

# Wstęp do CPS



(Wykorzystano materiały książki Marven C., Ewers G.: „Zarys cyfrowego przetwarzania sygnałów” Warszawa WKŁ 1999 oraz wykładu prof. A. Materki)

W niniejszych wykładach zostaną omówione tajniki cyfrowego przetwarzania sygnałów (CPS). Zdaję sobie sprawę z tego, że zawarty w wykładach materiał nie pokrywa pełnego zakresu wiedzy niezbędnego, na przykład, do zaprojektowania szybkiego modemu czy procesora CPS.

Daję Państwu jednak przejrzyste wprowadzenie do tego wysoce zaawansowanego technicznie tematu.

Dzięki tym wykładom zapoznacie się z **próbkowaniem sygnałów, filtrowaniem informacji, przekształceniami częstotli-wościowymi, analizą falkową**, i innymi zagadnieniami CPS. Umożliwią Wam one także podjęcie właściwej decyzji podczas borykania się z tą ciekawą tematyką.

Jednym z celów wykładów jest **dostarczenie studentowi podstawowego zakresu wiedzy niezbędnego przed przystąpieniem do lektury bardziej zaawansowanych opracowań** dotyczących cyfrowego przetwarzania sygnałów - opracowania te często mają zbyt wysoki pułap wiedzy dla początkującego.

## CO TO JEST CYFROWE PRZETWARZANIE SYGNAŁÓW ?

Otoczające nas środowisko pełne jest odczuwanych przez nas sygnałów, takich jak **dźwięk, temperatura** czy **światło**. W celu przetworzenia fali dźwiękowej w sygnały elektryczne dostarczane do naszego mózgu wykorzystujemy nasze uszy.

Następnie analizujemy niektóre właściwości dźwięku, takie jak **amplituda, częstotliwość** czy **zależności fazowe** w celu jego klasyfikacji oraz określenia kierunku rozchodzenia się. Łatwo możemy rozpoznać rodzaj dźwięku - **muzykę, mowę** czy też **odgłos młota pneumatycznego**.

Z kolei bodźce cieplne rozpoznawane są przez nasze nerwy umieszczone w wystawionych na oddziaływanie temperatury częściach naszej skóry. Nerwy te następnie przesyłają sygnały do mózgu. Analiza tych sygnałów jest prosta i może spowodować włączenie ogrzewania, czy otwarcie okna w pokoju.

Jak wiadomo nasz organizm jest wyposażony w **pięć** kategorii czujników: **sluchu, dotyku, smaku, zapachu** i **wzroku** oraz oczywiście w „superkomputer” jakim jest nasz mózg.

W większości przypadków niezbędne decyzje podejmujemy samodzielnie. Istnieje jednak wiele przypadków, w których celowe jest poproszenie o pomoc **maszyny**. Na podstawie swoich czujników związanych z niektórymi zmiennymi środowiska podejmuje **ona** decyzje niezależnie od nas.

Stosunkowo prostym zadaniem jest wyposażenie urządzenia w zestaw czujników przekształcających zmienną w sygnały elektryczne. Znacznie trudniejsze okazuje się przystosowanie rejestrującego sygnały komputera by reagował na nie w sposób podobny do naszego mózgu.

Nasz mózg funkcjonuje na podstawie elektrycznej reprezentacji nieustannie zmieniających się sygnałów, takich jak intensywność dźwięku, ciśnienie itp. Te nieprzerwanie zmieniające się sygnały nazywane są **sygnałami analogowymi**.

Nasz mózg może być uważany jako potężny **komputer analogowy**.

Aczkolwiek możliwe jest skonstruowanie elektronicznego komputera analogowego, tym niemniej jesteśmy bardziej przyzwyczajeni do komputerów cyfrowych, takich jak PC. Komputer osobisty doskonale sobie radzi z problemami obliczeniowymi wielu różnych zastosowań, jak na przykład baz danych czy arkuszy kalkulacyjnych, lecz nie za dobrze jest przystosowany do przetwarzania nieustannie zmieniających się sygnałów otaczającego nas świata.

Nasze kubki smakowe, **nos**, **uszy**, **oczy** i **skóra** przetwarzają odebrane sygnały zewnętrzne w energię elektryczną, którą dostarczają, za pośrednictwem nerwów do naszego mózgu. W podobny sposób, do przetworzenia takich wielkości, jak **ciśnienie**, **temperatura** czy **dźwięk** w sygnały elektryczne **możemy wykorzystać rozmaite czujniki elektroniczne**. Pozostaje jednak **problem konwersji sygnałów analogowych w wielkości cyfrowe** w celu przekazania wartości liczbowych komputerowi cyfrowemu do dalszej obróbki. Ten proces konwersji nazywany jest **przetwarzaniem analogowo-cyfrowym**, czy też przetwarzaniem **a/c**. Obróbka dostarczonych w tej postaci sygnałów jest dalej realizowana przez komputer cyfrowy i w związku z tym jest nazywana **cyfrowym przetwarzaniem sygnałów** lub w skrócie **DSP** (Digital Signal Processing).

We współczesnych systemach DSP w charakterze procesorów cyfrowych stosowane są jednokładowe mikrokomputery zaprojektowane specjalnie do tego celu i nazywane **procesorami DSP**.

### **Historia rozwoju DSP**

Cyfrowe przetwarzanie sygnałów było w oczywisty sposób uzależnione od stanu rozwoju techniki komputerowej. Większość aparatu matematycznego i algorytmów opracowano już w latach 50 poprzedniego stulecia. Mając na podorędziu odpowiednie algorytmy projektanci zaczęli poszukiwać takich architektur komputerowych, które najbardziej nadawały by się do ich efektywnej implementacji.

Podstawowym wymogiem był czas przetwarzania, czyli **możliwość dokonania przewidzianej symulacji w dopuszczalnym dla danego zastosowania czasie**.

Nie wiadomo dokładnie kiedy zastosowano komputery cyfrowe do przetwarzania w czasie rzeczywistym, lecz z tą chwilą podjęcie do przetwarzania i jego cele zmieniły się.

**Główną cechą systemów DSP stała się możliwość ich pracy w trybie czasu rzeczywistego**, czyli konieczność wykonania wszystkich operacji obliczeniowych i sterujących w czasie wystarczającym na niezakłócony przebieg trwającego procesu.

Wcześniejsze systemy były zdolne jedynie do zapisania przebiegu poszczególnych sygnałów w pamięci komputera i przetworzenia ich w **późniejszym czasie**.

Oczywiście takie systemy **nie mogły mieć wpływu na przebieg badanego procesu i nie mogły podjąć natychmiastowej decyzji** na podstawie zgromadzonych danych.

Zakres omawianego w naszych wykładach materiału obejmuje głównie **cyfrowe przetwarzanie sygnałów w czasie rzeczywistym**.

Związane z tym szczególne wymogi stymulowały przez ostatnie 60 lat szybki rozwój matematyki i technologii komputerowej przystosowanej do cyfrowego przetwarzania sygnałów.

**Podstawowy model** matematyczny opisujący **sygnały ciągłe** opiera się na sformułowanych w XIX wieku przekształceniach **Fouriera** i **Laplace'a**.

#### **Jean Baptiste Joseph Fourier (1768-1830)**

Urodził się 21 marca 1768 roku w Auxerre, we Francji. Pochodził z bardzo biednej rodziny. W 9 roku życia stracił obydwój rodziców. Brak pochodzenia i tytułu szlacheckiego uniemożliwiły mu naukę w szkole wyższej. Sytuację zmieniła dopiero Rewolucja Francuska.

Fourier zainteresowania matematyczne łączył z działalnością polityczną. Był dwukrotnie uwięziony, dwukrotnie udało mu się uniknąć gilotyny w okresie Rewolucji Francuskiej. Fourier współpracował z Naoleonem Bonaparte, który w 1802 mianował go prefektem dzielnicy Francji z siedzibą w Grenoble.

I właśnie w tym okresie Fourier opracował swoją teorię.

Fourier po raz pierwszy wykazał oraz uzasadnił możliwość reprezentacji dowolnej funkcji za pomocą nieskończonej sumy składowych harmonicznych.

W 1807r. praca Fouriera była recenzowana przez czterech znakomitych naukowców matematyków. Trzech z nich oceniło ją pozytywnie, lecz czwarty, J.L. Lagrange, podtrzymał swoje krytyczne stanowisko sprzed 50 lat. W efekcie praca ta nie została opublikowana. Po poprawkach i przeróbkach ukazała się w postaci książkowej 15 lat później, w 1822 r.

### **Pierre Simon, markiz de Laplace**

**Pierre Simon, markiz de Laplace** był wybitnym teoretykiem astronomii, prawdopodobnie największym od czasów Newtona. Urodził się 20 lat przed Fourierem, a wymyślony przez niego aparat matematyczny służył do opisu i lepszego zrozumienia ruchu planet.

Podobnie jak szeregi Fouriera przekształcenie Laplace'a zostało szeroko wykorzystane w najprzeróżniejszych dziedzinach nauki i techniki. Przy zastosowaniu nieznacznego rozszerzenia i odpowiedniej interpretacji przekształcenie Laplace'a staje się przekształceniem **Z**. Podobnie jak przekształcenie Laplace'a, również przekształcenie Fouriera ma swój odpowiednik dla sygnałów cyfrowych.

Dyskretne przekształcenie Fouriera (DFT) stało się popularne w latach 40. i 50. zeszłego wieku, jako narzędzie pomocne przy rozwijaniu cyfrowych technik obliczeniowych. Jednak możliwość jego użycia była ograniczona ze względu na duże potrzeby mocy obliczeniowej.

W roku 1965 pojawiła się publikacja pod niewiele obiecującym tytułem „**Algorytmy dla obliczeń komputerowych zespolonych szeregów Fouriera**” autorstwa **Cooleya i Tukeya**. Pozycja ta miała jednak przełomowe znaczenie, gdyż proponowała znaczne udoskonalenie DFT.

Autorzy wykorzystali niektóre szczególne właściwości DFT i opracowali algorytm powszechnie znany jako szybkie przekształcenie Fouriera (FFT). Podstawową przewagą nowego algorytmu była radykalna redukcja liczby niezbędnych do obliczeń operacji mnożenia.

Jak się wkrótce przekonamy **mnożenie** jest głównym czynnikiem ograniczającym możliwości obliczeniowe większości algorytmów DSP.

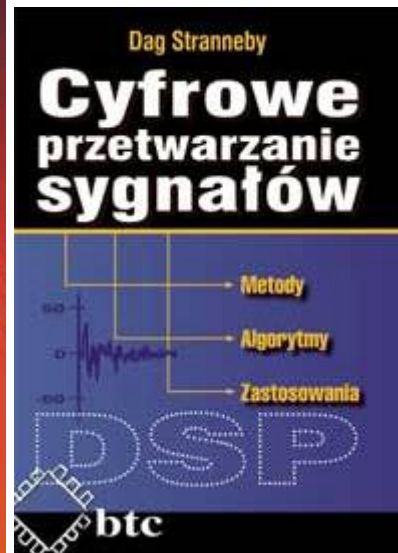
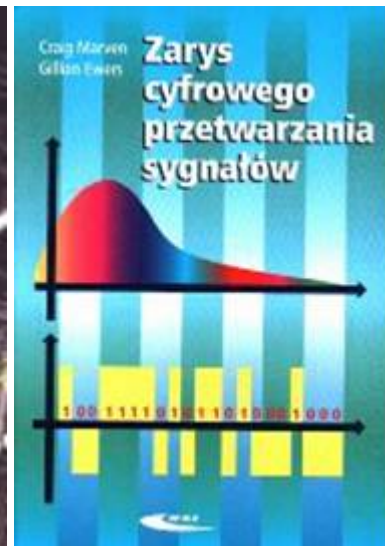
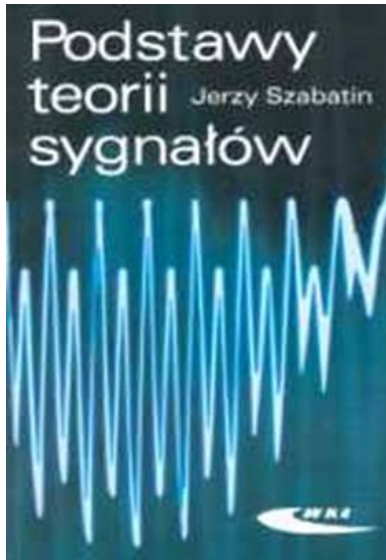
Redukując 100-krotnie liczbę mnożeń dla dużego (1024-punktowego) dyskretnego przekształcenia Fouriera, FFT stała się krokiem milowym w rozwoju cyfrowego przetwarzania sygnałów.

Istnieje kilka interesujących pozycji literaturowych dotyczących początków FFT.

Jako źródło swych inspiracji Cooley i Tukey podali jedynie pracę Gooda z roku 1958. Jednak inne badania wykazały, że za ojca FFT można uznać niemieckiego matematyka Rungego lub nawet Gaussa.

Ostatnie udoskonalenia teorii DSP nie są już tak przełomowe. Niemniej prace na ten temat są nieprzerwanie kontynuowane i zmierzają ku takim rozwiązaniom, jak zrównoleglenie algorytmów DSP, analiza falkowa, kompresja fraktalna i falkowa, steganografia cyfrowa, itp. Istnieje kilka fundamentalnych opracowań na temat podstaw teoretycznych DSP.

Dwoma klasycznymi pozycjami są książki „**Digital Signal Processing**” autorstwa **Oppenheima i Schafera** oraz „**Theory and Application of Digital Signal Processing**” napisana przez **Rabinera i Goulda**. Książki te zawierają szczegółowy opis teoretyczny DSP i nie są przeznaczone dla początkujących.



### Architektura komputerowa w aspekcie DSP

Jeśli przyjrzymy się chronologicznemu rozwojowi specjalnych procesorów jednocukładowych przystosowanych do realizacji algorytmów DSP, powinniśmy oddzielić od siebie ich architekturę logiczną i technologię wykonania.

W przypadku urządzeń półprzewodnikowych rozwój podstawowych struktur i rozwiązań logicznych zwykle znacznie wyprzedza możliwość ich fizycznej realizacji.

**Rozpatrzmy więc wpieryw rozwój architektury urządzeń DSP.**

Podstawowe typy architektury zarówno dużych komputerów, jak i mikroprocesorów podzielić można na **dwie kategorie**. Architektura pierwszego znaczącego komputera elektromechanicznego charakteryzowała się **oddzielnymi obszarami pamięci** dla programów i dla danych.

W ten sposób dostęp do kodu programu i danych realizowany być mógł jednocześnie.

Tego typu architektura nosi nazwę **harwardzkiej**, gdyż została opracowana w późnych latach 30. przez **Howarda Aikena**, fizyka Uniwersytetu Harvarda. Zbudowany tam komputer **Harvard Mark I** zaczął liczyć w roku 1944.

Pierwszym w pełni elektronicznym komputerem uniwersalnym był prawdopodobnie **ENIAC** budowany od roku 1943 do 1946 w Uniwersytecie Pensylwańskim.

Komputer ten miał architekturę zbliżoną do Harvard Mark I i posiadał oddzielne pamięci programu i danych.

**Z powodu złożoności wynikającej z konieczności stosowania dwóch odrębnych systemów pamięciowych architektura harwardzka nie zyskała popularności w rozwiązaniach uniwersalnych komputerów i w mikroprocesorach.**

Jednym z konsultantów projektu ENIAC był matematyk węgierskiego pochodzenia **John von Neumann**. Jest on uznawany za twórcę innej architektury komputerowej o doniosłym znaczeniu - zostało to udokumentowane w roku 1946 w pracy Burksa, Goldstina i von Neumanna.

Tak zwana architektura von Neumanna ustanowiła standard rozwoju systemów komputerowych w przeciągu ostatnich pięćdziesięciu lat.

Koncepcja owej architektury jest bardzo prosta i opiera się na dwóch podstawowych założeniach:

*po pierwsze:* nie istnieje zasadnicza różnica między kodem programu i danymi;

*po wtóre:* rozkazy programowe mogą być podzielane na dwie części, z których jedna zawiera kod operacyjny a druga adres operandu (danych, na których przeprowadzana jest operacja).

**W ten sposób stosowana być może pojedyncza przestrzeń adresowa, zawierająca zarówno kod programu jak i dane.**

Nowa architektura uprościła projektowanie komputerów, lecz zarazem wprowadziła pewne ograniczenie polegające na tym, że w danym czasie komputer może mieć dostęp albo do kodu programu albo do danych.

Praktyka pokazała jednak, że ograniczenie to nie stanowi istotnego problemu dla współczesnych komputerów uniwersalnych.

Powszechnie stosowane mikroprocesory, takie jak rodzina **68000** firmy Motorola, czy też rodzina **i86** firmy Intel wykorzystują właśnie architekturę von Neumanna.

Te i inne popularne mikroprocesory uniwersalne charakteryzują się także innymi cechami typowymi dla przeważającej większości komputerów ostatnich 50 lat.

Podstawowym blokiem obliczeniowym tych urządzeń jest **jednostka arytmetyczno-logiczna (ALU)** oraz **rejestr przesuwany**.

Takie operacje jak dodawanie, przesyłanie danych czy odejmowanie są łatwo wykonywane w przeciągu kilku cykli zegarowych.

Rozkazy złożone, jak **mnożenie** czy **dzielenie** tworzone są z sekwencji prostych operacji przesuwania i dodawania lub odejmowania. Urządzenia tego typu nazywane są komputerami o złożonym zestawie instrukcji (**CISC**).

Urządzenia CISC zawierają rozkazy „mnożenia”, lecz są one realizowane wewnątrz komputera jako ciąg rozkazów mikrokodowych, zaszytych w umieszczonej wewnątrz procesora pamięci stałej. Taka mikrokodowana instrukcja mnożenia wymaga jednak wielu cykli zegarowych.

**Cyfrowe przetwarzanie sygnałów jest realizowane od ponad 50 lat np. za pomocą minikomputerów. Duże koszty oraz mała szybkość obliczeń ograniczały liczbę użytkowników tych aplikacji.**

Z powodów, które staną się zrozumiałe nieco później cyfrowe przetwarzanie sygnałów wykorzystuje wiele cykli obliczeniowych o postaci:

$$A=BC+D$$

To proste wyrażenie zawiera operację mnożenia i operację dodawania.

Wszystkie te przekształcenia sygnału można zrealizować wykorzystując następujące działania elementarne:

- pamiętanie i opóźnianie,
- mnożenie,
- dodawanie.

Procesor sygnałowy (ang. DSP - digital signal processor) wykonuje te elementarne działania.

Podstawowa operacja arytmetyczna, wykonywana w jednym cyklu zegara w procesorze (operacja MAC - multiply and accumulate) :

$$acc = acc + x \times y$$

Główny problem CPS: wykonanie zadania w czasie między kolejnymi chwilami próbkowania (w czasie rzeczywistym).

**Z powodu długiego czasu wykonywania operacji mnożenia mikroprocesory CISC nie zabardzo nadają się do tego typu obliczeń.**

Tak więc niezbędne, z punktu widzenia technik DSP, jest urządzenie wyposażone w możliwość wykonywania jednoczesnej operacji mnożenia i dodawania w przeciągu jednego cyklu zegarowego.

Z tego powodu konieczne jest inne podejście do architektury komputerowej.

*Innymi słowy należy dostosować architekturę komputerową do naszych wymagań.*

Podczas przetwarzania sygnałów w czasie rzeczywistym głównym problemem jest ilość obliczeń jakie możemy wykonać przed nadejściem kolejnej porcji danych.

Wczesne systemy DSP były konstruowane z użyciem standardowych elementów, to znaczy rejestrów przesuwanych, układów mnożących i sumatorów.

Natychmiast okazało się, że operacja mnożenia była głównym czynnikiem ograniczającym wydajność tamtych komputerów.

Dokonał się jednak postęp i opracowano nowe układy mnożące wykorzystujące zasadę przetwarzania potokowego oraz nowe technologie.

**We wczesnych latach 70. zeszłego stulecia pojawił się pierwszy układ mnożący realizujący pełne mnożenie w jednym cyklu zegarowym - układ ten był oparty na szybkich elementach logicznych ze sprzężeniem emiterowym (ECL).**

Wiodącą organizacją w dziedzinie badań nad DSP były w tym czasie Laboratoria Lincolna.

Opracowany tam procesor „Lincoln FDP” ujrzał światło dzienne w roku 1971 i charakteryzował się czasem mnożenia równym 600 nanosekund, lecz zbudowany był z około 10000 oddzielnych obwodów scalonych. Miał on także trudności z wykonywaniem operacji równoległych, gdyż oparty był na architekturze von Neumanna.

Kolejny procesor o nazwie „Lincoln LSP/2” był zbudowany z wykorzystaniem zdobytych doświadczeń. Zastosowano architekturę harwardzką oraz całkowicie równoległą strukturę wewnętrzną. Dzięki temu był on czterokrotnie szybszy niż FDP - zużyto też nań trzykrotnie mniej obwodów scalonych.

W połowie lat 70. rozwijaniem technik DSP zaczęły się także zajmować inne renomowane instytuty badawcze.

Zbudowano układ mnożący o czasie mniejszym niż 200 ns, a wykorzystujące go procesory były już zdolne do przetwarzania w czasie rzeczywistym. Urządzenia te były

jednak tak ogromne gabarytowo i tak kosztowne, że ich komercyjne wykorzystanie było praktycznie niemożliwe.

**Osiągnięto jednak główny cel - stworzono podstawową architekturę komputera DSP.**

Na realizację powszechnie dostępnego procesora należało jeszcze poczekać. Niezbędne było przełamanie bariery technologicznej w produkcji urządzeń półprzewodnikowych, gdyż rzeczywiście przydatny procesor DSP nie mógł zajmować więcej niż kilka obwodów scalonych, a najlepiej jeden.

**Większość zadań CPS można sklasyfikować jako splot, korelację lub transformację.**

- **Splot jest podstawową operacją dla filtracji sygnałów. Filtracja pozwala na zwiększenie stosunku mocy sygnału do mocy zakłóceń.**
- **Korelacja pozwala na porównanie sygnału z przebiegiem odniesienia (wzorcem). Redukuje wpływ składowych losowych i pomaga wykryć składowe sygnału podobne do wzorca (filtr dopasowany).**
- **Transformacja służy do wyznaczenia widma sygnału, a ogólniej pozwala znaleźć charakterystyczne cechy sygnału w wybranej przestrzeni cech (np. w dziedzinie częstotliwości).**

#### **UKŁADY SCALONE DLA DSP**

W przeciągu lat 70. technologia półprzewodnikowa, a w szczególności technologia produkcji obwodów scalonych stawała się coraz bardziej skomplikowana i zaawansowana.

Głównymi stymulatorami jej rozwoju były potrzeby armii Stanów Zjednoczonych oraz rynek konsumencki elektronicznych urządzeń powszechnego użytku, szczególnie zaś urządzeń audio-wizualnych.

Pragnienie zminimalizowania gabarytów urządzeń, zmniejszenia ceny oraz znacznego zwiększenia szybkości działania stało się w tym czasie dominujące.

Doprowadziło to do opracowania pod koniec lat 70. nowego procesu technologicznego, tzn. wynalezienia struktury półprzewodnikowej **N-MOS**, czyli struktury metal-tlenek-półprzewodnik z kanałem *N*.

**Znaczący postęp w mikroelektronice pozwolił na realizację układów cyfrowych VLSI, których prędkość i dokładność spełniają wymagania CPS czasu rzeczywistego.**

Układy scalone oparte na nowej technologii mogły pracować z pojedynczym zasilaniem pięciowoltowym oraz mogły być produkowane z zachowaniem geometrii trzymikronowej.

Dzięki temu w jednym obwodzie scalonym mogło być już umieszczone ponad 100 000 tranzystorów.

**Na początku lat 80. dostępne stały się cztery typy jednoobwodowych układów do cyfrowego przetwarzania sygnałów.**

Palme pierwszeństwa w tej dziedzinie przyznaje się często firmie American Microsystems Inc. (AMI) za opracowanie urządzenia S2811.

Jest to jednak kwestia sporna.

Mniej więcej w tym samym czasie pojawiły się urządzenia I2920 firmy Intel oraz  $\mu$ PD7720 firmy Nippon Electric Company (NEC).

Nieco później, w roku 1982, firma Texas Instruments wprowadziła na rynek układ TMS32010.

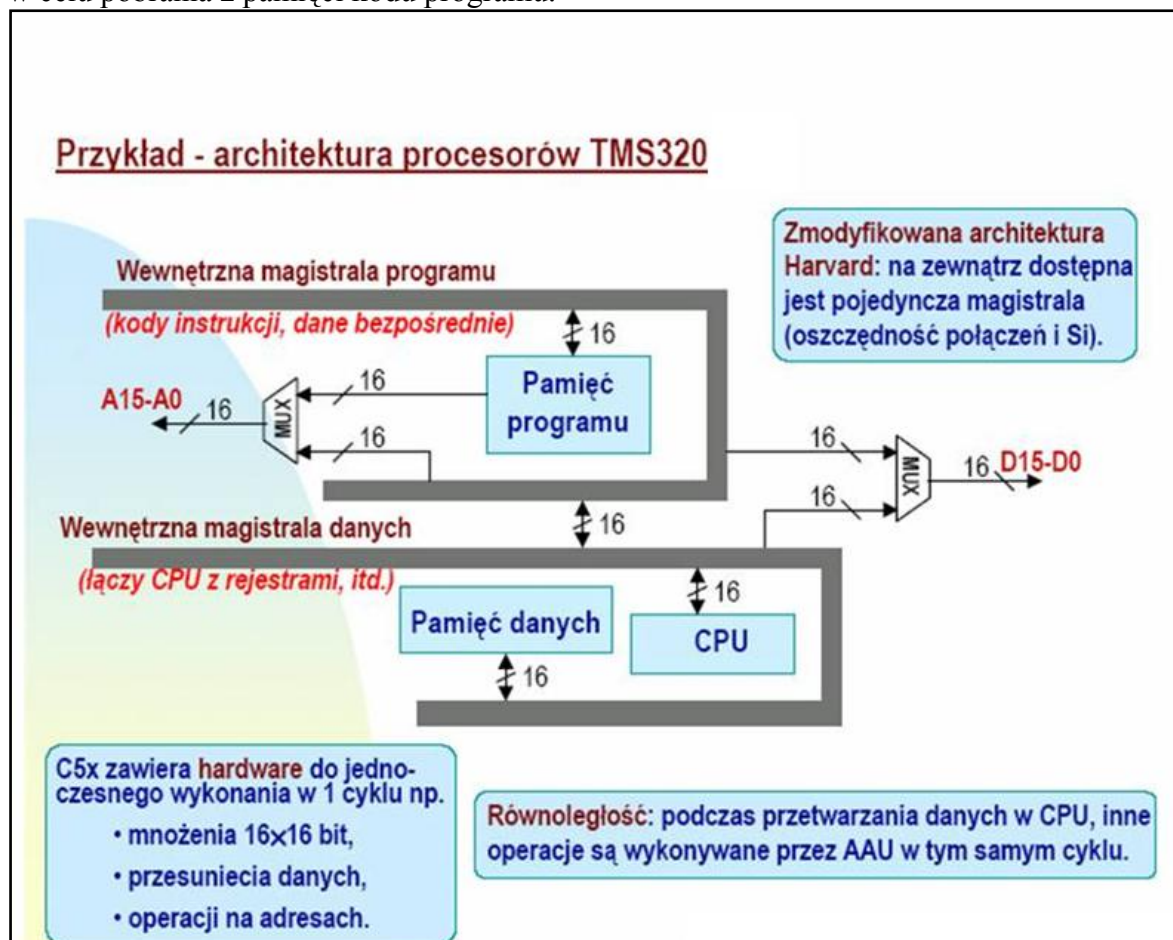
**Tak więc jednoobwodowa technika DSP stała się rzeczywistością.**

W celu oddzielenia pamięci programu od pamięci danych **wszystkie wczesne urządzenia DSP wykorzystywały architekturę harwardzką.**

Umożliwiało to jednoczesny dostęp do kodu programu i słów danych.

W systemach DSP pracujących w czasie rzeczywistym efektywny czas przepływu danych między procesorem a urządzeniami zewnętrznymi jest krytyczny.

Zastosowanie architektury harwardzkiej nie powoduje przerwania procesu przepływu danych w celu pobrania z pamięci kodu programu.

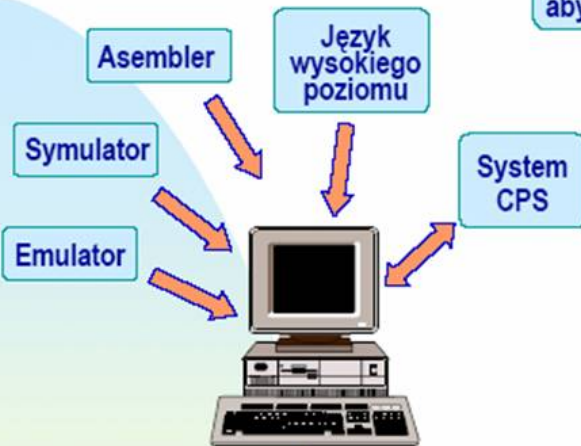


Trudno sobie wyobrazić wydajne stosowanie technik DSP bez odpowiednich narzędzi projektowych.

Texas Instruments uzupełnił swój mikroprocesor DSP o odpowiedni język assemblerowy, o zestaw ewaluacyjny i symulator, co upodobniło projektowanie systemów DSP do projektowania typowych systemów mikroprocesorowych.

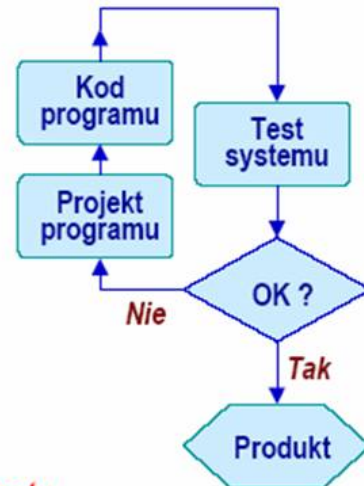
## Projektowanie systemów CPS

W odróżnieniu od układów analogowych, systemy DSP muszą zostać **zaprogramowane** aby mogły pełnić żądane funkcje.



### Cykl rozwojowy systemu

- ocena wykonalności (symulacja, parametry sprzętu, doświadczenie),
- specyfikacja funkcji systemu,
- projekt i testowanie programu



Pojęcie DSP jako „czarnej skrzynki” zostało wyeliminowane. Dalszy rozwój coraz bardziej szybkich urządzeń wspomaganych przez odpowiednie symulatory, programy uruchomieniowe, kompilatory C itp., pozwolił umieścić technikę DSP w głównym nurcie praktyki projektowania systemów mikrokomputerowych.

Burzliwie rozwijający się przemysł komputerów osobistych (PC) zgłaszał coraz to nowe wymagania dotyczące objętości pamięci.

To z kolei prowadziło do coraz mniejszych geometrii obwodów scalonych i umożliwiało umieszczanie większej liczby tranzystorów w jednym urządzeniu.

Proces technologiczny wytwarzania obwodów scalonych uzyskał kolejny skok jakościowy w momencie opracowania struktury półprzewodnikowej CMOS, czyli struktury komplementarnej MOS.

Stało się teraz możliwe stosowanie geometrii półprzewodnikowych w granicach **0,35** mikrona. Dzięki temu nie tylko zwiększyła się gęstość upakowania tranzystorów i bramek logicznych na jednostkę objętości, lecz również znacznie zmniejszyły się czasy przełączania i propagacji sygnałów wewnątrz obwodu scalonego, co w konsekwencji umożliwiło zwiększenie częstotliwości zegarowych oraz istotnie poprawiło przepustowość systemów.

W roku 1994 pojawiły się jednoukładowe procesory DSP zawierające ponad 4 miliony tranzystorów i wykonujące cykl mnożenia 32-bitowych danych zmiennoprzecinkowych w czasie 40 ns, a 16-bitowe mnożenie stałoprzecinkowe w czasie niższym niż 25ns.

**Współczesne procesory jednoukładowe (Lucent 1628, 0.12  $\mu\text{m}$ ) pozwalają na wykonanie mnożenia w czasie 10 ns przy napięciu zasilania 1 V i poborze mocy 20 mW.**

## Historia rozwoju procesorów sygnałowych

- 1978, S2811 (AMI): VMOS 4.5  $\mu\text{m}$ , mnożarka 16 b, cykl 300 ns (modemy),
- 1979, 2920 (Intel): NMOS 5  $\mu\text{m}$ , bez mnożarki (jedynie *shift and add*), cykl 400 ns,
- 1980, DSP-1 (Bell): NMOS 4.5  $\mu\text{m}$ , cykl 800 ns;  $\mu$  PD7720 (NEC): NMOS 4.5  $\mu\text{m}$ , cykl 250 ns, UV EPROM, środowisko sprzętowo-programowe (zyskał popularność),
- 1982, TMS32010 (Texas Instruments): NMOS 2.7  $\mu\text{m}$ , architektura harwardzka, cykl 200 ns, dane 16 b, mnożarka i akumulator 32 b, narzędzia uruchomieniowe (assembler, linker, symulator, emulator, karty); HD61810 (Hitachi): CMOS 3  $\mu\text{m}$ , cykl 300 ns, procesor zmiennoprzecinkowy (mantysa 12 b, cecha 4 b - nietypowy),
- 1983, MB8764 (Fujitsu): CMOS 2.3  $\mu\text{m}$ , cykl 100 ns,
- 1985, DSP-32 (Bell): CMOS 1.5  $\mu\text{m}$ , cykl 250 ns, zmiennoprzecinkowy, zgodny z IEEE (mantysa 24 b, cecha 8 b); TMS32020 (Texas Instruments): NMOS 2.4  $\mu\text{m}$ , cykl 200 ns, II generacja - większa RAM, rejestry pomocnicze,
- 1986, TMS320C25 (TI): ulepszony '20 - CMOS, ADSP2100 (Analog Devices), DSP56000 (Motorola),  $\mu$  PD77230 (NEC), PCD5010 (Philips), ...
- 1988-, TMS320C30 (TI): III generacja - zmiennoprzecinkowy, TMS320C40 (TI): IV gen. - praca równoległa, TMS320C50 (TI): V gen., stałoprzec., kompatybilny z 'C20.

Producent	Układ	Rok	MAC (ns)	# bitów stałoprzecink.	# bitów zmiennoprzecink.
Analog Devices	ADSP2100	1986	125	16/40	
	ADSP2100A	1988	80	16/40	
	ADSP210020	1991	50	32/40	40/80
AT&T	DSP32	1984	160	16	32/40
	DSP32C	1988	80	16/24	32/40
Motorola	DSP56001	1987	74	24/56	
	DSP96001	1989	75	32/64	44/96
Texas Instruments	TMS 32010	1982	390	16/32	
	TMS32020	1985	195	16/32	
	TMS320C25	1987	100	16/32	
	TMS320C30	1988	60	24/32	32/40
	TMS320C50	1990	35	16/32	

**W ostatnich latach zbiegły się następujące osiągnięcia:**

- opanowano technologię układów CMOS o submikrometrowej szerokości bramki (produkcja masowa układów scalonych procesora zmiennoprzecinkowego o cenie kilkunastu USD),
- opracowano nowe architektury procesorów pozwalające na pokonanie ograniczeń technologii (gęstość upakowania, moc zasilania, czas wykonania podstawowych działań) i uzyskanie zadowalającej prędkości przetwarzania sygnału,
- rozwinięto metody projektowania systemów DSP przy użyciu języków wysokiego poziomu oraz narzędzia wykorzystujące interfejsy graficzne (skrócenie fazy RD i szybsze wejście na rynek).