoje uzasadnienie. Do portów GPIO najczęściej dołączamy jakieś proste układy, na przykład diody LED, przyciski itp. Zgodnie z filozofią systemu Linux właśnie te dołączone urządzenia, czyli wspomniane diody LED i przyciski, powinny mieć swoje sterowniki udostępniające je aplikacjom użytkownika. Porty GPIO stanowią tylko mechanizm dołączania urządzeń, a nie urządzenie samo w sobie. To do jakiego wejścia lub wyjścia dane urządzenie jest dołączone oraz sposób jego sterowania nie powinien mieć znaczenia dla aplikacji. I to jest właśnie dodatkowe zadanie sterowników urządzeń, które stanowią dodatkową warstwę abstrakcji pomiędzy sprzętem a aplikacją użytkownika. Na przykład dioda LED fizycznie może być dołączona bezpośrednio do linii GPIO, ale równie dobrze może być do układu ekspandera I²C. Jednak to gdzie jest ona dołączona z punktu widzenia funkcjonalności aplikacji nie ma żadnego znaczenia.

Na początek zajmiemy się najbardziej eleganckim sposobem dostępu do linii GPIO, czyli sterownikami funkcjonalnymi, które używają pojedynczo linii GPIO. Są to mianowicie sterownik diod LED (LED class driver) oraz sterownik podsystemu Linux Input System dla klawiatury dołączonej do linii GPIO, czyli „GPIO Keys".

Sterownik diod LED (Led Class Driver)

Sterownik LED Class Driver, umożliwia kontrolę diod LED z aplikacji pracującej w przestrzeni użytkownika. Zapewnia on zaawansowany interfejs dla LED'ów umożliwiający ich: włączanie, wyłączanie, włączanie na określony czas bez interwencji aplikacji i sterowanie jasnością świecenia. Funkcja sterowania jasnością zależy od możliwości sprzętu i nie zawsze jest dostępna.

Aby używać sterownika LED Class Driver należy na etapie konfiguracji jądra (Device Drivers –> LED Devices) włączyć następujące opcje:

- LED Class support,
- LED Support for GPIO connected LEDS.

Pierwsza opcja włącza w kernelu obsługę LED Class Drivers, natomiast druga umożliwia używanie przez nią linii GPIO. Z uwagi na to, że w tego rodzaju urządzeniach nie istnieje system powiadamiania o dostępnych zasobach sprzętowych, takich jak BIOS, należy poinformować kernel, do której linii jest dołączona dana dioda oraz jaka jest jej nazwa. Przypisanie poszczególnych urządzeń jest realizowane w kodzie jądra Linuksa, odpowiedzialnym za inicjalizację maszyny. Na przykład, aby diodom o nazwach led0 oraz led1 przypisać sterowanie za pomocą portów PC2 oraz PC3, należy zadeklarować strukturę jak na listingu 8 oraz zarejestrować ją w systemie wywołując funkcję:

```
/* LEDs */
at91_gpio_leds(my_leds, ARRAY_
SIZE(my_leds));
```

Dopisanie diod LED w pliku konfiguracyjny maszyny będzie skutkowało tym, że w katalogu /sys/class/led0 oraz /sys/class/led1 pojawią się następujące pliki umożliwiające kontrolowanie dołączonych diod LED:

- brightness: umożliwia sterowanie jasnością diody LED. W wypadku implementacji za pomocą linii GPIO jak powyżej, wpisanie do tego pliku wartości 0 powoduje wyłączenie diody, natomiast wpisanie wartości różnej od 0 spowoduje jej włączenie.
- trigger: umożliwia ustawienie zaawansowanych wyzwalaczy np. mruganie

Listing 8. Struktura przypisująca diodom LED0 i LED1 sterowanie za pomocą wyprowadzeń PC2 i PC3
```
static struct at91_gpio_led my_leds[] = {
  {
    .name           = "led0",
    .gpio           = AT91_PIN_PC2,
    .trigger        = "timer",
  },
  {
    .name           = "led1",
    .gpio           = AT91_PIN_PC3,
    .trigger        = "timer",
  }
};
```

Listing 9. Deklaracja struktury, która spowoduje generowanie zdarzeń na skutek zbocza opadającego na liniach PC2 i PC3
```
static struct gpio_keys_button my_buttons[] = {
  { .desc = „Key0",
    .gpio = AT91_PIN_PC2,
    .active_low = 1,
    .code = KEY_C,
    .type = EV_KEY,
  },
  { .desc = „Key1",
    .gpio = AT91_PIN_PC2,
    .active_low = 1,
    .code = KEY_D,
    .type = EV_KEY,
  },
};

static struct gpio_keys_platform_data my_button_data = {
  .buttons = my_buttons,
  .nbuttons = ARRAY_SIZE(my_buttons),
};

static struct platform_device gpio_keys_dev = {
  .name = "gpio-keys",
  .id = 0,
  .dev =
  {
    .platform_data = &my_button_data,
  }
};
```

Lisiting 10. Rejestracja zdarzeń z listingu 1 w kodzie maszyny
```
for(i=0; i < ARRAY_SIZE(my_buttons); i++)
   at91_select_gpio(my_buttons[i].gpio, 0);
   platform_device_register(&gpio_keys_dev);
```

podczas aktywności interfejsu USB lub karty sieciowej bez ingerencji programu zewnętrznego np. wpisanie do tego pliku wartości eth0 spowoduje mruganie diody LED w momencie aktywności interfejsu sieciowego. Wpisanie wartości timer umożliwia sterowanie mruganiem diod LED.

– delay_on, delay_off: wpisanie do tych plików wartości włączonej w ms określa stosunek czasu włączenia do wyłączenia diody, co umożliwia jej miganie z wybranym wypełnieniem, gdy jako trigger wybrany został timer.

Na przykład, aby włączyć diodę led0 wystarczy z wiersza polecenia, wpisać:

```
echo 1 > /sys/class/led/led1/brightness
```

Sterownik klawiatury z użyciem linii GPIO

W Linuksie obsługa klawiatury jest realizowana za pośrednictwem podsystemu Linux Input System zapewniającego obsługę urządzeń wejściowych, takich jak myszki i klawiatury za pomocą jednolitego interfejsu. Istnieje możliwość uruchomienia sterownika GPIO Keys, który umożliwia współpracę prostych klawiszy dołączonych do linii GPIO z podsystemem Linux Input System. Dzięki uruchomieniu tego sterownika zyskujemy możliwość generowania zdarzeń w wyniku zmiany stanu linii GPIO. Jest to istotne, ponieważ aplikacja korzystająca z podsystemu Linux Input System może zostać uśpiona do czasu wystąpienia zdarzenia od wciśnięcia klawisza. Należy pamiętać, że pracujemy w systemie wielozadaniowym i należy unikać aktywnego odpytywania o stan linii, tak jak to robi się w przypadku aplikacji dla mikrokontrolerów. Aby umożliwić działanie klawiatury dołączonej do linii GPIO, należy w kernelu uaktywnić następujące opcje:

– Event Interface,
– Keyboyard,
– GPIO buttons.

Driver ten również wymaga zdefiniowania struktur w pliku inicjalizacyjnym maszyny zawierających informacje o rodzaju generowanych zdarzeń oraz o przypisanych do nich liniach GPIO. Na przykład, jeżeli chcemy, aby zbocze opadające na liniach PC2 i PC3 wygenerowało zdarzenie od klawiszy „C" i „D", należy zdefiniować struktury, które umieszczono na listingu 9, a następnie zarejestrować w kodzie inicjalizacyjnym maszyny (listing 10).

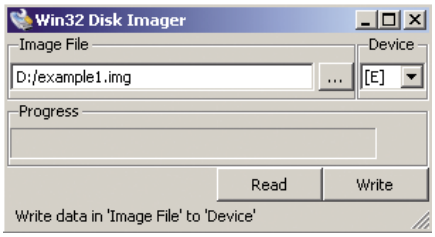
Od strony aplikacji przestrzeni użytkownika dostęp do wciśniętych klawiszy jest realizowany za pomocą urządzenia /dev/input/event0 udostępniającego ich kody. Za pomocą funkcji blokującej read można uśpić bieżący proces do momentu zaistnienia zdarzenia wciśnięcia klawisza, co jest bardzo ważne w przypadku środowiska, w którym może być uruchomionych wiele procesów.

Metoda brute-force, czyli /dev/mem

Przy omawiania sposobu dostępu do portów GPIO z powodów czysto porządkowych omówimy metodę, której nie powinno stosować się pod żadnym pozorem, a która bardzo często jest używana przez początkujących użytkowników. W Linuksie istnieje sterownik reprezentowany przez plik /dev/mem, którego zadaniem jest udostępnianie całej pamięci fizycznej. Dostęp do tego pliku ma standardowo tylko administrator (root). Użycie tego pliku umożliwia dostęp do fizycznej przestrzeni adresowej, a zatem również do portów GPIO. W Linuksie istnieje również możliwość użycia funkcji mmap, która umożliwia mapowanie pliku w przestrzeni adresowej procesu. Zatem stosując sterownik /dev/mem oraz funkcje mmap możemy uzyskać dostęp do dowolnych rejestrów mikrokontrolera. Przykład użycia /dev/mem wraz z mmap w celu uzyskania dostępu do pamięci fizycznej o adresie base i rozmiarze 1 strony przedstawiono na listingu 11.

Działanie funkcji sprowadza się do otwarcia pliku /dev/mem, a następnie mapowanie tego pliku w przestrzeni adresowej

Listing 11. Przykład użycia /dev/mem wraz z mmap w celu uzyskania dostępu do pamięci fizycznej o adresie base i rozmiarze 1 strony
```
uint32_t *mmap_base_address(unsigned base)
{
  int hwnd = open(„/dev/mem",O_RDWR|O_SYNC);
  if(hwnd<0)
  {
    //error handling
  }
  //Map to process space and calculate address
  unsigned long map_addr = base & ~(getpagesize()-1);
  void *mem = mmap(0,getpagesize(),PROT_READ|PROT_WRITE,MAP_SHARED,hwnd,map_
addr);
  if(!mem)
  {
    //Error handling
  }
  uint32_t *regs = (uint32_t*) mem + (base-map_addr)/4;
  return regs;
}
```



Rysunek 7. Okno wyboru identyfikatora dysku

brym rozwiązaniem w przypadku środowisk wielozadaniowych. Wówczas dużo lepszym rozwiązaniem byłoby użycie sterownika GPIO KEYS i podsystemu Input. Takie rozwiązanie zapewniałoby uśpienie procesu do momentu wciśnięcia klawisza bez konieczności cyklicznego sprawdzania stanu wejścia KEY.

Uruchomienie przykładu, na BF210

Skompilowany przykład wraz z podstawowym system Linux jest dostarczany w postaci obrazu karty SD, w postaci pojedynczego pliku example1.img W systemie Windows obraz z przykładem można nagrać na kartę SD za pomocą programu Win32diskImager, który można pobrać ze strony *https://wiki.kubuntu.org/Win32DiskImager*. Do czytnika kart SD należy włożyć kartę, a następnie podłączyć czytnik do komputera. Teraz uruchomić program i jako opcje image file wybrać plik example1.img, natomiast jako device należy wybrać identyfikator dysku, którym jest czytnik kart SD (rysunek 7). Kliknięcie na guzik Write powoduje rozpoczęcie zapisywania obrazu karty. Z poziomu systemu Linux, obraz można nagrać na kartę za pomocą standardowego polecenia dd (dd if=example.img of=/dev/identyfikator_urz_czytnika).

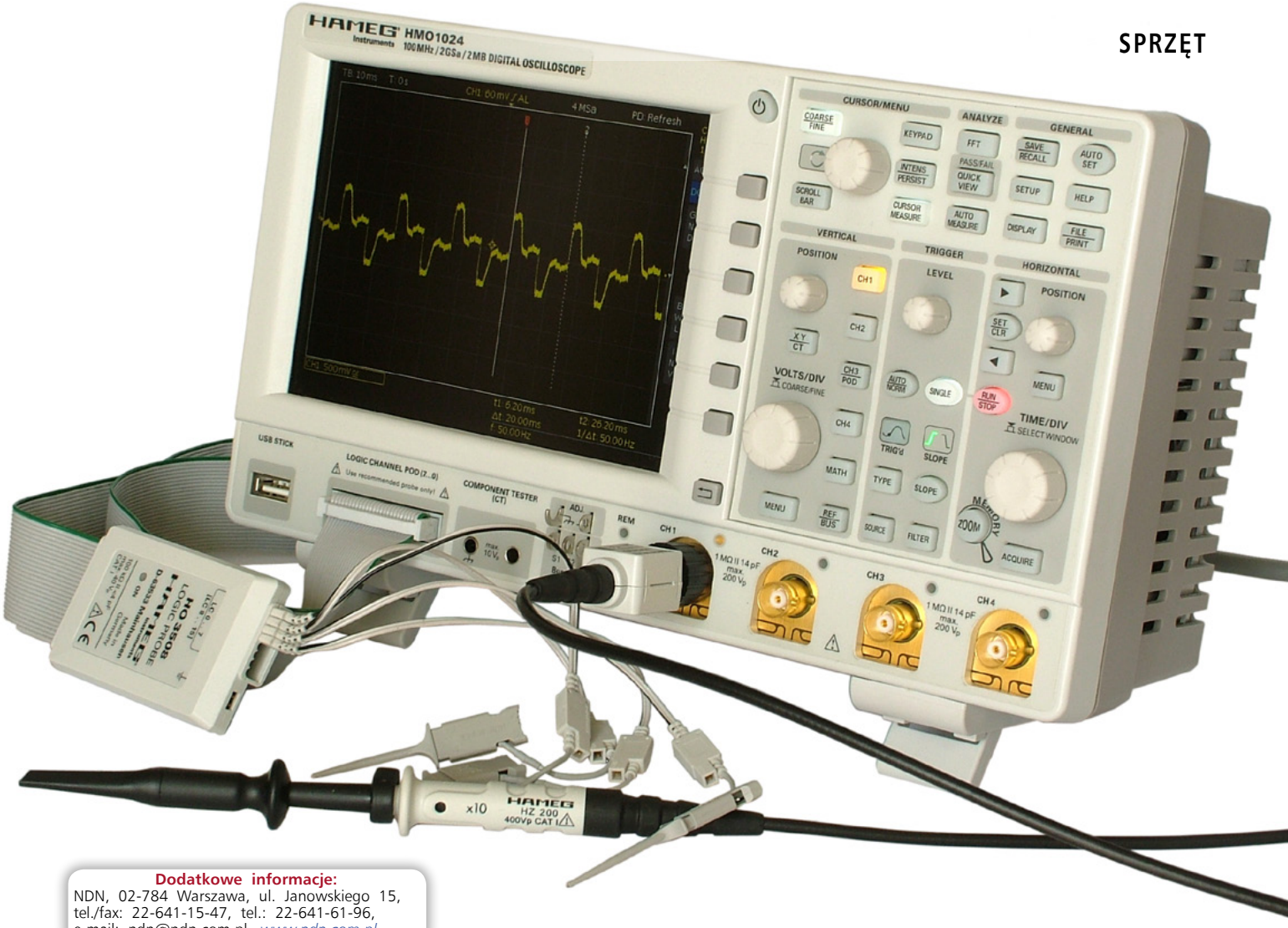
Po nagraniu obrazu karty należy ją umieścić w gnieździe karty komputera BF210, do portu DBGU należy dołączyć komputer PC, na którym powinien być uruchomiony program terminalowy (Hyperterminal, minicom itp.) skonfigurowany z następującymi parametrami transmisji 115200, n, 8, 1. Po włączeniu zasilania powinien być widoczny proces bootowania Linuksa, który powinien zakończyć się wyświetleniem zachęty do logowania bf210-at91 login:. Jako nazwę użytkownika należy wpisać root, a na pytanie o hasło wcisnąć klawisz Enter. Po zalogowaniu się należy wpisać polecenie sysgpio, co spowoduje uruchomienie opisanego wcześniej przykładu. Wciskając klawisz „1" możemy obserwować jak zmienia się stan diod LED1...LED4 reprezentujący licznik binarny. Zakończenie programu jest możliwe poprzez wciśnięcie kombinacji klawiszy CTRL+C, co spowoduje wypisanie komunikatu pożegnalnego oraz zakończenie pracy programu.

Kompilowanie programu ze źródeł

System Linux oraz wszystkie przykłady zostały przygotowane z wykorzystaniem pakietu OpenEmbedded. Sposób skompilowania powyższego przykładu oraz systemu wykracza poza ramy niniejszego artykułu. Dodatkowe pakiety dla OpenEmbedded oraz konfiguracja dla BF210 są dostarczane w postaci pliku oe.tar.gz. Aby skompilować powyższy przykład, należy skonfigurować OpenEmbeded zgodnie z dokumentacją dostępną na stronie projektu i dołączyć pakiety lokalne. Aby zbudować kompletny obraz dla przykładu, należy wydać polecenie bitbake ep-examples-image. Aby zbudować tylko pakiet z powyższym przykładem, należy wydać polecenie bitbake ep-example-sysfsgpio

Lucjan Bryndza, EP

REKLAMA



Dodatkowe informacje:
NDN, 02-784 Warszawa, ul. Janowskiego 15,
tel./fax: 22-641-15-47, tel.: 22-641-61-96,
e-mail: ndn@ndn.com.pl, *www.ndn.com.pl*

Hameg HMO1024
Oscyloskop edukacyjny

Sądząc po ilości ofert na oscyloskopy cyfrowe można przypuszczać, że ten sprzęt nie dożywa swojej naturalnej śmierci. Zanim wszystko w nim się zepsuje, ląduje na wysypisku śmieci, a jego miejsce zajmują kolejne, nowocześniejsze i bardziej wydajne modele. Tak szybki postęp nie stwarza użytkownikom komfortowej sytuacji, gdyż już w chwili zakupu nie są pewni czy dokonali odpowiedniego wyboru. Pewną gwarancję daje marka sprzętu. Hameg należy do tych firm, które cieszą się bardzo dobrą opinią użytkowników.

Dożyliśmy czasów, w których o oscyloskopach z klasyczną lampą obrazową pamiętają już tylko najstarsi elektronicy. Technika analogowa bezpowrotnie odchodzi w zapomnienie, wycofują się z niej najbardziej wytrwali producenci sprzętu. Takim „ostatnim Mohikaninem" jest dobrze u nas znana niemiecka firma Hameg. Wprawdzie nadal w jej ofercie znajdują się analogowe modele oscyloskopów, ale ich dni są już jednak policzone. W czerwcu 2010 roku inny niemiecki potentat w dziedzinie produkcji aparatury pomiarowej przeznaczonej głównie dla telekomunikacji, informatyki i laboratoriów EMC - firma Rhode&Schwarz, zaanonsowała swój pierwszy oscyloskop cyfrowy. Tak też złożyło się, że nie tak dawno R&S wykupił Hamega. Czy zatem decyzja o rezygnacji z produkcji oscyloskopów analogowych nie jest związana z przemianami własnościowymi? Oscyloskopy cyfrowe Hamega nadal jednak będą ukazywały się pod nazwą „Hameg", natomiast te modele, które zostały zaprojektowane w biurach konstrukcyjnych R&S będą produkowane z logo „Rhode&Schwarz". Zatem nic nie wskazuje na to, że marka Hameg zniknie z rynku, wręcz przeciwnie, ma ona mocne podstawy do dalszego istnienia. Oficjalna nazwa brzmi obecnie: Hameg Instruments A Rhode&Schwarz Company. Można oczekiwać, że połączenie sił znajdzie swoje pozytywne odzwierciedlenie w jakości, cenie i parametrach nowych przyrządów.

Aby nauka nie poszła w las

Opisywany w artykule oscyloskop ma dość interesującą genezę. Mianowicie kiedyś, w epoce sprzętu analogowego, nauczyciele prowadzący zajęcia praktyczne w pracowniach i laboratoriach szkół o profilu technicznym (elektronicznym), mieli swoje ulubione ćwiczenie, które zadawali biednym uczniom. Polegało ono na całkowitym,

re można umieścić w obudowie o kształcie dzwona, wokół rozrusznika silnika (**fotografia 3**).

Małe wartości ESL

Krótkie wyprowadzenia i odpowiednia budowa wewnętrzna kondensatorów zapewniają utrzymanie niskiej indukcyjności zastępczej. Jest ona zależna od modelu i wynosi tylko kilka nanohenrów. Jest to szczególnie istotne w przypadku występowania szybkich przełączeń obwodu, gdyż zapobiega powstawaniu bardzo wysokich napięć Dzięki temu nie ma potrzeby stosowania dodatkowych układów zabezpieczających tranzystory IGBT, które tłumiłyby nadmierne napięcia.

Ważna jest też wysoka wartość maksymalnego prądu chwilowego PCC. W przypadku pracy silnika w ruchu miejskim, częste ruszanie i zatrzymywanie pojazdu powoduje nagłe, chwilowe wzrosty prądów, o wartościach znacząco przekraczających znamionowe prądy kondensatorów (**rysunek 4**). Elementy stosowane w układach pośredniczących muszą być w stanie wytrzymywać takie prądy, bez doznawania jakichkolwiek uszkodzeń. Omawiane podzespoły marki Epcos wytrzymują prądy chwilowe o wartościach nawet do 2000 A (w zależności od wersji), tj. około 10 razy więcej niż ich prąd znamionowy.

Żywotność i wytrzymałość

Kondensatory PCC cechują się bardzo długim czasem życia, który wynosi 15 tys. godzin. Ponadto ich znamionowa temperatura pracy wynosi 110°C, ale mogą wytrzymywać jej krótkotrwałe wzrosty nawet do 125°C.

Tabela 2. Wybrane parametry kondensatorów PCC użytych do produkcji modułów HybridPACK1 i HybridPACK2

Parametr/Model	B25655J4307K001	B25655J4507K005
Użyty w module	HybridPACK1	HybridPACK2
Napięcie znamionowe [V_{DC}]	450	450
Pojemność [μF] ±10 %	300	500
Moc znamionowa [Ws]	30	50
Prąd znamionowy [A]	80	120
Temperatury pracy [°C]	od –40 do +110	od –40 do +110
ESR [mΩ]	≤8	≤10
ESL [nH]	≤25	≤15
Wymiary [mm]	140×72×50	237×72×50
Czas życia [h]	15000	15000

PCC są także bardzo bezpieczne. Ich ewentualne przeciążenie nie powoduje samozapłonu ani zniszczenia. Uszkodzenia warstw wewnątrz obudowy kondensatora nie prowadzą do zwarcia, a brak substancji łatwopalnych lub kwasów sprawia, że nie powodują one zagrożenia w przypadku wypadku drogowego.

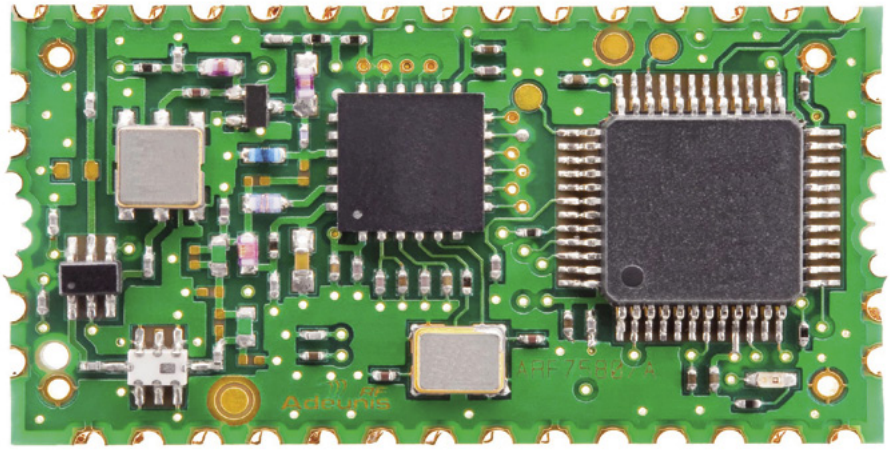
Wszystkie powyższe czynniki sprawiają, że kondensatory Power Capacitor Chip idealnie nadają się do zastosowań zarówno w motoryzacji, jak i do tworzenia niewielkich gabarytowo, przemysłowych przetwornic częstotliwości. Są to zarazem bardzo zaawansowane i sprawdzone już produkty.

Epcos

REKLAMA

STM32W
Cortex-M3 i zaawansowany tor RF

Rodzina STM32W poszerza obszary aplikacyjne nowoczesnych mikrokontrolerów 32-bitowych o systemy radiowe działające w pasmach ISM. Dzięki zintegrowaniu z rdzeniem Cortex-M3 kompletnego transceivera radiowego zgodnego z IEEE 802.15.4, otworem przed tymi mikrokontrolerami stoją aplikacje sieciowe jak na przykład ZigBee…



Układy STM32W…

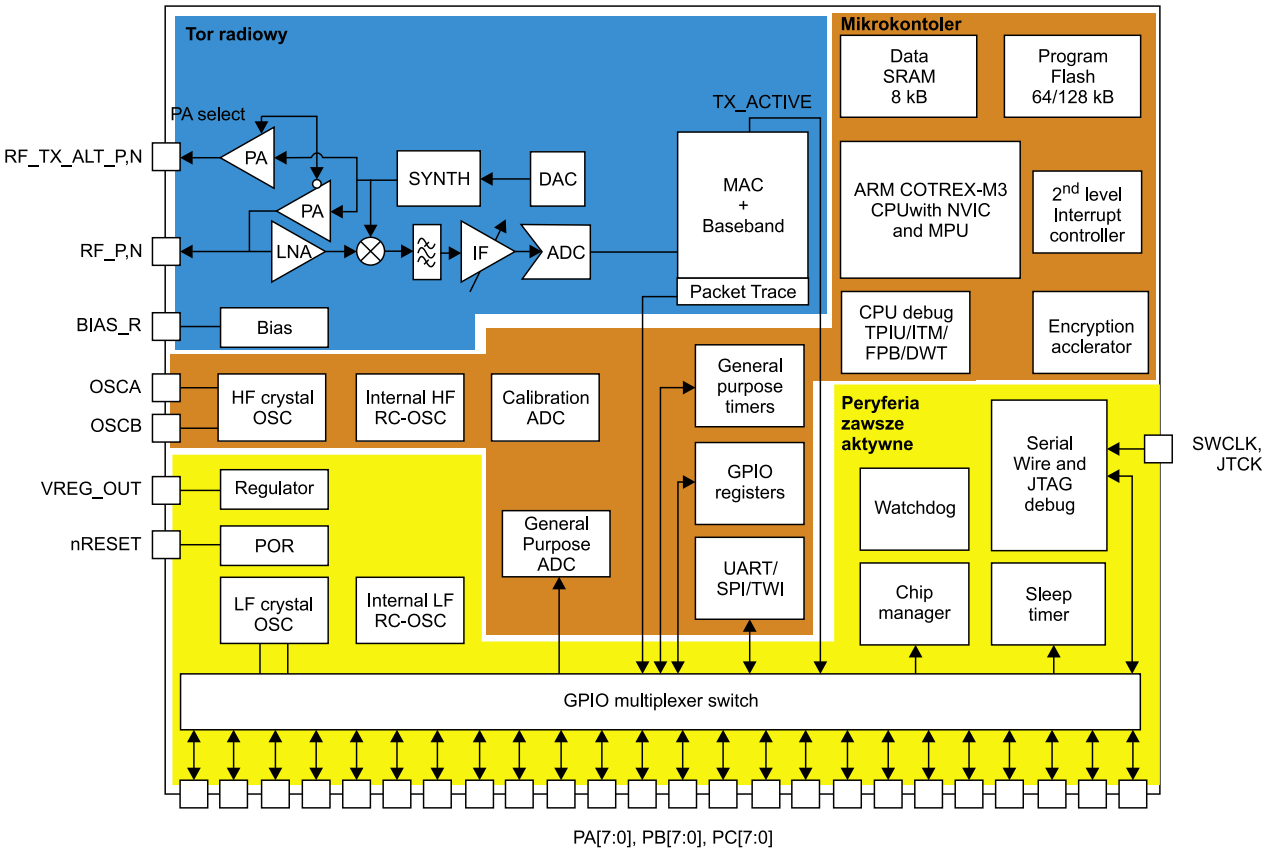
…to jednochipowe „kombajny" składające się z 32-bitowego mikrokontrolera z rdzeniem Cortex-M3 (jak klasyczne wersje STM32, taktowany do 24 MHz) zintegrowanego z kompletnym torem radiowym zgodnym ze specyfikacją IEEE802.15.4 na pasmo 2,4 GHz. Jego schemat blokowy pokazano na **rysunku 1**, a uproszczony widok budowy wewnętrznej – na **rysunku 2**.

Producent oferuje obecnie trzy wersje mikrokontrolerów STM32W108, z których każda jest wyposażona w 8 kB SRAM, timery, interfejsy komunikacyjne i inne bloki peryferyjne:

- STM32W108C8: 64 kB Flash, obudowa VQFPN48 (7×7 mm),
- STM32W108CB: 128 kB Flash, obudowa VQFPN48 (7×7 mm),
- STM32W108HB: 128 kB Flash, obudowa VQFPN40 (6×6 mm),

Dodatkowe informacje:
Dodatkowe informacje o układach przedstawionych w artykule są dostępne pod adresem www.st.com/stm32w

Tor nadawczy wbudowany w układy STM32W ma moc wyjściową do +7 dBm, zbudowano go na bazie syntezera DDS. W torze odbiorczym układu zastosowano heterodynę z niską częstotliwością pośrednią, dzięki czemu czułość dochodzi do –100 dBm.

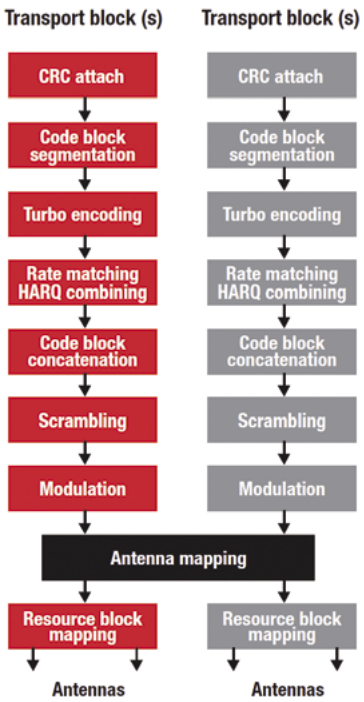


Rysunek 1. Schemat blokowy mikrokontrolerów STM32W108

Telekomunikacja czwartej generacji dzięki układom Texas Instruments

Wykładniczy wzrost ilości danych przesyłanych przez urządzenia mobilne sprawił, że operatorzy sieci komórkowych stanęli przed nowymi wyzwaniami. Na szczęście, metody transmisji bezprzewodowej wciąż ewoluują, a technologia LTE (Long Term Evolution) stała się światowym standardem komunikacji przyszłości.

LTE jest obecnie wprowadzana przez wszystkich wielkich operatorów sieci komórkowych na świecie. Pozwala na znacznie lepsze wykorzystanie dostępnego pasma częstotliwości, umożliwiające przesyłanie większej liczby bitów na sekundę w przeliczeniu na herc pasma, niż w przypadku starszych technologii. Jednakże implementacja LTE nie byłaby możliwa bez odpowiednich układów scalonych. W praktyce, konstruktorzy stacji bazowych Node B dla omawianej technologii stosują układy SoC (*System-On-Chip*), co pozwala zmniejszyć koszty działania gotowego urządzenia i zwiększyć jego niezawodność. Przykładem są układy SoC o wielordzeniowej architekturze KeyStone firmy Texas Instruments.



Rysunek 1. Procedura przetwarzania danych przesyłanych ze stacji bazowej do użytkownika w standardzie LTE

TI KeyStone

Układy z rodziny KeyStone zostały zaprojektowane tak, by zoptymalizować wydajność obsługę standardów WCDMA (*Wideband Code Division Multiple Access*) i LTE, przy jednoczesnym zmniejszeniu poboru mocy przez stację bazową. Co więcej, w przypadku instalacji omawianych układów wewnątrz Node B, duże znaczenie odgrywają wbudowane, konfigurowalne koprocesory obsługujące warstwę fizyczną (warstwę 1) standardów bezprzewodowych. Dzięki temu świetnie sprawdzają się w LTE.

LTE 3GPP

LTE to najnowszy standard mobilny opracowany przez zrzeszenie 3GPP (*Third Generation Partnership Project*). Uwzględniono w nim wiele nowoczesnych technologii, których zabrakło w tzw. telekomunikacji 3G. Teoretycznie LTE pozwala na transfer danych z szybkością przynajmniej do 100 Mb/s w kierunku: od stacji do użytkownika oraz przynajmniej do 50 Mb/s w przeciwnym, przy użyciu 20-megahercowego pasma. Realne wartości zależą od konkretnych realizacji i mogą być nawet wyższe. Co więcej, LTE pozwala elastycznie dostosowywać ilość używanego pasma w zakresie od 1,4 do 20 MHz oraz wybrać sposób multipleksowania różnych strumieni danych: z podziałem częstotliwości (FDD – *Frequency-Division Duplexing*) lub z podziałem czasu (TDD – *Time-Division Duplexing*).

Warstwy LTE

Podstawą stosu protokołu komunikacyjnego LTE jest warstwa fizyczna (PHY), czasami określana mianem warstwy pierwszej. To od niej zależy utrzymanie łącza pomiędzy stacją bazową a urządzeniem mobilnym bez przerw i zakłóceń. Kolejne warstwy to: druga – dostępu do medium (MAC – *Media Access Control*) i trzecia – kontroli zasobów radiowych (RRC – *Radio Resource Control*). Wraz z pierwszą, umożliwiają wyższym warstwom transfer danych przez łącze bezprzewodowe. Warstwa fizyczna obsługuje kodowanie kanałów, moduluje sygnał i automatyczne powtarza transmisję, gdy wystąpi taka potrzeba. Ponadto to od niej zależy obsługa wielu anten dla pojedynczego połączenia.

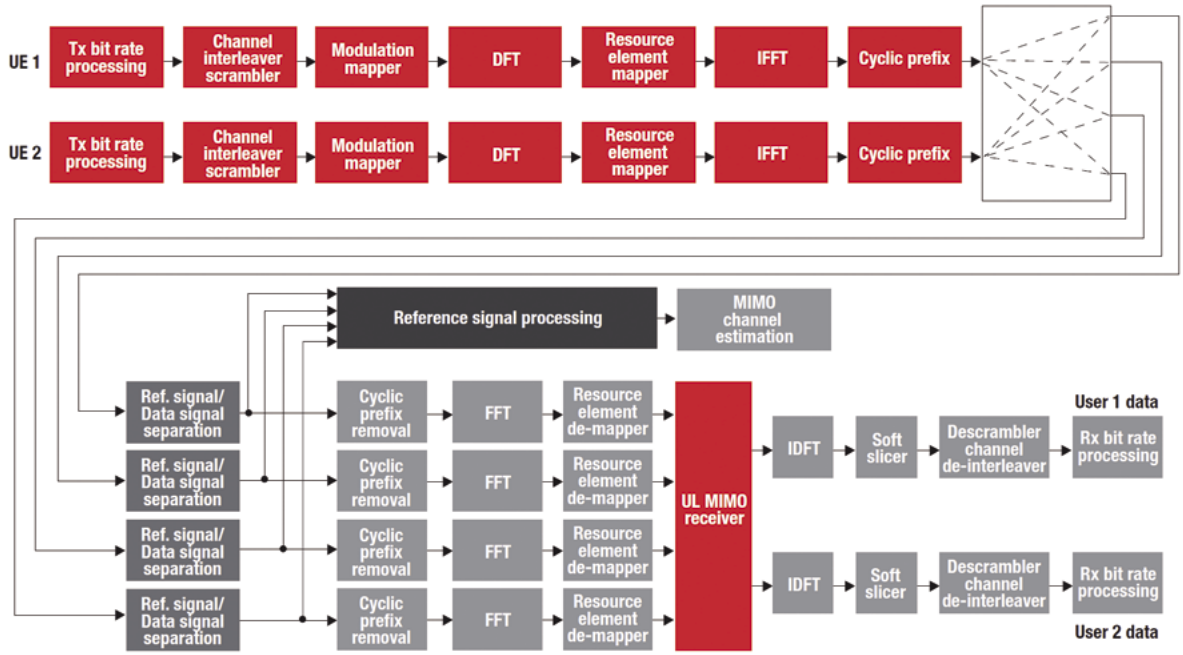
Dodatkowe informacje:
Bogatą ofertę produktową układów Texas Instruments można znaleźć na stronie internetowej http://pl.farnell.com/

Działanie warstwy fizycznej

Po odebraniu bloku danych z warstwy MAC, warstwa fizyczna oblicza jego sumę kontrolną (CRC – *Cyclic Redundancy Check*) i dołącza ją do bloku. Jeśli łączny rozmiar takiego pakietu przekracza maksymalny możliwy, tj. 6144 bity, jest on dzielony na segmenty. Dla każdego z nich obliczana jest nowa suma kontrolna. Następnie obliczany jest turbo-kod oraz dopasowywanie szybkości transmisji do dostępnych zasobów. Obsługiwany jest również protokół HARQ (*Hybrid Automatic Repeat Request*), który ułatwia odbiornikowi określenie, czy konieczna jest retransmisja pakietu. Po złączeniu ze sobą nadawanych bloków danych przeprowadzane jest kodowanie bitów, tak by skrócić wszelkie długie łańcuchy zer i jedynek, które mogłyby powodować problemy z synchronizacją odbiornika. Kolejnym krokiem jest modulacja: QPSK (*Quadrature Phase Shift Keying*), 16QAM (*Quadrature Amplitude Modulation*) lub 64QAM. Zmodulowany sygnał jest następnie wstępnie mapowany z uwzględnieniem transmisji za pomocą kilku anten, po czym jest on modulowany algorytmem OFDM, który przypisuje symbole danych dostępnym antenom do transmisji (**rysunek 1**).

OFDM

OFDM (*Orthogonal Frequency-Division Multiplexing*) to metoda stosowana w LTE do modulowania transmisji radiowej. Polega ona na jednoczesnym przesyle strumieni danych na ortogonalnych częstotliwościach



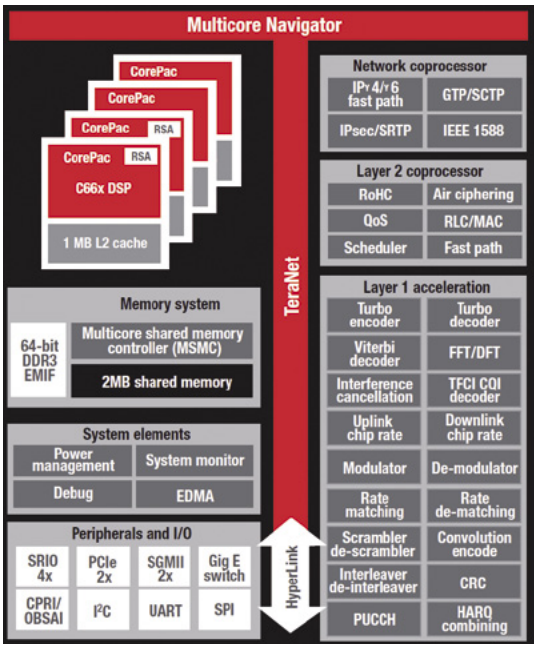
Rysunek 2. Obsługa technologii MIMO w LTE

nośnych. Zapewnia to dużą odporność na degradację sygnału w przypadku gdy w kanałach radiowych występuje zjawisko wielodrogowości. Umożliwia bardziej efektywne wykorzystanie dostępnego pasma i redukuje problemy związane z interferencjami międzysymbolowymi. Zmniejsza też problemy związane z synchronizacją czasową odbieranego sygnału. Implementacja modulacji OFDM jest dosyć prosta i wymaga zastosowania szybkiej transformaty Fouriera.

MIMO

Pełna obsługa LTE wymaga także implementacji technologii MIMO (*Multiple Input, Multiple Output*), zarówno po stronie nadajnika, jak i odbiornika. Polega ona użyciu więcej niż jednej anteny do odbioru i do nadawania sygnału. Jej użycie znacząco zwiększa maksymalną dostępną przepustowość łącza oraz obszar terenu, na którym sygnał jest wystarczający do prowadzenia komunikacji, bez potrzeby poszerzania pasma lub mocy nadawczej. W praktyce prowadzi ona do zwiększenia efektywności spektralnej łącza oraz jego niezawodności. Model przetwarzania danych nadajnika i odbiornika MIMO 2×4 (uplink) został przedstawiony na **rysunku 2**.

Użycie więcej niż jednej anteny do odbioru zwiększa stosunek SNR sygnału. Jeśli odbierany sygnał jest zaszumiony, warto stosować technikę MRC (*Maximum-Ratio Combining*). Jeśli większy problem niż szum stanowią interferencje, należy skorzystać z algorytmu MMSE (*Minimum Mean Square Error*). Użycie dowolnej z nich pozwala ustalić wagi, z jakimi sumowane są sygnały odbierane za pomocą poszczególnych anten. Wydajność tych operacji może być znacznie zwiększona, przy zastosowaniu przetwarzania zmiennoprzecinkowego w odbiorniku.



Rysunek 3. Schemat blokowy układu TMS320TCI6618

Układy SoC do LTE

Wszystkie powyższe procedury obsługuje układ TCI6618, który należy do rodziny wielordzeniowych procesorów sygnałowych TMS320C66x firmy Texas Instruments. Jego schemat blokowy został zilustrowany na **rysunku 3**. Zawiera cztery rdzenie C66x taktowane zegarem 1,2 GHz, zbudowane w oparciu o sprawdzoną architekturę. Dodatkowo zostały wzbogacone o możliwość przetwarzania zmiennoprzecinkowego oraz zmodernizowano sposób przekazywania danych wewnątrz jednostki. Cechy te sprawiają, że układ idealnie nadaje się do zastosowań w szybkich sieciach czwartej generacji.

TCI6618 obsługuje dwa pasma o szerokości 20 MHz każde i dla każdego z nich pozwala odbierać sygnały z dwóch anten i nadawać za pomocą dwóch anten, co w skutkuje maksymalnym transferem na poziomie 300 Mb/s w kierunku do użytkownika i 150 Mb/s od użytkownika.

Podsumowanie

Układy wykonane zgodnie z wielordzeniową architekturą KeyStone to pierwsze struktury na rynku, które zawierają wysokowydajny, zintegrowany procesor RISC oraz bloki DSP z odpowiednimi koprocesorami i podsystemami wejścia i wyjścia, potrzebnymi do komunikacji w standardzie LTE. KeyStone to zarazem pierwsza wielordzeniowa architektura, której wewnętrzna szyna danych umożliwia wystarczająco szybki, dostęp bez opóźnień do wszystkich rdzeni DSP, układów peryferyjnych, koprocesora oraz wejść i wyjść.

Texas Instruments udostępnia też oprogramowanie warstwy fizycznej LTE zoptymalizowane dla rdzeni C66x. Dzięki temu wszystkiemu, układy TMS320TCI6618 stanowią obecnie najbardziej zaawansowaną platformę sprzętowo-programową dla komunikacji bezprzewodowej czwartej generacji.

Zhihong Lin, Greg Wood
Texas Instruments

Artykuł został opracowany we współpracy z firmą Farnell

Programatory stacjonarne

Każdy, kto konstruuje, naprawia lub wytwarza urządzenie z układem programowalnym musi wcześniej czy później sięgnąć po programator. Dotyczy to zwłaszcza produkcji małoseryjnych, dla których nie opłaca się zamawiać układów programowanych na etapie produkcji oraz pracowni konstrukcyjnych, w których powstają prototypy urządzeń.

Fotografia 1. Programator EasyPro 90B

Programator to urządzenie, które służy do zapisania programu lub danych z pliku wynikowego o określonym, standardowym formacie do układu programowalnego. Pod tym pojęciem nie wolno rozumieć samych układów pamięci, aczkolwiek to właśnie ich najczęściej dotyczy operacja zapisu czy odczytu, obojętnie czy są to np. pamięci konfiguracji w układach FPGA czy analogowych czy też pamięci programu mikrokontrolerów. Łatwo domyślić się, że wymagania odnośnie do urządzeń programujących będą różne i zależne od warunków, w których są one użytkowane. Na zupełnie inne aspekty kładzie się bowiem nacisk w czasie budowania prototypu, produkcji jednostkowej, małoseryjnej czy wielkoseryjnej o czym można przeczytać w innych artykułach, oznaczonych jako „temat numeru".

Wybór programatora

Przy budowaniu prototypu podstawowymi kryteriami doboru programatora będą jego obszerna lista programowanych układów, możliwości (np. pracy jako debugger) oraz wygoda użytkowania. Na przykład jest ważne, aby programator współpracował z popularnymi kompilatorami języków programowania, współpracował z ich IDE i wbudowanym weń debuggerem. Jeśli firma wytwarza kilka lub nawet kilkadziesiąt sztuk danego wyrobu rocznie, to może posłużyć się tym samym programatorem, którego używano podczas budowy prototypu. Pomimo mniejszej ważności kryterium czasu programowania to mimo wszystko musi on być dla użytkownika akceptowalny.

W produkcji małoseryjnej, zależnie od kryterium, którym jest czas, używane są różne metody programowania. Niektórzy producenci programują układy przed wlutowaniem, inni po wlutowaniu w płytkę. Jeśli układy są programowane przed wlutowaniem, a firma często zmienia profil produkcji (np. produkcja kontraktowa), to ważnym kryterium będzie dostępność adapterów i liczba programowanych układów. Częstokroć niemniej ważny jest też czas programowania układu, a ten wydatnie skraca kolejna cecha funkcjonalna tj. możliwość równoległego programowania i weryfikacji wielu układów scalonych.

Niezależnie od metody programowania (szeregowe lub równoległe), programowanie w układzie (ISP) wymaga zastosowania pewnych specjalnych rozwiązań. Na przykład, na etapie projektowania płytki trzeba przewidzieć złącze dla programatora lub wykonać specjalne, niemaskowane pady oraz rodzaj pola kontaktowego z odpowiadającymi im igłami. Takie pole może się przydać również do wykonywania testów i nie musi być przeznaczone tylko do zaprogramowania układu. Ponadto, konstruktor musi umożliwić odseparowanie układu na czas programowania od jego otoczenia, tak aby nie powodowało ono błędów.

W produkcji wielkoseryjnej i kontraktowej istotne będą przede wszystkim czas programowania oraz elastyczność programatora rozumiana jako duża liczba programowanych układów, możliwość programowania wielokanałowego w jak najbardziej zautomatyzowanym procesie. Czasami będzie ważna również możliwość zabezpieczenia pliku hasłem po to, aby dostęp do niego miały wyłącznie osoby uprawnione.

Przed każdym z rodzajów omawianych programatorów stawiane są inne wymagania. Dla jednych kryterium będzie szybkość i możliwość automatyzacji procesu programowania, dla innych możliwość uruchamiania programu i zaprogramowania procesora. W tym artykule skupimy się na przeglądzie programatorów stacjonarnych, które można zwykle spotkać w warsztacie elektronika – konstruktora.

EasyPRO 90B

EasyPRO 90B (**fotografia 1**) to programator, który umożliwia zaprogramowanie ponad 6000 układów. Obsługiwane układy to:

- programowanie pamięci (E)EPROM z serii: 27, 28, 29, 39, 49,
- programowanie pamięci EEPROM i Flash w tym również pamięci szeregowych I²C, 24cxx , SPI 25xxx, MicroWire 93CXX,
- programowanie 16-bitowych pamięci Flash,
- programowanie mikrokontrolerów firm Philips, NXP, Atmel, Winbond, HY, MicroChip, Holtek, SST, ICSI, Syncoms, Maxim-Dallas,
- testowanie pamięci SRAM o pojemnościach od 2 k to 512 k,
- programowanie układów GAL/PLD/CPLD typu 16V8X, 20V8X,22V10X,
- testowanie funkcjonalne układów TTL/CMOS.

Oprogramowanie sterujące programatorem przez interfejs USB pracuje pod kontrolą systemu Windows. Aktualizacje są wykonywane przez Internet, w tym lista rozszerzanie listy programowanych układów (bezpłatne).

Wybrane parametry programatora:

- podstawka ZIF-40,
- napięcie programujące 2,8...18 V,
- napięcie zasilające programowane układy 1,8...10 V.

SmartPRO X5

Pokazany na **fotografii 2** programator SmartPRO oprócz programowania układów umieszczanych w podstawkach jest wyposażony w gniazdo, które pozwala na programowanie szeregowe z użyciem techniki ISP. Można powiedzieć, że cechy funkcjonalne tego programatora predysponują go nie tyle do użycia w serwisie, ile w pracowni konstrukcyjnej i w małym zakładzie produkcyjnym. Ten programator jest chwalony przez użytkowników przede wszystkim za system zabezpieczeń, który sprawdza, czy np. układ

Fotografia 2. Programator SmartPro X5

Fotografia 3. Programator ChipMax2

Fotografia 4. Programator TopMax2

został prawidłowo umieszczony w podstawce i czy jest połączenie wszystkich doprowadzeń podstawki z nóżkami programowanego układu.

Obsługiwane układy:
- pamięci (E)EPROM z serii: 27, 28, 29, 39, 49, pamięci szeregowe, Flash, EEPROM (podstawka ZIF-48),
- programowanie 16-bitowych pamięci EPROM i Flash np. 29F200, 29F400, 29F800,
- programowanie mikrokontrolerów Philips, NXP, Atmel, Winbond, HY, MicroChip, Holtek, SST, ICSI, Syncoms, Dallas-Maxim,
- testowanie pamięci SRAM,
- programowanie szeregowych pamięci EEPROM: I²C 24cxx , SPI 25xxx, MicroWire 93CXX,
- programowanie układów GAL/PLD/CPLD typu: 16V8X, 20V8X,22V10X,
- testowanie funkcjonalne układów TTL/CMOS.

Oprogramowanie sterujące programatorem przez interfejs USB pracuje pod kontrolą systemu Windows. Aktualizacje są wykonywane przez Internet, w tym lista rozszerzanie listy programowanych układów (bezpłatne).

Wybrane parametry programatora:
- interfejs USB 2.0,
- złącze ISP do programowania w układzie oraz uniwersalna podstawka ZIF-48,
- obsługa układów o niskim napięciu zasilania (od 1,8 V).
- programowanie seryjne użyteczne, gdy jest konieczność zaprogramowania większej liczby układów; w tym celu należy załadować do bufora programu plik, następnie umieszczać w podstawce jeden po drugim programowane układy; po wykryciu układu w podstawce programator sam rozpoczyna programowanie,
- możliwość tworzenia i zapisywania projektów (zapamiętywane są bity konfiguracyjne, plik wynikowy itp.).

ChipMax2

ChipMax2 (**fotografia 3**) to szybki, uniwersalny programator dla wymagających użytkowników o konstrukcji opartej na układzie FPGA. Dzięki temu programowanie pamięci Flash o pojemności 64 MBit trwa 42 sekundy. Programator ten umożliwia również obsługę nawet najstarszych pamięci EPROM o napięciu programowania do 25 V, dlatego dobrze nadaje się nie tylko do pracowni konstrukcyjnej, ale również do serwisowania różnych urządzeń.

Obsługiwane układy:
- programowanie pamięci EPROM o napięciu programujący do 25 V,
- programowanie pamięci równoległych i szeregowych,
- programowanie układów FPGA, CPLD, EPLD,
- programowanie pamięci Flash wykonanych w technologiach NOR i NAND,
- programowanie pamięci NVRAM, SPLD,
- programowanie pamięci 8-, 16 i 32-bitowych,
- programowanie mikrokontrolerów,
- testowanie funkcjonalne układów TTL/CMOS.

Programator sprawdza prawidłowość połączeń pomiędzy programowanym układem a podstawką. Również i ten model ma opcję programowania seryjnego do obsługi produkcji masowej. Dodatkowo, jego oprogramowanie umożliwia jednoczesną pracę 8 programatorów sterowanych przez pojedynczy komputer.

Wybrane parametry programatora:
- interfejs USB 2.0,
- konstrukcja programatora oparta o szybki układ FPGA,
- programowanie układów o napięciu zasilającym 5,0 V; 3,3 V; 2,7 V; 1,8 V,
- programowanie seryjne uruchamiane przyciskiem *Start*,
- możliwość tworzenia i zapisywania projektów.

TopMax2

Podobnie jak w ChipMax2, producent położył duży nacisk na szybkość programowania. Szybkość programatora jest jeszcze wyższa niż w poprzednim wyrobie. Programator TopMax2 (**fotografia 4**) umożliwia programowanie pamięci NAND i NOR Flash, pamięci EPROM o napięciu programowania do 25 V, szeregowych pamięci z interfejsami I²C, Mircrowire, SPI oraz mikrokontrolerów. Najniższe napięcie zasilania programowanych układów wynosi 1,5 V. Programator może pracować w trybie gang – do 8 programatorów nadzorowanych przez pojedynczy komputer. Metalowa, wytrzymała obudowa predysponuje go przede wszystkim do zastosowania w firmach produkcyjnych.

Cechy programatora:

Wellon VP190

Uniwersalny programator Wellon VP-190 (**fotografia 5**) jest idealnym urządzeniem do warsztatu elektronika – konstruktora lub serwisu. Komunikacja z komputerem odbywa się poprzez port USB 2.0. Programator obsługuje większość popularnych mikrokontrolerów, pamięci Flash, EEPROM i EPROM. Lista programowanych układów obejmuje około 5,5 tys. pozycji i jest stale aktualizowana przez Internet.

Obsługiwane układy:
- programowanie pamięci (E)EPROM z serii: 27,28,29,39,49,
- programowanie 8- i 16-bitowych pamięci EPROM,
- programowanie mikrokontrolerów Philips, NXP, Atmel, Winbond, HY, SST, ICSI, Syncoms, Maxim-Dallas,
- testowanie pamięci SRAM,
- programowanie pamięci szeregowych z serii 24, 25, 93CXX
- programowanie układów GAL/PLD/CPLD typu 16V8X, 20V8X,22V10X,
- programowanie wszystkich, popularnych układów producentów: AMD, AMIC, Maxim-Dallas, Atmel, Intel, Sony, Winbond i innych,

Mimo przeznaczenia, programator ma również tryb programowania masowego (programowanie po zamknięciu dźwigienki podstawki). Ciekawostką jest funkcja wyświetlania informacji technicznych o programowanym układzie.

Wybrane parametry programatora:
- programowanie układów zasilanych napięciem od 1,5 V,
- programowanie pamięci 8-, 16- i 32-bitowych,
- automatyczne rozpoznawanie rodzaju pamięci,

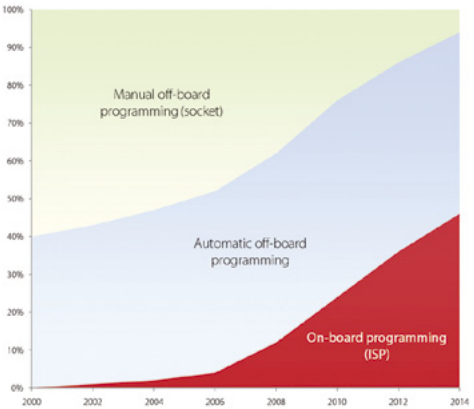
Produkcja robi to inaczej

Programowanie produkcyjne pamięci i mikrokontrolerów kieruje się nieco innymi zasadami niż programowanie laboratoryjne. Produkcją rządzi przede wszystkim wydajność, więc podstawowym kryterium przy doborze programatora nie jest jego uniwersalność i maksymalne pokrycie spektrum produkowanych na świecie układów programowalnych. Decydującego znaczenia w kontekście wielkość produkcji nabiera czas programowania pojedynczego układu. Wartość iloczynu wielkości produkcji i czasu programowania jest podstawowym parametrem, który implikuje metodę programowania i rodzaj stosowanego urządzenia.

Na linii produkcyjnej mamy w zasadzie do wyboru trzy podstawowe metody:
- Programowanie ręczne poza układem docelowym.
- Programowanie automatyczne poza układem docelowym.
- Programowanie w systemie docelowym (ISP – *In-System Programming*).

Pierwsza jest realizowana w trybie „off-line" poza główną linią produkcyjną w formie wstępnego przygotowania zaprogramowanych układów. Druga podobnie, chociaż istnieją w bardzo ograniczonym obszarze zastosowań rozwiązania do programowania automatycznego „on-line" w ciągu produkcyjnym. Trzecia, jest realizowana w fazie bezpośrednio poprzedzającej testowanie zmontowanej już płytki. W ostatnim przypadku typowa jest integracja procesów programowania i testowania oraz ich realizacja z poziomu komputera nadrzędnego za pośrednictwem odpowiedniego urządzenia sprzęgającego.

Jeszcze nie tak dawno wydawało się, że programowanie ISP całkowicie wyeliminuje tradycyjne programatory. Jednak coraz bardziej skomplikowane aplikacje oraz duże pojemności współcześnie stosowanych pamięci Flash spowodowały, że programatory skutecznie bronią się przed ofensywą relatywnie wolnych, choć z każdym rokiem coraz szybszych, opartych o transmisję szeregową rozwiązań ISP. Trend pokazany na **rysunku 1** wskazuje na zastępowanie, przy mniejszych wolumenach produkcji, programowania ręcznego programowaniem w systemie docelowym i stałego udziału w rynku programowania automatycznego „off-line".



Rysunek 1. Trendy w programowaniu układów

Łatwo zauważyć, że nadchodzi globalna automatyzacja procesu programowania w czasie produkcji. Spójrzmy więc na przykładzie wybranych urządzeń oferowanych przez firmę WG na to, co jest dostępne na rynku i spróbujmy dać odpowiedź co, gdzie i kiedy stosować.

Ręcznie w wielu gniazdach

Oczywistym jest, że przy małych i średnich wolumenach produkcji zakup automatu programującego nie ma sensu. Jesteśmy skazani na ręczne operacje lub – o ile to możliwe realizację programowania w systemie docelowym. Na początek zajmijmy się jednak typową sytuacją – mamy pewną liczbę układów scalonych do zaprogramowania.

W najprostszej sytuacji, gdy pojemność pamięci jest mała, czas jej programowania i weryfikacji (t_{pv}) jest znacząco krótszy od czasu obsługi (t_o) pojedynczego układu (tzn. czasu potrzebnego na wyjęcie układu z opakowania, umieszczenie go w podstawce, usunięcie po zaprogramowaniu z podstawki, zapakowanie do dalszej produkcji i chwilę oddechu operatora) wystarczy tradycyjny programator z jedną podstawką. Sytuacja zmienia się, gdy czasy t_{pv} i t_o są porównywalne. Na jednogniazdowym programatorze operator ma bezczynne przestoje, musi czekać na zaprogramowanie układu. Wyjściem jest zwielokrotnienie gniazd programujących poprzez użycie kilku programatorów jednogniazdowych lub jednego programatora wielogniazdowego. Szacunkową liczbę gniazd (n) potrzebnych dla jak najlepszego wykorzystania czasu operatora można oszacować jako $n=[t_{pv}/t_o]+1$. Jasno widać, że czas programowania z punktu widzenia optymalizacji wydajności powinien być jak najkrótszy, czyli programator możliwie szybki.

W przypadku, gdy chcemy ze względu na możliwość różnych konfiguracji produkcji stosować kilka jednogniazdowych programatorów alternatywnie do programatora wielogniazdowego, to dobrze jest, gdy oprogramowanie nadzorujące może jednocześnie obsłużyć kilka (typowo 8) takich urządzeń. Korzystne jest też, gdy urządzenie i oprogramowanie są łatwe w obsłudze i mogą być nadzorowane przez średnio wykwalifikowany personel. Stosowanie na produkcji programatorów ręcznych zazwyczaj wskazuje na krótkie serie produkcyjne różnych produktów, czyli częstą zmianę typów programowanych układów. W takich przypadkach programator powinien być uniwersalny, z wyjątkiem może tylko sytuacji programowania pamięci NAND Flash wielkiej pojemności, gdzie warto zastosować ultraszybkie rozwiązanie dedykowane.

Wymagania te spełniają w pełni produkty BeeProg2 i BeeHive204 firmy Elnec sterowane z komputera nadrzędnego przez USB, obsługujące ponad 60 tysięcy układów programowalnych! Bazujący na tym samym rozwiązaniu pojedynczego programatora tzw. site'a, najbardziej zaawansowany produkt BeeHive208 jest 8-podstawkowym rozwiązaniem obsługiwanym dodatkowo z poziomu wyświetlacza „touch screen". Korzystna na produkcji jest bowiem praca programatora bez nadzoru komputera jako urządzenia autonomicznego z buforową pamięcią danych, wyświetlaczem i klawiaturą. Wszystkie wymienione programatory firmy Elnec mają na panelu czołowym złącza interfejsów ISP pozwalające na wielokanałowe programowanie w układów systemie docelowym. Stanowią tym samym kompletne rozwiązanie problemu programowania podczas produkcji mało- i średnioseryjnej. Jeśli jednak jesteśmy w obszarze produkcji wielkoseryjnej to powinniśmy programować…

Automatycznie poza układem

Aby uzmysłowić problem, który stawia przed nami masowa produkcja, weźmy pod lupę gorący temat – programowanie pamięci NAND Flash. Przyjmijmy realne wielkości, tzn. że czas programowania i weryfikacji t_{pv}=10 minut dla pojemności 1 G×8 bitów w typowym jednogniazdowym programatorze, a produkcja miesięczna wynosi 48 tys.

sztuk. Wówczas miesięcznie potrzeba 1 tys. pełnych dniówek tylko na proces programowania i weryfikacji. Zakładając dalej, że będziemy programować w 16 gniazdach jednocześnie, bez żadnych przerw i przestojów, to potrzebujemy ponad 60 dniówek, czyli pełny miesiąc pracy na trzy zmiany. Oczywistym jest więc, że przy produkcji masowej najrozsądniejszymi rozwiązaniami są:

- Zastosowanie układów zaprogramowanych przez producenta (to jednak eliminuje konieczność użycia programatora).
- Zastosowanie jak najszybszych programatorów.
- Zwielokrotnienie liczby gniazd programujących i ewentualnie praca wielozmianowa.
- Automatyzacja procesu programowania.
- Programowanie „off-line" poza linią produkcyjną.

Ostatni punkt wymaga dodatkowego uzasadnienia. Są bowiem rozwiązania do programowania „on-line" na linii technologicznej, w których układy zapisywane na bieżąco są podawane do maszyny montażowej „pick&place". Wymagane, przykładowe 10 minut na programowanie, to jednak zbyt dużo, by nie powodować spowolnienia produkcji. Rozwiązania „on-line" są więc ograniczone tylko do pewnych typów układów o niedużej pojemności pamięci. Urządzenia te, chociaż dziś mogą być wystarczające, to jutro dla wchodzących do produkcji nowych typów programowanych układów mogą okazać się wąskim gardłem linii technologicznej. Ich możliwość adaptacji i rozbudowy jest bowiem bardzo ograniczona. Dlatego sugerujemy zastosowanie rozwiązań „off-line".

Kluczem jest szybkość i dlatego renomowani producenci programatorów wprowadzają kolejne generacje coraz szybszych urządzeń. W kontekście pamięci NAND Flash technologia FlashStream firmy BPM Microsystems bije wszystkich na głowę. W ręcznych czterogniazdowych programatorach BPM2800F (i oczywiście w automatach) stosuje się specjalizowany koprocesor *FlashStream Vector Engine*, który przyśpiesza czas programowania w porównaniu z klasycznymi metodami programowania o rząd wielkości. Biorąc pod uwagę przedstawione szacunkowe obliczenia jednoznacznie widać, że przy masowym programowaniu NAND Flash'y dopiero ultraszybkie programatory są w stanie rozwiązać problem.

Poza szybkością istotnymi czynnikami, które należy wziąć pod uwagę przy doborze produkcyjnych urządzeń technologicznych są:

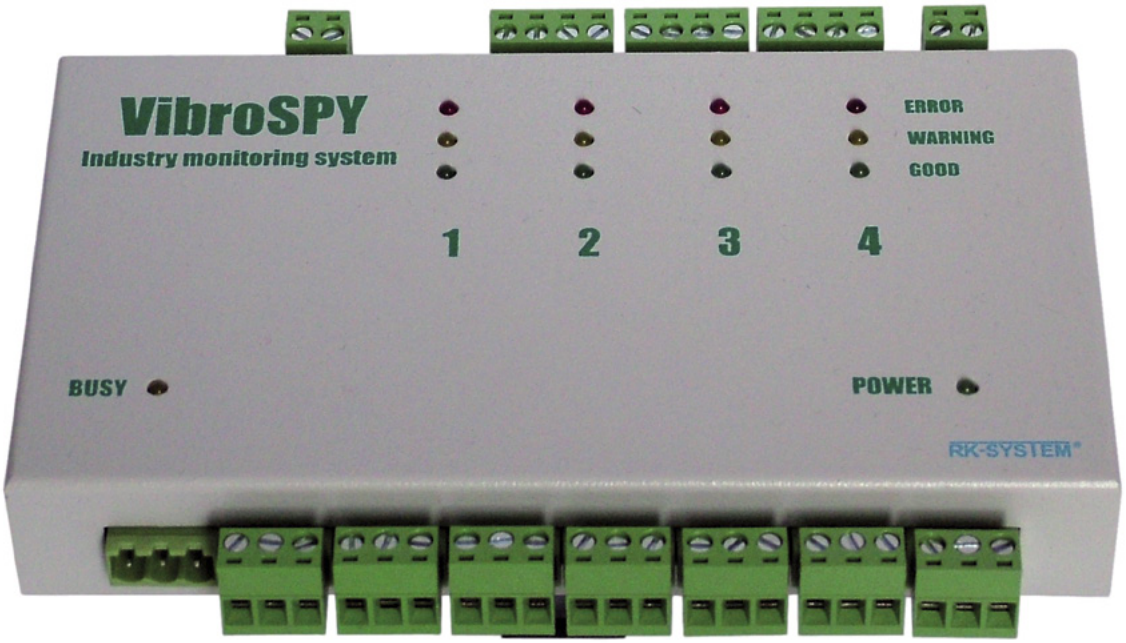
- Jakość i niezawodność urządzenia ze szczególnym uwzględnieniem trwałości podstawek.

Fotografia 2. Autonomiczny programatory BeeHive208 firmy Elnec

- Wsparcie techniczne ze strony producenta i dostępność lokalnego serwisu.
- Czas reakcji na uszkodzenie maszyny lub błąd algorytmu.
- Strategia, procedury i koszty dopisywania algorytmów programowania dla nowych układów.

Trzeba też zdać sobie sprawę, że automat programujący może być wąskim gardłem procesu produkcji, ponieważ często pracuje on na rzecz kilku linii produkcyjnych. Przestój w wyniku awarii może zatrzymać całą fabrykę! Dlatego też nie mogą to być urządzenia najtańsze, pochodzące od nieznа-

REKLAMA



VibroSPY
Przemysłowe podglądanie

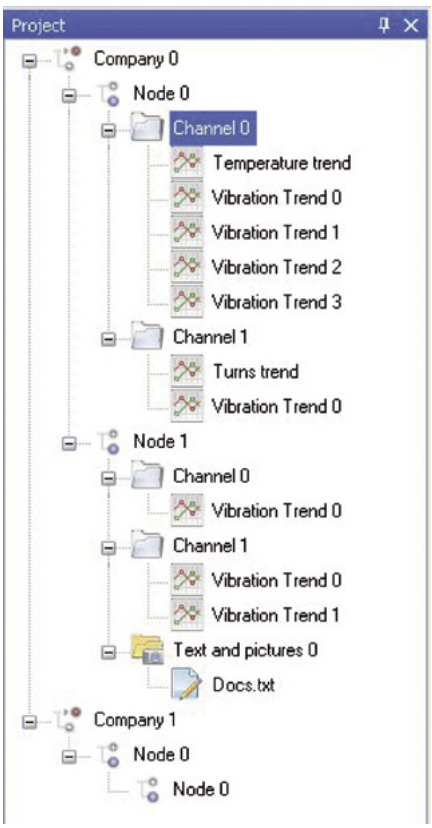
Na liniach produkcyjnych wielkich fabryk byli kiedyś inżynierowie odpowiedzialni za utrzymanie maszyn w gotowości produkcyjnej. Chodzili od stanowiska do stanowiska z podręcznym zestawem narzędzi i mierników, i badali czy urządzenia produkcyjne zachowują sprawność. Chociaż dzisiaj miejsca fabryk zajmują banki i centra handlowe, to jednak z produkcji zupełnie nie zrezygnowano. Współczesne linie („taśmy") wyglądają jednak nieco inaczej. Miejsce inżynierów zajęły czujniki łączone wzajemnie zaplatającymi się sieciami.

Dodatkowe informacje:
RK-System, 05-825 Grodzisk Mazowiecki, ul. Chełmońskiego 30, tel.: 22-724-30-39, faks 22-724-30-37, *www.rk-system.com.pl*, e-mail: sprzedaz@rk-system.com.pl

W coraz bardziej rozpędzonym współczesnym świecie, dla przemysłu ważna jest każda minuta. Z tą samą chwilą, w której staje jakaś linia produkcyjna, rozpoczyna się liczenie strat, nawet wtedy, gdy nic nie ginie i nie ulega destrukcji. A maszyny muszą się psuć, bo taka jest ich natura. Na razie jeszcze nikt nie wymyślił nie zużywających się elementów mechanicznych czy elektromechanicznych. Niezwykle istotna jest zatem ciągła obserwacja pracy, pozwalająca przewidywać sytuacje awaryjne i zawczasu na nie reagować. Umiejętnie przeprowadzona diagnoza pozwala skrócić czas przestoju do minimum. Ale diagnoza ta powinna być znana jeszcze w chwili pracy urządzenia.

VibroSPY – nowy produkt RK-System

Specjalizująca się w badaniach wyważania, drgań i wibracji firma RK-System opracowała nowy system przeznaczony do monitoringu urządzeń przemysłowych. VibroSPY, bo tak nazywa się ów system, jest przystosowany do pracy w sieci Ethernet, co oznacza, że może działać na praktycznie nieograniczonym obszarze, przekazując dane z czujników pomiarowych do bardzo odległych miejsc. Czujniki te są dołączane do lokalnych stacji pełniących funkcję urządzeń monitorujących. Ich liczba nie jest niczym ograniczona, ważne jest tylko, aby było możliwe dołączenie ich do sieci ethernetowej. Można to zrealizować zarówno za pośrednictwem połączenia przewodowego, jak i bezprzewodowo (Wi-Fi). Jedno urządzenie monitorujące dysponuje 4 wejściami analogowymi, 4 wejściami cyfrowymi, 4 wyjściami przekaźnikowymi. Do komunikacji z dowolnymi innymi urządzeniami może być wykorzystywany ponadto interfejs RS485. Stacja jest zasilana napięciem zewnętrznym 24 V_{DC}. Specjalny uchwyt na obudowie umożliwia mocowanie urządzenia na szynie DIN. Ze względu na



Rysunek 1. Drzewiasta struktura sieci systemu VibroSPY

Tabela 1. Podstawowe parametry i cechy urządzeń VibroSPY	
Rozdzielczość pomiaru	16 bitów
Zakres częstotliwości próbkowania	5 Hz...200 kHz
Pasmo analogowe	100 kHz
Napięcie zasilania	24 V
Pobór prądu	~300 mA
Liczba uniwersalnych kanałów analogowych	4
Liczba programowalnych kanałów cyfrowych	4
Liczba programowalnych kanałów przekaźnikowych	4
Obciążalność prądowa styków	5 A/250 V_{AC}, 5 A/30 V_{DC}
Interfejs komunikacyjny z innymi urządzeniami	RS485
Interfejs sieciowy	LAN 100 Mb
Diody sygnalizujące	OK, WARNNING, ERROR
Wyważanie elementów wirujących w łożyskach własnych	jedno lub dwupłaszczyznowe
Pomiar wielkości fizycznych	np.: wilgotność, stężenie gazu itp.
Liczba urządzeń dołączanych do sieci pomiarowej	nieograniczona
Mocowanie	szyna DIN
Praca zdalna i kontrola parku maszynowego w fabrykach dowolnie od siebie oddalonych	
Możliwość umieszczenia w hermetycznej, wodoszczelnej obudowie	
Zadawanie nastaw krytycznych wielkości mierzonych, po przekroczeniu których system może powiadamiać o problemie lub samoczynnie wykonać zaprogramowaną dla danego zdarzenia reakcję	
Możliwość dołączenia do systemu dowolnych urządzeń opartych o interfejs LAN	

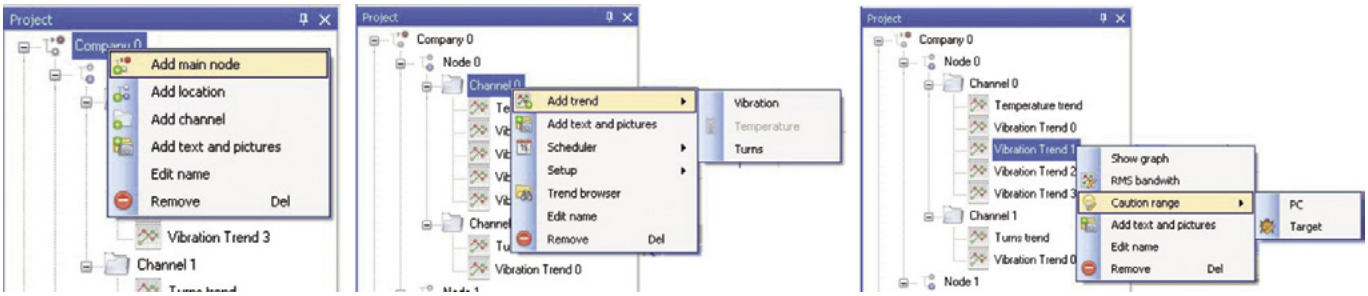
specyfikę zastosowań, do systemu będą najczęściej dołączane czujniki przyspieszenia (drgań), temperatury, wilgotności, ciśnienia, stężenia gazu itp. Generalnie są to czujniki z wyjściami analogowymi, czujniki ICP oraz czujniki z wyjściem typu pętla prądowa.

Ważną częścią systemu jest oprogramowanie. Jego zadaniem jest definiowanie sieci, gromadzenie i obróbka danych pomiarowych generowanych przez poszczególne stacje. Na podstawie zebranych informacji możliwe jest automatyczne tworzenie dokumentacji technicznej.

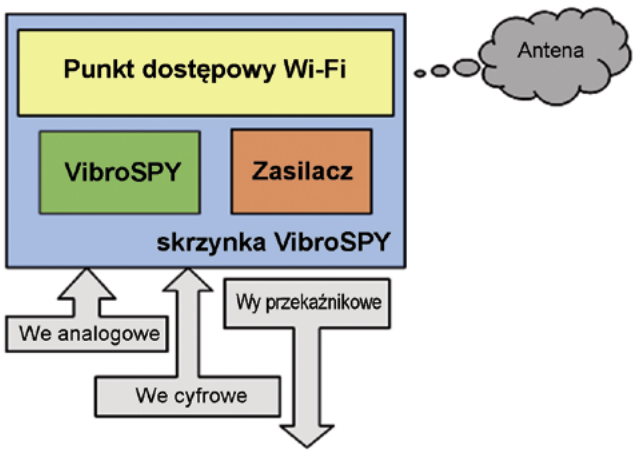
Komplet informacji dotyczących określonej grupy pomiarów jest zapisywany w tzw. projektach. Wydzielono w nich trzy strefy, zawierające: opis sieci uwzględniający ich hierarchiczną strukturę, bieżący stan urządzeń i czujników oraz trendy. Analiza trendów pozwala prognozować występowanie awarii obserwowanych urządzeń. Z tego względu jest to najważniejsza funkcja programu.

Sieć ma strukturę drzewiastą (**rysunek 1**). Jej główne węzły są utworzone z firm, w których z kolei można wyróżnić budynki, a w nich kanały pomiarowe. Na najniższym poziomie struktury jest gromadzona dokumentacja elektroniczna, na którą składają się pliki tekstowe i graficzne. Definiowanie sieci jest bardzo proste. Przebiega z wykorzystaniem rozwijanych menu wywoływanych kliknięciem prawego przycisku myszki na danej pozycji struktury. Na ekranie pojawiają się wtedy listy z różnymi opcjami do wyboru (**rysunek 2**). Sieć, o której jest mowa w kontekście systemu VibroSPY jest w pewnym sensie terminem umownym, można bowiem sobie wyobrazić, że pomiary będą wykonywane tylko przez jedno urządzenie monitorujące. Mimo to, dane mogą być przesyłane w odległe miejsca poprzez Ethernet. Schematycznie taki przypadek przedstawiono na **rysunku 3.** Bardziej rozbudowaną konfigurację, typowo sieciową, przedstawiono na **rysunku 4**.

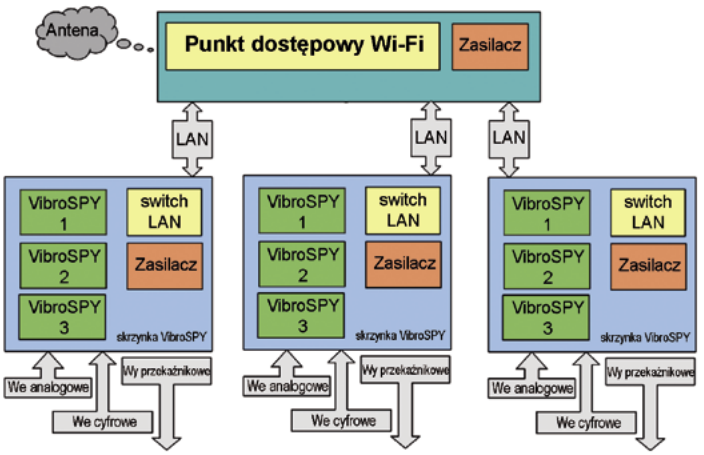
Czujniki systemu monitoringu VibroSPY są (w nomenklaturze sieci) dopinane do kanałów pomiarowych. Ich wyboru dokonuje się z bazy danych o czujnikach, któ-



Rysunek 2. Menu wykorzystywane podczas definiowania sieci



Rysunek 3. Sieć VibroSPY wykorzystująca jedno urządzenie monitorujące

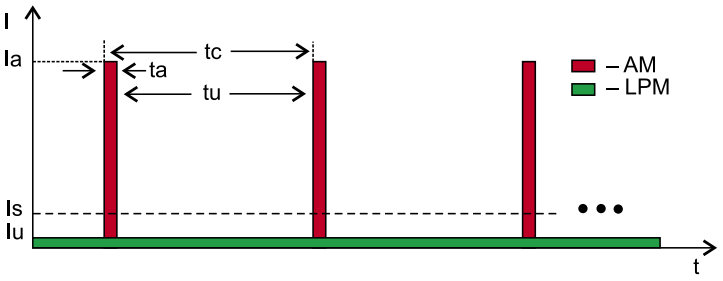


Rysunek 4. Sieć VibroSPY wykorzystująca wiele urządzeń monitorujących

Energooszczędne MSP430
Nowa seria 0xx

Rozwój rodziny MSP430 nie spowalnia, a wręcz przeciwnie – nabiera rozpędu. Liczba wyprodukowanych modeli mikrokontrolerów przekroczyła 300. Powstała także nowa niskonapięciowa seria układów o oznaczeniu 0xx (Low Voltage Series). W artykule omówiono mikrokontrolery z serii 0xx.

MSP430 mają 5 bądź 6 energooszczędnych trybów uśpienia LPM (*Low Power Mode*). W trybach LPM3, LPM4 i LPM5 (głębokie uśpienie) pobór prądu przez mikrokontroler może zostać obniżony do wartości rzędu nanoamperów. W trybie pracy aktywnej AM (*Active Mode*) pobór prądu MSP430 znacząco wzrasta i oscyluje w granicach setek mikroamperów, a nawet pojedynczych miliamperów. Z tej przyczyny, aby wykonać urządzenie energooszczędne, programista musi sterować pracą mikrokontrolera zgodnie ze schematem zamieszczonym na **rysunku 1.** Jednym słowem mikrokontroler musi jak najmniej czasu spędzać w trybie pracy aktywnej (duży pobór prądu), a jak najwięcej w trybie uśpienia (mały pobór prądu). Im, krótszy będzie czas pracy MSP430 w trybie aktywnym, tym średnia wartość pobieranego prądu będzie mniejsza, a urządzenie zasilane z baterii będzie pracować dłużej (**ramka 1**).

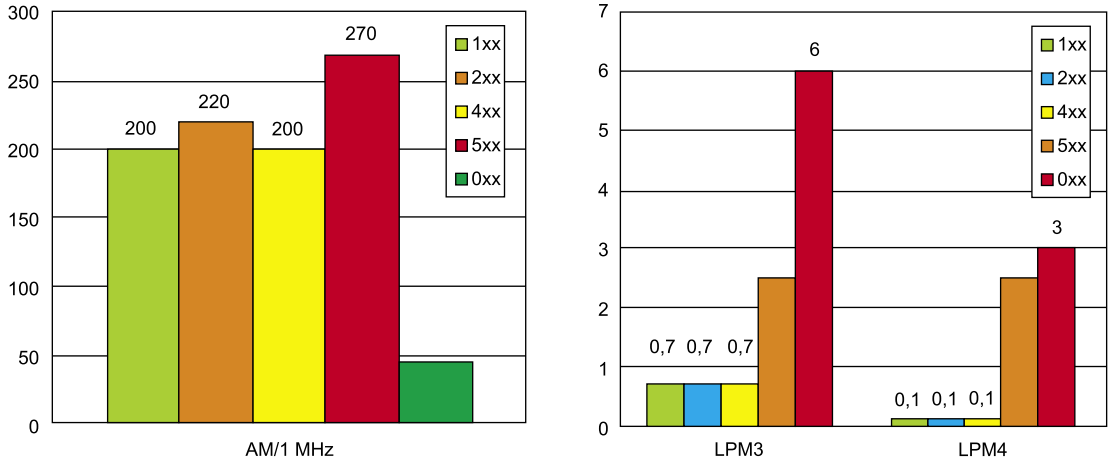


Rysunek 1. Optymalne sterowanie pracą MSP430

Maksymalny czas pracy mikrokontrolera na jednym komplecie baterii Tp[h] można obliczyć korzystając ze wzoru **Tp=Cb/Is**, gdzie: **Cb** – pojemność (ładunek elektryczny) baterii [mAh], **Is** – średni pobór prądu [mA].
W przypadku sterowania jak na rysunku 1, średni pobór prądu mikrokontrolera Is [mA] można obliczyć korzystając ze wzoru **Is=(ta*Ia + tu*Iu)/tc**, gdzie: **ta** – czas pracy w trybie aktywnym AM [s], **tu** – czas pracy w trybie uśpienia LPM [s], **tc** – czas trwania pojedynczego cyklu pracy [s], **Ia**, **Iu** – pobór prądu w trybie aktywnym, uśpienia [mA].
Przykład:
Urządzenie wyposażone jest w mikrokontroler MSP430f1232. Mikrokontroler pracuje w cyklach jednosekundowych, gdzie przez 100 ms przebywa w trybie aktywnym (pobór prądu 300 μA), a pozostałe 900 ms jest uśpiony w trybie LPM3 (pobór prądu 800 nA). Urządzenie jest zasilane z baterii typu CR2032 o pojemności 240 mAh.
Średni pobór prądu wynosi: Is=(0,1 s*0,3 mA+0,9 s*0,0008 mA)/1s = 0,03072 mA
Maksymalny czas pracy na baterii wynosi: Tp=240 mAh/0,03072 mA = 7812,5 h. Po przeskalowaniu jednostek z otrzymujemy ponad 325 dni pracy urządzenia.
W przypadku, gdy czas pracy MSP430f1232 w trybie aktywnym zmniejszymy z 100 do 20 ms, to średni pobór prądu zmaleje do wartości 0,006784 mA , a maksymalny czas pracy na baterii wydłuży się do ponad 4 lat.

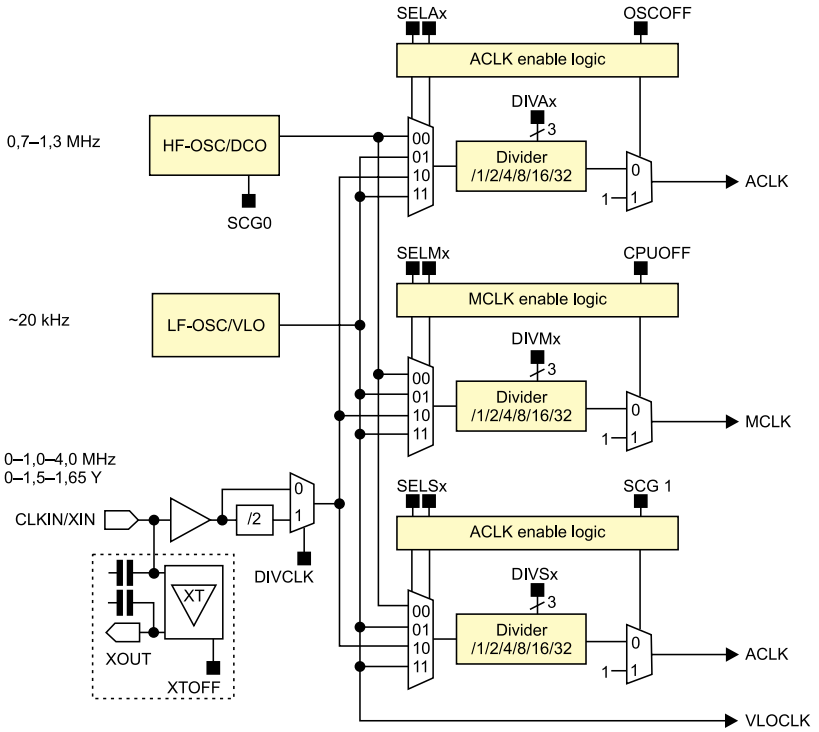
Mikrokontrolery MSP430 z serii 0xx

Mniejszy pobór prądu układów serii 0xx w trybie pracy aktywnej uzyskano przez obniżenie napięcia zasilania mikrokontrolera

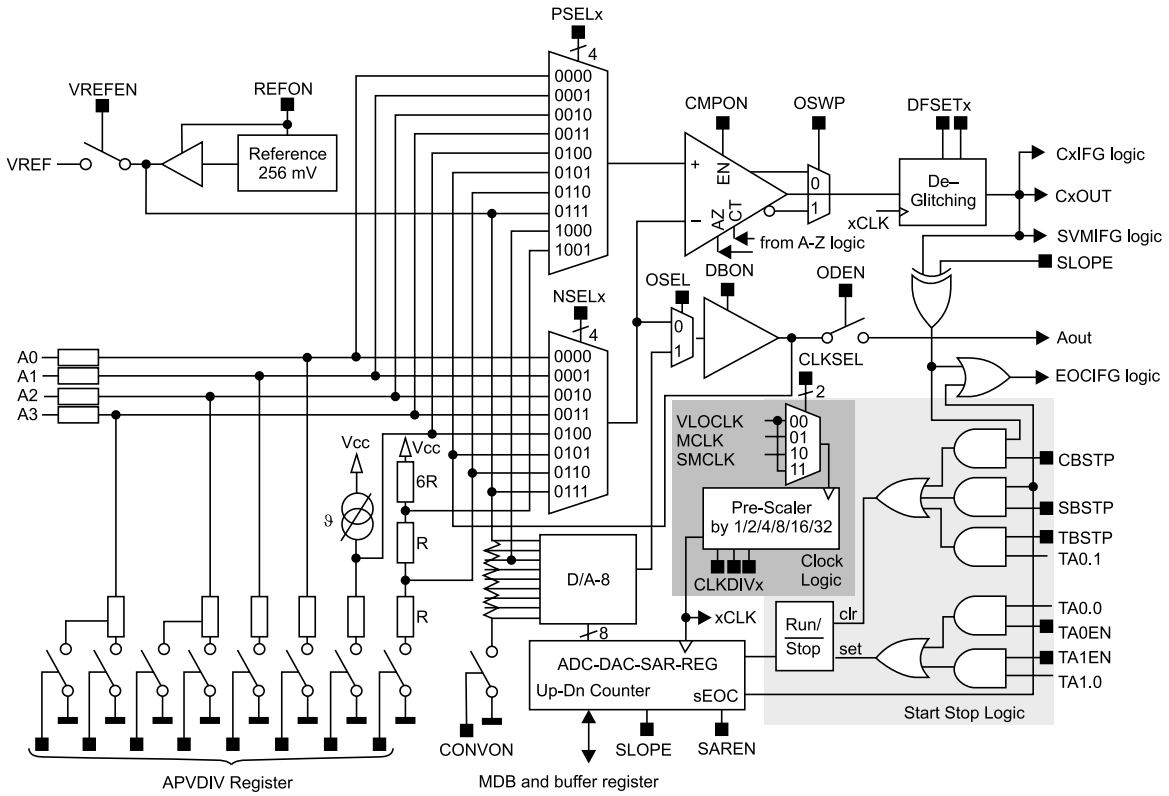


Rysunek 2. Pobór prądu MSP430 (μA). W seriach 1xx, 2xx, 4xx przy zasilaniu napięciem 2,2 V, w serii 5xx – 3 V (rdzeń 2 V/ 12 MHz), a w serii 0xx – 1,3 V

Tabela 1. Parametry mikrokontrolerów z serii 0xx				
Procesor	**MSP430C091**	**MSP430C092**	**MSP430L092**	
pamięć	RAM	128 B	128 B	2 KB
	ROM	1 KB	2KB	---
wejścia – wyjścia	11			
układ Watchdog	moduł Watchdog Timer w trybie pracy Watchdog			
system zegarowy	CCS (2 źródła wewnętrzne,1 źródło zewnętrzne, 4 sygnały zegarowe)			
liczniki	2 liczniki TAR, moduł Watchdog Timer w trybie pracy Timer			
peryferia	8-bitowy DAC, 8-bitowy ADC, komparator, generator napięcia referencyjnego 256 mV, dzielnik napięcia zasilania, czujnik temperatury			



Rysunek 3. Schemat blokowy układu zegarowego CCS



Rysunek 4. Schemat blokowy modułu analogowego A-Pool

(ramka 2). Wcześniej produkowane MSP430 mogły być zasilane napięciem z zakresu 1,8...3,6 V (w serii 3xx 2,5...5,5 V), w nowych układach serii 0xx napięcie zasilania obniżono do wartości 0,9...1,65 V. Zmniejszenie napięcia zasilania poniżej 1,8 V spowodowało, że w serii 0xx nie są produkowane układy z pamięcią EEPROM/FLASH. Produkowane są wyłącznie procesory wyposażone w pamięć ROM/RAM.

Na dzień dzisiejszy wyprodukowano 3 układy serii 0xx. Wszystkie procesory oferowane są w 14-pinowych obudowach typu TSSOP. Programista ma do dyspozycji 2 porty wejścia-wyjścia P1, P2 obsługujące 11 linii. Każda linia portu P1, P2 może pracować w trybie funkcyjnym, bądź w trybie we-wy. W przypadku, gdy linia pracuje w trybie we-wy programista może ustawić kierunek linii (wejście/wyjście), odczytać stan wejścia, ustawiać stan wyjścia, obsłużyć linię w trybie przerwań, oraz użyć rezystorów podciągających.

Specjalnie na potrzeby serii 0xx zaprojektowano nowy układ zegarowy CCS (*Compact Clock System*). Schemat blokowy układu ilustruje **rysunek 3**. Układ zegarowy wyposażono w dwa wewnętrzne źródła zegarowe: LF-OSC (oscylator niskiej częstotliwości odpowiednik znanego z serii 2xx, G2xx, 5xx/6xx, CC430 oscylatora VLO), HF-OSC (oscylator wysokiej częstotliwości odpowiednik występującego we wszystkich seriach MSP430 oscylatora DCO). Oscylator LF-OSC wytwarza sygnał o częstotliwości około 20 kHz i jest zawsze aktywny (nie można go wyłączyć). Oscylator HF-OSC wytwarza sygnał

Zestawy ewaluacyjne dla µC z rdzeniami Cortex-M (2)

W numerze majowym omówiliśmy dużą część zestawów ewaluacyjnych dla mikrokontrolerów z rdzeniami Cortex-M. Ograniczona objętość tamtego artykułu nie pozwoliła opisać wszystkich ciekawych i ważnych produktów. Z tego powodu publikujemy drugą część przeglądu, w której omówione zostały zestawy, które nie zmieściły się w części pierwszej.

Dodatkowe informacje:
- pierwsza część artykułu

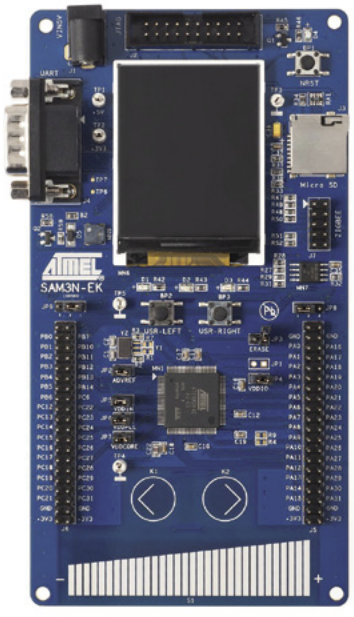
Atmel

Firma Atmel oferuje 3 rodzaje mikrokontrolerów z rdzeniem Cortex-M3 i dla każdego z nich po jednym zestawie ewaluacyjnym, w których zainstalowano najsilniejsze z poszczególnych rodzin układów. W poprzedniej części omówiliśmy jeden z nich – SAM3S-EK. Jednakże i dwa pozostałe, SAM3U-EK i SAM3N-EK zasługują na uwagę czytelników.

SAM3N-EK

Zestaw ten kosztuje około 870 zł netto.

Podstawowe cechy:
- Mikrokontroler SAM3N4C (256 kB Flash, 24 kB SRAM)
- Zewnętrzna pamięć NAND Flash
- Kolorowy wyświetlacz LCD-TFT 2”
- Interfejs UART
- Interfejs kart microSD
- 2 przyciski użytkownika oraz przyciski QTouch, przycisk resetu
- 2 diody LED użytkownika, dioda LED zasilania
- Interfejs JTAG/ICE

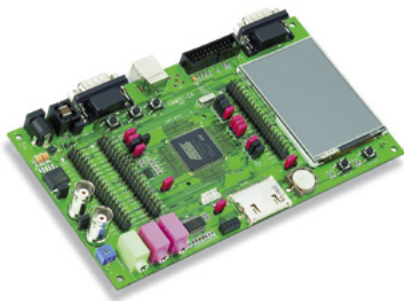


- Złącze ZigBee
- Dwa 32-bitowe interfejsy PIO i jeden 15-bitowy PIO

SAM3U-EK

Jest to najbardziej rozbudowany zestaw firmy Atmel do układów z rdzeniami Cortex-M.

Podstawowe cechy:
- Mikrokontroler SAM3U4E (256 kB Flash, 48 kB SRAM)
- Interfejsy UART i USART
- Wejście i wyjścia audio: stereofoniczne słuchawkowe i wejście liniowe oraz monofoniczne wejście mikrofonowe
- Interfejs kart SD/MMC
- Kolorowy wyświetlacz LCD TFT o przekątnej 2,8” z panelem dotykowym
- Czujnik temperatury
- Akcelerometr trójosiowy
- Dwa przyciski użytkownika oraz trzy przyciski resetu i budzenia
- Dwie diody LED użytkownika, dioda LED zasilania
- Interfejs USB (device)
- Interfejs JTAG
- Złącze BNC wraz z potencjometrem jako wejście konwertera analogowo-cyfrowego
- Złącze interfejsu ZigBee
- Trzy 32-bitowe interfejsy PIO



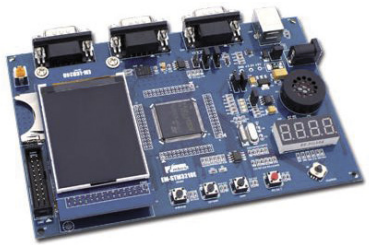
Embest

W poprzedniej części artykułu nie omówiliśmy zestawów firmy Embest. Produkuje ona 5 ciekawych płytek deweloperskich wyposażonych w wyświetlacze. Są one oparte na układach marki STMicroelectronics, NXP i Atmel. Oto trzy z nich:

EM-STM3210E

Cena tego zestawu wynosi około 380 zł netto.

Podstawowe cechy:
- Mikrokontroler STM32F103ZE (512 kB Flash, 64 kB SRAM)
- Interfejs szeregowy UART



Większość przytoczonych cen pochodzi z zagranicznych sklepów i zależy od kursu walut oraz nie uwzględnia kosztów przesyłki. Ceny w polskich sklepach mogą odbiegać od zaprezentowanych.

Gdzie kupić?
Niżej prezentujemy nazwy niektórych dystrybutorów producentów zestawów deweloperskich omówionych w artykule:

Analog Devices
- Alfine
- Farnell
- Silica/Avnet
- Arrow

Atmel
- Arrow Electronic
- EBC Elektronik
- JM Elektronik
- MSC Polska
- Kamami
- Seguro

Cypress
- Future Electronics

Embest
- Kamami

Energy Micro
- Avnet
- Codico

Microsemi
- Future Electronics
- MSC Polska

Nuvoton
- PDW Marthel

Texas Instruments
- Farnell
- Arrow
- Silica/Avnet
- EBV Elektronik
- Contrans TI

– Interfejs szeregowy CAN
– Interfejs USB (złącze typu B)
– Kolorowy wyświetlacz LCD TFT o rozdzielczości 240×320 pikseli
– Wbudowana bateria podtrzymująca zegar RTC
– Zewnętrzna pamięć RAM (1 MB) i SPI Flash (8 MB)
– Interfejs kart microSD
– Interfejsy ETM i JTAG
– Zasilanie napięciem 5 V z USB

W zestawie, oprócz płytki znajduje się także kabel RS-232 i kabel USB.

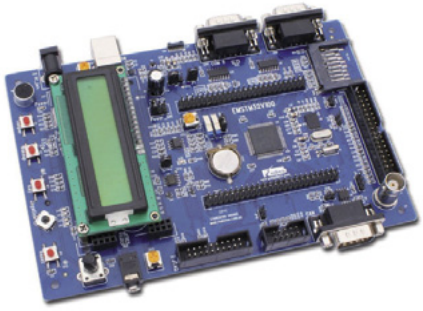
EM-STM32V100

Drugi, nieco prostszy zestaw Embestu z układem STMicroelectronics kosztuje ok. 370 zł netto.

Podstawowe cechy:

– Mikrokontroler STM32F103VBT6 (128 kB Flash, 20 kB SRAM)
– 8 MB zewnętrznej pamięci Flash
– Alfanumeryczny wyświetlacz LCD 16×2
– Interfejs USB (device)
– Interfejs CAN 2.0
– Dwa interfejsy szeregowe RS-232
– Transceiver IrDA
– Złącze kart SD
– Złącze kart Smart Card
– Jedno złącze do podłączania silnika elektrycznego
– Czujnik temperatury
– 4 przyciski i joystick
– 4 diody LED
– Mikrofon i wyjście słuchawkowe
– Bateria do podtrzymania zegara RTC
– Interfejsy JTAG i SWD
– Zestaw może być zasilany z trzech różnych źródeł: z gniazda 5 V_{DC}, z portu USB lub poprzez wyprowadzone tzw. goldpiny

W zestawie, oprócz płytki znajdują się także kabel: RS-232, kabel USB i kabel ethernetowy.



STM32F107

Trzeci z zestawów Embestu z układem marki STM kosztuje 450 zł netto.

Podstawowe cechy:

– Mikrokontroler STM32F107VC (256 kB Flash, 64 kB SRAM)
– Kolorowy wyświetlacz LCD TFT o rozdzielczości 320×240 pikseli
– Złącze miniUSB (OTG)
– Interfejs Fast Ethernet
– Interfejs kart microSD
– Interfejsy szeregowe SPI i I²C
– Potencjometr
– Czujnik temperatury
– Głośnik i gniazdo słuchawkowe
– Interfejs JTAG
– Joystick i przycisk resetu
– 4 diody LED oraz 4-cyfrowy wyświetlacz diodowy
– Zasilanie napięciem 5 V z USB

W zestawie, oprócz płytki znajdują się także: kabel RS-232, kabel USB, kabel ethernetowy i płyta z oprogramowaniem.

Freescale

Bardzo ciekawe zestawy dla układów z rdzeniami Cortex-M4 ma w swojej ofercie Freescale. Zawierają one procesory z rodziny Kinetis. Mogą być używane samodzielnie lub w ramach platformy Freescale Tower System.

Kinetis KwikStik

Ten zestaw jest szczególnie godny polecenia ze względu na swoją cenę. Kosztuje około 100 zł.

Podstawowe cechy:

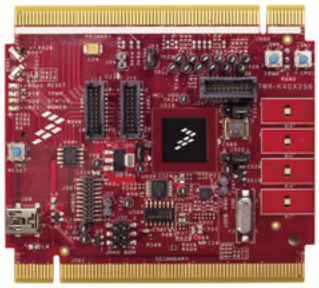
– Mikrokontroler K40X256 (256 kB Flash, 256 kB SRAM)
– Wyświetlacz LCD o 306 segmentach
– 2 złącza microUSB
– Wyjście audio
– Złącze J-Link
– Interfejs podczerwieni
– Złącze kart SD
– Interfejs panelu pojemnościowego
– Złącze systemu Freescale Tower
– Akumulatorek do podtrzymania zasilania

Kinetis TWR-K60N512-KIT i TWR-K40X256-KIT

Są to dwa podobne do siebie zestawy, zgodne z platformą Freescale Tower System. Cena pierwszego to około 590 zł netto, a drugiego – 560 zł netto.

Podstawowe cechy wspólne:

– Mikrokontroler K60N512 dla TWR-K60N512-KIT i K40X256 w przypadku TWR-K40X256-KIT. Oba mają po 256 kB Flash i 64 kB SRAM
– 4 pojemnościowe przyciski dotykowe
– Interfejs kart SD
– 3-osiowy akcelerometr (MMA7660)
– 2 przyciski i potencjometr
– 4 diody LED
– Port podczerwieni



– Interfejs JTAG
– Złącze systemu Freescale Tower
– Możliwość użycia interfejsów USB, Ethernet, RS-232/485, CAN, SPI, I²C i Flexibus poprzez Freescale Tower System

Ponadto model K40X256-KIT zawiera 28-segmentowy wyświetlacz LCD.

Cypress Semiconductor

Własne zestawy dla układów z rdzeniami Cortex-M3 ma również firma Cypress Semiconductor. Zawierają one mikrokontrolery PSoC.

CY8CKIT-014

Jest to interesujący zestaw z dużą liczbą czujników. Jego cena wynosi około 170 zł netto.

Podstawowe cechy:

– Mikrokontroler Cypress CY8C5588A-XI-060 (256 kB Flash, 64 kB SRAM)
– Czujnik zbliżeniowy marki Samtec
– 3-osciowy akcelerometr (Kionix)
– 5-segmentowy pojemnościowy suwak zbliżeniowy
– 1 przycisk użytkownika
– 8 diod LED
– Termistor marki Murata
– Złącze USB
– 22 złącza GPIO
– Złącze baterii 9 V

PIC32
Korzyści płynące ze stosowania 32-bitowych mikrokontrolerów

Minęło niemal 40 odkąd powstał pierwszy w pełni zintegrowany procesor, zawierający pamięć oraz układy wejść i wyjść. O ile był on wtedy postrzegany jako nietypowe urządzenie sterujące, szybko powstała cała gama tego typu układów, w efekcie czego ustalił się podział pomiędzy mikrokontrolery i mikroprocesory. Przez lata ich ścieżki rozwoju biegły osobno – mikrokontrolery stały się podstawą aplikacji wbudowanych, podczas gdy główny nacisk na rozwój stawiany był na procesory.

Mikroprocesory od lat miały przewagę pod względem wydajności nad mikrokontrolerami. Jednakże te drugie dominowały w dziedzinie wszechstronności. Obecnie na rynku widać duże zapotrzebowanie na podzespoły zarówno bardzo wydajne, jak i wyposażone w liczne interfejsy. Wiele z nowoczesnych mikrokontrolerów obsługuje zaawansowane interfejsy kablowe i bezprzewodowe, ale często brakuje im odpowiednio dużej mocy obliczeniowej potrzebnej do przetwarzania danych zdobywanych dzięki dostępowi do sieci komputerowych. Co więcej, duża wydajność potrzebna jest także do zapewnienia bezpieczeństwa komunikacji. Z tych względów w ostatnim czasie powstał trend, który sprawia, że na rynku pojawiają się nowe mikrokontrolery wzbogacone o zaawansowane podukłady przetwarzania danych.

Czy 32-bitowa wydajność jest potrzebne?

Na rynku dostępnych jest wiele dosyć wydajnych, nie tylko 32-bitowych mikrokontrolerów, wystarczających do licznych zaawansowanych aplikacji. Jednakże decyzja o użyciu układu 32-bitowego pozwala mieć nieco mocy „w zapasie", co ułatwia aktualizację urządzenia w przyszłości. Bardzo często okazuje się, że zastosowanie bardziej wydajnego mikrokontrolera tylko nieznacznie podnosi koszt całkowity urządzenia, ale za to istotnie zwiększa jego funkcjonalność. Powszechne obecnie, aktualizowanie oprogramowania urządzeń pozwala więc na szybkie wytworzenie produktu, który z czasem zostanie rozbudowany o dodatkowe funkcje – o ile tylko użyto w nim odpowiednio wydajnej jednostki przetwarzania danych.

Jak przejść na 32-bitową architekturę?

W praktyce okazuje się, że bardzo wiele firm zauważyło korzyści płynące ze stosowania 32-bitowych mikrokontrolerów i zdecydowało się na zmianę architektury używanych układów. Jednakże w wielu przypadkach zmiana ta może nie być łatwa, gdyż tylko niektórzy producenci µC oferują swoim klientom prostą ścieżkę ewolucji produktów. W efekcie, pomimo licznej konkurencji, wybór mikrokontrolera 8- lub 16-bitowe-

go może okazać się dosyć prosty w porównaniu do wyboru układu 32-bitowego. Warto jednak zauważyć, że wybór ten powinien być rozpatrywany nie tylko w kontekście konkretnych urządzeń, ale przede wszystkim jako selekcja producenta i jego narzędzi.

Firma Microchip wprowadziła na rynek pierwsze mikrokontrolery 32-bitowe, rodzinę PIC32, w 2007 roku. W przeciwieństwie do wielu konkurentów, skoncentrowała się na zapewnieniu bezproblemowej możliwości migracji pomiędzy dowolnymi układami PIC. W efekcie, decyzja o wyborze architektury 32-bitowej nie musi być jednoznaczna z rezygnacją z 8- lub 16-bitowej (**rysunek 1**). Z tego powodu, firmy które stosują mikrokontrolery Microchipa zamiast stawiać sobie pytanie: „jak trudno będzie przenieść kod na układy 32-bitowe?", mogą zastanawiać się, jak dołączyć urządzenia oparte na tych wydajnych podzespołach do obecnej oferty produktowej.

Wspólne narzędzia, wspólne biblioteki

Odpowiedź na postawione powyżej pytanie będzie oczywiście zależeć od konkretnego producenta, ale niezależnie od niej, należy mieć świadomość, że zazwyczaj decyzja o zmianie architektury powoduje powstanie dodatkowych kosztów. Najczęściej konieczny jest zakup nowych narzędzi deweloperskich, środowiska programistycznego i bibliotek programowych. Wyjątkiem są sytuacje, gdy producent układu scalonego zapewnia kompatybilność wsteczną swoich narzędzi, bez poświęcenia kompatybilności z przyszłymi mikrokontrolerami. Przykładem takiego działania jest właśnie Microchip, którego większość narzędzi programistycznych przeznaczonych do układów 32-bitowych jest zgodna z istniejącymi układami 8- i 16-bitowymi.

Jest to możliwe m.in. dzięki temu, że darmowe biblioteki Microchipa są napisane w języku C. Raz napisana w środowisku MPLAB aplikacja może częstokroć być bardzo szybko skompilowana zarówno na układ 8-, 16-, jak i 32-bitowy, nie tracąc na wydajności W efekcie, gdy kod danej aplikacji rośnie z czasem, w pewnym momencie możliwe jest po prostu przekompilowanie go dla mikrokontrolera o nowszej architekturze. To samo dotyczy zestawów deweloperskich, takich jak PICTail Daughter Board.

Warto zauważyć, że układy PIC32 są kompatybilne z innymi rodzinami PIC także sprzętowo. Oznacza to, że wiele z podzespołów użytych np. w 16-bitowej aplikacji będzie równie dobrze współpracowała z PIC32. Żaden inny producent mikrokontrolerów nie zapewnia tak prostej migracji pomiędzy swoimi układami, nawet jeśli są one oparte o własne, autorskie architektury.

Wpływ wydajności na cenę

Kolejnym pytaniem, jakie mogą zadawać sobie producenci urządzeń dotyczy wydajności, jakiej faktycznie będą potrzebować w nowych produktach. Należy jednak pamiętać, że obecnie koszt wytworzenia scalonego układu krzemowego nie jest już na tyle duży, by miał główny wpływ na całkowity koszt produkcji gotowego urządzenia. Dlatego pytania o wydajność nie należy traktować jako głównego czynnika wpływającego na cenę.

Firma Microchip stosuje w rodzinie PIC32 rdzeń MIPS32 M4K, którego wydajność w optymalnych warunkach, w przeliczeniu na megaherc zegara wynosi 1,5 DMPIS/MHz (1,5 mln operacji stałoprzecinkowych na megaherc). Jest to wynik lepszy niż w każdym innym z obecnie dostępnych 32-bitowych mikrokontrolerów.

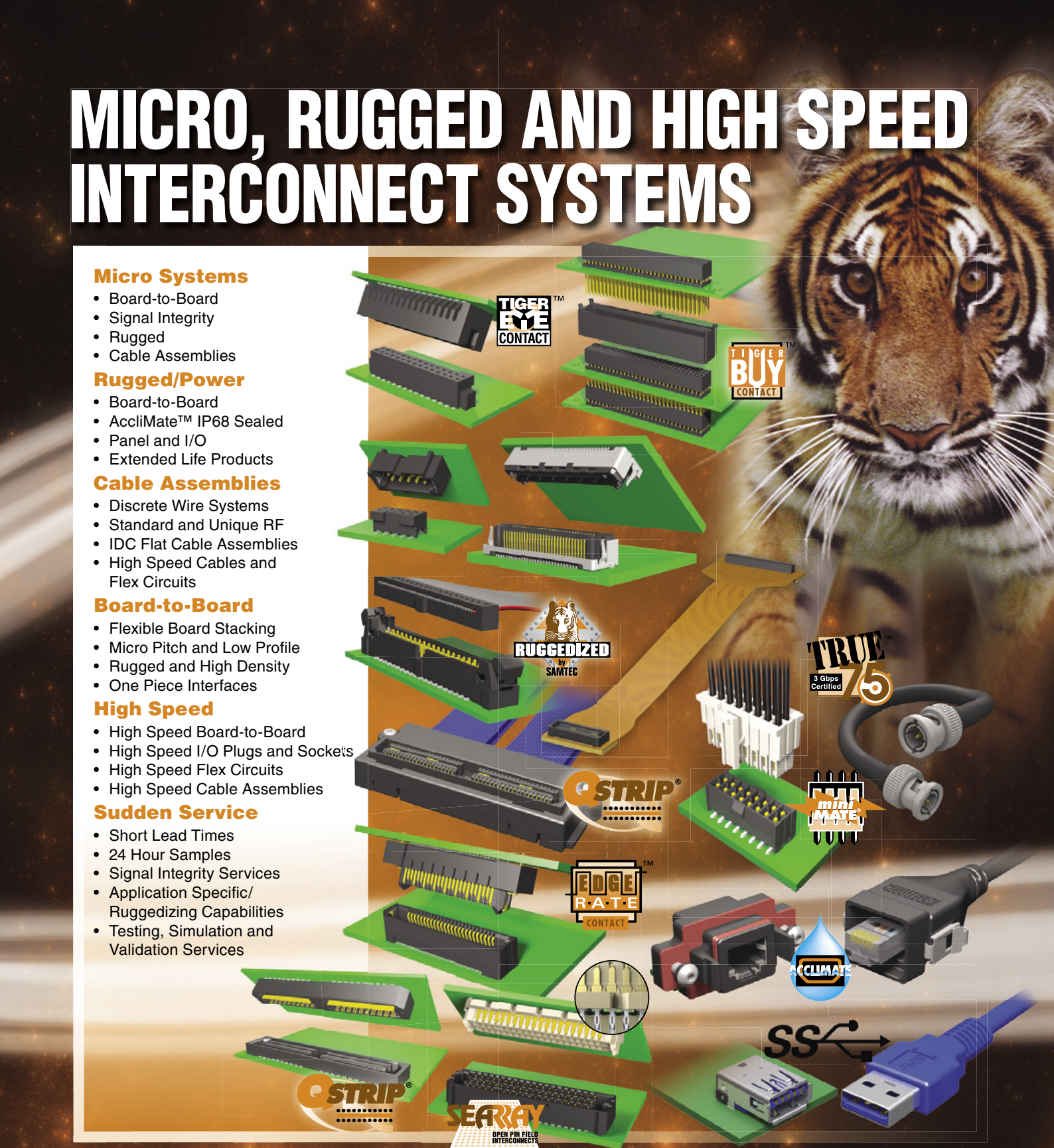
Wydajność układów PIC32 jest ponad dwukrotnie wyższa niż w przypadku 16-bitowych PIC24, dzięki czemu otwiera przed konstruktorami szereg nowych możliwości zastosowań. Aby okiełznać ten potencjał, Microchip współpracuje z dużą grupą firm trzecich, które tworzą cały „ekosystem" narzędzi deweloperskich zgodnych z architekturą MIPS. Uzupełniają one i tak duży zestaw bibliotek Microchipa i tym bardziej ułatwiają migrację pomiędzy układami.

32-bitowe możliwości

Korzyści płynące z zastosowania 32-bitowej architektury wynikają m.in. z dostępności 32-bitowych rejestrów oraz 32-bitowej jednostki arytmetyczno-logicznej zdolnej do wykonywania operacji typu „pomnóż i zsumuj" (MAC – *multiply and accumulate*) w trakcie jednego cyklu. W rezul-

ZAJRZYJ NA TE STRONY
www.wobit.com.pl
silniki.pl
silniki.com
enkodery.pl
www.cyfronika.com.pl
elektronika dla wszystkich
sklep internetowy
wszystko dla elektroniki
www.cyfronika.com.pl
renex
NARZĘDZIA DLA ELEKTRONIKÓW
www.renex.com.pl
www.piekarz.pl
Hurtownia części elektronicznych
firma@piekarz.pl tel. 022-835-50-37 fax 022-213-92-82
MS Elektronik
Dystrybutor Elementów Elektronicznych
Tel. (58) 629 24 69
Faks: (58) 629 32 00
E-mail: info@mselektronik.com.pl
Oferta czynnych i biernych elementów elektronicznych renomowanych producentów
www.mselektronik.com.pl
GAMMA
info@gamma.pl
www.gamma.pl
PODZESPOŁY ELEKTRONICZNE
HUMA Co.
www.humasklep.pl
www.maszczyk.pl
MASZCZYK
05-071 Sulejówek-Miłosna
ul. Mickiewicza 10
tel.: 22 783 45 20
fax: 22 783 90 85
maszczyk@maszczyk.pl
MERSERWIS
ul. Gen. Wł. Andersa 10
00-201 Warszawa
fax/tel:+48 22 831 42 56
aparatura kontrolno pomiarowa, elementy automatyki, serwis
www.merserwis.pl
info@feryster.pl
sklep.INDUCTORS.pl
ELEMENTY INDUKCYJNE

MICRO, RUGGED AND HIGH SPEED INTERCONNECT SYSTEMS



Micro Systems
- Board-to-Board
- Signal Integrity
- Rugged
- Cable Assemblies

Rugged/Power
- Board-to-Board
- AccliMate™ IP68 Sealed
- Panel and I/O
- Extended Life Products

Cable Assemblies
- Discrete Wire Systems
- Standard and Unique RF
- IDC Flat Cable Assemblies
- High Speed Cables and Flex Circuits

Board-to-Board
- Flexible Board Stacking
- Micro Pitch and Low Profile
- Rugged and High Density
- One Piece Interfaces

High Speed
- High Speed Board-to-Board
- High Speed I/O Plugs and Sockets
- High Speed Flex Circuits
- High Speed Cable Assemblies

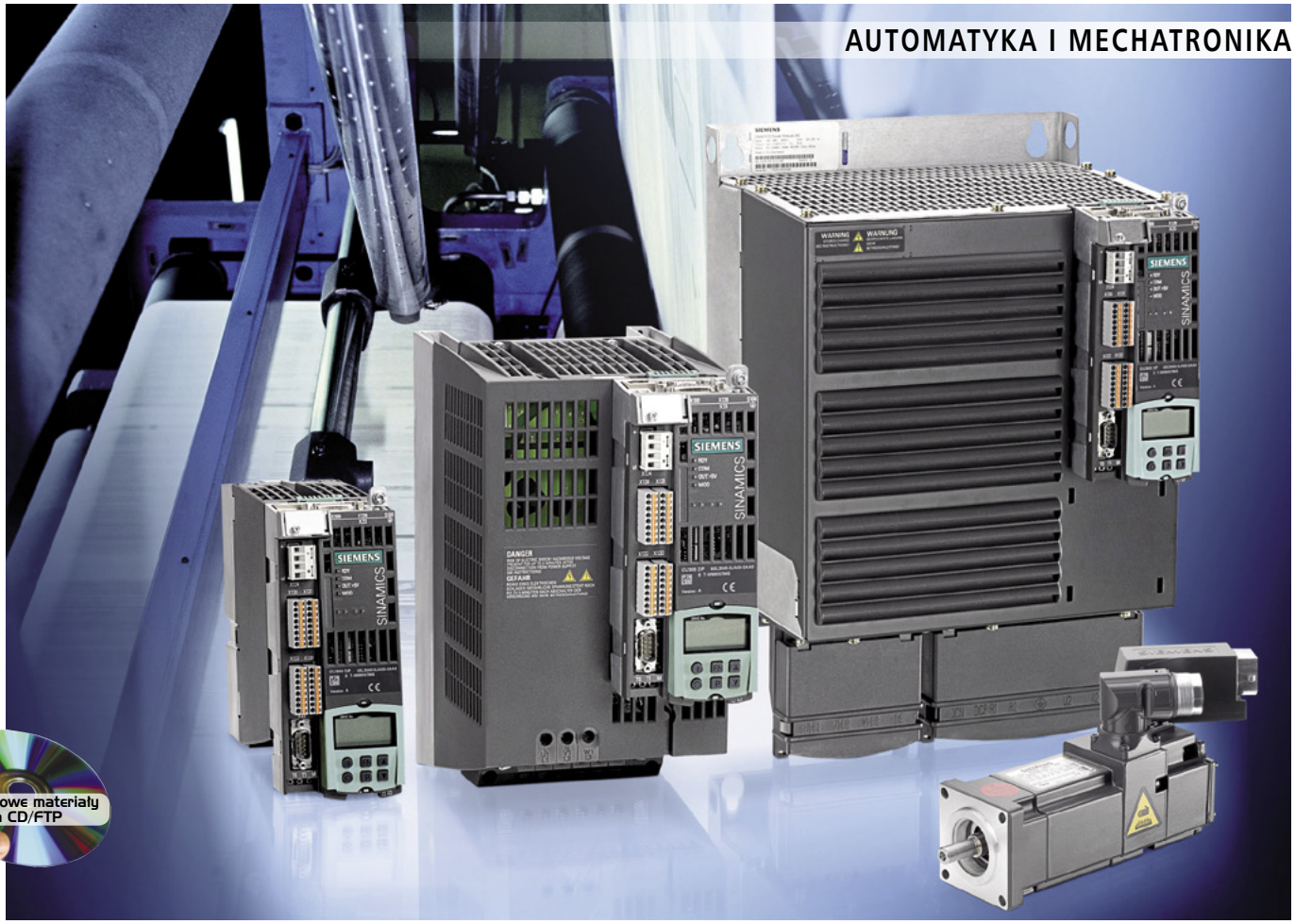
Sudden Service
- Short Lead Times
- 24 Hour Samples
- Signal Integrity Services
- Application Specific/ Ruggedizing Capabilities
- Testing, Simulation and Validation Services

samtec
SUDDEN SERVICE

eltronika
www.eltronika.pl

ELTRONIKA Sp. z o.o.
ul. Warszawska 41 pok. 7 • 05-092 Lomianki
tel. +48 22 751 9744 • fax: +48 22 751 9774
E-mail: info@eltronika.pl

ELTRONIKA - oddz. Katowice
ul. Dabrowskiego 1 • 40-032 Katowice
tel./fax: +48 32 257 2484
E-mail: katowice@eltronika.pl



Pozycjonowanie napędu SINAMICS (2)

W przykładzie pokażemy, w jaki sposób można pozycjonować silnik serwomechanizmu za pomocą sterownika napędu Siemens Sinamics S110 z użyciem interfejsu impulsowego sterownika S7-1200 CPU1214C. Zarówno napęd serwomechanizmu, jak i S7-1200 CPU mają indywidualne wewnętrzne liczniki impulsów, których zawartość reprezentuje bieżącą pozycję napędu. Przed przemieszczeniem do pozycji bezwzględnej, licznik w S7-1200 CPU musi zostać zsynchronizowany z pozycją fizyczną osi silnika.

Dodatkowe materiały na CD/FTP:
ftp://ep.com.pl, user: 10925, pass: 87thc181
- pierwsza część artykułu

Obliczanie maksymalnej częstotliwości dopuszczalnej dla silnika

W celu zapewnienia, że nie nastąpi wymuszenie pracy silnika z prędkością większą niż nominalna, trzeba określić maksymalną częstotliwość silnika, która może zostać wygenerowana przez interfejs impulsowy modułu S7-1200 CPU. Do wykonania tego trzeba znać nominalną prędkość pracy silnika oraz liczbę impulsów przypadającą na jeden obrót.

Szczególną cechą modułu Sinamics S110 jest możliwość zmiany liczby impulsów na jeden obrót silnika. Możliwy jest więc wybór między większą dokładnością pozycjonowania a największą możliwą dynamiką napędu. Oznacza to, że gdy dokładność pozycjonowania wzrasta (większa liczba impulsów na obrót), to wał silnika porusza się mniejszymi skokami kąta. Występuje w tym przypadku ograniczenie maksymalnej prędkości.

Jeżeli dynamika jest większa (mniejsza liczba impulsów na obrót), to szybciej można osiągnąć nominalną prędkość obrotową silnika (a nawet ją przekroczyć). Jednakże wiąże się to ze wzrostem kąta obrotu silnika na jeden impuls. Pozycjonowanie jest wówczas mniej dokładne.

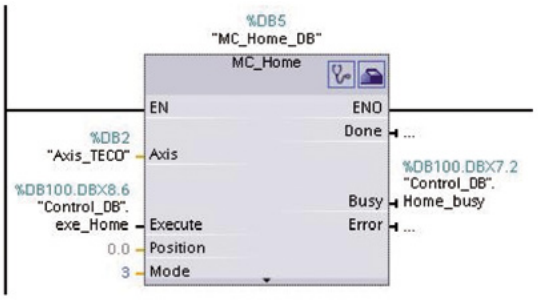
W tym przykładzie konfiguracji celem jest osiągnięcie nominalnej prędkości silnika. Zatem należy obliczyć liczbę impulsów na obrót. Rezultatem tego powinna być maksymalna częstotliwość silnika (nominalna prędkość) równa 3000 obrotów na minutę, co odpowiada maksymalnej możliwej dla S7-1200 częstotliwości impulsów wynoszącej 100000 impulsów na sekundę.

W tym wypadku obliczenie liczby impulsów dla napędu przebiega następująco:

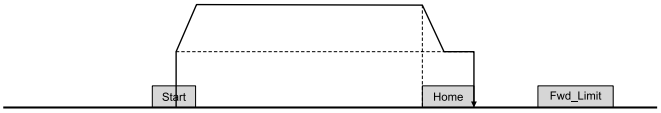
Dane:

Nominalna prędkość silnika serwomechanizmu (T_{Motor}) = 3000 rpm
Maksymalna częstotliwość impulsów z CPU (f_{CPU}) = 100000 pps

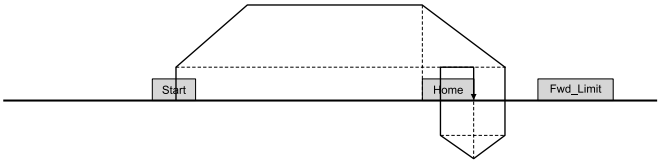
Obliczenia:



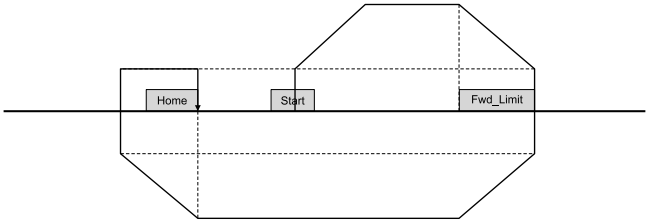
Rysunek 20. Blok „MC_Home"



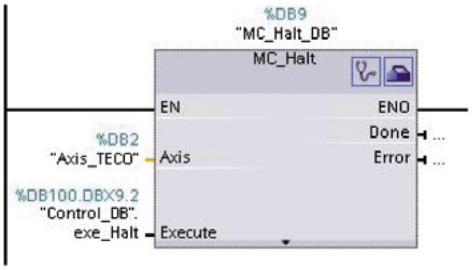
Rysunek 21. Pozycja początkowa na lewo od punktu odniesienia, a spowolnienie ruchu przed osiągnięciem zbocza ujemnego



Rysunek 22. Pozycja początkowa na lewo od punktu odniesienia, a spowolnianie ruchu nie jest zakończone przed osiągnięciem zbocza ujemnego



Rysunek 23. Pozycja początkowa na prawo od punktu odniesienia



Rysunek 24.

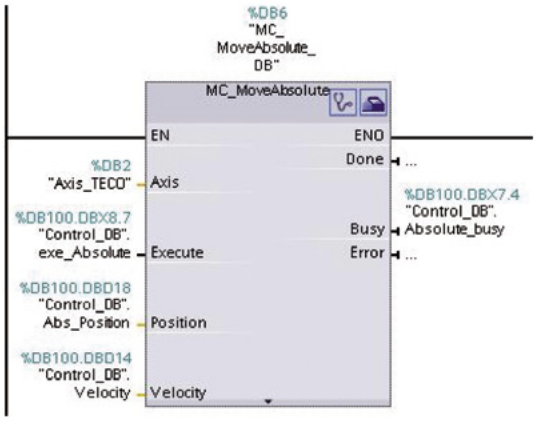
Przerwanie zadania (MC_Halt)

Każde aktywne zadanie, tzn. każdy aktywne przesuwanie osi, może zostać zatrzymane za pomocą bloku „MC_Halt" (**rysunek 24**). Na dodatnim zboczu sygnału na wejściu „Execute" oś zostaje zatrzymana z pewnym opóźnieniem. Pozycja, w której oś się zatrzymuje, nie jest określona.

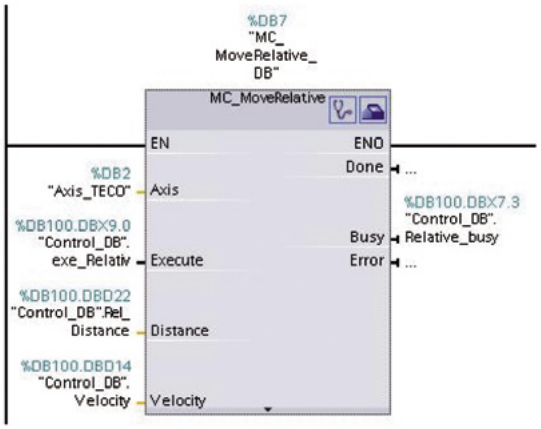
Poza tym każde aktywne zadanie może zostać przerwane przez uruchomienie nowego zadania. Zawsze tylko ostatnio uruchomione zadanie jest aktywne. Przykład: oś porusza się z określoną prędkością. Jeżeli zostanie uaktywnione zadanie sterowania ręcznego, to zadanie ze stałą prędkością zostaje skasowane, a uaktywnione zadanie z ręcznym sterowaniem.

Pozycjonowanie bezwzględne (MC_MoveAbsolute)

Dzięki możliwości ustawienia w pozycji spoczynkowej, znana jest bieżąca pozycja osi. Za pomocą bloku „MC_MoveAbsolute" (**rysunek 25**) można osiągnąć dowolną pozycję w [mm] w granicach



Rysunek 25. Blok MC_Halt



Rysunek 26. Blok MC_MoveRelative

krańców mechanicznych, specyfikując rzeczywistą pozycję. Poza tym trzeba podać wartość prędkości ruchu.

Jeżeli wykonanie bloku rozpoczyna się dodatnim zboczem sygnału podanego na wejście „Execute", następuje obliczenie wymaganej liczby impulsów, koniecznej do przemieszczenia osi do pozycji docelowej. Następnie silnik zostaje przyspieszony, jeśli to możliwe, do określonej prędkości, a następnie zatrzymany, z pewnym opóźnieniem, w pozycji docelowej.

Pozycjonowanie względne (MC_MoveRelative)

Oprócz pozycjonowania bezwzględnego, istnieje też opcja przesunięcia względnego na dowolną odległość, w wybranym kierunku i z wybraną prędkością za pośrednictwem bloku „MC_MoveRelative" (**rysunek 26**).

Po uruchomieniu bloku dodatnim zboczem na wejściu „Execute", oś przesuwa się na określoną odległość z wybraną prędkością. Kierunek jest określany na podstawie znaku (+/–) wartości odległości.

Zerowanie pozycji (Clear Position)

Zerowanie pozycji napędu jest stosowane dla osi obrotowych w celu uniknięcia przemieszczenia poza maksymalną możliwą pozycję, a więc uniknięcia powodowanych tym zakłóceń. Jeżeli aktywne jest wyjście „Clr", liczniki dla nastawionej i aktualnej pozycji napędu zostają ustawione na „0". Jest to konieczne dla eliminacji możliwej różnicy zawartości tych liczników.

Jeżeli pojawi się błąd pozycjonowania wynikający ze znacznej różnicy między nastawioną i aktualną pozycją, jedyną możliwością korekcji tego błędu jest zresetowanie pozycji. Przy włączaniu napędu, wyjście cyfrowe „CLR" jest automatycznie uaktywniane przez jedną sekundę, ponieważ, w zależności od typu enkodera, po włączenia zasilania może pojawić się różnica między wartością ustawioną i aktualną.

Andrzej Gawryluk

Opracowano na podstawie materiałów firmy Siemens.

Gotowe moduły liniowe
Aplikacje związane z paletyzacją i przenoszeniem produktów

Firma WObit od 1991 roku funkcjonuje na polskim rynku automatyki przemysłowej, nie tylko jako dystrybutor, ale również jako producent całego szeregu rozwiązań. W portfolio produkcyjnym WObitu znajdują się takie urządzenia, jak sterowniki oraz zadajniki trajektorii dla silników krokowych, sterowniki silników DC, różnego rodzaju urządzenia pomiarowe pozwalające na podłączenie czujników monitorujących szereg wielkości fizykalnych (np. drogę, siłę, moment obrotowy itp.). W 2009 rozpoczęto produkcję autorskiego rozwiązania mechatronicznego, jakim jest inteligentna oś liniowa.

Rysunek 2. Zadajnik trajektorii

Moduł liniowy MLA napędzany paskiem zębatym

Moduł liniowy MLA, przedstawiony na **rysunku 1** jest ekonomicznym urządzeniem do realizacji przemieszczeń liniowych wymagających pozycjonowania. Moduł bazuje na profilu aluminiowym o zastrzeżonym wzorze przekroju i szynowej prowadnicy liniowej, po której toczący się wózek napędzany jest paskiem zębatym. Pozycjonowanie paskiem zębatym jest tańszą choć mniej dokładną alternatywą dla drogich rozwiązań bazujących na śrubach pociągowych.

Moduł napędzany jest standardowo silnikiem krokowym, a ruch obrotowy kół zębatych zamieniany jest na przemieszczenie liniowe wózka prowadnicy za pomocą paska zębatego.

Zintegrowany układ sterowania silnikiem krokowym zamknięty w kompaktowej obudowie eliminuje nadmiar kabli i sprawia, że konstrukcja jest zwarta i nie należy martwić się o dodatkowe elementy sterowania. Dostępna jest elektronika ze zwykłym sterowaniem mikrokrokowym kontrolowanym wejściami impulsowymi i kierunku oraz bardziej zaawansowana elektronika sterująca zawierająca zadajnik trajektorii (**rysunek 2**). Interfejs USB umożliwia łatwe połączenie z każdym komputerem PC. Układ sterowania posiada dodatkowo wejścia/wyjścia, które mogą być dowolnie skonfigurowane i wykorzystane np. do synchronizacji z drugim modułem, bądź do sterowania innymi elementami wykonawczymi.

Rysunek 1. Moduł liniowy MLA

Tabela 1. Porównanie parametrów modułu liniowego MLA i MLAS

Moduł liniowy	Maks. prędkość liniowa [m/s]*	Precyzja pozycjonowania [mm]*	Maks. siła udźwigu [N]*	Przełożenie napędu	Zastosowana prowadnica	Zastosowany wózek
MLA	1	0,1	40	1 [obr]/150 [mm]	HGR15	HGH15
MLAS	0,2	0,01	1000	1 [obr]/4 [mm]	EGR15	EGW15CC

* wartość parametru zależy od zastosowanego napędu

Oś liniowa na pasku zębatym została stworzona do określonych rozwiązań. Jej główną zaletą jest łatwość aplikacji, kompaktowa budowa i bardzo atrakcyjna cena wynikająca z zastosowanej technologii. Niestety przy napędzie wykorzystującym pasek zębaty można przenosić elementy o stosunkowo niskiej masie (do 4 kg w układzie pionowym). Przedstawione warunki predestynują moduł liniowy MLA do przenoszenia np. czujników pomiarowych, skanerów, a także niewielkich towarów. Jej zastosowanie w paletyzacji definiuje zatem przede wszystkim waga przenoszonego produktu. Należy przy tym uwzględnić wagę zastosowanego chwytaka mechanicznego lub pneumatycznego.

Moduł liniowy MLAS napędzany śrubą kulową

W 2010 roku w wyniku zapotrzebowania rynku na napęd liniowy o zwiększonej nośności oraz precyzji, firma WObit zaprojektowała i wyprodukowała moduł liniowy napędzany śrubą kulową. Moduł MLAS został przedstawiony na **rysunku 3**. Nowe rozwiązanie przy zachowaniu założeń związanych z ergonomią i ekonomią, stwarza możliwość przenoszenia znacznie większych ciężarów (do 100 kg w układzie pionowym) i umożliwia pozycjonowanie z dokładności