

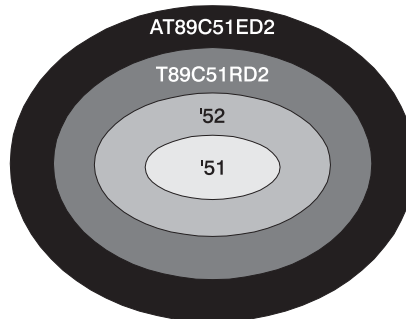
## Wstęp

Mikrokontrolery z rodziny MCS51, mimo upływu lat, nadal cieszą się dużą popularnością. Jak na razie nic nie wskazuje na schyłek ich popularności, a wręcz przeciwnie – producenci cały czas wprowadzają do oferty nowe mikrokontrolery z tej rodziny. Wystarczy tu wspomnieć, że większość mikrokontrolerów z wbudowanym kontrolerem USB to procesory z rdzeniem '51. Podobna sytuacja występuje w układach ze zintegrowanym modulem do transmisji radiowej (*Chipcon, Nordic*). W ofertach producentów pojawiają się również mikrokontrolery o bardzo dużej mocy obliczeniowej pracujące z sygnałem zegarowym o częstotliwości 200 MHz. Nowe odmiany procesorów z coraz bardziej rozbudowanymi modułami peryferyjnymi oraz potrzeba tworzenia aplikacji w krótkim czasie powodują, że coraz częściej do pisania oprogramowania dla mikrokontrolerów są wykorzystywane języki wysokiego poziomu. W tej dziedzinie niekwestionowanym liderem jest język C. O zaletach programowania w tym języku napisano już wiele książek, więc w zasadzie nikogo nie trzeba do niego przekonywać. Oczywiście, na rynku są dostępne kompilatory innych języków wysokiego poziomu (np. Pascal, Basic, Fortran), które są wykorzystywane do tworzenia oprogramowania dla mikrokontrolerów. Część z tych kompilatorów (np. Bascom – kompilator języka Basic dla mikrokontrolerów AVR i '51) zdobyła dość dużą popularność, jednak śmiało można powiedzieć, że jedynie język C stał się swego rodzaju standardem.

Przykłady zamieszczone w niniejszej książce były pisane z wykorzystaniem kompilatora Keil C. Wybrano go dlatego, że jest on obecny na rynku już od kilkunastu lat i przez ten czas stał się solidnym (i co bardzo ważne: stabilnym) standardem. Bardzo często można spotkać się z opiniami, że program napisany we wcześniejszej wersji jakiegoś kompilatora, nie daje się skompilować przy wykorzystaniu aktualnej wersji tego samego kompilatora. Takich obaw nie muszą mieć użytkownicy Keil C – stwierdzenie to jest wynikiem kilkunastoletniego używania różnych wersji tego kompilatora. Co więcej, Keil C – jako dobrze dopracowany produkt – wraz ze zmianą wersji nie wymusza na użytkowniku konieczności poznawania nowego środowiska pracy. Oznacza to, że raz zdobyta wiedza i doświadczenie w użytkowaniu kompilatora procentuje przez wiele lat. Wszystkie prezentowane dalej programy mogą być bez zmian kompilowane ostatnią wersją kompilatora Keil 7.5 mimo, że podczas przygotowywania książki używałem wersji Keil 7.0.

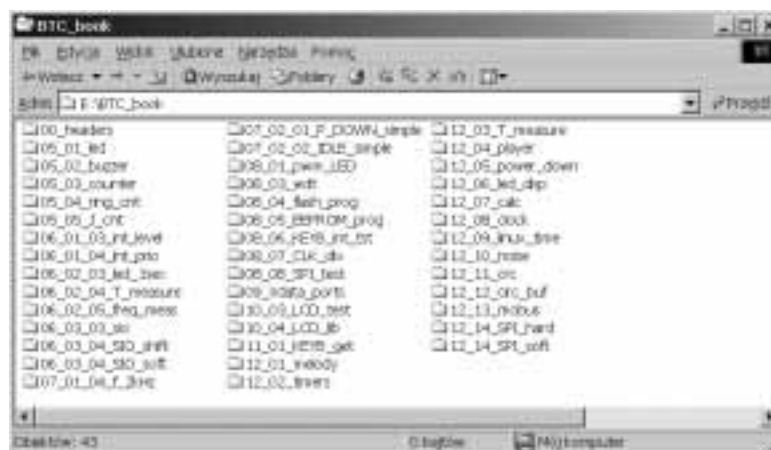
Przykłady przedstawione w książce wykonano z wykorzystaniem zestawu uruchomieniowego ZL2MCS51 (<http://www.btc.pl/index.php?id=zl2mcs51>). Zestaw ten wykorzystuje mikrokontroler AT89C51RD2. Jest to procesor z pamięcią Flash wyposażony w tzw. *bootloader*. *Bootloader* to program umożliwiający programowanie mikrokontrolera przez złącze transmisji szeregowej RS232. Dzięki temu mikrokontroler ten można programować bez jakiegokolwiek dodatkowego sprzętu. Wystarczy komputer PC, bezpłatny program dostępny na witrynie firmy Atmel (<http://www.atmel.com>) o nazwie *Flip* oraz kabel do transmisji szeregowej. Wybrany mikrokontroler jest bardzo rozbudowany – jest on wyposażony m.in. w wewnętrzną pamięć RAM, moduł UART, dodatkowe układy czasowe (timery).

Niniejsza książka ma budowę taką, jak pokazano na rysunku na następnej stronie. Najpierw przedstawiono możliwości procesorów typu '51. Następnie pokazano to,



co ulepszono w procesorach typu '52. Kolejne udoskonalenia dodała firma Temic, tworząc procesory typu T89C51RD2. Produkt ten przejęła firma Atmel i dalej udoskonała, czego efektem są procesory AT89C51RD2 i AT89C51ED2. Należy pamiętać, że nawet te najbardziej zaawansowane procesory po włączeniu zasilania zachowują się jak typowe, najprostsze procesory serii '51. Posługując się takim rozbudowanym mikrokontrolerem można przygotować programy i sprawdzić, jak działają nawet te najstarsze, najprostsze mechanizmy występujące we wszystkich procesorach z rodziny '51. Nie ma obowiązku korzystania z tego co zawarto w procesorach AT89C51RD2 i AT89C51ED2, choć wprowadzone udogodnienia są bardzo funkcjonalne przy atrakcyjnej cenie procesora.

Wersję demonstracyjną kompilatora Keil C wraz ze zintegrowanym środowiskiem programisty można pobrać ze strony producenta: <http://www.keil.com>. Ponieważ są to dość duże pliki, możliwe jest również zamówienie u producenta płyty CD-ROM z oprogramowaniem demonstracyjnym. Pliki źródłowe wszystkich przykładów zamieszczonych w książce można pobrać ze strony <http://www.btc.pl/index.php?id=c51pk> w postaci pliku *c51pk.zip*. Założono, że plik z przykładami o nazwie *c51pk.zip* zostanie rozpakowany do katalogu *E:/BTC\_book* – zawartość tego katalogu przedstawiono poniżej.



Podkatalogi folderu *E:\BTC\_book* zawierają projekty i kody źródłowe programów opisanych w dalszych rozdziałach. Na przykład folder *E:\BTC\_book\12\_10\_noise* zawiera wszystkie pliki źródłowe opisane w podrozdziale 12.10. Dodatkowy napis *noise* jest skrótowym wyjaśnieniem czego dotyczy program (generuje on pseudolosowy szum *noise*).

Większość przedstawionych w książce projektów można kompilować i symulować w wersji demonstracyjnej kompilatora (ograniczenie wielkości kodu wynikowego do 2 kB). W książce starałem się przedstawić sposoby wykorzystania większości zasobów mikrokontrolera AT89C51ED2. Zaprezentowane przykłady są, w miarę możliwości, proste. Wychodzę z założenia, że Czytelnik potrzebuje prostych i zrozumiałych przykładów, by posługując się nimi napisać swoje własne, bardziej rozbudowane programy.

Na zakończenie chciałbym złożyć podziękowania dla Pana dra Krzysztofa Kardacha za dotychczasową współpracę. Pan dr Jędrzej Ułasiewicz pomógł mi w przygotowaniu rozdziału nt. transmisji szeregowej i buforów kołowych.

*Jacek Majewski*