

Nowości w układach programowalnych

Zwiększający się udział układów programowalnych we współczesnej elektronice znajduje odbicie zarówno w dynamicznym rozwoju ich technologii produkcji, jak i w ich nowych możliwościach. Postęp jest tak szybki, że zacierają się różnice między układami ASIC i FPGA o dużej skali integracji. Układy programowalne zdobywają nowe obszary zastosowań, także w technice analogowej. W artykule przedstawiamy kilka interesujących rozwiązań, jakie zastosowano w układach programowalnych w kilku ostatnich miesiącach.

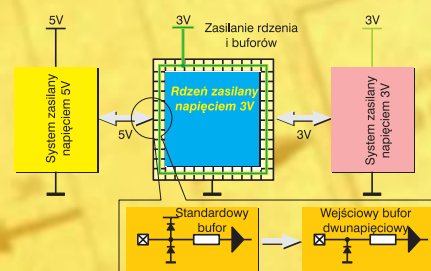
Wzrost zastosowania układów programowalnych w ostatnich latach jest tak gwałtowny i podąża tak wieloma ścieżkami, że praktycznie nie jest możliwe precyzyjne prześledzenie wszystkich trendów, jakie występują na rynku. W ostatnich trzech latach pojawiło się bardzo wiele pomysłów mających na celu udoskonalenie układów programowalnych, lecz do dzisiaj wykrystalizowało się i przetrwało zaledwie kilka z nich. Są to idee dotycząca zarówno poprawy parametrów elektrycznych (statycznych i dynamicznych), a także „logicznych“ możliwości konfiguracyjnych.

W tym, z konieczności krótkim, przeglądzie zostały pominięte rozwiązania, które są stosowane co najmniej od początku roku 1999, jak np.: możliwość programowania struktur w systemie, interfejs JTAG, duplikowanie nieulotnych matryc programowalnych EEPROM szybszymi odpowiednikami SRAM, itp. Nie zostały także uwzględnione pomysły - efemerydy, które pomimo długotrwałych prac są ciągle w stadium prób i testów, jak np. rekonfiguracja w locie układów programowalnych oraz układy z matrycami multikonkwestowymi.

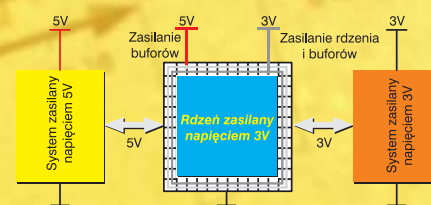
Obniżenie poboru mocy

Jednym z ważniejszych kierunków rozwoju współcześnie produkowanych układów programowalnych jest modyfikacja struktur, mająca na celu zminimalizowanie pobieranej przez nie mocy. Zabiegi te są zgodne z trendem występującym we współczesnej elektronice, który stymuluje konstruktorów urządzeń i - pośrednio - technologów, aby większość wysiłków twórczych skierować właśnie w tę stronę.

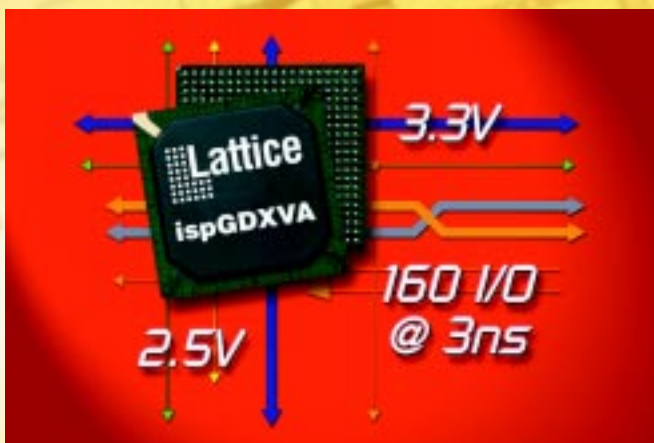
Jedną z najprostszych i najczęściej wykorzystywanych dróg osiągnięcia tego celu jest zasilanie układu napięciem niższym niż standardowe (za takie uważam 5V). Ze względu na konieczność umożliwienia pracy takiego układu w systemach zbudowanych w oparciu o standardowe układy cyfrowe TTL i CMOS stosowane są dwa rozwiązania:

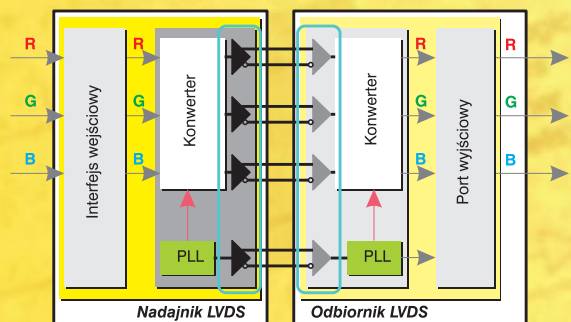


Rys. 1. Współpraca z otoczeniem układu z niskonapięciowymi buforami i rdzeniem.



Rys. 2. Współpraca z otoczeniem układu z rozdzielonym zasilaniem rdzenia i buforów.





Rys. 3. Tor przesyłania danych LVDS.

- Zasilenie całej struktury napięciem o obniżonej wartości, a przystosowanie tylko buforów I/O do współpracy z otoczeniem zasilanym napięciem standardowym (rys. 1). Rozwiązanie to jest aplikacyjnie proste, ponieważ w urządzeniu docelowym szyny zasilające są wyraźnie rozdzielone.
- Zasilenie rdzenia układu wraz z programowalną matrycą niższym napięciem (1,8..3,3V), a buforów I/O napięciem o standardowej wartości (rys. 2). Takie rozwiązanie nieco komplikuje budowę systemu docelowego, ale pozwala ujednolicić konstrukcję buforów I/O, co wprost przekłada się na powierzchnię struktury układu, a w konsekwencji jego cenę.

Obniżenie napięcia zasilającego było możliwe dzięki zastosowaniu do produkcji układów nowocześniejszych technologii wytwarzania półprzewodnikowych układów scalonych z wielokrotnym procesem metalizowania (ścieżki przewodzące).

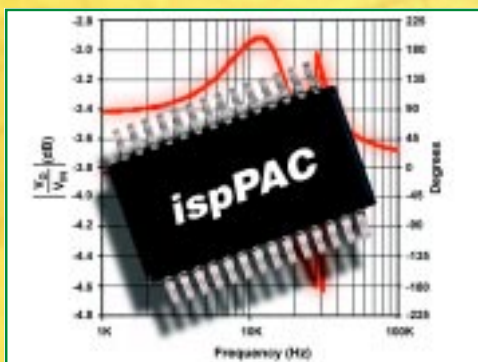
Obecnie wszyscy liczący się na świecie producenci oferują w wersji niskonapięciowej układy małej, średniej i dużej skali integracji. Prekursorem tego trendu jest firma Lattice, która jako pierwsza wprowadziła na rynek układy GAL22V10XLV przystosowane do zasilania napięciem 2,5V. Obserwuje się próby wykreowania nowego standardu zasilania napięciem 1,8V.

Wielostandardowe porty wejścia-wyjścia

Bardzo ważną cechą współczesnych struktur programowalnych jest zdolność do bardzo szybkiej wymiany z otoczeniem sygnałów cyfrowych. Wiąże się to z koniecznością wbudowywania w układy programowalne analogowych transceiverów różnicowych LVDS (ang. Low Voltage Differential Signalling). Przykładowo można tu wymienić rozwiązania oferowane przez firmę National Semiconductors w strukturach AMPA7K (rys. 3). Moduły nadawczo-odbiorcze wykonano w technice analogowej, co umożliwia przesyłanie sygnałów o amplitudzie zaledwie 800mV i częstotliwości do 750MHz (a nawet 1GHz w MPA10x). Obniża się w ten sposób

pobór mocy i znacznie redukuje poziom emitowanych do otoczenia zakłóceń EM. Niektórzy producenci (np. Texas Instruments oraz QuickLogic) promują jeszcze bardziej zaawansowany standard LVDS-XL, w którym transmisja sygnałów cyfrowych pomiędzy układami odbywa się różnicowym łączem analogowym z amplitudą sygnału zaledwie 200mV.

W niektórych zaawansowanych strukturach FPGA (APEX firmy Altera, DY8000 firmy Dyna Semiconductors i Spartan II firmy Xilinx) oraz CPLD (CoolRunner firmy Xilinx) wykorzystuje się różnicowe przesyłanie danych także wewnątrz struktury logicznej. W przypadku układów CoolRunner zastosowanie różnicowego systemu odczytu matrycy EEPROM i przesyłania danych wewnątrz niektórych fragmentów układu pozwoliło radykalnie obniżyć poziom mocy pobieranej statycznie (blisko 100 razy) w stosunku do standardowych odpowiedników wykonanych w tradycyjnej technologii.



Większość wprowadzanych na rynek w ostatnich miesiącach układów FPGA jest wyposażona w rozbudowane transceivery I/O, obsługujące różne standardy przyjęte w specjalizowanych układach cyfrowych. Takie rozwiązania zastosowano na przykład w rodzinie układów FPGA Spartan II firmy Xilinx. Porty I/O tych układów są przystosowane do współpracy z układami cyfrowymi zgodnymi ze standardami: LVCMOS, LVTTTL (napięciowo i prądowo), PCI, GTL, GTL+, HSTL c. I/II/IV, SSTL3 c. I/II, CTT oraz AGP.

Wielostandardowe porty wejścia-wyjścia

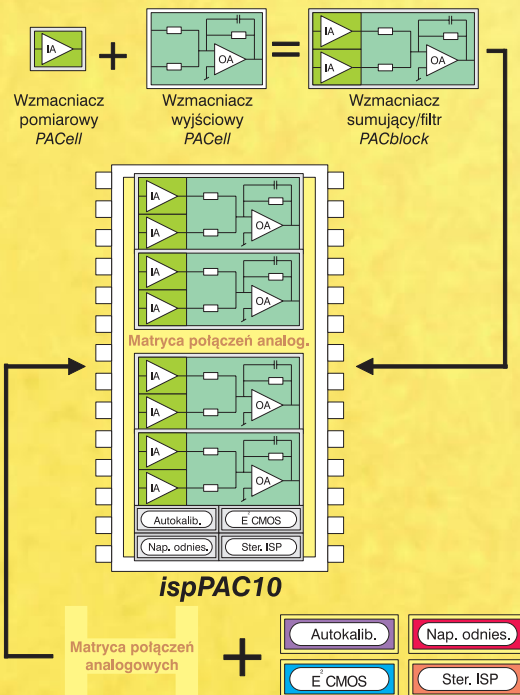
Analogowe układy programowalne

Podjęwane od początku lat 90. przez producentów układów programowalnych próby opracowania analogowych struktur programowalnych zaowocowały powstaniem rodziny układów ispPAC (opracowanie firmy Lattice).

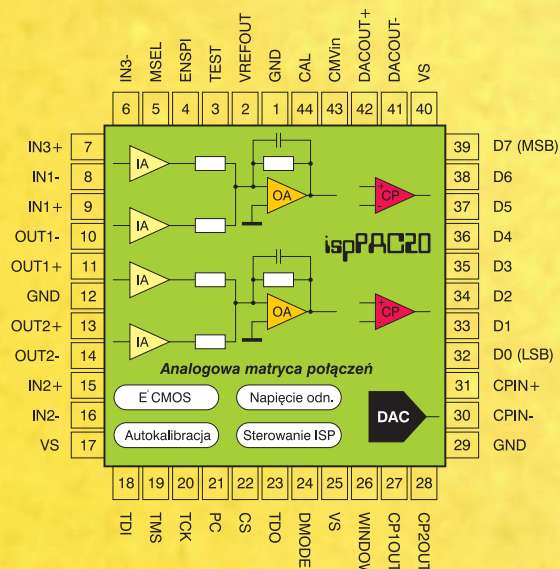
Obecnie są dostępne takie układy w trzech wersjach: ispPAC10, ispPAC20 i ispPAC80. Różnią się one znacznie wewnętrzną budową, w związku z czym zakres ich stosowania jest dość szeroki.

Architektura układów ispPAC10 (rys. 4) jest - oczywiście w pewnym przybliżeniu - analogowym odpowiednikiem architektury PLD znanej np. z GAL16V8 i podobnych struktur. Budowa analogowych makrocel oparta jest na dwóch blokach analogowych, których większość parametrów użytkownik może samodzielnie konfigurować. W ten sposób poprzez odpowiednie łączenie i konfigurowanie bloków analogowych możliwe jest tworzenie układów realizujących większość typowych funkcji analogowych, tzn. wielu rodzajów filtrów o łatwo modyfikowalnych charakterystykach amplitudowo-fazowych i wzmocnieniu, układów próbkująco-pamiętających o dużej szybkości działania, a także układów „dopasowujących” przetwarzany sygnał do wymagań przetwornika A/C itp.

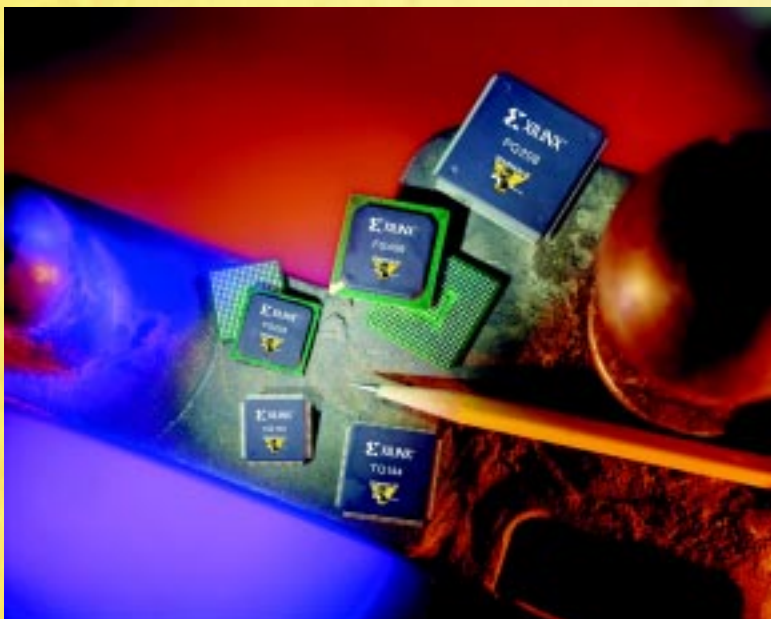
Strukturę o nieco innej koncepcji Lattice zastosował w układach ispPAC20 (rys. 5). Podstawowym elementami wewnętrznej struktury układu są dwa konfigurowalne bloki wzmacniaczy-filtrów identycznych z zastosowanymi w ispPAC10. Dodatkowo w układ wbudowano analogowe komparatory o programowanej histerezie oraz uniwersalny przetwornik C/A. Dzięki przemyślanej konstrukcji, sterowanie pracą przetwornika jest



Rys. 4. Budowa układu ispPAC10.



Rys. 5. Budowa układu ispPAC20.



zoptymalizowane pod kątem systemów mikroprocesorowych, do których te układy są przede wszystkim przeznaczone. Układ ispPAC80 jest z kolei programowalnym w systemie filtrem aktywnym, o projektowanej przez użytkownika charakterystyce amplitudowo-fazowej i odpowiedzi impulsowej.

Analogowe układy programowalne ispPAC wyposażono w interfejs ISP oraz nieulotną matrycę pamięciową, w której przechowywane są wszystkie nastawy konfiguracji układu. Bogate wyposażenie, łatwość konfiguracji oraz doskonałe parametry poszczególnych bloków układów ispPAC powodują, że są one alternatywne do standardowych rozwiązań analogowych torów obróbki sygnałów.

Struktury typu „combo“

Kolejnym zauważalnym trendem w rozwoju układów programowalnych jest integracja w jednej półprzewodnikowej strukturze matrycy programowalnej oraz mikrokontrolera wraz z peryferiami lub samych rozbudowanych peryferii.

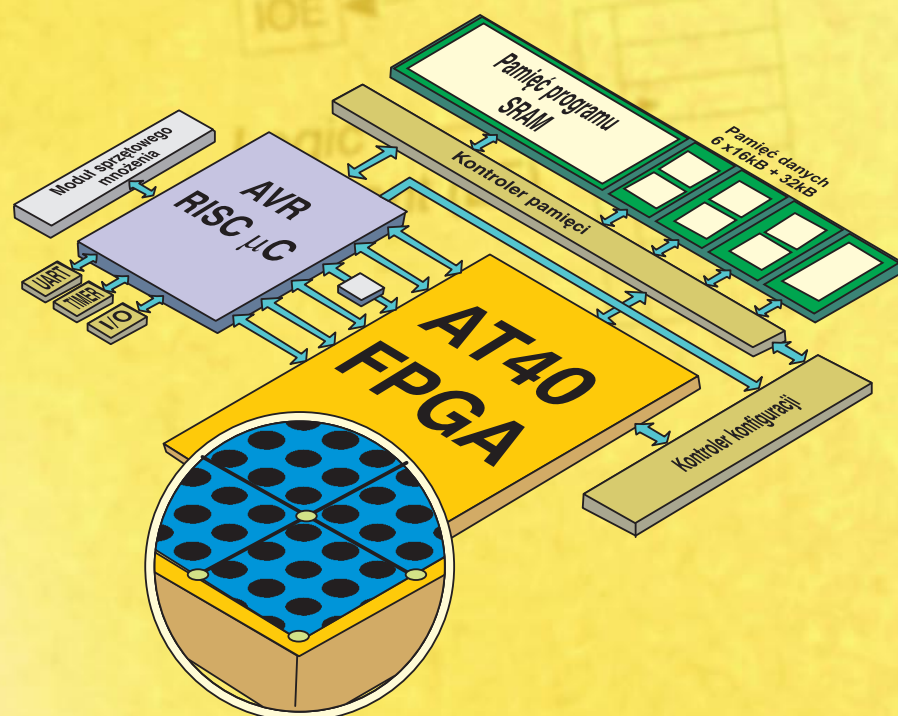
Amerykańska firma Waferscale jest producentem rozbudowanych układów peryferyjnych serii PSD8xx/9xx (opisane w EP7/2000), integrujących w jednej strukturze szereg portów I/O, pamięć danych RAM, pamięć programu Flash oraz niewielką matrycę programowalną, która służy do samodzielnego wykonania przez użytkownika dekodera adresowego, może także służyć do rekonfiguracji interfejsu, poprzez który mikrokontroler współpracuje z peryferiami. Matryca programowalna zintegrowana w układach PSD9xx jest programowana w systemie, a jej konfigurację można zmieniać, dopasowując do aktualnych wymagań systemu.

Inną filozofię przyjęli twórcy układów FPSLIC typu AT94K firmy Atmel (opisane w EP6/2000) - rys. 6. Opracowali oni bowiem układ integrujący szybki mikrokontroler RISC z rodziny AVR oraz jedną z bardziej uniwersalnych matryc FPGA - AT40. Mikrokontroler wyposażono w szereg modułów peryferyjnych (timery-liczniki, UART, I²C, programowane porty I/O) oraz pamięć danych i programu, których fragmenty mogą być wykorzystane do rekonfiguracji matrycy FPGA. Wymiana informacji pomiędzy FPGA i rdzeniem procesora jest ułatwiona dzięki 16 przerwaniom przypisanym do obsługi tej współpracy. Bloki funkcjonalne im-

plementowane w FPGA mogą być tak-towane 8 globalnymi sygnałami zegarowymi, spośród których dwa mogą być wytwarzane przez peryferia mikrokontrolera lub programowo.

Zwiększanie elastyczności

Nowe układy programowalne są budowane w oparciu o nieustannie udoskonalane elementy (makrocele, LUT, matryce łączeniowe), które zwiększają elastyczność architektury. Przykładem nowych trendów jest wprowadzanie trójstanowych magistral wewnątrz matryc łączeniowych, co proponuje m.in. Lattice w układach ispLSI8000 i ispLSI8000V.



Rys. 6. Schemat blokowy układów FPSLIC firmy Atmel.

Inną metodą zwiększenia elastyczności konfiguracji jest wprowadzenie pośrednich matryc połączeniowych, które pozwalają rozdzielić zagrzebane makrocele wejściowe od wyjściowych, a także od buforów wyjściowych. Dodatkową zaletą tego rozwiązania jest zmniejszenie rozmiarów matryc łączeniowych, co powoduje zmniejszenie pasożytniczych pojemności, które głównie ograniczają szybkość działania całej struktury. Rozwiązanie tego typu oferuje w układach ispLSI5000 firma Lattice (matryce PTSA) oraz Xilinx w XC9500.

Układy FPGA produkowane przez firmę Dyna Semiconductors wyznaczają z kolei trend polegający na specjalizowaniu makrocel do pewnych rodzajów aplikacji, przy czym względna uniwersalność architektury została zachowana. Firma ta oferuje układy, w których tablice LUT konfigurują makrocele w sposób optymalny dla cyfrowych filtrów FIR stosowanych w systemach DSP, dla systemów adresowania, gdzie pojedyncza makrocela może spełniać rolę 4-bitowego programowanego licznika, możliwa jest także optymalizacja pod ką-

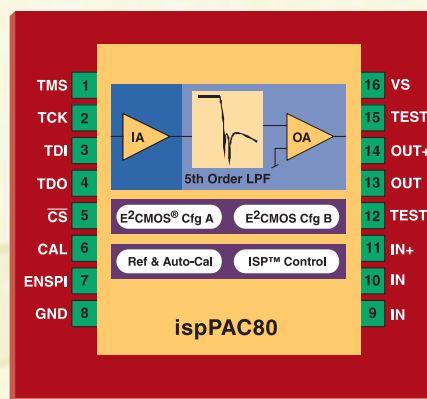
tem modułów liczących złożone sumy kontrolne (rejstry z programowanym przesunięciem i funkcją ExOR na wyjściu). Takie podejście do architektury układów obniża w pewnym stopniu ich elastyczność, ale ułatwia dobór układu pod kątem wymagań aplikacji.

Nowe języki opisu struktur programowalnych

W ciągu kilku ostatnich miesięcy silnie zaznacza się promowanie nowych (dla układów programowalnych) języków opisu. Szczególny nacisk jest kładziony na kompilatory języka ANSI C oraz C+ dla układów programowalnych, przy czym zauważalne są dwa główne kierunki rozwijania tego typu narzędzi:

- budowa kompilatorów C zintegrowanych z konwerterem C/VHDL, czego przykładem może być IAR-PLD Lab,
- tworzenie uniwersalnych konwerterów C/VHDL, czego przykładem może być ART Builder opracowany przez firmę Frontierd.

Niezależnie od sposobu realizacji konwertera, stosowana jest jedna metoda translacji kodu oparta na algorytmie



mie Cordic. Podejmowane są także próby wprowadzenia konwerterów PASCAL/VHDL, lecz obecnie nie są dostępne wersje testowe tego typu narzędzi.

Pomimo znacznych ograniczeń obecnie dostępnych kompilatorów ANSI-C i PASCALa dla układów PLD, ich stopniowy rozwój ma szansę zachęcić liczne grono konstruktorów do szerokiego stosowania układów programowalnych w nowych opracowaniach.

Piotr Zbysiński, AVT
piotr.zbysinski@ep.com.pl