

Procesory sygnałowe DSP

semestr zimowy 2012/2013, E-3, WIEiK, PK

1

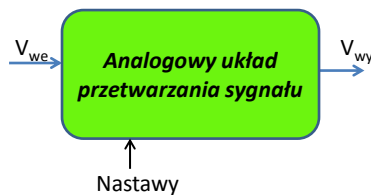
Przetwarzanie sygnałów

- Przetwarzanie sygnałów zajmuje się wykonywaniem pewnych operacji na sygnałach oraz interpretacją tychże sygnałów.
- Przetwarzania sygnałów jest dziedziną matematyki stosowanej, która dotyczy użytecznych operacji na sygnałach lub analizy tych sygnałów odbywającej się w **czasie ciągłym** lub na **dyskretnych próbkach** tych sygnałów.
- W zależności od zastosowań takimi użytecznymi operacjami mogą być działania kontrolne, **kompresja danych, transmisja danych, pozbywanie się szumów i sygnałów zakłócających, predykcja zachowania się sygnału, filtrowanie, wygładzanie, odtwarzanie zniekształconego sygnału z jego "zarysów", rekonstrukcja tomograficzna, identyfikacja, klasyfikacja i wiele innych tego typu operacji.**
- Przetwarzane sygnały mogą zawierać dźwięk, obrazy, wielkości mierzone zależne od czasu i dane pochodzące z czujników pomiarowych, na przykład dane z procesów biologicznych, takich jak elektrokardiogramy, sygnały z systemów kontroli, sygnały transmisji telekomunikacyjnych, np. sygnały radiowe i telewizyjne oraz wiele innych.

semestr zimowy 2012/2013, E-3, WIEiK, PK

2

Tor analogowego przetwarzania sygnałów



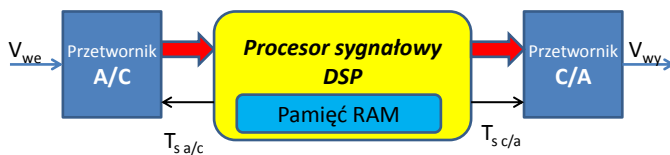
Przykłady analogowej obróbki sygnału

- Filtrowanie sygnału (filtr dolnoprzepustowy, górnoprzepustowy, pasmowo-zaporowy, pasmowo-przepustowy)
- Całkowanie, różniczkowanie sygnału
- Wyznaczanie wartości skutecznej sygnału, wartości średniej, minimalnej, maksymalnej
- Wyznaczanie widma sygnału za pomocą filtrów
- Odszumianie sygnału
- Kompresja sygnału, dekompresja sygnału
- Modulacja sygnału, demodulacja sygnału

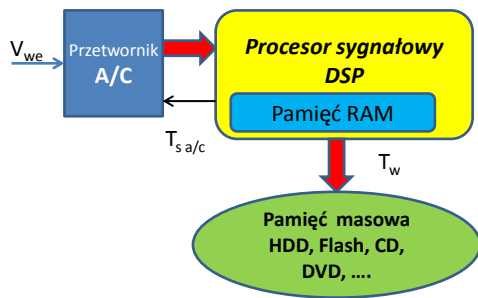
semestr zimowy 2012/2013, E-3, WIEiK, PK

3

Tor cyfrowego przetwarzania sygnałów



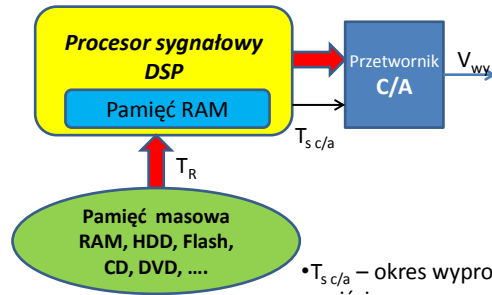
- $T_{s\ a/c}$ – okres próbkowania sygnału wejściowego
- $T_{s\ a/c}$ – okres wyprowadzania sygnału wyjściowego
- T_w – okres zapisu sygnału wejściowego



semestr zimowy 2012/2013, E-3, WIEiK, PK

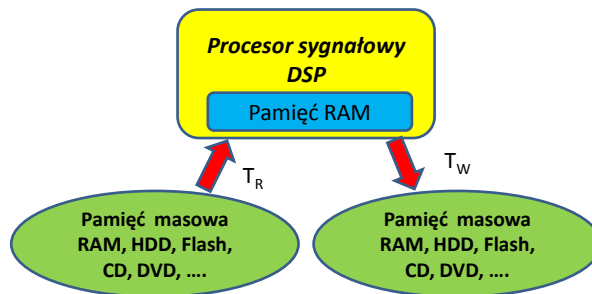
4

Tor przetwarzania DSP

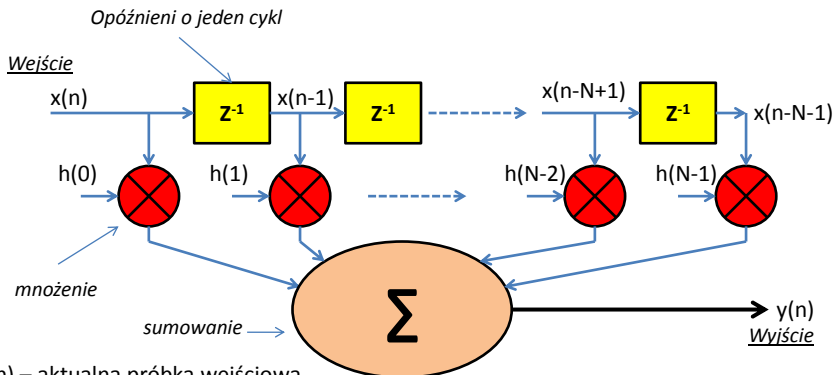


- $T_{s\ c/a}$ – okres wyprowadzania sygnału wyjściowego
- T_R – okres odczytu danych dla sygnału wyjściowego

Tor przetwarzania DSP



Podstawowe algorytmy DSP – filtr o Skończonej Odpowiedzi Impulsowej SOI (FIR – Finite Impulse Respose)



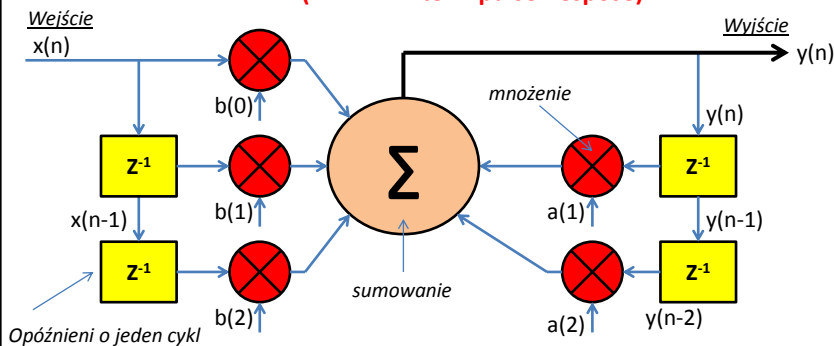
$x(n)$ – aktualna próbka wejściowa
 $y(n)$ – aktualna próbka wyjściowa
 $h(n)$ – współczynniki filtru
 N - rząd filtru

$$y(n) = h(n) \otimes x(n) = \sum_{k=0}^{N-1} h(k) \cdot x(n-k)$$

Operacja splotu

Np. dla $N=5$ $y(n) = x(n)h(0) + x(n-1)h(1) + x(n-2)h(2) + x(n-3)h(3) + x(n-4)h(4)$

Podstawowe algorytmy DSP – filtr o Nieskończonej Odpowiedzi Impulsowej NOI (IIR – Infinite Impulse Respose)



$x(n)$ – aktualna próbka wejściowa
 $y(n)$ – aktualna próbka wyjściowa
 $a(n), b(n)$ – współczynniki filtru
 N - rząd filtru

$$y(n) = \sum_{k=0}^M b(k) \cdot x(n-k) - \sum_{k=1}^N a(k) \cdot y(n-k)$$

Np. dla $N=3$ $y(n) = x(n)b(0) + x(n-1)b(1) + x(n-2)b(2) - y(n-1)a(1) - y(n-2)a(2)$

Podstawowe algorytmy DSP –

DFT - Dyskretne przekształcenie Fouriera (Discrete Fourier Transform - DFT)
FFT – Szybkie przekształcenie Fouriera

$$X(f) = \int_{-\infty}^{\infty} x(t) \cdot e^{-2\pi f t} dt$$

Ciągłe przekształcenie Fouriera

$$x(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega) \cdot e^{j(\omega T_s n)} d(\omega T_s)$$

$$X(m) = \sum_{n=0}^{N-1} x(n) \cdot e^{-j \frac{2\pi m n}{N}}$$

Współrzędne biegunowe

$$X(\omega) = \sum_{n=-\infty}^{\infty} x(n) \cdot e^{j(\omega T_s n)}$$

Dyskretne przekształcenie Fouriera

$$X(m) = \sum_{n=0}^{N-1} x(n) \cdot \left[\cos\left(\frac{\pi m n}{N}\right) - j \sin\left(\frac{\pi m n}{N}\right) \right]$$

Współrzędne prostokątne

$X(m)$ - m -ta składowa wyjściowa DFT, tj. $X(0)$, $X(1)$, $X(2)$, $X(3)$ itd.,

m - indeks próbek wyjściowych DFT w dziedzinie częstotliwości, $m=0,1,2,3,\dots,N-1$

$X(n)$ - ciąg próbek wejściowych, $x(0)$, $x(1)$, $x(2)$, $x(3)$ itd.,

n - indeks próbek wejściowych w dziedzinie czasu, $n = 0,1,2,3,\dots,N-1$,

$j = \sqrt{-1}$ oraz

N - liczba próbek ciągu wejściowego = liczba próbek ciągu wyjściowego oraz równa liczbie punktów częstotliwości w ciągu wyjściowym DFT.

semestr zimowy 2012/2013, E-3, WIEiK, PK

9

Podstawowe algorytmy DSP –

FFT – Szybkie przekształcenie Fouriera

$$X(m) = \sum_{n=0}^{N-1} x(n) \cdot \left[\cos\left(\frac{\pi m n}{N}\right) - j \sin\left(\frac{\pi m n}{N}\right) \right]$$

- **Szybka transformacja Fouriera** (ang. **FFT - Fast Fourier Transform**) to [algorytm liczenia dyskretnej transformaty Fouriera DFT](#) oraz transformaty do niej odwrotnej.
- Obliczanie tych sum za pomocą powyższego wzoru zajęłoby $O(N^2)$ operacji.
- Algorytmy (jak algorytm [Cooleya-Tukeya](#)) obliczające szybką transformację Fouriera bazują na metodzie dziel i zwyciężaj, rekurencyjnie dzieląc transformatę wielkości $N = N_1 N_2$ na transformaty wielkości N_1 i N_2 z wykorzystaniem $O(N)$ operacji mnożenia.
- Najpopularniejszą wersją FFT jest *FFT o podstawie 2*. Jest to bardzo efektywna operacja, jednak wektor próbek wejściowych musi mieć długość $N = 2^k$, (np. 4, 8 16, 32, 64, 128, 256, 512, 1024, ...) gdzie k to liczba naturalna.
- Wynik otrzymuje się na drodze schematycznych przekształceń, opartych o tak zwane *struktury motylkowe*.

semestr zimowy 2012/2013, E-3, WIEiK, PK

10

Podstawowe algorytmy DSP – Dyskretna Transformata Kosinusowa

- **DCT** (ang. **d**iscrete **c**osine **t**ransform, czyli dyskretna transformacja cosinusowa) - jedna z najpopularniejszych blokowych transformacji danych. Jest szczególnie popularna w stratnej kompresji danych.
- DCT przekształca skończony ciąg N liczb rzeczywistych $g(0), \dots, g(N-1)$ w ciąg liczb rzeczywistych $G(0), \dots, G(N-1)$ zgodnie z zależnościami:

$$G(0) = \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} g(m) \quad \text{Dla } k=1, 2, 3, \dots, N-1$$

$$G(k) = \sqrt{\frac{2}{N}} \cdot \sum_{m=0}^{N-1} g(m) \cdot \cos \frac{\pi k(2m-1)}{2N}$$

Processor DSP

- Procesory Sygnałowe (**DSP- Digital Signal Processor**) są to układy mikroprocesorowe przewidziane do szybkich obliczeń matematycznych w czasie rzeczywistym.
- W większości przypadków obliczenia wykonuje się na dyskretnych próbkach reprezentujących jakieś sygnały (mowy, dźwięku, obrazu, itp.).
- Obliczenia mogą być wykonywane na liczbach rzeczywistych jak i zespolonych.
- Architektura procesorów sygnałowych jest tak zbudowana, aby zminimalizować czasy wykonywania operacji mnożenia i sumowania, które występują w prawie we wszystkich algorytmach do cyfrowego przetwarzania sygnałów, np. w filtrach FIR i IIR, DFT lub FFT. Skrócone są też czasy pobierania i zapisywania danych do pamięci.
- Pierwsze procesory DSP powstały dla potrzeb techniki wojskowej, dziedziny, która wniosła największy wkład w analizę i syntezę dźwięku, czyli hydrolokacji.

Procesor DSP

- Zwłaszcza synteza dźwięku stanowiła pole do popisu dla wczesnych konstrukcji DSP - układ, który potrafił wygenerować dźwięki odpowiadające dokładnie charakterystyce akustycznej atomowego okrętu podwodnego (i to konkretnej jednostki!) stanowił niebywałą atrakcją dla wszystkich flot, dzięki możliwości produkcji tanich pozoratorów, stanowiących istotny element systemu obronnego tych okrętów.
- A po zaspokojeniu potrzeb flot i uzbrojeniu odpowiedniej ilości pozoratorów i samonaprowadzających torped przyszła kolej na zastosowanie DSP w technice cywilnej, oczywiście przede wszystkim do syntezy i obróbki dźwięku.
- Był to akurat czas, kiedy komputery zaczęły mieć głos, dzięki czemu zapotrzebowanie na procesory dźwięku przybrało skalę pozwalającą na znaczny spadek cen, zwłaszcza, że zakres zastosowań DSP stał się bardzo szeroki.

Procesor DSP

- Obecnie procesory DSP są prawie "wszechobecne". Znajdujemy je w kartach dźwiękowych, w których umożliwiają uzyskiwanie dźwięku 3D i służą jako synteзаторы MIDI, w wyższej klasy sprzęcie audio i "głośno mówiących" aparatach telefonicznych, gdzie służą przede wszystkim do eliminacji wpływu akustyki pomieszczenia na jakość dźwięku.
- DSP zastosowane w modemach umożliwiły uzyskanie szybkości transmisji znacznie wyższych od tych, jakie wynikałyby z pasma częstotliwości przenoszonego przez łącza telefoniczne.
- W telefonach komórkowych, a także w dyktafonach cyfrowych, procesory sygnałowe służą do kompresji dźwięku tak, aby zachować jego jakość przy możliwie małym natężeniu strumienia danych do przesłania lub zapisu.
- Również technologia dysków twardych nie jest w stanie obejść się bez procesorów sygnałowych - wysoką gęstość zapisu, uzyskiwaną w technologii PRML, zawdzięczamy właśnie zastosowaniu procesora sygnałowego w układzie odczytu danych.

Procesor DSP

- Rozwój technologiczny, który pozwolił na znaczne zwiększenie szybkości przetwarzania procesorów, umożliwił ich zastosowanie również w technice wideo - dzięki odpowiedniej mocy DSP mogły powstać np. cyfrowe kamery wideo.
- W tym ostatnim zastosowaniu potrzebne są już gigantyczne moce - np. realizowana już od dawna w trybie on-line przez procesory DSP kompresja sygnału wideo do formatu MPEG-2 dla PC jest możliwa dopiero w przypadku zastosowania 500-megahercowego Pentium III, a i to tylko dzięki temu, że jednostka SIMD tego procesora zdolna jest do równoległej obróbki czterech argumentów.

Procesor DSP

- Typowy procesor sygnałowy, przeznaczony np. do cyfrowej obróbki dźwięku, ma stosunkowo ograniczony zakres zadań: wykonywanie, zgodnie z programem, operacji arytmetycznych na kolejnych próbkach sygnału.
- Taki zakres zadań kształtuje odpowiednią architekturę - większość współczesnych DSP to procesory RISC, wyspecjalizowane do stałoprzecinkowych operacji arytmetycznych na ciągach danych.
- Specyfika zastosowań DSP pozwala na zastosowanie w ich konstrukcji architektury Harvard, (lub zmodyfikowanej architektury typu von Neumanna) z rozdzielonymi przestrzeniami (pamięciami) danych i instrukcji. Architektura ta pozwala na znaczne uproszczenie konstrukcji procesora, a równocześnie zapewnia mu dużą wydajność.

Zastosowanie procesorów sygnałowych

- obróbka dźwięku:
 - korektory
 - efekty specjalne (echo, pogłos, dodawanie głębi),
 - filtracja
 - usuwanie echa (ang. *echo cancellation*) (telefony komórkowe, łączność cyfrowa)
- przetwarzanie dźwięku:
 - zniekształcanie
 - kodowanie (CD, MP3, telefon i radio cyfrowe)
 - rozpoznawanie mowy
 - synteza mowy
 - systemy aktywnego wyciszania (słuchawki obsługi technicznej lotnisk, dźwięk transformatora energetycznego, hałas wentylatora w komputerze PC,
 - echolokacja i lokalizacja bierna: sonary ultradźwiękowe, wykrywanie, lokalizacja i identyfikacja obiektów (przykład „szyk mikrofonów”)

Zastosowanie procesorów sygnałowych

- obróbka obrazu:
 - regulacja parametrów (barwa, nasycenie, kontrast)
 - „obraz w obrazie”
 - korektory
 - przechwytywanie i zatrzymywanie
- przetwarzanie obrazu:
 - kodowanie/kompresja (JPG, DIVX)
 - rozpoznawanie obrazów (medycyna, „oczy” robotów)
 - synteza obrazu (w prostszych systemach graficznych)
 - zmiana rozdzielczości, interpolacja

Zastosowanie procesorów sygnałowych

-sterowanie maszyn elektrycznych:

- wieloosiowe frezarki i tokarki numeryczne
- nowoczesny napęd pojazdów elektrycznych
- sterowanie silnikami elektrycznymi (falowniki)
- roboty przemysłowe

-inne:

- układy sztucznej inteligencji
- autopilot
- układy sterowania ruchu robotów
- aparatura medyczna
- kontrola poprawności działania procesów przemysłowych
- itp.

Zastosowanie procesorów sygnałowych

- urządzenia radiokomunikacji lądowej dla przemysłowego, komercyjnego i wojskowego użytku, włączając w to aparaturę systemów trunkingowych, komórkowych i przywoławczych,
- telekomunikacyjne systemy satelitarne,
- systemy światłowodowe i aparatura łączności optycznej (uwzględniając podczerwień),
- systemy i aparatura telemetryczna,
- aparatura łączności faksymile,
- osprzęt nowoczesnych central telefonicznych,
- urządzenia radiolokacyjne i radionawigacyjne (włączając aparaturę stacji poszukiwania, rozpoznawania, wykrywania, naprowadzania i automatycznego śledzenia),
- systemy oraz aparatura optyczno-elektroniczna dla wyszukiwania i obserwacji, wraz z aparaturą kontroli, testowania oraz inny osprzęt wspomagający,
- aparatura hydrolokacyjna i hydroakustyczna, systemy poszukiwania, rozpoznania i Podążania,
- systemy „wizji maszynowej” i dedykowanego monitoringu,
- radioelektroniczna aparatura meteorologiczna i radioastronomiczna,
- aparatura do pomiarów geofizycznych,

Zastosowanie procesorów sygnałowych

- sprzęt przeciwdziałania radiowego (włączając stacje generowania zakłóceń radiowych i aparaturę podwodną),
- wyspecjalizowane urządzenia dla „wywiadu radiotechnicznego”,
- systemy nawigacyjne i aparatura samolotów, okrętów, pojazdów lądowych (autopiloty,
- latarnie radiowe, pelengatory, aparatura systemów nawigacji krótko dystansowej);
- aparatura cyfrowej telewizji i audycji radiowej,
- systemy astroorientacji rakiet oraz statków kosmicznych;
- aparatura rejestracji i odtwarzania materiałów dźwiękowych,
- aparatura kontroli bezinwazyjnej metali i materiałów (defektoskopy, przyrządy do pomiarów emisji akustycznej),
- karty i układy multimedialnego wspomaganie komputerów PC (procesory geometryczne, akceleratory, karty graficzne, wideoprocessory itd.)

Procesor DSP

- Jądro typowego DSP zawiera jedną lub kilka stałoprzecinkowych jednostek arytmetycznych, zespół rejestrów pełniących rolę akumulatorów w operacjach arytmetyczno-logicznych oraz oczywiście dekodery instrukcji i logikę sterowania i adresów.
- Jak widać, podstawowa architektura jest prosta. Jednak dla bardziej zaawansowanych zadań, jak np. obróbka sygnału wideo, oferowane przez podstawową architekturę wydajności (50-100 MIPS) są zdecydowanie za małe. Droga zwiększenia wydajności jest tu przede wszystkim przetwarzanie potokowe oraz zastosowanie rozwiązań super- i hiperskalarnych, czyli równoległych jednostek arytmetycznych lub równoległych kanałów wykonawczych z wieloma jednostkami arytmetycznymi każdy.
- Charakter obrabianych danych określa długość słowa, z jakim pracuje procesor. Popularne DSP do obróbki dźwięku operują na danych 16-bitowych, zapewniających wystarczającą jakość w typowych zastosowaniach. Ale stosuje się również konstrukcje o dłuższym, 32-bitowym słowie, pozwalające na większą precyzję przetwarzania. Również 64-bitowe procesory sygnałowe nie są rzadkością (np. w karcie Sound Blaster AWE 64).

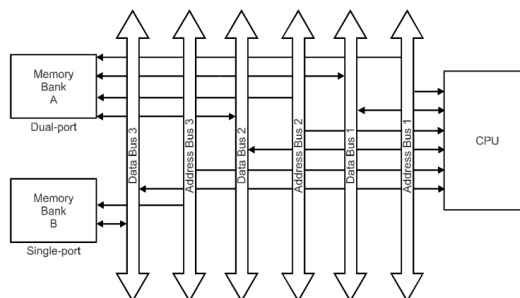
Elementy i właściwości procesora DSP

- sprawna jednostka przetwarzająca (CPU),
- wysokowydajna jednostka arytmetyczno-logiczna (ALU), sprzętowy układ mnożący, może być kilka ALU,
- dodatkowa specjalna jednostka arytmetyczna zwana MAC (), mnożenie i dodawanie w jednym cyklu rozkazowym,
- szerokość szyny danych 16, 20, 24, 32 bity,
- możliwość wykonywania obliczeń na liczbach stałoprzecinkowych (-1 do +1) i zmiennoprzecinkowych (mantissa $\times 2^{\text{exponent}}$), mantissa -1 do +1 exponent liczba całkowita,
- mechanizmy nasycania wyniku, zaokrąglania,
- mechanizmy do szybkiego indeksowania wektorów (tablic) i macierzy,
- szerokość akumulatora i jednostki ALU i jednostki typu MAC większa niż szerokość szyny danych w celu uniknięcia błędu przepełnienia lub przekroczenia zakresu,
- wielokrotny dostęp do pamięci programu i danych, tak aby w trakcie wykonywania jednej instrukcji można było pobrać kolejną instrukcję, daną lub kilka danych,
- kilka szyn danych i szyn adresowych umożliwiających wielodostęp do pamięci,
- segmentacja pamięci danych na banki ,

semestr zimowy 2012/2013, E-3, WIEiK, PK

23

Budowa procesorów DSP



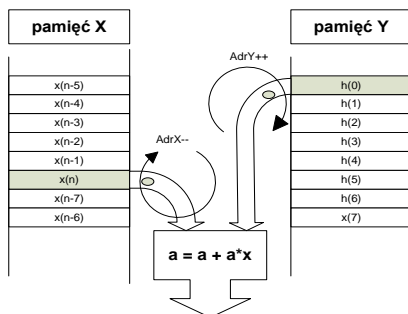
Procesor DSP z podwójną magistralą danych w architekturze Harvard

semestr zimowy 2012/2013, E-3, WIEiK, PK

24

Rejestr kołowy

- W wielu algorytmach DSP wartość próbki wyjściowej jest uzależniona od próbki aktualnej oraz wartości próbek poprzednich sygnału wejściowego i wyjściowego oraz współczynników, a więc muszą one być gdzieś pamiętane.
- Dlatego cechą charakterystyczną procesorów sygnałowych jest duża ilość rejestrów do pamiętania, które wspomagają realizację tzw. buforów kołowych.
- W buforze kołowym każda przychodząca próbka nadpisuje najstarszą próbkę, w ten sposób nie zajmuje się niepotrzebnie czasu procesora na przesuwanie próbek w pamięci tak jak ma to miejsce w buforze przesuwalnym, w którym każda nowa próbka jest wpisywana po przesunięciu próbek poprzednich o jedną pozycję



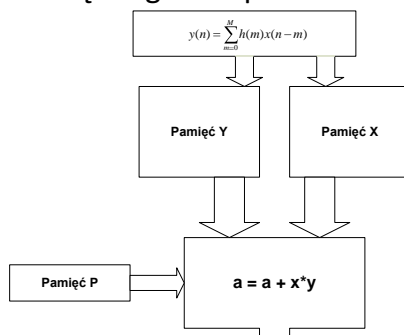
Do wykonywania tych operacji w procesorach sygnałowych zaimplementowano układy adresujące pamięć tzw. DAG (ang. Data Address Generators). Adresowanie to, jak i możliwość mnożenia z akumulacją w jednym cyklu zegara, stanowią główną zaletę jaka decyduje o wydajności procesorów DSP w przypadku przetwarzania sygnałów.

semestr zimowy 2012/2013, E-3, WIEIK, PK

25

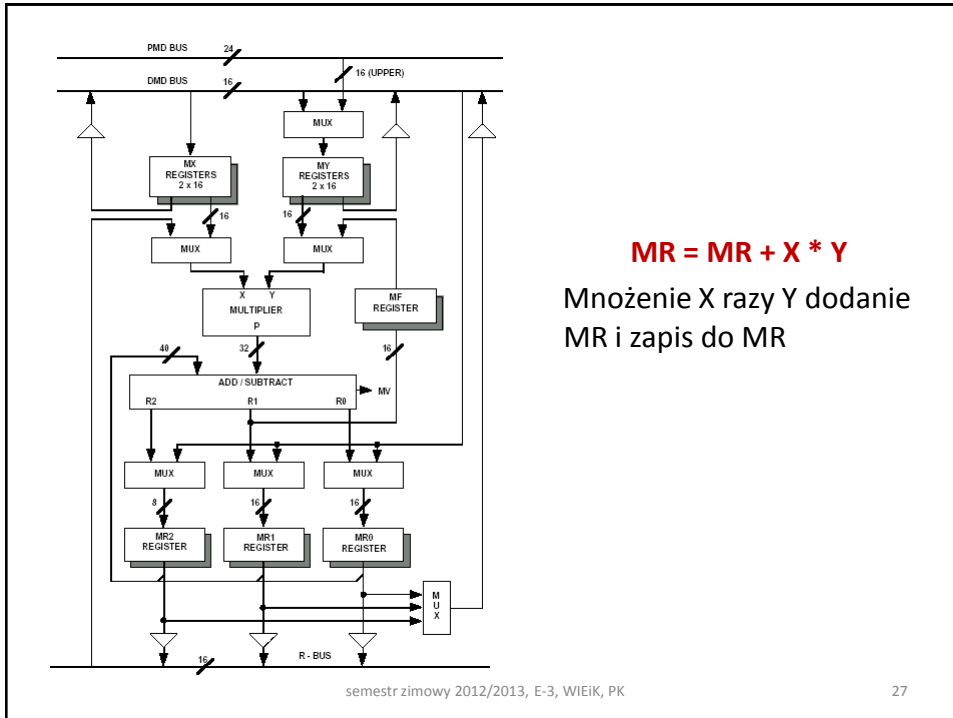
Jednostka MAC

- Możliwość równoległego mnożenia z akumulacją w jednym cyklu realizowana jest przez tzw. jednostkę MAC (ang. *Multiple and Accumulate*).
- Wykonuje ona kilka operacji: pobiera dane i współczynniki, mnoży je, dodaje do siebie, zapamiętuje oraz modyfikuje adres, wszystko to odbywa się automatycznie, odciążając programistę od zbędnego komplikowania kodu.



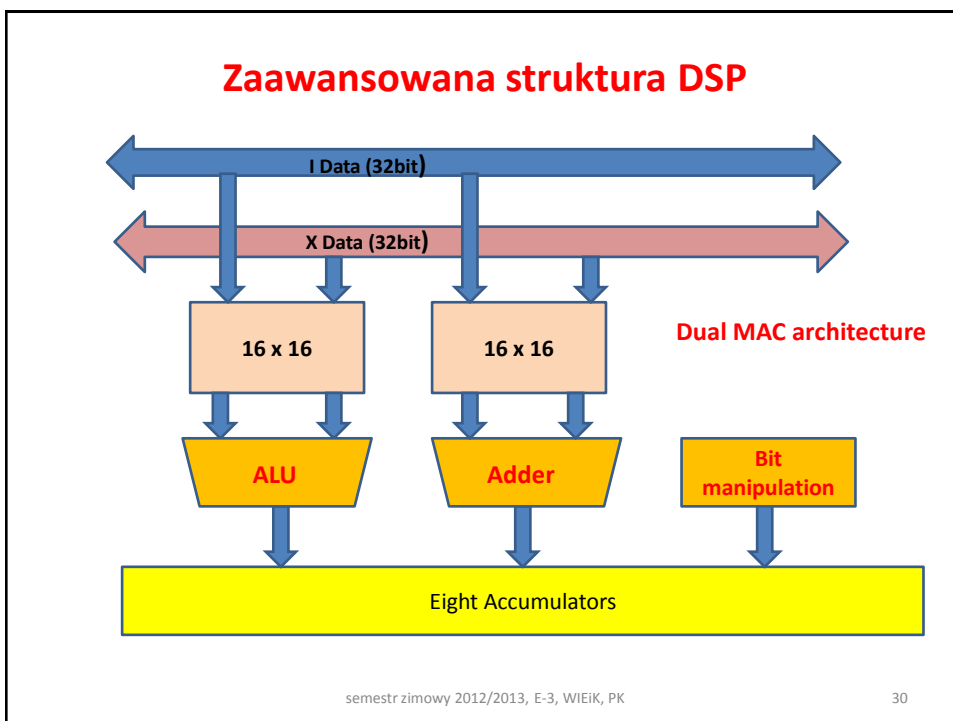
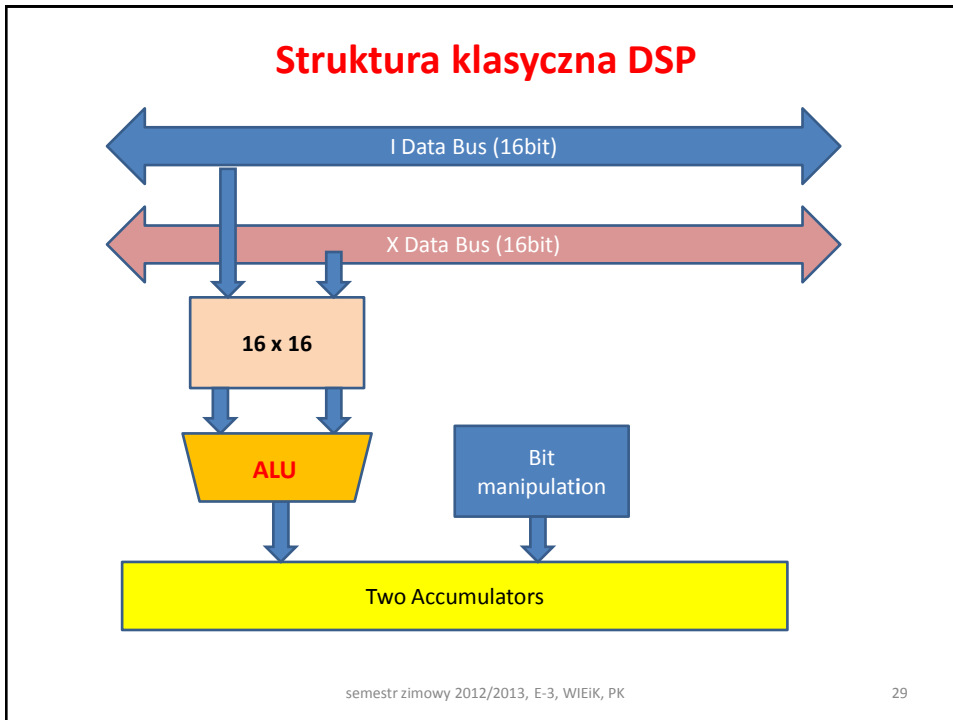
semestr zimowy 2012/2013, E-3, WIEIK, PK

26



Elementy i właściwości procesora DSP

- procesory DSP mają możliwość wykonywania instrukcji typu **SIMD** - pojedyncza instrukcja wiele danych wejściowych (single instruction, multiple data),
- specjalne moduły adresowania przyspieszające obliczenia FFT,
- jeden lub kilka układów do adresowania pamięci programu lub danych,
- kontroler typu DMA (bezpośredni dostęp pamięci)
- szybkie interfejsy szeregowy,
- szybkie porty wejścia/wyjścia szeregowy lub równoległe do komunikacji z A/C i C/A,
- możliwość wykonywania kilku instrukcji równoległe,
- jedna instrukcja wykonywana w ciągu jednego cyklu zegarowego,
- specjalny zestaw instrukcji arytmetycznych lub logicznych operujący na wektorach danych,
- szybkie instrukcje skoków i pętli,



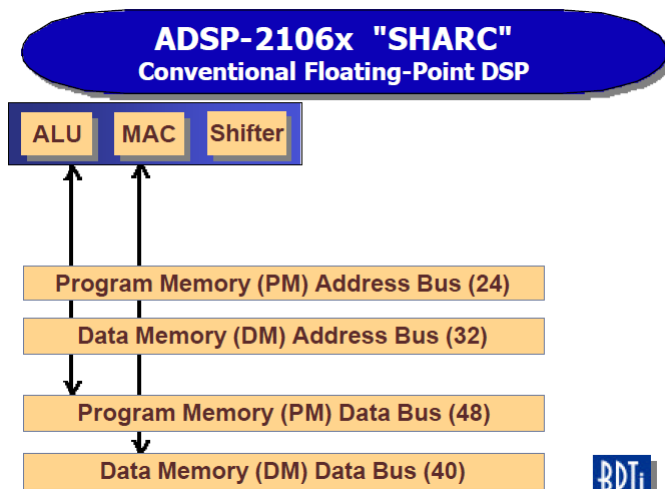
SIMD

- **SIMD** (ang. *Single Instruction, Multiple Data*) – jeden z podstawowych rodzajów architektur komputerowych według taksonomii Flynna, obejmujący systemy, w których przetwarzanych jest wiele strumieni danych w oparciu o pojedynczy strumień rozkazów. Architektura SIMD jest charakterystyczna dla komputerów wektorowych.
- Pierwsze komputery o architekturze SIMD stosowano głównie do obliczeń naukowo-technicznych (np. Cray X-MP czy Thinking Machines CM-1 i CM-2). Obecnie jednostki realizujące zadania zgodnie z metodologią SIMD obecne są także w stosowanych w domowych komputerach procesorach opartych o architekturę x86. Procesory te oferują listę rozkazów poszerzoną o zestawy rozkazów typu SIMD, takie jak: MMX, 3DNow!, SSE, SSE2, SSE3, SSSE3, SSE4, SSE5, AVX, AltiVec.

SIMD

- Aby można było wykorzystać opcję SIMD musi być możliwość równoległego przetwarzania danych.
- SIMD nadaje się do obróbki dużej liczby danych

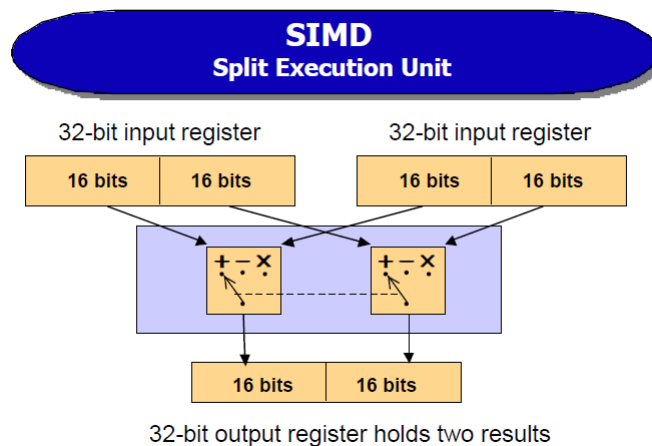
Pojedyncze jednostki wykonawcze



semestr zimowy 2012/2013, E-3, WIEIK, PK

33

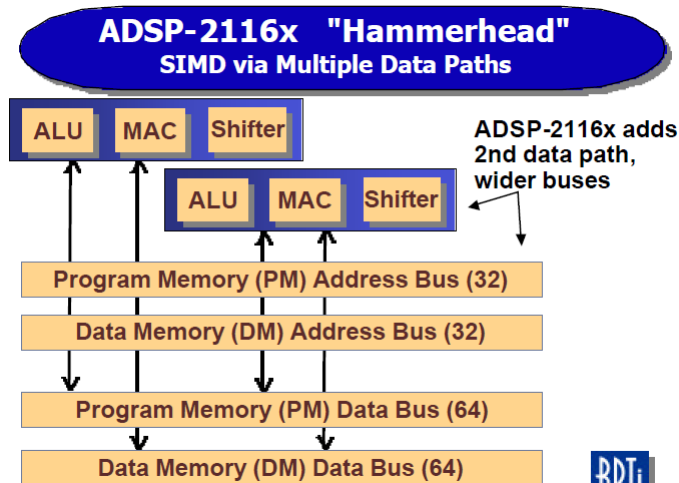
SIMD



semestr zimowy 2012/2013, E-3, WIEIK, PK

34

Zwielokrotnienie jednostek wykonawczych



35

Potokowość - pipelining

- Podstawowym mankamentem techniki potoku są [rozkazy skoku](#), powodujące w najgorszym wypadku potrzebę przeczyszczenia całego potoku i wycofania rozkazów, które następowały zaraz po instrukcji skoku i rozpoczęcie zapełniania potoku od początku od adresu, do którego następował skok.
- Taki rozkaz skoku może powodować ogromne opóźnienia w wykonywaniu programu – tym większe, im większa jest długość potoku.
- Dodatkowo szacuje się, że dla [modelu programowego x86](#) taki skok występuje co kilkanaście rozkazów. Z tego powodu niektóre architektury programowe (np. [SPARC](#)) zakładały zawsze wykonanie jednego lub większej ilości rozkazów następujących po rozkazie skoku, tzw. skok opóźniony. Stosuje się także skomplikowane metody predykcji skoku lub metody [programowania](#) bez użycia skoków.

semestr zimowy 2012/2013, E-3, WIEIK, PK

36

Potokowość

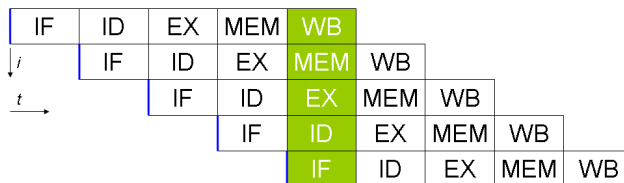
- **Potokowość** ([ang. pipelining](#)) – technika budowy [procesorów](#) polegająca na podziale logiki procesora odpowiedzialnej za [proces](#) wykonywania programu ([instrukcji](#) procesora) na specjalizowane grupy w taki sposób, aby każda z grup wykonywała część pracy związanej z wykonaniem rozkazu.
- Grupy te są połączone sekwencyjnie – potok ([ang. pipe](#)) – i wykonują pracę równocześnie, pobierając dane od poprzedniego elementu w [sekwencji](#).
- W każdej z tych grup rozkaz jest na innym stadium wykonania. Można to porównać do taśmy produkcyjnej. W uproszczeniu, potok wykonania instrukcji procesora może wyglądać następująco:

semestr zimowy 2012/2013, E-3, WIEiK, PK

37

Potokowość

- Pobranie instrukcji z pamięci – [ang. instruction fetch \(IF\)](#)
- Zdekodowanie instrukcji – [ang. instruction decode \(ID\)](#)
- Wykonanie instrukcji – [ang. execute \(EX\)](#)
- Dostęp do pamięci – [ang. memory access \(MEM\)](#)
- Zapisanie wyników działania instrukcji – [ang. store; write back \(WB\)](#)



W powyższym 5-stopniowym potoku, przejście przez wszystkie stopnie potoku (wykonanie jednej instrukcji) zabiera co najmniej 5 cykli zegarowych. Jednak ze względu na jednoczesną pracę wszystkich stopni potoku, jednocześnie wykonywanych jest 5 rozkazów procesora, każdy w innym stadium wykonania.

Oznacza to, że taki procesor w każdym cyklu zegara rozpoczyna i kończy wykonanie jednej instrukcji i statystycznie wykonuje rozkaz w jednym cyklu zegara. Każdy ze stopni potoku wykonuje mniej pracy w porównaniu do pojedynczej logiki, dzięki czemu może wykonać ją szybciej – z większą [częstotliwością](#) – tak więc dodatkowe zwiększenie liczby stopni umożliwia osiągnięcie coraz wyższych częstotliwości pracy.

semestr zimowy 2012/2013, E-3, WIEiK, PK

38

VLIW

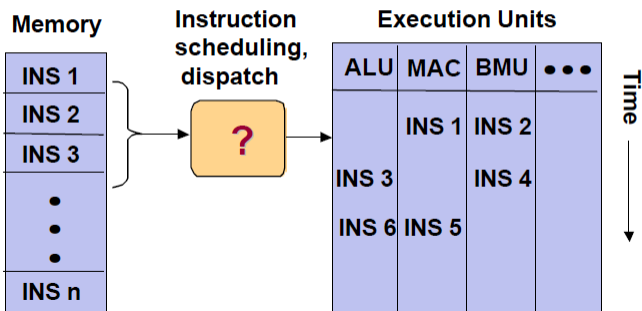
- **VLIW** ([ang. Very Long Instruction Word](#)) – nazwa architektury mikroprocesorów z bardzo długim słowem instrukcji.
- Obecnie procesory VLIW są oparte na architekturze [RISC](#), zazwyczaj z czterema lub maksymalnie ośmioma jednostkami obliczeniowymi. Po normalnej kompilacji programu, kompilator VLIW porządkuje kod na ścieżki, które wprost nie posiadają jakichkolwiek zależności. Następnie są one dzielone na cztery lub więcej części (jeden dla każdej jednostki obliczeniowej CPU) i pakowane razem w większe instrukcje z dodatkową informacją odnośnie jednostki, na której ma być wykonywana. Rezultatem tego jest pojedynczy wielki [op-code](#) (stąd nazwa "Very Long").
- Z tego powodu pojedyncze rozkazy posiadają w sobie już zdekodowane (lub wstępnie zdekodowane) sygnały sterujące, dane oraz, co najważniejsze, "instrukcje" dedykowane dla konkretnych jednostek wykonawczych mikroprocesora. Tak stworzona pojedyncza instrukcja procesora VLIW ma wielkość rzędu setek bitów – 256 lub więcej.
- Procesor [TriMedia](#) firmy [Philips](#) jest przedstawicielem architektury VLIW, tak samo jak [Intel Itanium IA-64](#).

Superscalar DSPs

- **Superskalarność** ([ang. Superscalar](#)) – jest to cecha [mikroprocesorów](#) oznaczająca możliwość jednoczesnego ukończenia kilku instrukcji w pojedynczym cyklu zegara. Jest to możliwe dzięki zwielokrotnieniu jednostek wykonawczych, co umożliwia [obliczenia równoległe](#).
- Pierwszym [procesorem Intela](#) z rodziny [x86](#) wykorzystującym fragmentaryczną superskalarność był procesor [Pentium](#), który posiadał dwie jednostki wykonawcze, z czego jedną zubożoną, mogącą wykonywać tylko proste instrukcje; [Pentium Pro](#) posiadał już 3 jednostki wykonawcze. Większość procesorów superskalarnych nie ma w pełni zduplikowanej jednostki wykonywania kodu - mogą mieć wiele [ALU](#), jednostek zmiennopozycyjnych i tak dalej, wobec czego pewne instrukcje będą wykonywane bardzo szybko, a inne nie.
- Pełne wykorzystanie wszystkich jednostek wykonawczych zależy od tego, czy w programie nie występują zależności między kolejnymi instrukcjami - tj. czy kolejna instrukcja jako argumentu nie potrzebuje wyników poprzedniego rozkazu.
- Ponadto współczesne procesory, np. Pentium Pro i nowsze, mogą zmieniać [kolejność wykonywanych instrukcji](#) (zachowując oczywiście zależności między instrukcjami) - aby w pełni wykorzystać jednostki wykonawcze wyszukują instrukcje niezależne od siebie i wykonują je równoległe.

Rozdział rozkazów

Multi-Issue Approaches: Superscalar vs. VLIW

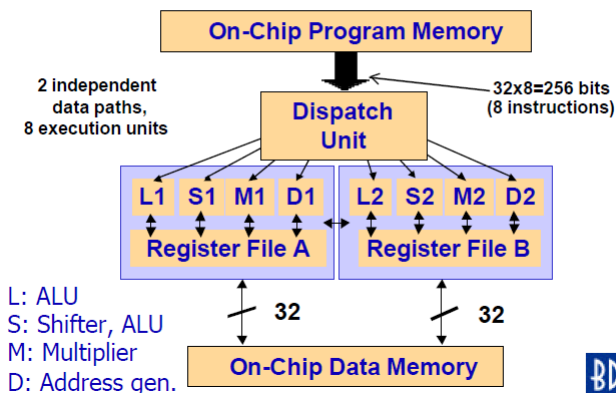


semestr zimowy 2012/2013, E-3, WIEIK, PK

41

Procesory DSP typu VLIW

Example VLIW Data Path ('C62xx)



semestr zimowy 2012/2013, E-3, WIEIK, PK

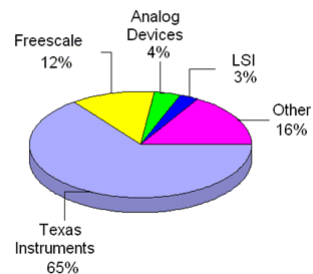
42

Producenci procesorów sygnałowych

- Texas Instruments (C2000, C5000, C6000)
- Freescale (Motorola) (MSC81xx, StarCore Architecture)
- Analog Devices (SHARC, Blackfin)
- LSI
- Agere Systems
- NXP (Philips) (TriMedia VLIW)

TRADITIONAL DSP VENDOR MARKET SHARES

CY 2008: \$6.5 Billion



semestr zimowy 2012/2013, E-3, WIEiK, PK

43

Historia procesorów sygnałowych

- Pierwsze systemy DSP składały się z oddzielnych układów (np. jednostka CPU, układ mnożący, pamięci, układy I/O)
- Jednym z pierwszych w pełni scalonych procesorów sygnałowych był układ firmy Texas Instruments TMS32010, wprowadzony na rynek w 1982r.
- Był to układ 16-bitowy, operacje mnożenia z dodawaniem wykonywał w 390ns.
- Bardzo udanym układem był DSP56000, firmy Motorola

semestr zimowy 2012/2013, E-3, WIEiK, PK

44

Procesory DSP, firmy Texas Instruments

- [TMS320C1x](#), pierwsza generacja 16-bit [fixed point](#) DSPs.
- [TMS32010](#), pierwszy procesor DSP przedstawiony 1983, używający zewnętrznej pamięci
 - [TMS320M10](#), procesor z wewnętrzną pamięcią [ROM](#) , 3KB
 - [TMS320C10](#), [TMS320C15](#) etc.
- [TMS320C3x](#), zmiennie-przecinkowy
 - [TMS320C33](#)
- [TMS320C4x](#), zmiennie-przecinkowy
- [TMS320C8x](#), układ wieloprocessorowy
 - [TMS320C80 MVP](#) (multimedia video processor) zmiennie-przecinkowy 32-bit "master processor" i cztery 32-bit fixed-point "parallel processors".

Procesory DSP, firmy Texas Instruments

- TMS320 **C2000** seria lub TMS320C2x, fixed point
- TMS320 **C5000** seria 16-bit fixed point
- TMS320 **C6000** seria, lub TMS320C6x: VLIW based DSP's. fixed point i floating point
- The **DaVinci Series**, lub TMS320DM64xx seria [SoCs](#) z układem z serii C6000 DSP, ARM9 GPPs, i układami Digital Media.
- **OMAP** odmiana, zawiera procesor typu ARM na tym samym układzie
- **DA** odmiana (do zastosowań "Internet audio") seria DA25x jest z procesorem ARM procesorem DSP typu C55x.
- **DM** odmiana: DM270 ma procesory ARM7 i ARM9

Porównanie szybkości działania DSP

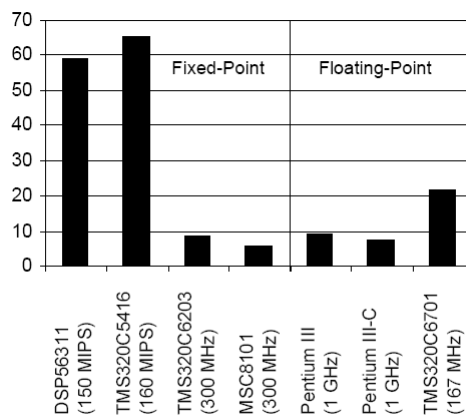


FIGURE 1. Execution times for a 256-point complex FFT, in microseconds (lower is better).

semestr zimowy 2012/2013, E-3, WIEIK, PK

47

Kod programu w języku C:

```
#include <stdio.h>
int tab1[]={1,2,3,4};
int tab2[]={8,6,4,2};
int main(void)
{
    int i;
    int wynik =0;
    for (i = 0; i < 4; i++)
        wynik += tab1[i]*tab2[i];
    printf("%d\n", wynik);
    return 0;
}
```

Realizacja prostego filtra SOI -Pentium

$$y = \sum_{i=0}^3 data[i] * coeff[i]$$

Procesor Pentium aby wykonać ten program musi:

1. Wpisać do wskaźnika1 adres data[0]
2. Wpisać do wskaźnika2 adres coeff[0]
3. Odczytać zawartość komórki pamięci wskazującej na data[0]
4. Odczytać zawartość komórki pamięci wskazującej na coeff[0]
5. Wymnożyć data[0]*coeff[0]
6. Dodać wynik mnożenia do poprzedniego wyniku
7. Ustawić wskaźnik1 na następny element tablicy data
8. Ustawić wskaźnik2 na następny element tablicy coeff
9. Zwiększyć i o jeden
10. Jeśli i < 4 wrócić do kroku 3

semestr zimowy 2012/2013, E-3, WIEIK, PK

48

Realizacja prostego filtra SOI - DSP

Procesor DSP wykonuje punkty 3-8 jako jedną operację, więc jedno sumowanie zajmuje łącznie tylko 3 cykle maszynowe. Kroki 3-8 są to tzw. podstawowe operacje DSP. Używając identycznego kodu źródłowego dla procesora Texas Instruments TMS320F2812 uzyskujemy kod maszynowy w postaci:

```

Address      MCode      Assembly Instruction
0x8000      FF69          SPM 0
0x8001      8D04 0000R  MOVL XAR1,#data
0x8003      76C0 0000R  MOVL XAR7,#coeff
0x8005      5633          ZAPA
0x8006      F601          RPT #1
0x8007      564B 8781    || DMAC ACC:P,*XAR1++,*XAR7++
0x8009      10AC          ADDL ACC,P<<PM
0x800A      8D04 0000R  MOVL XAR1,#y
0x800B      1E81          MOVL *XAR1,ACC
  
```

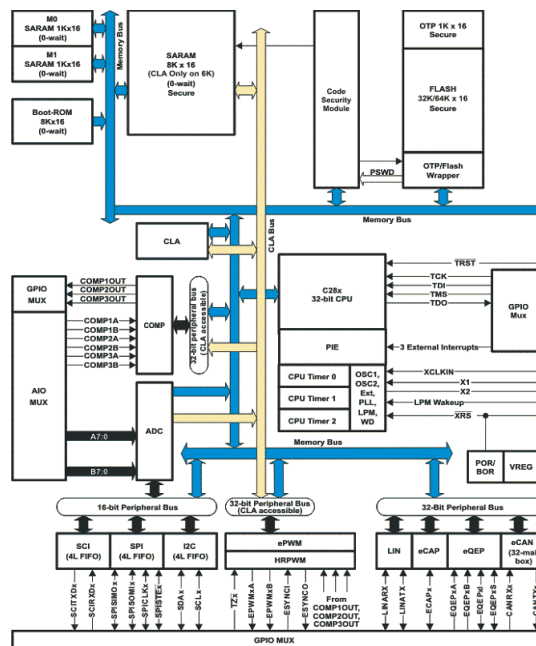
Space : 12 Code Memory ; 9 Data Memory

Execution Cycles : 10 @ 150MHz = 66 ns

semestr zimowy 2012/2013, E-3, WIEIK, PK

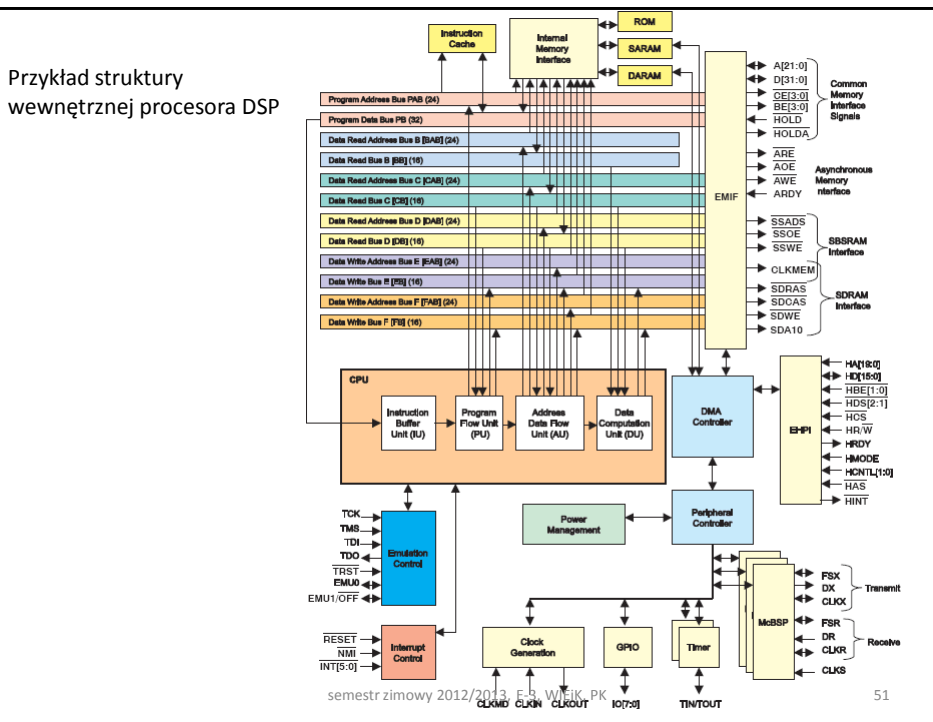
49

Przykład struktury
wewnętrznej procesora DSP
z firmy Analog Devices



semestr zimowy 2012/2013, E-3, WIEIK, PK

50



Kontrolery sygnałowe DSC – Digital Signal Controller

- Digital Signal Controller – DSC – jest to połączenie mikrokontrolera i procesora sygnałowego w jeden układ scalony - tzw. kontroler sygnałowy.
- Elementy, które są typowe dla procesorów DSP, np. jednostka MAC, szybki rejestr przesuwany (ang. barrel shifters), rozszerzony akumulator, szybki układ przerwań sprzętowych są obecnie wstawiane do mikrokontrolerów.
- Lista rozkazów kontrolerów DSC jest też powiększana o rozkazy typu DSP, przyspieszające obliczenia
- Można też powiedzieć, że układ DSC to procesor sygnałowy z elementami mikrokontrolera, wyposażony w np. szybkie i wielokanałowe modulatory PWM, interfejsy szeregowo, przetwornik ADC.
- Układy DSC głównie mają zastosowanie w układach sterowania silników elektrycznych, układach przetwarzania energii, układach przetwarzania sygnałów z czujników gdzie algorytmy numeryczne są skomplikowane i wymagają odpowiedniej mocy obliczeniowej.
- Pojęcie DSC zostało po raz pierwszy użyte przez firmę Microchip, w roku 2002.
- Największymi producentami układów DSC są firmy Microchip, Freescale, Texas Instruments.
- Inni producenci oferują podobne układy ale traktowane są te układy jako mikrokontrolery.

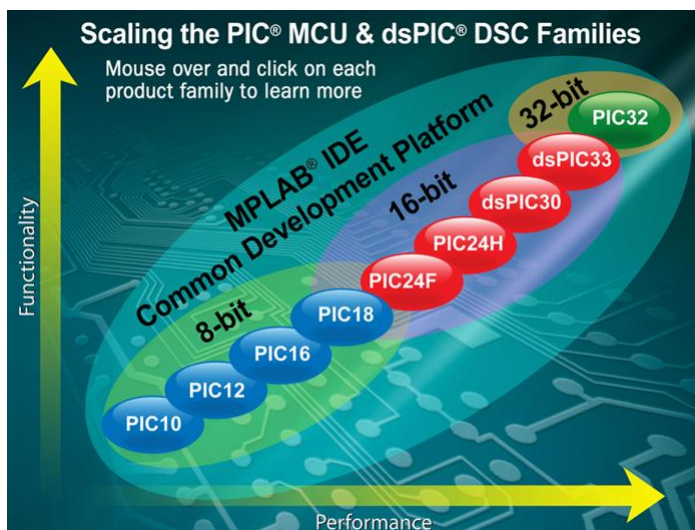
Popularne układy DSC

Producent	układ	Zegar (MHz)	Flash (kB)	Kanały PWM, rozdzielczość, Min. wsp. Wypełnienia
Microchip	dsPIC30F	30	6-144	4-8 (16 bits, 1 or 16.5 ns depending on part)
	dsPIC33F	40	12-256	8 (16 bits, 12.5 ns)
Texas Instruments	TMS320F280x	60-100	32-256	16 PWM (13 bits, 150 ps)
	TMS320LF240x	40	16-64	7-16 PWM (11 bits, 150 ps)
Freescale	MC56F83x	60	48-280	12 PWM (15 bits, 10 ns)
	MC56F80x	32	12-64	5-6 PWM (15 bits, 10 ns)
	MC56F81x	40	40-572	12 PWM (15 bits, 10 ns)

semestr zimowy 2012/2013, E-3, WIEIK, PK

53

Procesory DSC, firmy Microchip



semestr zimowy 2012/2013, E-3, WIEIK, PK

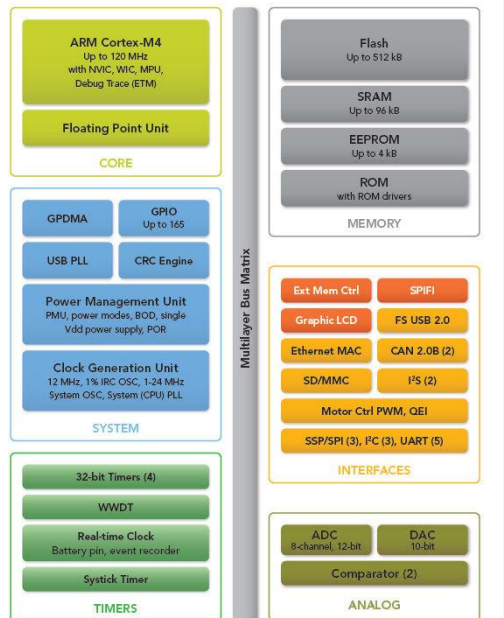
54

Struktura układu DSC firmy NXP, opartego o rdzeń ARM Cortex – M4

LPC4000 Digital Signal Controller Family



LPC407x/8x



semestr zim