Politechnika Wrocławska

Architektura Systemów Komputerowych

Wykład 4

Dr inż. Radosław Michalski
Katedra Inteligencji Obliczeniowej, Wydział Informatyki i Zarządzania
Politechnika Wrocławska

Wersja 1.1, wiosna 2018

Źródła i licencja

Najbardziej aktualna wersja tego wykładu znajduje się tu:
https://github.com/rmhere/lecture-comp-arch-org

Zawartość tego wykładu

Hierarchia pamięci

Architektury systemów komputerowych

Hierarchia pamięci

Wprowadzenie

- w pamięci zlokalizowane są instrukcje i dane
- każdy chciałby posiadać nieograniczoną pamięć z krótkimi czasami dostępu
- jednak redukcja czasu dostępu skutkuje wzrostem kosztu
- **wobec tego pojawia się pojęcie hierarchii pamięci**

Hierarchia pamięci

Tło historyczne

We are therefore forced to recognize the possibility of constructing a hierarchy of memories, each of which has greater capacity than the preceding but which is less quickly accessible[1].

John von Neumann, public domain

[1] John Von Neumann and Goldstine Brucks. *Preliminary discussion of the logical design of an electronic computing instrument*. 1946.

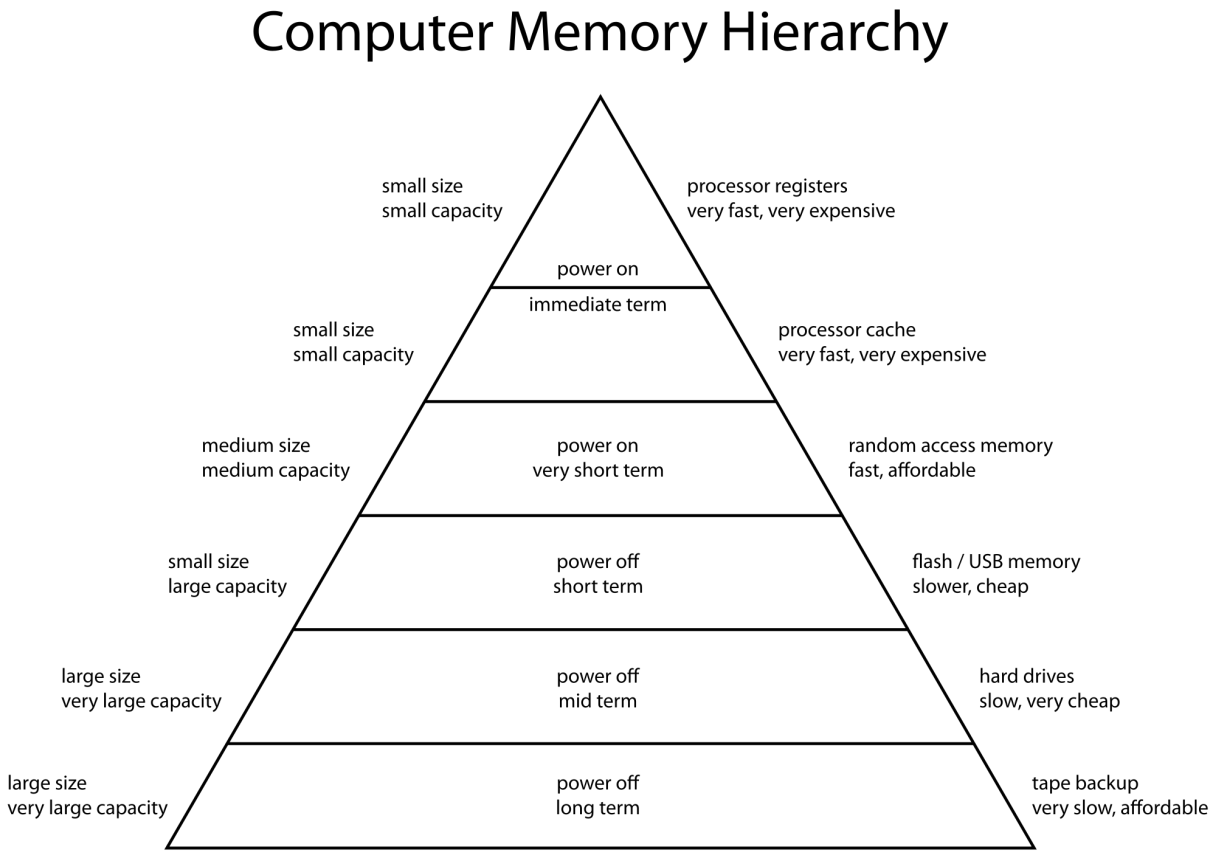
Hierarchia pamięci

Podstawowe założenia

- **pamięć systemu zorganizowana jest jako hierarchia**
- **cała przestrzeń adresowa** jest dostępna **w największej** i przez to najwolniejszej pamięci
- **mniejsze i szybsze pamięci**, każda zawierająca podzbiór danych z pamięci poniżej, **ulokowane są coraz bliżej procesora**
- patrząc z perspektywy **procesora**, korzysta on tylko **z najszybszego rodzaju pamięci**

Hierarchia pamięci

Schemat



Computer memory hierarchy, public domain

Hierarchia pamięci

Od góry do dołu

Rejestry procesora:

- na szczycie hierarchii
- najszybsze, szybkością zsynchronizowane z CPU
- duże zużycie energii
- mała liczba, małych rozmiarów

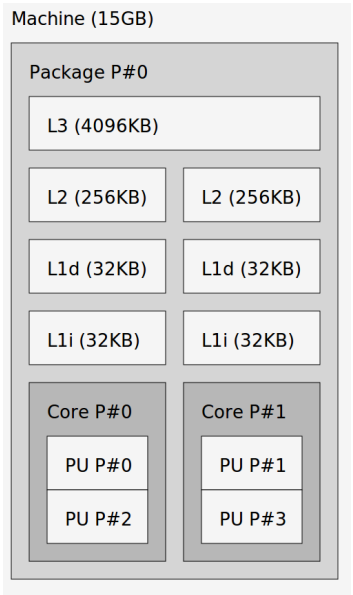
Pozostałe rodzaje pamięci (niżej w hierarchii):

- wolniejsze
- mniejsze zużycie energii
- tańsze

Hierarchia pamięci

Hierarchia pamięci - przykład #1

Lenovo™ ThinkPad® X260, Intel® Core™ i7-6500U, 16 GB RAM



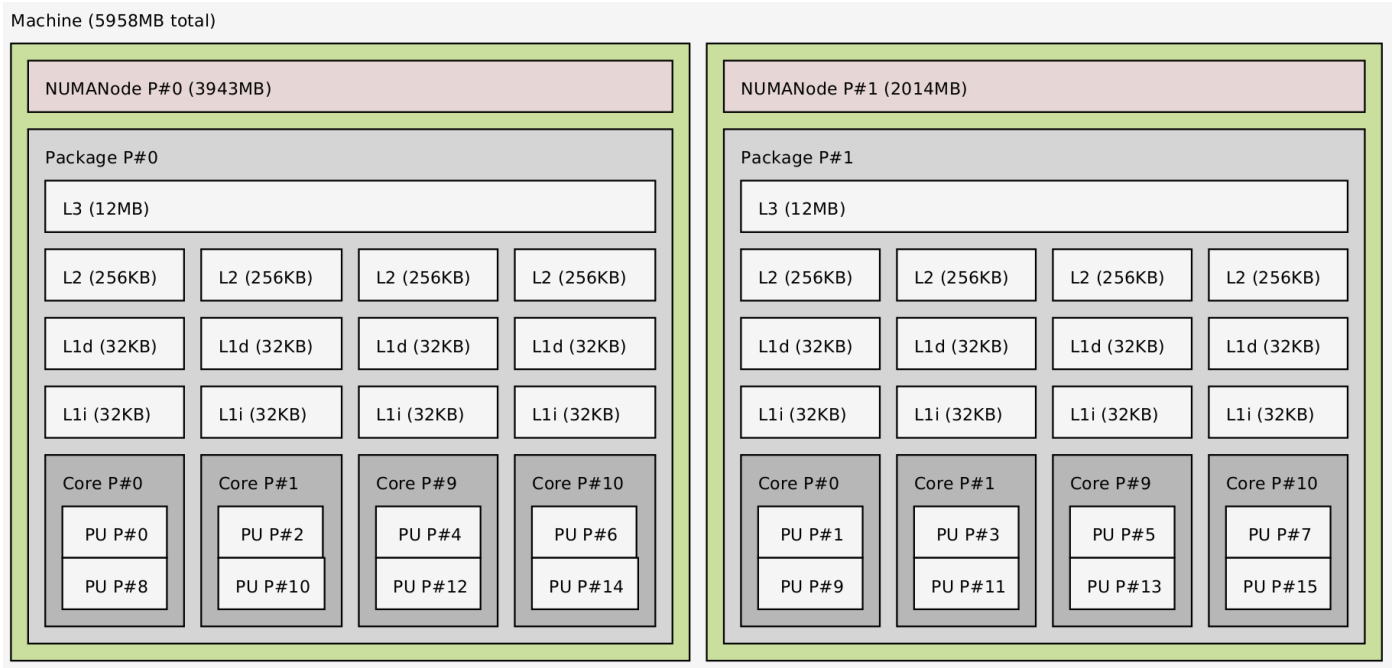
Zrzut ekranu z aplikacji *lstopo* (pakiet Portable Hardware Locality)

Lenovo, ThinkPad and ThinkStation are trademarks of Lenovo in the United States, other countries, or both. Intel, Intel Core and Xeon are trademarks of Intel Corporation or its subsidiaries in the U.S. and/or other countries.

Hierarchia pamięci

Hierarchia pamięci - przykład #2

Lenovo™ ThinkStation® D20, Intel® Xeon™ E5640, 6 GB RAM



Zrzut ekranu z aplikacji *lstopo* (pakiet Portable Hardware Locality)

Hierarchia pamięci

Trwałość

Jak długo dane rezydują w pamięci?

- **do wyłączenia**: rejestry, cache, RAM
- **do usunięcia**: dyski twarde, flash, pamięć taśmowa

Jednak należy mieć na uwadze, że nawet nietrwałe dane mogą pozostać przez pewien czas po wyłączeniu komputera[2].

[2] J Alex Halderman et al. "Lest we remember: cold-boot attacks on encryption keys". In: *Communications of the ACM* 52.5 (2009), pp. 91–98.

Hierarchia pamięci

Lokalność

W jaki sposób uzyskujemy dostęp do danych w pamięci?

- najczęściej - **nie losowo**
- **lokalność przestrzenna** - odczytywanie sąsiadujących komórek pamięci
- **lokalność czasowa** - odczytywanie komórek pamięci jeśli również niedawno były odczytywane

Świadomość tych właściwości sugeruje wprowadzenie pojęć **zimnego i gorącego przechowywania** i różnych **strategii zarządzania pamięcią**.

Hierarchia pamięci

Zarządzanie pamięcią

W jaki sposób mądrze zarządzać pamięcią?

- na początku, **zapoznaj się z hierarchią pamięci własnego systemu**
- umieść dane, **do których odwołujesz się najczęściej w górnych warstwach** hierarchii pamięci
- put **dane rzadziej odczytywane w dolnych warstwach**
- miej świadomość, że **lokowanie danych na różnych warstwarch to kompromis pomiędzy czasem dostępu a kosztem**
- **obserwuj w jaki sposób system komputerowy korzysta z pamięci** w obecnym układzie i wprowadzaj zmiany na bieżąco

Hierarchia pamięci

Procesor i cache

W jaki sposób procesor korzysta z cache?

- **procesor szuka danych w pamięci cache**, jeśli ich tam nie ma, ładowane są z niższych warstw hierarchii
- jeśli procesor znajdzie blok w pamięci cache, mówimy o **trafieniu** (cache hit), w przeciwnym razie o **nietrafieniu** (cache miss)
- **współczynnik trafień** (hit rate) to proporcja trafień do wszystkich żądań dostępu do pamięci (we współczesnych systemach hit rate dla L1 wynosi ponad 90%[3])
- **współczynnik nietrafień** (miss rate) wynosi *1 - hit rate*

[3] How L1 and L2 CPU Caches Work, and Why They're an Essential Part of Modern Chips, Joel Hruska, ExtremeTech

Hierarchia pamięci

Podsumowanie

- hierarchia pamięci wprowadza porządek wedle typów pamięci
- wyższe poziomy - szybsze ale droższe
- niższe poziomy - wolniejsze ale tańsze
- trwałość pamięci - czas życia danych
- lokalność pamięci - jakie dane będą odczytywane
- relacja pomięcy procesorem i pamięcią cache
- jeśli chcesz wiedzieć więcej, zapoznaj się z tym dokumentem

To do: użyj narzędzia **lstopo** z pakietu Portable Hardware Locality aby dowiedzieć się o organizacji pamięci własnego komputera.

Hierarchia pamięci

Źródła i polecane materiały

- T. Schwarz, *"Introduction to Information Storage Technology"*, Santa Clara University, CA, United States (materiały do kursu)
- D. Patterson, J. Hennessy, *"Computer Architecture: A Quantitative Approach"*, Elsevier (książka)
- R. Bryant, G. Ganger, *"15-213: Introduction to Computer Systems"*, Carnegie Mellon School of Computer Science, PA, United States (materiały do kursu)

Architektury systemów komputerowych

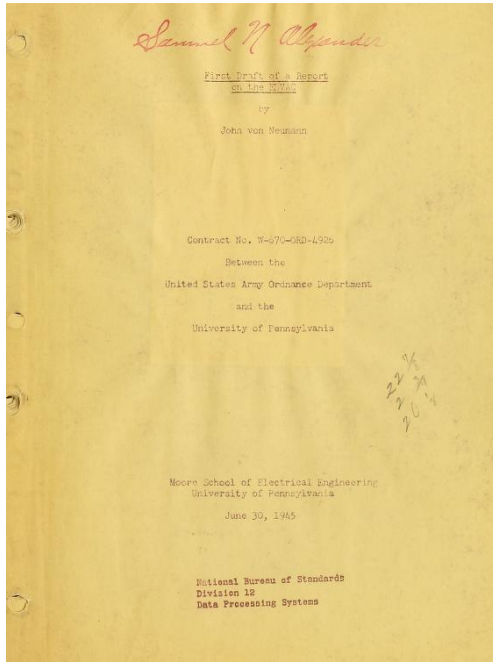
Wprowadzenie

- **Czym jest architektura komputera?**
  Architektura komputera jest projektem orgazniacji i interakcji kluczowych komponentów systemu komputerowego.
- **Najważniejsze architektury komputerowe:**
  - Princeton (von Neumann)
  - Harvard
  - Harvard-Princeton

Architektury systemów komputerowych

Architektura Princeton - wprowadzenie

- Zaproponowana przez Johna von Neumanna w 1945 roku
- Architektura Princeton składa się z:
  - pamięci
  - jednostki arytmetyczno-logicznej ALU
  - jednostki sterującej (CU)
  - urządzeń we-wy (I/O)



EDVAC report, public domain

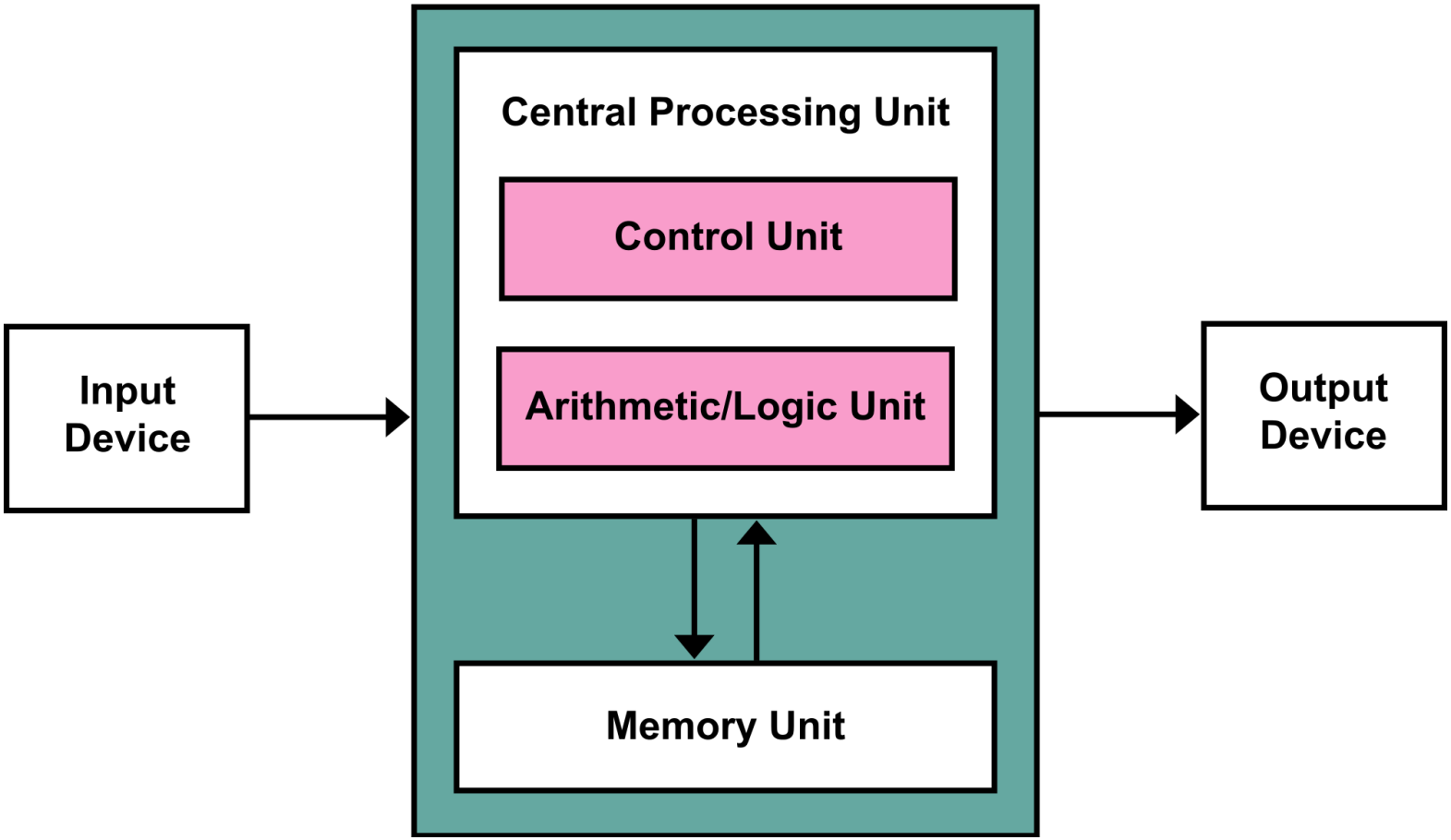
Architektury systemów komputerowych

Architektura Princeton - szczegóły

- **pamięć i urządzenia kontrolowane są przez procesor**
- wszystkie komponenty łączy **szyna systemowa**
- **dane** poruszają się po szynie w trybie **półdupleksu**
- **pamięć** przechowuje **instrukcje i dane** (stored-program concept)

Architektury systemów komputerowych

Architektura Princeton - schemat



Kapooht - Von Neumann Architecture, CC BY-SA 3.0

Architektury systemów komputerowych

Architektura Princeton - zalety i wady

Zalety:

- CU pozyskuje instrukcje i dane w ten sam sposób z jednej pamięci
- dostęp do danych z urządzeń i pamięci realizowany jest także w ten sam sposób
- programiści mogą zorganizować przestrzeń pamięci wedle swojej wizji

Wady:

- szeregowe przetwarzanie instrkcji (brak zrównoleglenia)
- pojedyncza szyna danych jest wąskim gardłem
- skoro instrukcje dzielą pamięc z danymi, mogą być nadpisane

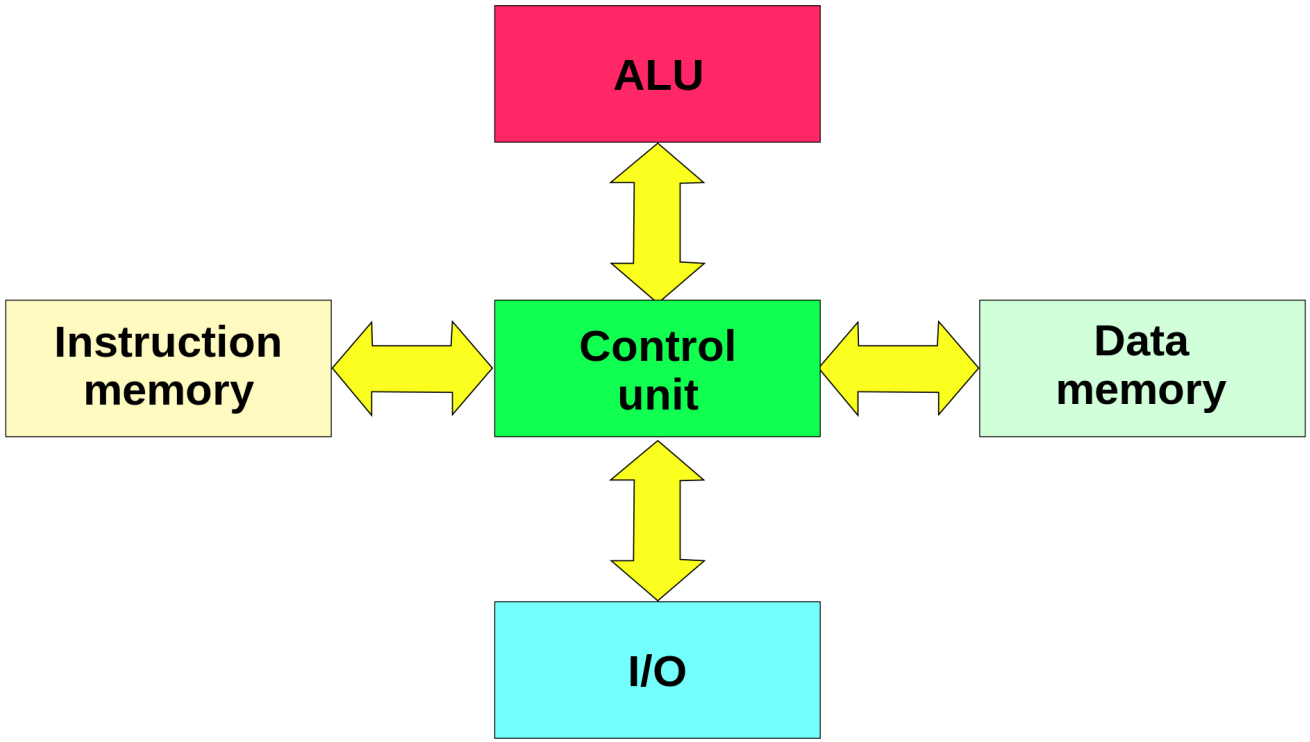
Architektury systemów komputerowych

Architektura Harvard - wprowadzenie

- Architektura Harvard została wprowadzona w komputerze Harvard Mark I (1944)
- podstawowa różnica: **dane i instrukcje są rozłączne**
- dzięki temu można uzyskiwać do nich dostęp **jednocześnie**
- **program nie może sam się zmienić**

Architektury systemów komputerowych

Architektura Harvard - schemat



Architektury systemów komputerowych

Architektura Harvard - zalety i wady

Zalety:

- równoległy dostęp do danych i instrukcji
- dostęp do danych i instrukcji w ten sam sposób
- możliwy różny rozmiar komórek pamięci

Wady:

- wolna pamięć nie może być alokowana dla innego typu pamięci
- droższa, ponieważ potrzebne są dwie szyny danych

Architektury systemów komputerowych

Architektura Harvard-Princeton

- łączy obie architektury próbując redukować ich wady
- **osobne wyższe warstwy hierarchii pamięci** i **wspólne niższe**
- przynajmniej **jeden poziom cache jest osobny dla danych i instrukcji** (pamiętasz wynik działania lstopo?)
- **szybka** dzięki zrównolegleniu tego poziomu pamięci
- pozostawia **elastyczność dla programistów**

Większość komputerów ogólnego przeznaczenia zaprojektowana jest wedle tej architektury (zwanej także zmodyfikowaną architekturą Harvard).

Architektury systemów komputerowych

Podsumowanie

- Architektury Harvard i Princeton różnią się ze względu na sposób dostępu do pamięci aby uzyskać dane i instrukcje
- Harvard posiada osobne szyny dla instrukcji i danych
- Princeton (von Neumann) posiada tylko jedną szynę
- Harvard-Princeton (znodyfikowana architektura Harvard) jest połączeniem dwóch podejść
- jeśli chcesz wiedzieć więcej, zapoznaj się z tym dokumentem

Architektury systemów komputerowych

Źródła i polecane materiały

- P. Dudzik, A. Guzik, *"Architektury komputerów i procesorów"*, AGH University of Science and Technology, Kraków, Poland (materiały do kurs)
- O. Matunga, *"Micro Computer Architecture"* (prezentacja)
- D. Patterson, J. Hennessy, *"Computer Architecture: A Quantitative Approach"*, Elsevier (książka)

Slajd końcowy

Pytania? Komentarze?

Jeśli masz pomysł jak poprawić lub wzbogacić te wykłady,
proszę zgłoś to jako issue w tym repozytorium:

https://github.com/rmhere/lecture-comp-arch-org