

Architektura komputerów

Wykład 1

Wprowadzenie

Wojciech Kordecki

Collegium Witelona
Wydział Nauk Technicznych i Ekonomicznych
Zakład Informatyki

Semestr letni 2023/24



Plan wykładu

- 1 Wprowadzenie.



Plan wykładu

- 1 Wprowadzenie.
- 2 Od 8086 do ... ?



Plan wykładu

- 1 Wprowadzenie.
- 2 Od 8086 do ... ?
- 3 Procesory ARM i smartfony. Raspberry Pi.



Plan wykładu

- 1 Wprowadzenie.
- 2 Od 8086 do ... ?
- 3 Procesory ARM i smartfony. Raspberry Pi.
- 4 Cykl rozkazowy procesora.



Plan wykładu

- 1 Wprowadzenie.
- 2 Od 8086 do ... ?
- 3 Procesory ARM i smartfony. Raspberry Pi.
- 4 Cykl rozkazowy procesora.
- 5 Pamięć.



Plan wykładu

- 1 Wprowadzenie.
- 2 Od 8086 do ... ?
- 3 Procesory ARM i smartfony. Raspberry Pi.
- 4 Cykl rozkazowy procesora.
- 5 Pamięć.
- 6 Układy otoczenia procesora (chipset).



Plan wykładu

- 1 Wprowadzenie.
- 2 Od 8086 do ... ?
- 3 Procesory ARM i smartfony. Raspberry Pi.
- 4 Cykl rozkazowy procesora.
- 5 Pamięć.
- 6 Układy otoczenia procesora (chipset).
- 7 Układy wejścia/wyjścia. DMA.



Plan wykładu

- 1 Wprowadzenie.
- 2 Od 8086 do ... ?
- 3 Procesory ARM i smartfony. Raspberry Pi.
- 4 Cykl rozkazowy procesora.
- 5 Pamięć.
- 6 Układy otoczenia procesora (chipset).
- 7 Układy wejścia/wyjścia. DMA.
- 8 Koprocesory numeryczne.



Plan wykładu

- 1 Wprowadzenie.
- 2 Od 8086 do ... ?
- 3 Procesory ARM i smartfony. Raspberry Pi.
- 4 Cykl rozkazowy procesora.
- 5 Pamięć.
- 6 Układy otoczenia procesora (chipset).
- 7 Układy wejścia/wyjścia. DMA.
- 8 Koprocesory numeryczne.
- 9 Karty rozszerzeń.



Plan wykładu

- 1 Wprowadzenie.
- 2 Od 8086 do ... ?
- 3 Procesory ARM i smartfony. Raspberry Pi.
- 4 Cykl rozkazowy procesora.
- 5 Pamięć.
- 6 Układy otoczenia procesora (chipset).
- 7 Układy wejścia/wyjścia. DMA.
- 8 Koprocesory numeryczne.
- 9 Karty rozszerzeń.
- 10 Łącza i złącza



Plan wykładu

- 1 Wprowadzenie.
- 2 Od 8086 do ... ?
- 3 Procesory ARM i smartfony. Raspberry Pi.
- 4 Cykl rozkazowy procesora.
- 5 Pamięć.
- 6 Układy otoczenia procesora (chipset).
- 7 Układy wejścia/wyjścia. DMA.
- 8 Koprocesory numeryczne.
- 9 Karty rozszerzeń.
- 10 Łącza i złącza
- 11 Procesory 64-bitowe i wielordzeniowe.



Plan wykładu

- 1 Wprowadzenie.
- 2 Od 8086 do ... ?
- 3 Procesory ARM i smartfony. Raspberry Pi.
- 4 Cykl rozkazowy procesora.
- 5 Pamięć.
- 6 Układy otoczenia procesora (chipset).
- 7 Układy wejścia/wyjścia. DMA.
- 8 Koprocesory numeryczne.
- 9 Karty rozszerzeń.
- 10 Łącza i złącza
- 11 Procesory 64-bitowe i wielordzeniowe.
- 12 *Superkomputery.*



Plan wykładu

- 1 Wprowadzenie.
- 2 Od 8086 do ... ?
- 3 Procesory ARM i smartfony. Raspberry Pi.
- 4 Cykl rozkazowy procesora.
- 5 Pamięć.
- 6 Układy otoczenia procesora (chipset).
- 7 Układy wejścia/wyjścia. DMA.
- 8 Koprocesory numeryczne.
- 9 Karty rozszerzeń.
- 10 Łącza i złącza
- 11 Procesory 64-bitowe i wielordzeniowe.
- 12 *Superkomputery.*
- 13 *Komputery kwantowe.*



Plan wykładu

- 1 Wprowadzenie.
- 2 Od 8086 do ... ?
- 3 Procesory ARM i smartfony. Raspberry Pi.
- 4 Cykl rozkazowy procesora.
- 5 Pamięć.
- 6 Układy otoczenia procesora (chipset).
- 7 Układy wejścia/wyjścia. DMA.
- 8 Koprocesory numeryczne.
- 9 Karty rozszerzeń.
- 10 Łącza i złącza
- 11 Procesory 64-bitowe i wielordzeniowe.
- 12 *Superkomputery.*
- 13 *Komputery kwantowe.*
- 14 *Maszyna Turinga.*



Plan wykładu

- 1 Wprowadzenie.
- 2 Od 8086 do ... ?
- 3 Procesory ARM i smartfony. Raspberry Pi.
- 4 Cykl rozkazowy procesora.
- 5 Pamięć.
- 6 Układy otoczenia procesora (chipset).
- 7 Układy wejścia/wyjścia. DMA.
- 8 Koprocesory numeryczne.
- 9 Karty rozszerzeń.
- 10 Łącza i złącza
- 11 Procesory 64-bitowe i wielordzeniowe.
- 12 *Superkomputery.*
- 13 *Komputery kwantowe.*
- 14 *Maszyna Turinga.*
- 15 Podsumowanie.



Wymagania

Wymagania dotyczą wykładu.



Wymagania

Wymagania dotyczą wykładu.
Egzamin pisemny i ustny.



Wymagania

Wymagania dotyczą wykładu.

Egzamin pisemny i ustny.

Zaliczenie laboratorium na ocenę ≥ 4.0 zwalnia z egzaminu z taką samą oceną.

Ocenę można na egzaminie poprawić bez obawy o jej obniżenie.

Szczegółowa lista zagadnień będzie podane na ostatnim wykładzie.



Wymagania

Wymagania dotyczą wykładu.

Egzamin pisemny i ustny.

Zaliczenie laboratorium na ocenę ≥ 4.0 zwalnia z egzaminu z taką samą oceną.

Ocenę można na egzaminie poprawić bez obawy o jej obniżenie.

Szczegółowa lista zagadnień będzie podane na ostatnim wykładzie.

- Podstawy architektury komputerów.



Wymagania

Wymagania dotyczą wykładu.

Egzamin pisemny i ustny.

Zaliczenie laboratorium na ocenę ≥ 4.0 zwalnia z egzaminu z taką samą oceną.

Ocenę można na egzaminie poprawić bez obawy o jej obniżenie.

Szczegółowa lista zagadnień będzie podane na ostatnim wykładzie.

- Podstawy architektury komputerów.
- Zasadnicze układy komputerów



Wymagania

Wymagania dotyczą wykładu.

Egzamin pisemny i ustny.

Zaliczenie laboratorium na ocenę ≥ 4.0 zwalnia z egzaminu z taką samą oceną.

Ocenę można na egzaminie poprawić bez obawy o jej obniżenie.

Szczegółowa lista zagadnień będzie podane na ostatnim wykładzie.

- Podstawy architektury komputerów.
- Zasadnicze układy komputerów
- Podstawy teoretyczne architektury komputerów.



Wymagania

Wymagania dotyczą wykładu.

Egzamin pisemny i ustny.

Zaliczenie laboratorium na ocenę ≥ 4.0 zwalnia z egzaminu z taką samą oceną.

Ocenę można na egzaminie poprawić bez obawy o jej obniżenie.

Szczegółowa lista zagadnień będzie podane na ostatnim wykładzie.

- Podstawy architektury komputerów.
- Zasadnicze układy komputerów
- Podstawy teoretyczne architektury komputerów.
- Układy.



Wymagania

Wymagania dotyczą wykładu.

Egzamin pisemny i ustny.

Zaliczenie laboratorium na ocenę ≥ 4.0 zwalnia z egzaminu z taką samą oceną.

Ocenę można na egzaminie poprawić bez obawy o jej obniżenie.

Szczegółowa lista zagadnień będzie podane na ostatnim wykładzie.

- Podstawy architektury komputerów.
- Zasadnicze układy komputerów
- Podstawy teoretyczne architektury komputerów.
- Układy.
- Peryferia.



Wymagania

Wymagania dotyczą wykładu.

Egzamin pisemny i ustny.

Zaliczenie laboratorium na ocenę ≥ 4.0 zwalnia z egzaminu z taką samą oceną.

Ocenę można na egzaminie poprawić bez obawy o jej obniżenie.

Szczegółowa lista zagadnień będzie podane na ostatnim wykładzie.

- Podstawy architektury komputerów.
- Zasadnicze układy komputerów
- Podstawy teoretyczne architektury komputerów.
- Układy.
- Peryferia.
- Nowe trendy.



Literatura

- [Me] P. Metzger. *Anatomia PC*. wyd. 11, Helion 2007.
- [Ta] A. S. Tanenbaum, H. Bos. *Systemy operacyjne*. Helion 2016.
- [BB] J. G. Brookshear, D. Brylow. *Informatyka w ogólnym zarysie*. PWN 2022.
- [St] W. Stallings. *Organizacja i architektura systemu komputerowego*. Tom 1 i 2, PWN 2022.



Uwagi o literaturze

Książka [Me] – w dużej części przestarzała, zawiera jednak wiele informacji podstawowych i pożytecznych do teraz.

Książki [Ta] oraz [BB] są w wielu fragmentach bliskie temu wykładowi.

Najbliższa tematyce wykładu jest dwutomowa monografia [St] , a zwłaszcza jej drugi tom.



Uwagi o literaturze

Książka [Me] – w dużej części przestarzała, zawiera jednak wiele informacji podstawowych i pożytecznych do teraz.

Książki [Ta] oraz [BB] są w wielu fragmentach bliskie temu wykładowi.

Najbliższa tematyce wykładu jest dwutomowa monografia [St] , a zwłaszcza jej drugi tom.

Ze względu na szybki rozwój techniki, internet jest niezastąpionym źródłem wiedzy. Do wiadomości z tego źródła należy jednak podchodzić bardzo krytycznie.

Źródłem najpewniejszym, choć najtrudniejszym, są dokumentacje techniczne producentów układów.



Najwybitniejsi naukowcy, których prace przyczyniły się do powstania komputerów:

- Blaise Pascal (kalkulator od nazwiska konstruktora zwany Pascaliną, 1642)
- Gottfried Leibniz (system binarny, żywa ława do obliczeń)
- Abraham Stern (maszyna licząca)
- Charles Babbage (maszyna różnicowa, maszyna analityczna)
- Ada Lovelace (prace teoretyczne, wizjonerskie w jej czasach koncepcje wykorzystania komputerów)
- Claude Shannon (teoretyczne podstawy budowy komputerów)
- Alan Turing (teoretyczne podstawy informatyki, maszyna Turinga i uniwersalna maszyna Turinga)

Źródło: Wikipedia

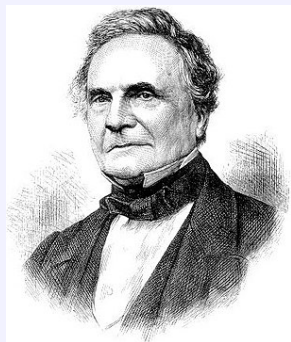


Charles Babbage (1791–1871)

Angielski matematyk, astronom i mechanik, autor tablic logarytmicznych, konstruktor mechanicznych maszyn liczących.

W 1817 roku ukończył kolegium Peterhouse na Uniwersytecie w Cambridge, a potem wykładał w nim nauki ścisłe.

Babbage zajmował się też astronomią, mechaniką, organizacją fabryk. Owładnięty ideą gromadzenia i opracowania liczb i faktów, poświęcił 35 lat życia na konstruowanie maszyn liczących.



Źródło: Wikipedia



Charles Babbage – maszyna różnicowa

Mechaniczne, automatyczne urządzenie, służące do tworzenia tablic matematycznych dla funkcji wielomianowych. Po raz pierwszy zaproponowane zostało w 1822 przez angielskiego matematyka i wynalazcę, Charlesa Babbage'a. W związku ze złożonością projektu, oraz zbyt dużymi wymaganiami finansowymi, konstrukcja ta nie została w pełni zrealizowana za życia jej twórcy. Dopiero osiągnięcie w XX wieku dostatecznie zaawansowanego postępu technologicznego pozwoliło na praktyczne zrealizowanie i potwierdzenie pomysłu Babbage'a. Rozwinięciem idei maszyny różnicowej była zaproponowana przez Charlesa Babbage'a maszyna analityczna będąca mechanicznym odpowiednikiem współczesnych komputerów.

Źródło: Wikipedia



Replika maszyny różnicowej



Źródło: Wikipedia



Charles Babbage – maszyna analityczna (1)

Maszyna analityczna miała być konstrukcją mechaniczną, napędzaną silnikiem parowym. Program oraz dane miały być wprowadzane za pośrednictwem kart perforowanych (pomysł opracowany wcześniej przez Josepha Jacquarda i zastosowany w krośnie jego konstrukcji). Dane wyjściowe miały być drukowane na drukarce, rysowane przy pomocy urządzenia kreślarskiego (prekursora ploterów) lub zapisywane na kartach perforowanych.

Źródło: Wikipedia



Charles Babbage – maszyna analityczna (2)

Konstrukcja maszyny analitycznej zakładała rozdzielenie pamięci („magazynu”, ang. store) i jednostki obliczeniowej („młyna”, ang. mill) – podobnie jak we współczesnych komputerach. Urządzenie pozwalało na wykorzystanie konstrukcji znanych z dzisiejszych języków programowania, takich jak pętle, instrukcje warunkowe czy przetwarzanie równoległe, było więc kompletne w sensie Turinga. Co znamienne, maszyna analityczna – choć nigdy nie skonstruowana – była pierwszym urządzeniem, dla którego zostały napisane programy komputerowe.

Pierwszy z nich, autorstwa (nie jest to pewne) Ady Lovelace, miał obliczać liczby Bernoulliego.

<https://servecom.pl/blog/ada-lovelace-pierwsza-programistka/>

Źródło: *Wikipedia*



Liczby Bernoulliego

Rozwinięcie w szereg Taylora funkcji:

$$\frac{x}{e^x - 1} = \sum_{n=0}^{\infty} \frac{B_n x^n}{n!}.$$

Współczynniki B_n , to liczby Bernoulliego.



Program – liczby Bernoulliego

Diagram for the computation by the Engine of the Numbers of Bernoulli. See Note G. (page 727 of seq.)

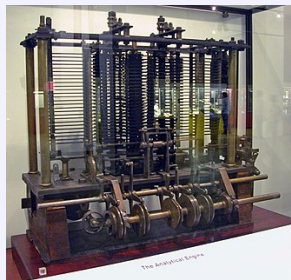
Number of Operation.	Name of Operation.	Variables used upon.	Variables receiving results.	Indication of change in the value on any Variable.	Statement of Results.	Data												Working Variables.												Result Variables.																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																						
						V_1	V_2	V_3	V_4	V_5	V_6	V_7	V_8	V_9	V_{10}	V_{11}	V_{12}	V_{13}	V_{14}	V_{15}	V_{16}	V_{17}	V_{18}	V_{19}	V_{20}	V_{21}	V_{22}	V_{23}	V_{24}	V_{25}	V_{26}	V_{27}	V_{28}	V_{29}	V_{30}	V_{31}	V_{32}	V_{33}	V_{34}	V_{35}	V_{36}	V_{37}	V_{38}	V_{39}	V_{40}	V_{41}	V_{42}	V_{43}	V_{44}	V_{45}	V_{46}	V_{47}	V_{48}	V_{49}	V_{50}	V_{51}	V_{52}	V_{53}	V_{54}	V_{55}	V_{56}	V_{57}	V_{58}	V_{59}	V_{60}	V_{61}	V_{62}	V_{63}	V_{64}	V_{65}	V_{66}	V_{67}	V_{68}	V_{69}	V_{70}	V_{71}	V_{72}	V_{73}	V_{74}	V_{75}	V_{76}	V_{77}	V_{78}	V_{79}	V_{80}	V_{81}	V_{82}	V_{83}	V_{84}	V_{85}	V_{86}	V_{87}	V_{88}	V_{89}	V_{90}	V_{91}	V_{92}	V_{93}	V_{94}	V_{95}	V_{96}	V_{97}	V_{98}	V_{99}	V_{100}	V_{101}	V_{102}	V_{103}	V_{104}	V_{105}	V_{106}	V_{107}	V_{108}	V_{109}	V_{110}	V_{111}	V_{112}	V_{113}	V_{114}	V_{115}	V_{116}	V_{117}	V_{118}	V_{119}	V_{120}	V_{121}	V_{122}	V_{123}	V_{124}	V_{125}	V_{126}	V_{127}	V_{128}	V_{129}	V_{130}	V_{131}	V_{132}	V_{133}	V_{134}	V_{135}	V_{136}	V_{137}	V_{138}	V_{139}	V_{140}	V_{141}	V_{142}	V_{143}	V_{144}	V_{145}	V_{146}	V_{147}	V_{148}	V_{149}	V_{150}	V_{151}	V_{152}	V_{153}	V_{154}	V_{155}	V_{156}	V_{157}	V_{158}	V_{159}	V_{160}	V_{161}	V_{162}	V_{163}	V_{164}	V_{165}	V_{166}	V_{167}	V_{168}	V_{169}	V_{170}	V_{171}	V_{172}	V_{173}	V_{174}	V_{175}	V_{176}	V_{177}	V_{178}	V_{179}	V_{180}	V_{181}	V_{182}	V_{183}	V_{184}	V_{185}	V_{186}	V_{187}	V_{188}	V_{189}	V_{190}	V_{191}	V_{192}	V_{193}	V_{194}	V_{195}	V_{196}	V_{197}	V_{198}	V_{199}	V_{200}	V_{201}	V_{202}	V_{203}	V_{204}	V_{205}	V_{206}	V_{207}	V_{208}	V_{209}	V_{210}	V_{211}	V_{212}	V_{213}	V_{214}	V_{215}	V_{216}	V_{217}	V_{218}	V_{219}	V_{220}	V_{221}	V_{222}	V_{223}	V_{224}	V_{225}	V_{226}	V_{227}	V_{228}	V_{229}	V_{230}	V_{231}	V_{232}	V_{233}	V_{234}	V_{235}	V_{236}	V_{237}	V_{238}	V_{239}	V_{240}	V_{241}	V_{242}	V_{243}	V_{244}	V_{245}	V_{246}	V_{247}	V_{248}	V_{249}	V_{250}	V_{251}	V_{252}	V_{253}	V_{254}	V_{255}	V_{256}	V_{257}	V_{258}	V_{259}	V_{260}	V_{261}	V_{262}	V_{263}	V_{264}	V_{265}	V_{266}	V_{267}	V_{268}	V_{269}	V_{270}	V_{271}	V_{272}	V_{273}	V_{274}	V_{275}	V_{276}	V_{277}	V_{278}	V_{279}	V_{280}	V_{281}	V_{282}	V_{283}	V_{284}	V_{285}	V_{286}	V_{287}	V_{288}	V_{289}	V_{290}	V_{291}	V_{292}	V_{293}	V_{294}	V_{295}	V_{296}	V_{297}	V_{298}	V_{299}	V_{300}	V_{301}	V_{302}	V_{303}	V_{304}	V_{305}	V_{306}	V_{307}	V_{308}	V_{309}	V_{310}	V_{311}	V_{312}	V_{313}	V_{314}	V_{315}	V_{316}	V_{317}	V_{318}	V_{319}	V_{320}	V_{321}	V_{322}	V_{323}	V_{324}	V_{325}	V_{326}	V_{327}	V_{328}	V_{329}	V_{330}	V_{331}	V_{332}	V_{333}	V_{334}	V_{335}	V_{336}	V_{337}	V_{338}	V_{339}	V_{340}	V_{341}	V_{342}	V_{343}	V_{344}	V_{345}	V_{346}	V_{347}	V_{348}	V_{349}	V_{350}	V_{351}	V_{352}	V_{353}	V_{354}	V_{355}	V_{356}	V_{357}	V_{358}	V_{359}	V_{360}	V_{361}	V_{362}	V_{363}	V_{364}	V_{365}	V_{366}	V_{367}	V_{368}	V_{369}	V_{370}	V_{371}	V_{372}	V_{373}	V_{374}	V_{375}	V_{376}	V_{377}	V_{378}	V_{379}	V_{380}	V_{381}	V_{382}	V_{383}	V_{384}	V_{385}	V_{386}	V_{387}	V_{388}	V_{389}	V_{390}	V_{391}	V_{392}	V_{393}	V_{394}	V_{395}	V_{396}	V_{397}	V_{398}	V_{399}	V_{400}	V_{401}	V_{402}	V_{403}	V_{404}	V_{405}	V_{406}	V_{407}	V_{408}	V_{409}	V_{410}	V_{411}	V_{412}	V_{413}	V_{414}	V_{415}	V_{416}	V_{417}	V_{418}	V_{419}	V_{420}	V_{421}	V_{422}	V_{423}	V_{424}	V_{425}	V_{426}	V_{427}	V_{428}	V_{429}	V_{430}	V_{431}	V_{432}	V_{433}	V_{434}	V_{435}	V_{436}	V_{437}	V_{438}	V_{439}	V_{440}	V_{441}	V_{442}	V_{443}	V_{444}	V_{445}	V_{446}	V_{447}	V_{448}	V_{449}	V_{450}	V_{451}	V_{452}	V_{453}	V_{454}	V_{455}	V_{456}	V_{457}	V_{458}	V_{459}	V_{460}	V_{461}	V_{462}	V_{463}	V_{464}	V_{465}	V_{466}	V_{467}	V_{468}	V_{469}	V_{470}	V_{471}	V_{472}	V_{473}	V_{474}	V_{475}	V_{476}	V_{477}	V_{478}	V_{479}	V_{480}	V_{481}	V_{482}	V_{483}	V_{484}	V_{485}	V_{486}	V_{487}	V_{488}	V_{489}	V_{490}	V_{491}	V_{492}	V_{493}	V_{494}	V_{495}	V_{496}	V_{497}	V_{498}	V_{499}	V_{500}	V_{501}	V_{502}	V_{503}	V_{504}	V_{505}	V_{506}	V_{507}	V_{508}	V_{509}	V_{510}	V_{511}	V_{512}	V_{513}	V_{514}	V_{515}	V_{516}	V_{517}	V_{518}	V_{519}	V_{520}	V_{521}	V_{522}	V_{523}	V_{524}	V_{525}	V_{526}	V_{527}	V_{528}	V_{529}	V_{530}	V_{531}	V_{532}	V_{533}	V_{534}	V_{535}	V_{536}	V_{537}	V_{538}	V_{539}	V_{540}	V_{541}	V_{542}	V_{543}	V_{544}	V_{545}	V_{546}	V_{547}	V_{548}	V_{549}	V_{550}	V_{551}	V_{552}	V_{553}	V_{554}	V_{555}	V_{556}	V_{557}	V_{558}	V_{559}	V_{560}	V_{561}	V_{562}	V_{563}	V_{564}	V_{565}	V_{566}	V_{567}	V_{568}	V_{569}	V_{570}	V_{571}	V_{572}	V_{573}	V_{574}	V_{575}	V_{576}	V_{577}	V_{578}	V_{579}	V_{580}	V_{581}	V_{582}	V_{583}	V_{584}	V_{585}	V_{586}	V_{587}	V_{588}	V_{589}	V_{590}	V_{591}	V_{592}	V_{593}	V_{594}	V_{595}	V_{596}	V_{597}	V_{598}	V_{599}	V_{600}	V_{601}	V_{602}	V_{603}	V_{604}	V_{605}	V_{606}	V_{607}	V_{608}	V_{609}	V_{610}	V_{611}	V_{612}	V_{613}	V_{614}	V_{615}	V_{616}	V_{617}	V_{618}	V_{619}	V_{620}	V_{621}	V_{622}	V_{623}	V_{624}	V_{625}	V_{626}	V_{627}	V_{628}	V_{629}	V_{630}	V_{631}	V_{632}	V_{633}	V_{634}	V_{635}	V_{636}	V_{637}	V_{638}	V_{639}	V_{640}	V_{641}	V_{642}	V_{643}	V_{644}	V_{645}	V_{646}	V_{647}	V_{648}	V_{649}	V_{650}	V_{651}	V_{652}	V_{653}	V_{654}	V_{655}	V_{656}	V_{657}	V_{658}	V_{659}	V_{660}	V_{661}	V_{662}	V_{663}	V_{664}	V_{665}	V_{666}	V_{667}	V_{668}	V_{669}	V_{670}	V_{671}	V_{672}	V_{673}	V_{674}	V_{675}	V_{676}	V_{677}	V_{678}	V_{679}	V_{680}	V_{681}	V_{682}	V_{683}	V_{684}	V_{685}	V_{686}	V_{687}	V_{688}	V_{689}	V_{690}	V_{691}	V_{692}	V_{693}	V_{694}	V_{695}	V_{696}	V_{697}	V_{698}	V_{699}	V_{700}	V_{701}	V_{702}	V_{703}	V_{704}	V_{705}	V_{706}	V_{707}	V_{708}	V_{709}	V_{710}	V_{711}	V_{712}	V_{713}	V_{714}	V_{715}	V_{716}	V_{717}	V_{718}	V_{719}	V_{720}	V_{721}	V_{722}	V_{723}	V_{724}	V_{725}	V_{726}	V_{727}	V_{728}	V_{729}	V_{730}	V_{731}	V_{732}	V_{733}	V_{734}	V_{735}	V_{736}	V_{737}	V_{738}	V_{739}	V_{740}	V_{741}	V_{742}	V_{743}	V_{744}	V_{745}	V_{746}	V_{747}	V_{748}	V_{749}	V_{750}	V_{751}	V_{752}	V_{753}	V_{754}	V_{755}	V_{756}	V_{757}	V_{758}	V_{759}	V_{760}	V_{761}	V_{762}	V_{763}	V_{764}	V_{765}	V_{766}	V_{767}	V_{768}	V_{769}	V_{770}	V_{771}	V_{772}	V_{773}	V_{774}	V_{775}	V_{776}	V_{777}	V_{778}	V_{779}	V_{780}	V_{781}	V_{782}	V_{783}	V_{784}	V_{785}	V_{786}	V_{787}	V_{788}	V_{789}	V_{790}	V_{791}	V_{792}	V_{793}	V_{794}	V_{795}	V_{796}	V_{797}	V_{798}	V_{799}	V_{800}	V_{801}	V_{802}	V_{803}	V_{804}	V_{805}	V_{806}	V_{807}	V_{808}	V_{809}	V_{810}	V_{811}	V_{812}	V_{813}	V_{814}	V_{815}	V_{816}	V_{817}	V_{818}	V_{819}	V_{820}	V_{821}	V_{822}	V_{823}	V_{824}	V_{825}	V_{826}	V_{827}	V_{828}	V_{829}	V_{830}	V_{831}	V_{832}	V_{833}	V_{834}	V_{835}	V_{836}	V_{837}	V_{838}	V_{839}	V_{840}	V_{841}	V_{842}	V_{843}	V_{844}	V_{845}	V_{846}	V_{847}	V_{848}	V_{849}	V_{850}	V_{851}	V_{852}	V_{853}	V_{854}	V_{855}	V_{856}	V_{857}	V_{858}	V_{859}	V_{860}	V_{861}	V_{862}	V_{863}	V_{864}	V_{865}	V_{866}	V_{867}	V_{868}	V_{869}	V_{870}	V_{871}	V_{872}	V_{873}	V_{874}	V_{875}	V_{876}	V_{877}	V_{878}	V_{879}	V_{880}	V_{881}	V_{882}	V_{883}	V_{884}	V_{885}	V_{886}	V_{887}	V_{888}	V_{889}	V_{890}	V_{891}	V_{892}	V_{893}	V_{894}	V_{895}	V_{896}	V_{897}	V_{898}	V_{899}	V_{900}	V_{901}	V_{902}	V_{903}	V_{904}	V_{905}	V_{906}	V_{907}	V_{908}	V_{909}	V_{910}	V_{911}	V_{912}	V_{913}	V_{914}	V_{915}	V_{916}	V_{917}	V_{918}	V_{919}	V_{920}	V_{921}	V_{922}	V_{923}	V_{924}	V_{925}	V_{926}	V_{927}

Maszyna analityczna – model próbny

Babbage przedstawił pierwszy projekt swojego urządzenia w 1837 roku i rozwijał go aż do śmierci w 1871 roku. Ze względu na brak środków finansowych maszyna nie została zbudowana.

W 2011 grupa z Wielkiej Brytanii ogłosiła 10-letni projekt rekonstrukcji maszyny.

Źródło: Wikipedia



W 2010 r. ogłoszono projekt o nazwie Plan 28, w ramach którego grupa pasjonatów podjęła się trudnego zadania budowy całej konstrukcji na bazie planów Babbage'a. (...) Dopiero w grudniu 2021 r. projekt osiągnął długo oczekiwany punkt definitywny. Ukończono pierwsze szkice, zawierające opis projektów silnika analitycznego, a także analizę specyfiki wdrożenia.

Projekt Plan 28



Teraźniejszość – codzienność

Omówimy ją przy okazji na następnych, zdalnych wykładach 2 – 11.



Teraźniejszość – codzienność

Omówimy ją przy okazji na następnych, zdalnych wykładach 2 – 11.

- PC-ty.
- Laptopy
- Tablety.
- Smartfony



Moc obliczeniowa

FLOPS (od ang. floating point operations per second, operacje zmiennoprzecinkowe na sekundę) jednostka mocy obliczeniowej komputerów, używana szczególnie w zastosowaniach naukowych.



Moc obliczeniowa

FLOPS (od ang. floating point operations per second, operacje zmiennoprzecinkowe na sekundę) jednostka mocy obliczeniowej komputerów, używana szczególnie w zastosowaniach naukowych. Dawniej MIPS – (million instructions per second) – liczba rozkazów procesora wykonanych na sekundę.



Moc obliczeniowa

FLOPS (od ang. floating point operations per second, operacje zmiennoprzecinkowe na sekundę) jednostka mocy obliczeniowej komputerów, używana szczególnie w zastosowaniach naukowych. Dawniej MIPS – (million instructions per second) – liczba rozkazów procesora wykonanych na sekundę.

Mnożniki

Nazwa	Skrót	Wartość
Kilo	K	1024
Mega	M	1024^2
Giga	G	1024^3
Tera	T	1024^4
Peta	P	1024^5
Exa	E	1024^6



Teraźniejszość – superkomputery

Superkomputery Wikipedia



Teraźniejszość – superkomputery

Superkomputery Wikipedia

2016:

[Sunway TaihuLight](#) (Wuxi,ChRL) – moc obliczeniowa 93 PFLOPS.



Teraźniejszość – superkomputery

Superkomputery Wikipedia

2016:

[Sunway TaihuLight](#) (Wuxi, ChRL) – moc obliczeniowa 93 PFLOPS.

2018:

[Summit](#) (IBM, USA) – moc obliczeniowa 200 petaflopsów (200 PFLOPS). Pobiera 15 MW mocy.



Teraźniejszość – superkomputery

Superkomputery Wikipedia

2016:

[Sunway TaihuLight](#) (Wuxi, ChRL) – moc obliczeniowa 93 PFLOPS.

2018:

[Summit](#) (IBM, USA) – moc obliczeniowa 200 petaflopsów (200 PFLOPS). Pobiera 15 MW mocy.

2020:

[Fugaku](#), (Japonia, Fujitsu) – moc obliczeniowa 416 PFLOPS.



Terażniejszość – superkomputery

Superkomputery Wikipedia

2016:

[Sunway TaihuLight](#) (Wuxi, ChRL) – moc obliczeniowa 93 PFLOPS.

2018:

[Summit](#) (IBM, USA) – moc obliczeniowa 200 petaflopsów (200 PFLOPS). Pobiera 15 MW mocy.

2020:

[Fugaku](#), (Japonia, Fujitsu) – moc obliczeniowa 416 PFLOPS.

2022:

[Frontier](#) (USA, Cray) – moc obliczeniowa 1102 PFLOPS.



Najbliższa przyszłość – Jupiter

JUPITER (Joint Undertaking Pioneer for Innovative and Transformative Exascale Research) zostanie uruchomiony w 2024 r. w Centrum Superkomputerowym w Jülich w Niemczech przez EuroHPC JU.

(...)

Jednostka ma być oparta na niestandardowych chipach bazujących na architekturze ARM, spiętych w około 6 tys. węzłów obliczeniowych. Platforma wykorzysta superwydajne jednostki obliczeniowe Nvidia GH100, z rdzeniami CPU Nvidia Grace oraz układem graficznym Hopper. Każdy węzeł będzie liczył cztery zestawy superchipów GH200, czyli 288 rdzeni ARM.

Komputer Świat



Nagroda Nobla z fizyki

Laureatami Nagrody Nobla z fizyki w 2022 r. zostali: Alain Aspect, John F. Clauser i Anton Zeilinger. To naukowcy, którzy po raz pierwszy eksperymentalnie potwierdzili „dziwactwa” mechaniki kwantowej i byli pionierami informatyki kwantowej.

<https://wyborcza.pl/7,75400,28983046,nagroda-nobla-z-fizyki-za-potege-mechaniki-kwantowej.html#S.TD-K.C-B.4-L.1.duzy>



Przyszłość – komputery kwantowe (1)

Podstawowymi elementami budowy kwantowego komputera są kwantowe bramki logiczne. Kwantowy bit, tzw. *kubit*, zgodnie z prawami mikroświata nie będzie miał ustalonej wartości 1 lub 0, tak jak bit w standardowym komputerze. W trakcie obliczeń będzie się znajdował w jakimś stanie pośrednim. Rządzi tym prawo prawdopodobieństwa, podobnie jak położeniem elektronu w atomie. Kubit jest kwantową superpozycją zera i jedynki. Pojedynczy wynik obliczeń komputera kwantowego będzie niepewny. Istotne staje się wykonanie całej serii obliczeń i dopiero ich średnia wartość z dużą dokładnością określi prawidłowy wynik – tym dokładniejszy, im więcej komputer dokona obliczeń. Kubit niesie w sobie naraz o wiele więcej informacji niż zero-jedynkowy bit. Dlatego jest w stanie wykonać równolegle wiele obliczeń.

Źródło: Wikipedia



Przyszłość – komputery kwantowe (2)

Rejestr kwantowy to np. zespół atomów, z których każdy realizuje jeden z kubitów. Każdy ciąg zer i jedynek, o długości równej rozmiarom rejestru, daje się zapisać w kubitach tego układu (tak samo jak w komórkach pamięci rejestru konwencjonalnego, ale w rejestrze takim w danej chwili może być zapisany tylko jeden ciąg zero-jedynkowy). Rejestr kwantowy, jako złożony z kubitów, może być w stanie będącym dowolną superpozycją wielu ciągów zero-jedynkowych. Jeśli w takim rejestrze kwantowym zapisana by została jakaś duża baza danych, wykonanie pewnej operacji na kubitach tego rejestru byłoby równoznaczne z wykonaniem tej operacji na wszystkich danych naraz.

Źródło: Wikipedia



Komputer jaki jest . . .



Komputer jaki jest . . .

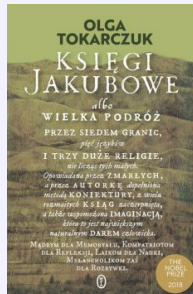
Benedykt Chmielowski, Nowe Ateny, Lwów 1745, t. I, s. 475.
Pierwsza polska encyklopedia.



Komputer jaki jest ...

Benedykt Chmielowski, Nowe Ateny, Lwów 1745, t. I, s. 475.
Pierwsza polska encyklopedia.

Od historii jej tworzenia zaczyna się
powieść Olgi Tokarczuk:
*Księgi Jakubowe albo Wielka podróż
przez siedem granic, pięć języków i trzy
duże religie, nie licząc tych małych.*

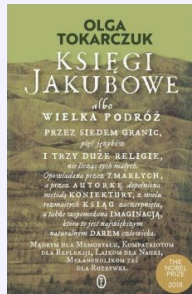


Komputer jaki jest ...

Benedykt Chmielowski, Nowe Ateny, Lwów 1745, t. I, s. 475.
Pierwsza polska encyklopedia.

Od historii jej tworzenia zaczyna się
powieść Olgi Tokarczuk:
*Księgi Jakubowe albo Wielka podróż
przez siedem granic, pięć języków i trzy
duże religie, nie licząc tych małych.*

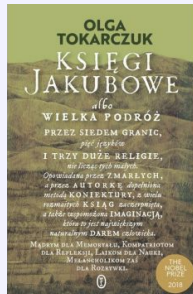
Początek hasła: *Koń jaki jest, każdy widzi.*



Komputer jaki jest ...

Benedykt Chmielowski, Nowe Ateny, Lwów 1745, t. I, s. 475.
Pierwsza polska encyklopedia.

Od historii jej tworzenia zaczyna się
powieść Olgi Tokarczuk:
*Księgi Jakubowe albo Wielka podróż
przez siedem granic, pięć języków i trzy
duże religie, nie licząc tych małych.*



Początek hasła: *Koń jaki jest, każdy widzi.*

Komputer jaki jest, każdy widzi, ale szczegóły będzie musiał
poznać i widzieć na koniec semestru – weryfikacja na egzaminie.



PC, laptop

W roku 1981 firma IBM wprowadziła na rynek swój pierwszy komputer stacjonarny – komputer osobisty lub PC. Opierał się na szesnastobitowym procesorze firmy Intel.

Podstawowe oprogramowanie opracowała nowo utworzona forma Microsoft.



PC, laptop

W roku 1981 firma IBM wprowadziła na rynek swój pierwszy komputer stacjonarny – komputer osobisty lub PC. Opierał się na szesnastobitowym procesorze firmy Intel.

Podstawowe oprogramowanie opracowała nowo utworzona forma Microsoft.

Obecnie PC-ty, w wersjach przenośnych laptopy, są produkowane przez różne firmy. Większość z oprogramowaniem firmy Microsoft. Architektura takich komputerów będzie zasadniczą częścią wykładu.



Smartfon

Smartfony, to zminiaturyzowane komputery, oparte najczęściej na procesorach nieintelowskich, zwykle z rodziny ARM.



Smartfon

Smartfony, to zminiaturyzowane komputery, oparte najczęściej na procesorach nieintelowskich, zwykle z rodziny ARM.

Jedną z wielu funkcji smartfonów jest telefonia.



Smartfon

Smartfony, to zminiaturyzowane komputery, oparte najczęściej na procesorach nieintelowskich, zwykle z rodziny ARM.

Jedną z wielu funkcji smartfonów jest telefonia.

Moc obliczeniowa obecnych smartfonów jest większa niż dawniejszych superkomputerów. Przykładowo iPhone XS (rok 2018) w teście LINPACK uzyskuje wynik 7.5 GFLOPS, co w 1993 roku stawiałoby go na 30 miejscu wśród najszybszych superkomputerów świata.



Elementy komputera

- Procesor
- Magistrale
- Chipset
- Płyta główna
- Pamięć EPROM
- Pamięć RAM
- Pamięć masowa
- Zegar
- Porty
- Inne urządzenia wewnętrzne



Architektura harwardzka

Pamięć danych programu jest oddzielona od pamięci rozkazów.

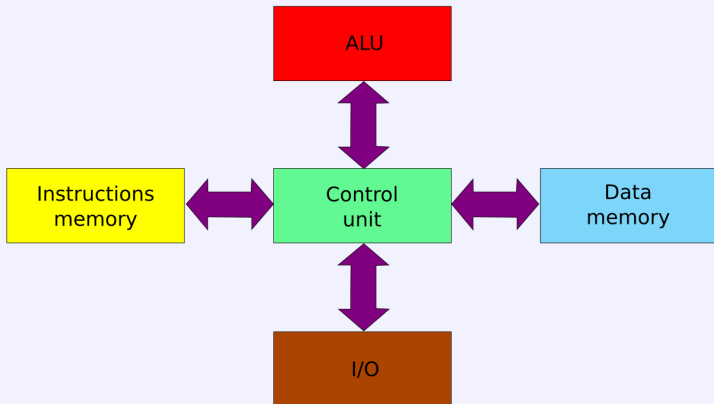
Podstawowa architektura komputerów zerowej generacji i początkowa komputerów pierwszej generacji.

Prostsza w stosunku do architektury von Neumanna struktura, powoduje większą szybkość działania.

Oddzielenie pamięci danych od pamięci rozkazów powoduje, że architektura harwardzka jest obecnie powszechnie stosowana w mikrokontrolerach jednoukładowych, w których dane programu są zapisane w nieulotnej pamięci ROM (EPROM/EEPROM) lub flash, a dane tymczasowe są umieszczane w pamięci RAM.



Architektura harwardzka – schemat



Źródło – internet

Architektura von Neumanna

Architektura opracowana przez Johna von Neumanna, Johna W. Mauchly'ego oraz Johna Presper Eckerta w 1945 roku. Dane przechowywane są wspólnie z instrukcjami.



Architektura von Neumanna

Architektura opracowana przez Johna von Neumanna, Johna W. Mauchly'ego oraz Johna Presper Eckerta w 1945 roku. Dane przechowywane są wspólnie z instrukcjami.

Komponenty.

- Pamięć komputerowa przechowująca dane programu oraz instrukcje programu; każda komórka pamięci ma adres.



Architektura von Neumanna

Architektura opracowana przez Johna von Neumanna, Johna W. Mauchly'ego oraz Johna Presper Eckerta w 1945 roku. Dane przechowywane są wspólnie z instrukcjami.

Komponenty.

- Pamięć komputerowa przechowująca dane programu oraz instrukcje programu; każda komórka pamięci ma adres.
- Jednostka sterująca odpowiedzialna za pobieranie danych i instrukcji z pamięci oraz ich sekwencyjne przetwarzanie.



Architektura von Neumanna

Architektura opracowana przez Johna von Neumanna, Johna W. Mauchly'ego oraz Johna Presper Eckerta w 1945 roku. Dane przechowywane są wspólnie z instrukcjami.

Komponenty.

- Pamięć komputerowa przechowująca dane programu oraz instrukcje programu; każda komórka pamięci ma adres.
- Jednostka sterująca odpowiedzialna za pobieranie danych i instrukcji z pamięci oraz ich sekwencyjne przetwarzanie.
- Jednostka arytmetyczno-logiczna odpowiedzialna za wykonywanie podstawowych operacji arytmetycznych.



Architektura von Neumanna

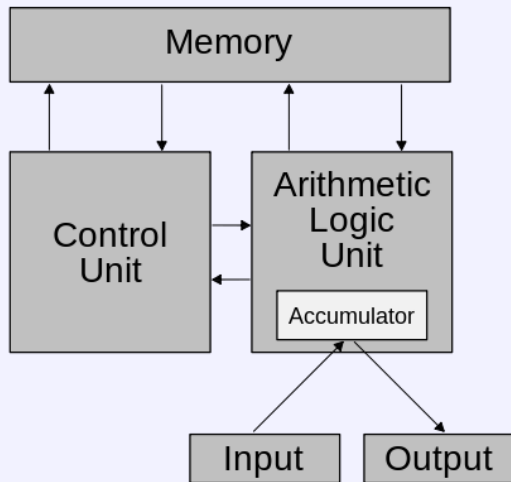
Architektura opracowana przez Johna von Neumanna, Johna W. Mauchly'ego oraz Johna Presper Eckerta w 1945 roku. Dane przechowywane są wspólnie z instrukcjami.

Komponenty.

- Pamięć komputerowa przechowująca dane programu oraz instrukcje programu; każda komórka pamięci ma adres.
- Jednostka sterująca odpowiedzialna za pobieranie danych i instrukcji z pamięci oraz ich sekwencyjne przetwarzanie.
- Jednostka arytmetyczno-logiczna odpowiedzialna za wykonywanie podstawowych operacji arytmetycznych.
- Urządzenia wejścia/wyjścia służące do interakcji z operatorem.



Architektura von Neumanna – schemat



System komputerowy

Control Unit + Arithmetic Logic Unit = Processor.



System komputerowy

Control Unit + Arithmetic Logic Unit = Processor.

Pamięć programu i danych jest wspólna.



System komputerowy

Control Unit + Arithmetic Logic Unit = Processor.

Pamięć programu i danych jest wspólna.

W architekturze von Neumanna system ma:



System komputerowy

Control Unit + Arithmetic Logic Unit = Processor.

Pamięć programu i danych jest wspólna.

W architekturze von Neumanna system ma:

- skończony i pełny zbiór rozkazów,



System komputerowy

Control Unit + Arithmetic Logic Unit = Processor.

Pamięć programu i danych jest wspólna.

W architekturze von Neumanna system ma:

- skończony i pełny zbiór rozkazów,
- program wprowadzony do pamięci komputera tak jak danych,



System komputerowy

Control Unit + Arithmetic Logic Unit = Processor.

Pamięć programu i danych jest wspólna.

W architekturze von Neumanna system ma:

- skończony i pełny zbiór rozkazów,
- program wprowadzony do pamięci komputera tak jak danych,
- instrukcje i dane jednakowo dostępne dla procesora.



System komputerowy

Control Unit + Arithmetic Logic Unit = Processor.

Pamięć programu i danych jest wspólna.

W architekturze von Neumanna system ma:

- skończony i pełny zbiór rozkazów,
- program wprowadzony do pamięci komputera tak jak danych,
- instrukcje i dane jednakowo dostępne dla procesora.
- przetwarzanie informacji przez sekwencyjne wykonywanie instrukcji.



John von Neumann (1903–1957)

John von Neumann – węgierski matematyk, inżynier chemik, fizyk i informatyk, pracujący głównie w Stanach Zjednoczonych. Wniósł znaczący wkład do wielu dziedzin matematyki – w szczególności był głównym twórcą teorii gier, teorii automatów komórkowych (w które pewien początkowy wkład miał także Stanisław Ulam) i stworzył formalizm matematyczny mechaniki kwantowej. Uczestniczył w projekcie Manhattan. Przyczynił się do rozwoju numerycznych prognoz pogody.

Źródło – Wikipedia



Plagiat?

Mauchly i Eckert oskarżali von Neumanna o przywłaszczenie wspólnych wyników prac. Sam von Neumann uważał ich pracę za dzieło zbiorowe o licencji public domain (projekt był współfinansowany przez rząd amerykański).

Źródło – Wikipedia



Procesory CISC – Complex Instruction Set Computer

W architekturze CISC pojedynczy rozkaz mikroprocesora wykonuje kilka operacji niskiego poziomu – przykład pobranie z pamięci, operację arytmetyczną i zapisanie do pamięci.



Procesory CISC – Complex Instruction Set Computer

W architekturze CISC pojedynczy rozkaz mikroprocesora wykonuje kilka operacji niskiego poziomu – przykład pobranie z pamięci, operację arytmetyczną i zapisanie do pamięci.

Przykłady rodzin procesorów o architekturze CISC:

- IBM System/360
- VAX
- PDP-11
- x86



Historia rodziny x86

Wprowadzony w roku 1987 procesor 8086 z zegarem 5 MHz miał 29 000 tranzystorów. Wprowadzony w roku 2013 sześciordzeniowy Core i7 EE 4960X pracuje z częstotliwością 4 GHz i ma 1.86 miliarda tranzystorów ([St] str.26).

Do tej historii będziemy wracać w ciągu semestru.



Procesory RISC – Reduced Instruction Set Computer

- Liczba rozkazów zredukowana do niezbędnego minimum.
- Zredukowane tryby adresowania.
- Ograniczenie komunikacji pomiędzy pamięcią a procesorem, tylko load i store. Inne rozkazy tylko na rejestrach.
- Zwiększenie liczby rejestrów.



Procesory RISC – Reduced Instruction Set Computer

- Liczba rozkazów zredukowana do niezbędnego minimum.
- Zredukowane tryby adresowania.
- Ograniczenie komunikacji pomiędzy pamięcią a procesorem, tylko load i store. Inne rozkazy tylko na rejestrach.
- Zwiększenie liczby rejestrów.

Współczesne procesory zgodne z x86 produkowane przez firmy Intel, AMD i VIA przetwarzają rozkazy procesora x86 na proste mikropolecenia pracujące według idei RISC, często wykonujące się równolegle.

Z punktu widzenia programisty rozkazy są widziane jako CISC, ale ich rdzeń jest RISC-owy. Rozkazy CISC są rozbijane na mikrorozkazy (ang. microops), które są następnie wykonywane przez RISC-owy blok wykonawczy.



Rodzina ARM

Architektura ARM (Advanced RISC Machine, pierwotnie Acorn RISC Machine) – 32-bitowa oraz 64-bitowa (Apple A7, 2013 r.) architektura (model programowy) procesorów typu RISC.

Różne wersje rdzeni ARM są szeroko stosowane w systemach wbudowanych i systemach o niskim poborze mocy, ze względu na ich energooszczędną architekturę.



Rodzina ARM

Architektura ARM (Advanced RISC Machine, pierwotnie Acorn RISC Machine) – 32-bitowa oraz 64-bitowa (Apple A7, 2013 r.) architektura (model programowy) procesorów typu RISC.

Różne wersje rdzeni ARM są szeroko stosowane w systemach wbudowanych i systemach o niskim poborze mocy, ze względu na ich energooszczędną architekturę.

Procesory z architekturą ARM są jednymi z najczęściej stosowanych procesorów na świecie. Używa się ich między innymi w dyskach twardych, telefonach komórkowych, routerach, kalkulatorach, a nawet w zabawkach dziecięcych. Obecnie zajmują one ponad 75% rynku 32-bitowych CPU dla systemów wbudowanych.



RISC-V

RISC-V – otwarty model programowy procesora (ISA – instruction set architecture) oparty o zasady RISC.

Projekt RISC-V ISA

Źródło – Wikipedia

Botland – przewodnik

RISC-V vs ARM

