Hardwarové technologie

[Principy počítačů 3](#_Toc162464181)

[Historický vývoj 3](#_Toc162464182)

[První počítače 3](#_Toc162464183)

[První univerzální počítač 3](#_Toc162464184)

[První číslicový počítač 3](#_Toc162464185)

[Generace počítačů 3](#_Toc162464186)

[Binární logika 4](#_Toc162464187)

[AND 4](#_Toc162464188)

[NAND 4](#_Toc162464189)

[OR 5](#_Toc162464190)

[NOR 5](#_Toc162464191)

[XOR 5](#_Toc162464192)

[Modulace signálu 5](#_Toc162464193)

[Útlum 5](#_Toc162464194)

[Šum 5](#_Toc162464195)

[Druhy modulace 5](#_Toc162464196)

[Architektura počítače 7](#_Toc162464197)

[Von Neumannovo a Harvardské schéma 7](#_Toc162464198)

[Von Neumannova architektura 7](#_Toc162464199)

[Harvardská architektura 8](#_Toc162464200)

[Flynnova taxonomie 8](#_Toc162464201)

[SI 8](#_Toc162464202)

[MI 9](#_Toc162464203)

[SD 9](#_Toc162464204)

[MD 9](#_Toc162464205)

[Kombinace 9](#_Toc162464206)

[Základní deska 9](#_Toc162464207)

[Ovlivňuje 10](#_Toc162464208)

[Rozšiřující sloty 10](#_Toc162464209)

[Blokové schéma 10](#_Toc162464210)

[Form Factor 11](#_Toc162464211)

[Konektory 11](#_Toc162464212)

[Propojky (jumpery) 11](#_Toc162464213)

[Přepínače 11](#_Toc162464214)

[BIOS 11](#_Toc162464215)

[UEFI 11](#_Toc162464216)

[Procesor + Mikro architektura Procesoru 12](#_Toc162464217)

[Dělení architektury 12](#_Toc162464218)

[Registry 13](#_Toc162464219)

[ALU 13](#_Toc162464220)

[Vnitřní šířka mikroprocesoru 13](#_Toc162464221)

[Instrukční sada 13](#_Toc162464222)

[Systémové přerušení 13](#_Toc162464223)

[Správa paměti 13](#_Toc162464224)

[Cache paměť 14](#_Toc162464225)

[Vnitřní frekvence 14](#_Toc162464226)

[Vnější frekvence 14](#_Toc162464227)

# Principy počítačů

## Historický vývoj

### První počítače

* Pascalina
  + Mechanická kalkulačka z ozubených koleček pro sčítání a odčítání
* G. W. Leibnity
  + Kroková kalkulačka
  + Mechanická kalkulačka založená na dvojkové soustavě
* Arithmometer
  + První sériově vyráběné kalkulačky

### První univerzální počítač

* Charles Babbage
  + Matematik, který vytvořil Analytický stroj
    - Univerzální ve smyslu, že dokáže simulovat jakýkoliv jiný stroj, algoritmus bez potřeby HW přestavby
    - Stroj nebyl nikdy sestaven, ale přesto pro něj byly napsány programy

### První číslicový počítač

* Z1
  + Rok 1936
  + Paměť 1408 bitů složená z mechanických posuvných prvků
  + Jedinou elektrickou součástí byl motor
  + Frekvence 1 Hz
  + 22bitová aritmetika
* Z2
  + Používal elektromagnetická relé
  + Pracoval pouze s celými čísly – 16 bitů
* Z3
  + 22bitový počítač
  + Používán pro výpočet balistických drah raket
  + 2600 elektromagnetických relé
* Mark 1
  + Rok 1937
  + Laboratoře firmy IBM
  + Reléové obvody
  + První plně elektrický funkční počítač

### Generace počítačů

#### Generace

* Počítače se řadí podle toho, jaké základní stavební prvky obsahují
* Zejména dominovaly elektromagnetické relé

#### Generace

* Relé nahradily elektronky
* Operační rychlost kolem tisíce operací za sekundu
* Programuje se výhradně ve strojovém kódu, neexistují programovací jazyky a překladače
* Neexistoval jednotný software
* ENIAC

#### Generace

* Místo elektronek se používají tranzistory
* Začíná se programovat ve Fortranu
* Vynalezeno přerušení
* Vznikají první operační systémy

#### Generace

* Integrované obvody
* Vznikají první magnetické disky
* Objevují se první operační systémy
* První superpočítače
* Výroba PC v tisícových sériích
* Jednotlivé typy PC jsou nekompatibilní
* První mikro a mini PC

#### Generace

* Od vzniku IBM PC/XT
* Mikroprocesor Intel 8086
* Éra systému MS-DOS
* Cena prudce klesá, výkon prudce stoupá
* Snaha o kompatibilitu systémů

## Binární logika

* Zabývá se logickými operacemi ve dvojkové soustavě

### AND

|  |  |  |
| --- | --- | --- |
| **A** | **B** | **Y** |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

### NAND

|  |  |  |
| --- | --- | --- |
| **A** | **B** | **Y** |
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

### OR

|  |  |  |
| --- | --- | --- |
| **A** | **B** | **Y** |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

### NOR

|  |  |  |
| --- | --- | --- |
| **A** | **B** | **Y** |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

### XOR

|  |  |  |
| --- | --- | --- |
| **A** | **B** | **Y** |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

## Modulace signálu

* Proces v čase, kterým se mění charakter nosného signálu modulujícím signálem
* Finální signál je vždy degradován kvůli útlumu nebo šumu

### Útlum

* Při přenosu dochází k energetickým ztrátám
  + Amplituda přenášeného signálu slábne úměrně ke vzdálenosti

### Šum

* Vznik nežádoucích signálů, než je přenášející signál
* Způsobeno typem vodiče, množstvím stínění, okolními signály

### Druhy modulace

#### Analogová modulace

* Využíváno nosné vlny
* No sinusový signál se nanáší modulační signál – informace
* Modulovaný signál může mít různou amplitudu

#### Amplitudová modulace

* Patří mezi jednoduché spojité modulace
* V závislosti na změně modulačního signálu se mění amplituda nosného signálu
* Frekvence ani fáze se nemění
* Hlavní parametr je hloubka modulace

Obsah obrázku řada/pruh

Popis byl vytvořen automaticky

#### Frekvenční modulace

* Kmitočet nosné vlny je ovlivňován pomocí modulačního signálu
* Nosná vlna se mění o frekvenční zdvih

Obsah obrázku řada/pruh

Popis byl vytvořen automaticky

#### Fázová modulace

* Princip podobný frekvenční modulaci
* Dochází pouze ke změně fáze
* Okamžitá hodnota fáze nosného signálu se mění v závislosti na strmosti amplitudy modulačního signálu

Obsah obrázku řada/pruh, anténa

Popis byl vytvořen automaticky

# Architektura počítače

## Von Neumannovo a Harvardské schéma

### Von Neumannova architektura

* Hlavní myšlenka
  + Struktura je nezávislá od zpracovávaných problémů
  + Na řešení problému se musí zvenčí zavést návod na zpracování a program
  + Program i data jsou uložena do paměti společně
* Využití dvojkové soustavy
* 5 funkčních jednotek
  + Řídící jednotka
  + Aritmeticko-logická jednotka
  + Paměť
  + Vstupní zařízení
  + Výstupní zařízení
* Po sobě jdoucí instrukce programu se uloží do paměťových buněk jdoucích po sobě
* Instrukcemi skoku se dá odklonit od zpracování instrukcí v uloženém pořadí

Obsah obrázku text, diagram, snímek obrazovky, řada/pruh

Popis byl vytvořen automaticky

#### Řadič

* Řídící jednotka celého PC
* Generuje řídící signály pro ostatní bloky
* Rozumí strojovému kódu

#### ALU

* Aritmeticko-logická jednotka
* Vykonává operace s daty
* Obsahuje binární sčítačku, komparátory, příznakové bity…
* Řadič dává ALU povel, jakou operaci má vykonat s daty na jejím vstupu

#### Paměť

* Obecně cokoliv, kam lze uložit data a později je opět vyzvednout
* Myšlenkou u Von Neumanna je adresovatelná paměť
  + Všechny části paměti jsou očíslovány
  + Všechny buňky jsou jednoznačně identifikovány
* Při práci s pamětí lze vybrat konkrétní adresu, na kterou zapisovat nebo z ní číst

#### Rozdíl oproti dnešním PC

* Von Neumanova architektura pracuje vždy s jedním programem
  + Špatné využití strojového času
* Von Neuman pracoval pouze s chráněným režimem
* Program se dnes nemusí do paměti zavádět celý, ale pouze jeho část a zbytek nechat na disku
* Dnes může počítat disponovat i více procesory

#### Reálný režim

* Základní režim (procesor se v něm nachází po zapnutí)
* Pokud procesor operuje v tomto režimu je plně kompatibilní s i8086
* Používá 20bitové adresy, lze adresovat 1MB paměti
* Není možnost multitaskingu
* Všechny procesory jsou v tomto režimu, dokud není zaveden operační systém, který je přepne do jiného režimu

#### Chráněný režim

* Podpora pro multitasking
  + Iluze paralelního zpracovávání
  + Mezi úlohami je pouze velice rychle přepínáno
    - Při přepnutí se musí uložit stav rozpracované úlohy
    - Za přepínání je odpovědné jádro OS
    - Každá úloha musí běžet izolovaně
* Vzájemná ochrana úloh
* Privilegování operačního systému
* Podpora pro práci s virtuální pamětí

### Harvardská architektura

* Není potřeba mít paměť stejných parametrů pro data a pro program
* Paměti mohou být naprosto odlišné, mít jinou délku slova, časování, technologii…
* Paměť dat je typu RWM, paměť programu je nezávislá na napájení (ROM, FLASH, …)

Obsah obrázku text, diagram, snímek obrazovky, řada/pruh

Popis byl vytvořen automaticky

## Flynnova taxonomie

* Dělení počítačů podle schopnosti paralelního zpracování

### SI

* Single Instruction Stream
* V čase řešení problému běží jeden program

### MI

* Multiple Instruction Stream
* Během řešení běží více programů najednou

### SD

* Single Data Stream
* Jeden zpracovávaný tok dat

### MD

* Multiple Data Stream
* Více zpracovávaných toků dat

### Kombinace

#### SISD

* Klasický Von Neumannův počítač
* Jeden program a jeden sériový tok dat

#### MISD

* Hypotetická kombinace několika programů zpracovávajících jeden tok dat
* Takový princip se zatím nepoužívá

#### SIMD

* Větší počet funkčních jednotek pracujících na řešení téhož programu
* Všechny jednotky provádí stejnou instrukci, ale nad jinými daty
* Řídící jednotka vysílá stejný kód ke všem procesorům

#### MIMD

* Obecný typ paralelního systému
* Flexibilnější, ale mohou nastat problémy s alokací zdrojů, deadlockem…
* Množina procesorů současně provádí různé instrukce nad různými daty

Obsah obrázku text, snímek obrazovky, Písmo, číslo

Popis byl vytvořen automaticky

## Základní deska

* Účelem je propojit jednotlivé součástky počítače a poskytnout jim napájení
* Fyzicky jde o desku plošného spoje

### Ovlivňuje

* Typ mikroprocesoru
* Typ, rychlost, velikost operační paměti
* BIOS
* Integrované řadiče
* Integrovaná I/O zařízení
* Integrovaný USB controller
* ….

### Rozšiřující sloty

* Umožňují připojit další zařízení
* ISA
  + dříve pro GPU
* EISA, VESA
  + Novější ISA, dnes také nepoužíváno
* PCI
  + Dříve standardem pro všechny rozšiřující karty, dnes nahrazován PCI-E
* APG
  + Slot navržen speciálně pro GPU
  + Rychlejší než PCI, ale vytlačen PCI-E
* PCI-E
  + Nástupce PCI a APG
  + Univerzální sběrnice pro připojení jakéhokoliv standardního typu přídavných karet
  + Mnohem rychlejší než předchůdci, větší šířka pásma…

### Blokové schéma

Obsah obrázku text, snímek obrazovky, diagram, Písmo

Popis byl vytvořen automaticky

### Form Factor

* ATX
* microATX
* PC-XT
* AT form factor, ETX, BTX, mini-ITX, NLX

### Konektory

* Interní a externí

#### Interní

* Na ploše základní desky
* IDE, SATA, FLOPPY
* Napájecí konektory
* Konektory pro ventilátory, zvukové karty, USB, kabely předního krytu

#### Externí

* Na zadním panelu desky
* USB, PS/2, eSATA, COM, DVI, HDMI, VGA, LAN, audio

### Propojky (jumpery)

* Umožňují nastavit některé parametry základní desky
* Jde o skupinu pinů
* Nastavení je uvedeno v dokumentaci desky

### Přepínače

* Jde o switche na základní desce
* Stav ON/OFF
* Každý je očíslován a funkce je v dokumentaci desky
* Spíše u starších desek

### BIOS

* Basic Input Output Systém
* Spojuje HW, umožňuje komunikaci OS s HW
* Uložen v ROM paměti
* Dnes spíše UEFI

### UEFI

* Unified Extensible Firmware Interface
* Za účelem překonat omezení BIOSu
  + 16bit režim CPU
  + 1 MB adresovatelného místa
* První specifikace 12. prosince 2000
* Umožňuje využití grafického rozhraní

## Procesor + Mikro architektura Procesoru

### Dělení architektury

* CISC
  + Compley Instruction Set Computer
* RISC
  + Reduced Instruction Set Computer
* VLIW
  + Very Long Instruction Word
* MISC
  + Minimum Instruction Set Computer

#### CISC

* Obsáhlá instrukční sada
* Proměnlivá délka kódu i výpočtu
  + Složitý řadič
    - Instrukce trvají i desítky taktů
  + Roste celková složitost mikroprocesoru
* Obsaženo
  + Binární aritmetika
  + Instrukce pro práci s textem
  + Různé numerické formáty
  + Podpora BCD aritmetiky
* Použito v řadách x86

#### RISC

* Důvodem vzniku bylo nevyužívání všech dostupných instrukcí
* Jednoduché instrukce
* Menší množství adresních režimů
* Velké množství pracovních registrů
  + Slouží jako cache paměť
  + Všechny aritmetické a logické instrukce se provádí na těchto registrech
* U ARM nebo v integrovaných obvodech

#### VLIW

* Speciální formát operačního kódu
* V jedné instrukci uloženy operační kódy pro všechny jednotky
* Konstanta v instrukčním slově je použita pro naplnění vybraného registru
* Horizontální formát umožňuje řadiči zůstat jednoduchý
* Paralelní práce jednotek
* Instrukce mají pevnou délku

##### Přednosti

* Velká jednoduchost řadiče
* Možnost zpracovávání instrukcí paralelně

##### Nevýhody

* Veškeré optimalizace se musí provádět v závislosti na mikroprocesoru
* Není vhodné při používání interpretovaných kódů
* Konflikty jednotlivých operací nad registry

#### MISC

* Alternativa k RISC nebo CISC
* Instrukce bez operandů
  + Ty jsou známy implicitně
* Menší nárok na rychlost operační paměti
* Rychlejší reakce na přerušení

### Registry

* Paměť, která umí uložit vždy jedno slovo
  + Šířka slova od 4 bitů do 128 bitů (nejčastěji 8, 16, 32, 64)
* Kapacita je velice malá
* Paměť využívána všemi instrukcemi

### ALU

* Výpočet základních aritmeticko-logických operací
* Na vstup přijdou dvě n-bitová čísla
  + Zvolení operace
* Na výstup je po zpoždění zasláno m-bitové číslo
  + Výsledek operace
* Na výstupu mohou být další signály (flags)
  + Příznak nulovosti, nebo přetečení
* Součet, rozdíl, dvojkový doplněk, porovnání, negace, logický součet/rozdíl, součin

### Vnitřní šířka mikroprocesoru

* Schopnost zpracovat najednou určité množství informací
* Dnes 32bitové, 64bitové

### Instrukční sada

* Sada instrukcí pro přesuny dat mezi pamětí a registry, aritmetické a logické instrukce
* U nových mikroprocesorů i instrukce pro koordinaci víceprocesorových prostředí

### Systémové přerušení

* Signál vysílaný HW nebo programem
* Vysláním si prvek snaží zabrat procesor pro sebe
  + Dnes vektorový systém přerušení
    - Každé přerušení je identifikováno číslem
    - V paměti je uložena tabulka přerušení, kde vektor ukazuje na adresu s obslužným programem přerušení

### Správa paměti

* Tato jednotka stojí mezi adresy generovanými programem a skutečnými adresami v paměti
  + Důvodem překladu je lepší využití paměti a zabezpečení

### Cache paměť

* Mezisklad mezi různě rychlými komponentami PC
* Načte data se sběrnice, která čekají až je bude mikroprocesor potřebovat
  + Ten si je pak načte

### Vnitřní frekvence

* Taktovací impulsy, které určují pracovní tempo
  + Na desce je generátor taktů, který určuje takt pro mikroprocesor
  + Z této externí frekvence je určena vnitřní frekvence
  + Mezi sběrnicí a mikroprocesorem pracuje násobička, která převádí externí tak na vyšší vnitřní
* Mikroprocesory mají takt pevně určen

### Vnější frekvence

* Generována základní deskou
* Pracují s ní všechny součásti desky

## Paměti

* Zařízení, používané k ukládání programu nebo dat pro okamžitou nebo trvalou potřebu
* Mikroprocesor z ní čte programy, kterými je řízen a také do ní ukládá výsledky operací

### Parametry

* Kapacita
* Přístupová doba v ns
* Přenosová rychlost
* Statičnost / dynamičnost
* Volatilita
* Destruktivnost při čtení
* Spolehlivost
* Cena za bit

### ROM

* Read Only Memory
* Není energeticky závislá (data zůstanou i po vypnutí PC)
* Pro uložení BIOSu
  + Přenos z ROM do RAM nazýváme stínování (shadowing)

#### ROM

* Výrobce buňky přepálí, ty nosí logickou 1, ostatní vedou proud a jsou nositelem logické 0

#### PROM

* Podobné jako ROM, ale informace nezapisuje výrobce, ale uživatel pomocí programátoru ROM

#### EPROM

* Lze do nich opakovaně zapisovat, smazání se provádí pomocí UV světla, zápis pomocí elektrického náboje

#### EEPROM

* Mazatelná paměť pomocí elektrických impulsů, počet zápisů i mazání je omezen

#### Flash-EPROM

* Nejrychlejší typ, počet cyklů kolem 1000, programovatelná přímo z PC

### RAM