

ELEKTRONIK ELEKTOR

MIESIĘCZNIK DLA ELEKTRONIKÓW

co miesiąc w
Elektronice Praktycznej

System rozwojowy AVR-RISC

eksperyment z najnowszymi procesorami RISC Atmela

8-bitowy procesor AVR-RISC AT90S1200 firmy Atmel, dostępny od jakiegoś czasu, dał się już poznać jako bardzo szybki. Cena też jest dobra, gdyż mniej niż 3 funty, to tanio nawet na rynku hobbystów. Tak więc zastąpienie kilku układów logicznych (z podstawkami) układem AVR może się opłacać. Jednak najbardziej interesującą cechą tego procesora jest to, że ma interfejs szeregowy z bezpośrednim dostępem do wewnętrznej pamięci. Ogólna koncepcja procesora AVR była już omawiana w EE w styczniu 1998. Celem niniejszego artykułu jest przedstawienie kilku bardziej praktycznych zastosowań układu 90S1200 z wykorzystaniem łatwo dostępnego sprzętu i oprogramowania.

Aby zapewnić swojej nowej rodzinie procesorów dobry start, Atmel przygotował sporo narzędzi programowych, dostępnych bezpłatnie na jego stronie WWW. Jest tam, na przykład, kompletny asembler (dla DOS-a, jak również dla Windows) i symulator. Odpowiednie dane techniczne i dokumentację oprogramowania można również ściągać za darmo. A więc, jeśli chcesz się zapoznać z tym nowym procesorem bez wyrzekania się dotychczasowego sprzętu, radzimy zerknąć na stronę www.atmel.com, gdzie znajdziesz mnóstwo interesujących treści, włącznie z:

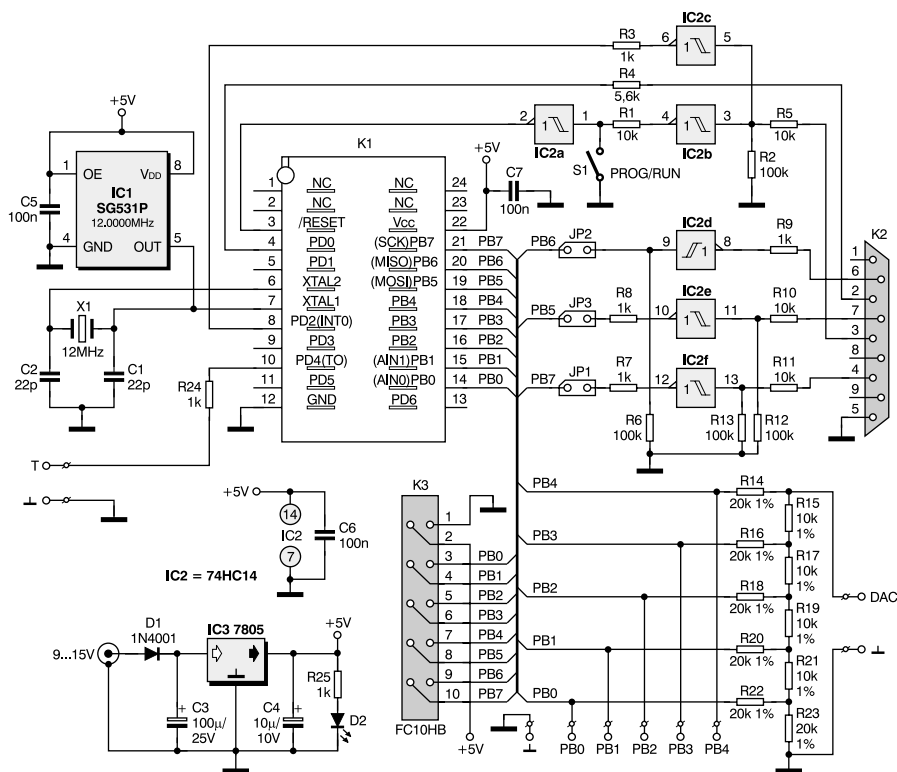
- asemblerem rodziny Atmel AVR;
- symulatorem rodziny Atmel AVR;
- przykładami programów ilustrujących prostotę arytmetyki oraz sposób korzystania z pamięci EEPROM.

System rozwojowy

Atmel udostępnił również płytkę demonstracyjną i związane z nią oprogramowanie sterujące. Inaczej jednak niż wspo-

mniane wcześniej oprogramowanie i dokumentacja, materiał ten nie jest wolny od opłat. Ponieważ miesięcznik *Elektor Electronics* stale kieruje się w stronę aktywnych (to jest lutujących) entuzjastów elektroniki, zaczęliśmy rozwijać nasz własny system eksperymentalny - programator dla procesorów AVR-RISC.

Jego schemat elektryczny przedstawiono na rys. 1. Najważniejszym elementem programatora jest sześciokrotny przerzutnik Schmitta, który zamienia sygnały interfejsu szeregowego PC (RS232) na poziomy, które mogą być przetwarzane przez interfejs szeregowy procesora, który nie jest kompatybilny ani z RS232, ani z innym asynchronicznym. Dlatego jest tam przełącznik umożliwiający przełączanie pomiędzy ładowaniem oprogramowania (ang. *software download*) a wykonywaniem programu (ang. *run program*) oraz możliwość wmontowania kwarcu lub modułu oscylatora kwarcowego, generującego sygnał zegarowy procesora.



Rys. 1. Schemat elektryczny systemu rozwojowego AVR.

Jeśli przewidujesz zastosowania wymagające dużej precyzji (na przykład pomiary częstotliwości lub czasu), najlepszym wyborem jest moduł oscylatora (osadzony w podstawie). W takim przypadku montowany jest tylko oscylator IC1, natomiast X1, C1 i C2 należy po prostu pominąć. Przy mniej krytycznych aplikacjach kwarc jest oczywiście bardziej oszczędną opcją. Jeśli zamierzasz programować mnóstwo procesorów, zaleca się stosowanie gniazda ZIF (ang. *zero-insertion force* - o zerowej sile wtykania). To jest to! Ze względu na swoją prostotę, układ może być zamontowany na kawałku płytki uniwersalnej.

Program narzędziowy ładowania szeregowego

Aby umożliwić ładowanie do procesora programów tworzonych przez Atmel Assembler, potrzebujesz specjalnego oprogramowania ładującego. Ponieważ autor nie znalazł odpowiedniego programu na stronie WWW Atmela, zaproponował własne rozwiązanie. Ten program i plik jego kodu źródłowego (napisanego w Pascalu) jest dostępny na dyskietce o numerze katalogowym 986020-1. Dyskietka ta i oprogramowanie ściągnięte na

stępie ze strony Atmela tworzą kompletny pakiet umożliwiający rozpoczęcie pracy z zestawem.

Przykładowe programy

Dobrym sposobem uczenia się programowania mikroprocesorów jest testowanie w pełni funkcjonujących programów, a następnie ich modyfikowanie i rozszerzanie. W przypadku nowych opracowań procesorów, decydującym czynnikiem jest często to, czy gotowe moduły programowe są dostępne, czy też nie. Moduły takie mogą uwolnić Cię od ciężaru pisania, powiedzmy, własnych podprogramów wprowadzania/wyprowadzania znaków, aby poprzestać tylko na tym jednym przykładzie.

Dyskietka projektu zawiera pewną liczbę takich modułów (patrz **tab. 1**), co powinno pomóc niedoświadczonym początkującym, w koncentrowaniu się na bardziej zasadniczych tematach podczas procesu uczenia się. Niestety, pełne omówienie wszystkich modułów z dyskietki przekracza rozmiary niniejszego artykułu.

Tych z Was, którzy interesują się szczegółami programowania, odsyłamy do komentarzy plików kodów źródłowych na dyskietce i do pliku XAVR.DOC, który

Tab. 1.

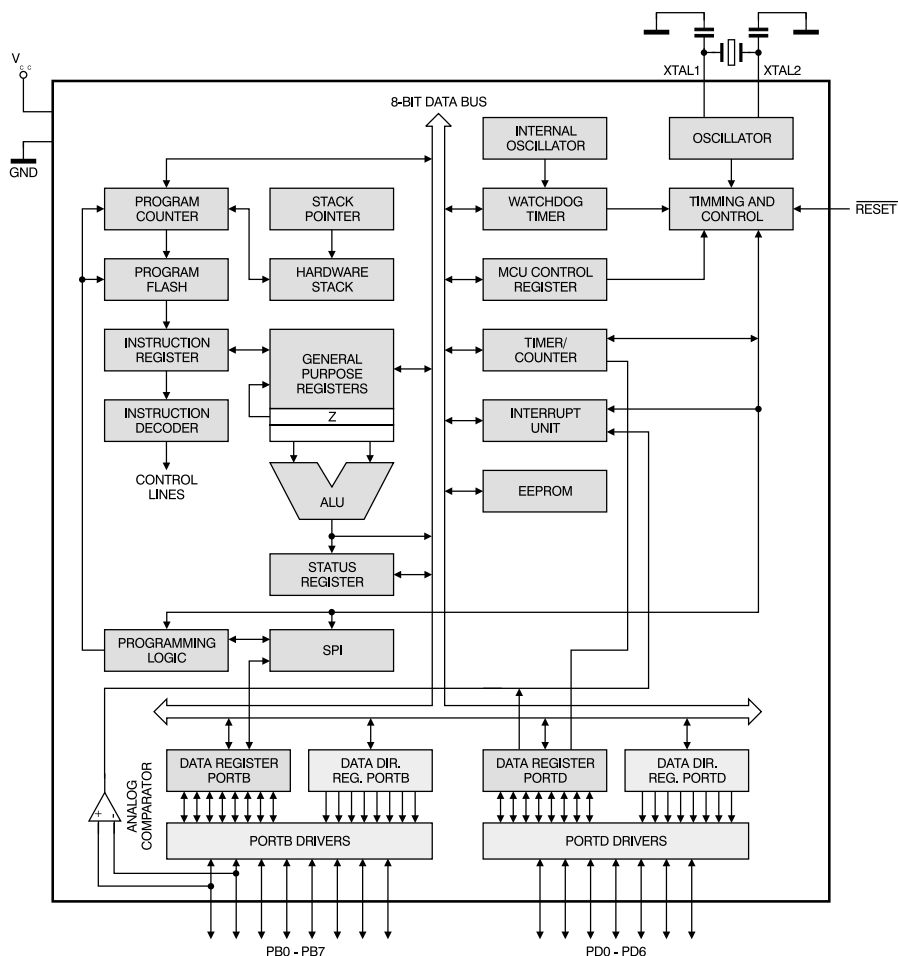
Programy PC

- SER90
program narzędziowy ładowania programów płytki rozwojowej. Na dyskietce plik źródłowy w Pascalu.
- V24COM
program komunikacji szeregowej V24. Na dyskietce załączono plik źródłowy w Pascalu.
- AVRASM.EXE
assembler AVR (wersja DOS). Załączony dzięki uprzejmości Atmel UK.

Programy przykładowe AT90S1200

- XSEROUT1
wyprowadza znaki za pośrednictwem RS232, 9600 bitów/s
- XHEXOUT1
wyprowadza rejestry za pośrednictwem RS232, format heksadecymalny
- XHEXOUT1
wyprowadza wszystkie 32 rejestry (zrzut) za pośrednictwem RS232
- XDECOUT1
wyprowadza 24-bitowe wartości (0 - 16777215), format dziesiętny, za pośrednictwem RS232
- XSERIN
wprowadza znaki za pośrednictwem RS232, 9600 bitów/s
- XFMES1
pomiar częstotliwości do 5MHz, bramka 1s, wyjście RS232, format dziesiętny, 9600 bitów/s
- XTMES1
pomiar czasu, 1ms do 8s, rozdzielczość 0,5ms, wyjście RS232, 9600 bitów/s
- XNCO1
oscylator sterowany numerycznie, wyjście 77,5kHz przy zastosowaniu kwarcu 12MHz
- XNCO2
generator pily sterowany numerycznie, programowany za pośrednictwem RS232
- XFRQDIV1
preskaler dzielący przez N, N programowane za pośrednictwem RS232
- XPATGEN1
prosty 8-kanalowy generator wzorca
- XPATGEN2
prosty 8-kanalowy generator wzorca, wzorec korygowany za pośrednictwem RS232
- XDIV625
preskaler dzielący przez 625, impulsy nie nakładające się

przedstawia przegląd dostępnych programów, uzupełniony krótkimi opisami i przykładami zastosowań. W tym artykule przedstawiamy tylko kilka głównych pomysłów, które mogą być zrealizowane za pośrednictwem AT90S1200. Naszym celem jest zebranie tych zastosowań, w których szybkość jest czynnikiem decydującym. Ostatecznie, nie ma sensu, aby



WYKAZ ELEMENTÓW

Rezystory

- R1, R5, R10, R11: 10kΩ
- R2, R6, R12, R13: 100kΩ
- R3, R7, R8, R9, R24, R25: 1kΩ
- R4: 5,6kΩ
- R14, R16, R18, R20, R22, R23: 20kΩ 1%
- R15, R17, R19, R21: 10kΩ 1%

Kondensatory

- C1, C2: 22pF, ceramiczne (patrz tekst)
- C3: 100nF/25V, stojący
- C4: 10nF/10V, stojący
- C5, C6, C7: 100nF, ceramiczny

Półprzewodniki

- D1: 1N4001
- D2: LED, czerwona wysokosprawna
- IC1: SG531P12.0000MHz EPSON (Eurodis), patrz tekst
- IC2: 74HC14
- IC3: 7805

Różne

- X1: 12MHz, patrz tekst
- K1: 24-stykowy Aries 0,3-0,6 cala (Famel)
- JP1, JP2, JP3: 2-stykowe zwory
- S1: przełącznik suwakowy, 1-pozycyjny, do druku
- K2: 9-stykowe gniazdo SUB-D
- K3: złącze 10-stykowe (opcjonalne)
- K4: gniazdo zasilacza sieciowego, do druku
- Dyskietka: nr katalogowy 986020-1

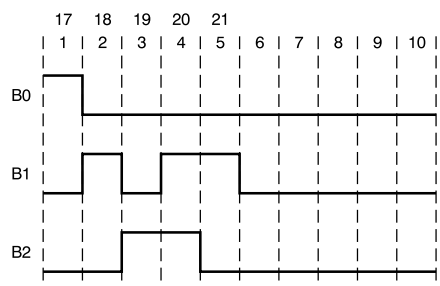
Rys. 2. Architektura procesora Atmel AT90S1200.

tak szybka bestia, jak procesor AVR-RISC włączała i wyłączała lampę co kilka sekund! Takich zastosowań nie znajdziesz w tym artykule. Przeciwnie, nauczysz się, na przykład, jak tworzyć różne pochodne sygnały zegarowe ze źródła 10MHz. Takie zastosowanie jest na pewno zbyt wielkim wymaganiem dla procesora 8051, natomiast AT90S1200 może mu poddać z łatwością!

Instrukcja co 66ns: zastosowania

Procesor AT90S1200 może pracować z częstotliwością zegara aż do 16MHz. Ponieważ większość instrukcji wykonuje się w jednym cyklu impulsów zegara, jest możliwe wykonywanie instrukcji z szybkością 16 milionów na sekundę. A więc jeśli chcesz, by pewien podprogram był wykonywany milion razy na sekundę, odnośna sekcja programu może mieć długość od 10 do 16 instrukcji. Za pośrednictwem takiej liczby instrukcji można wykonać wcale niemało. Oczywiście, wyłącznie na poziomie sprzętowym, któż bowiem może wyobrazić sobie program dla Windows o długości tylko kilkudziesięciu bajtów?

Naprawdę szybki procesor może być często użyty dla zaoszczędzenia kilku cyfrowych układów scalonych, gdy wymagana szybkość nie jest aż tak duża, by stosować kosztowne programowalne układy logiczne. Wielką zaletą takiego mikrokontrolera jest to, że



Rys. 3. Dzięki swojej wydajności kontrolery Atmel są w stanie generować złożone wzorce sygnałów cyfrowych z częstotliwościami znacznie powyżej 10MHz.

ma dostęp do całej armii rejestrów i do akumulatora zdolnego wykonywać sumowanie! Ci spośród was, którzy kiedykolwiek próbowali zmusić programowalne układy logiczne do prostych obliczeń, docenią to ułatwienie.

AT90S1200: krótki przegląd

Głównie z powodu swojej architektury RISC, struktura procesora AT90S1200 pozostaje cudownie prosta. Jej zarys przedstawiono na rys. 2. Sekcją szczególnie nas interesującą jest matryca 30 rejestrów 8-bitowych, które mogą być wykorzystane do przeprowadzania obliczeń i przechowywania wyników. Dalej, są tam dwa porty wejścia/wyjścia (Port B i Port D), których każde wyprowadzenie jest indywidualnie konfigurowalne jako wejście lub jako wyjście.

Wykonywanie programu przebiega pod kontrolą sygnału taktującego o częstotliwości zegara centralnego, która może mieć dowolną wartość pomiędzy 0 a 16MHz. Ponadto, jest tam system sterowania przerwami i mały 8-bitowy timer/licznik.

Kompletny opis procesora AT90S1200 obejmuje około 50 stron, a zainteresowanym czytelnikom usilnie polecamy poszukiwanie go w Internecie lub u lokalnego dystrybutora firmy Atmel.

A oto przykład programowania umożliwiający zapoznanie się z kontrolerem AVR.

Prosty cyfrowy generator słów

Ponieważ AT90S1200 jest w stanie pracować z dowolną częstotliwością zegara pomiędzy 0Hz (praca statyczna) a 16MHz, doskonale nadaje się do wykorzystania jako generator złożonych sygnałów wzorcowych, opartych na tylko jednej częstotliwości centralnego zegara. Zazwyczaj gene-

rator taki jest budowany przy wykorzystaniu modułów licznika i dekoderek, a końcowy układ może być dość skomplikowany, ponieważ może wymagać wielu dodatkowych elementów, aby przeciwdziałać powstawaniu niepożądanych efektów impulsowych (pików).

Przy częstotliwościach poniżej 16MHz jest jednak doskonale możliwe zatrudnienie procesora RISC. Na przykład, wzór impulsów, przedstawiony na **rys. 3** jest tworzony przy pomocy programu zamieszczonego na **list. 1**. Program ten jest krótki, a jego działanie ogranicza się do ładowania rejestrów od *r17* do *r22* pożądaną kombinacją sygnałów. Następnie, do wyprowadzania wartości z maksymalną szybkością poprzez Port B zastosowano pętlę.

Program ten może być modyfikowany na wiele sposobów. Na przykład, poprzez włączenie pętli opóźnienia możesz tworzyć sygnały o bardzo długich okresach, taktowanych z dokładnością kwarcu lub synchronicz-

nie z sygnałami zegara. Nie powinien być uznany za zbyt trudną taką modyfikację programu, by kształt sygnału nie zmieniał się w trakcie sześciu początkowych cykli zegara, a następnie, na przykład, po odczekaniu dokładnie 428376233 okresów zegara cały cykl zaczynał się od nowa. Wszystko, czego potrzebujesz dla realizacji tej funkcji, to licznik zdolny utrzymać ślad pożądaną liczbę cykli zegara. Do tego celu wystarczą cztery rejestry procesora, połączone w jedną całość o długości 32 bitów i kilka dodatkowych instrukcji. Nie jest potrzebne żadne lutowanie, gdy z jakiegokolwiek powodu należy zmienić liczbę cykli. Wszystko, co musisz zrobić, to zmiana kilku stałych w programie i to jest to. Elastyczność tego rodzaju jest po prostu niedostępna w tradycyjnej logice połączeń drutowych, a bardzo kosztowna w logice programowalnej (w szczególności narzędzia rozwojowe).

Kolekcja przykładowych programów, znajdująca się na dyskietce projektu, obejmuje prosty generator sygnałów wzorcowych odczytujący swoją informację wzoru z szeregowego interfejsu (za pośrednictwem XPATGEN2.EXE i XPATGEN2.PAT).

EE

Artykuł publikujemy na podstawie umowy z redakcją miesięcznika "Elektor Electronics".

Editorial items appearing on pages 21..24 are the copyright property of (C) Segment B.V., the Netherlands, 1998 which reserves all rights.