

KATEDRA INFORMATIKY
PŘÍRODOVĚDECKÁ FAKULTA
UNIVERZITA PALACKÉHO

STRUKTURA POČÍTAČŮ

JIŘÍ HRONEK, JIŘÍ MAZURA



VÝVOJ TOHOTO UČEBNÍHO TEXTU JE SPOLUFINANCOVÁN
EVROPSKÝM SOCIÁLNÍM FONDEM A STÁTNÍM ROZPOČTEM ČESKÉ REPUBLIKY

Olomouc 2007

Abstrakt

Tento učební text seznamuje čtenáře s problematikou základů technických prostředků personálních počítačů. Rozsah a úroveň detailů je zvolena tak, aby respektovala nároky a rozdíly studijních programů. Doporučením je doplňkové studium specializovaných a systematictějších materiálu, jejichž prostudování spolu s praktickým cvičením by měly problematiku prohloubit a doplnit.

Cílová skupina

Text je primárně určen pro posluchače prvního ročníku bakalářského a kombinovaného studijního programu Aplikovaná informatika na Přírodovědecké fakultě Univerzity Palackého v Olomouci.. Text předpokládá znalosti základních pojmů matematiky, informatiky a fyziky.

Obsah

1	Reprezentace dat, Boolova algebra a logické funkce	9
1.1	Reprezentace dat	9
1.1.1	Zobrazení čísel se základem 2	9
1.1.2	Úvod do logických operátorů – výroková logika	10
1.2	Základní operátory Boolovy algebry	11
1.2.1	Logické proměnné a logické funkce.....	11
1.2.2	Logické operátory.....	11
1.2.3	Vlastnosti základních logických operátorů.....	13
1.2.4	Princip fyzické realizace.....	15
1.3	Definice logické funkce	17
1.3.1	Pravdivostní tabulka	17
1.3.2	Základní zápis logické funkce	17
1.4	Logické funkce n proměnných.....	18
1.4.1	Funkce jedné proměnné.....	19
1.4.2	Funkce dvou proměnných	19
1.4.3	Funkce více než dvou proměnných	20
1.4.4	Základní nutné operátory	20
1.5	Zjednodušování zápisu logické funkce.....	20
1.5.1	Algebraická minimalizace	21
1.5.2	Karnaughova metoda.....	21
1.5.3	Algoritmické metody.....	22
2	Základní logické operátory.....	24
2.1	Úplný systém logických funkcí.....	24
2.2	Minimální úplný systém logických funkcí.....	24
2.2.1	Realizace operátorů <i>NOT</i> , <i>AND</i> a <i>OR</i> pomocí <i>NAND</i>	25
2.2.2	Realizace operátorů <i>NOT</i> , <i>AND</i> a <i>OR</i> pomocí <i>NOR</i>	25
2.2.3	Realizace operátoru nonekvivalence pomocí <i>NAND</i>	26
2.2.4	Realizace logické funkce pomocí logických operátorů <i>NAND</i>	27
3	Logické obvody	29
3.1	Kombinační logické obvody	29
3.1.1	Komparátor.....	29
3.1.2	Multiplexor (Multiplexer).....	30
3.1.3	Binární dekodér	31
3.1.4	Sčítačka.....	32

4	Zobrazení čísel, kódy	34
4.1	Přímé kódy	34
4.1.1	Princip změny základu	35
4.2	Aditivní kód	37
4.3	Doplňkový kód pro záporná čísla	37
4.4	Reprezentace čísel s řádovou čárkou	39
4.4.1	Čísla s fixní řádovou čárkou	39
4.4.2	Čísla s pohyblivou řádovou čárkou	39
4.5	Kódování textu	41
4.6	Kontrolní kódy	43
4.6.1	Grayův kód	43
4.6.2	Kódy p z n	44
4.6.3	Kontrola paritou	44
4.7	Samoopravný kód	45
4.8	Detekce chyb při sériovém záznamu dat	47
5	Sekvenční obvody	50
5.1	Klopné obvody	51
5.1.1	Klopný obvod RS	51
5.1.2	Klopný obvod D	52
5.1.3	Klopný obvod JK	54
5.1.4	Přenos informací v systému	55
5.1.5	Typické sekvenční obvody v počítačích	56
6	Hardware počítače	60
6.1	Generace počítačů	60
6.1.1	Historie počítačů	60
6.1.2	Koncepce Johna von Neumanna	61
6.1.3	Harwardská koncepce	62
6.1.4	Počítače PC	63
6.2	Základní deska PC	63
6.2.1	Sběrnice (bus)	65
6.2.2	Technologie Plug & Play (PnP)	71
6.3	Procesor	71
6.4	Procesory Intel	75
6.5	PAMĚTI	79
6.5.1	Paměti typu ROM	81
6.5.2	Paměti typu RAM	83
6.5.3	Cache paměti	86

6.5.4	CMOS paměť	90
6.6	PEVNÝ DISK.....	91
6.6.1	Geometrie pevných disků	92
6.6.2	Disková pole.....	99
6.7	PRUŽNÝ DISK	100
6.7.1	Modulace dat při záznamu na magnetická média.....	100
6.8	CD A DVD	102
6.8.1	Compact Disk - CD	102
6.9	Mechaniky ZIP.....	104
6.10	Videokarty	104
6.10.1	Typy videokaret.....	106
6.11	Síťová karta.....	114
6.12	Karty pro příjem rádia a televize.....	115
6.13	Monitor	116
6.14	VSTUPNÍ ZAŘÍZENÍ	120
6.15	Připojování vstup/výstupních zařízení	122
6.15.1	Rozhraní Centronics (EPSON).....	123
6.15.2	Rozhraní V.24 I. (RS-232C)	124
6.15.3	USB (Universal Serial Bus)	126
6.15.4	IEEE 1394 (FireWire).....	128
7	Závěr.....	130
8	Seznam literatury.....	131
9	Seznam obrázků	132

1 Re prezentace dat, Boolova algebra a logické funkce

Studijní cíle: Po prostudování kapitoly by měl student zvládnout úvod do problematiky reprezentace dat ve dvojkové soustavě a teoretické základy kombinačních obvodů – základy Boolovy algebry, definice a úpravy logických funkcí.

Klíčová slova: Data, informace, číselné soustavy, výroky, Boolova algebra, logické proměnné, funkce a operátory, normální formy - ÚDNF a ÚKNF, minimalizace, Karnaughova mapa

Potřebný čas: 4 hodiny

1.1 Re prezentace dat

Pracujeme-li s obvyklými čísly, používáme symboly, jimž přiřazujeme deset různých hodnot od nuly (symbol 0) do devíti (symbol 9). Používáme soustavu zobrazení čísel se základem deset. Dá se přepokládat, že prapůvodním důvodem používání dekadické soustavy je existence deseti prstů na lidských rukách - prvotní početní pomůcky. Někdy můžeme zaznamenat i používání jiných soustav, než je dekadická. Příkladem může být soustava šedesátková pro záznam minut a vteřin či dvanáctková a dvacítková pro původní anglická platidla – šilink a penci.

První stroje na zpracování čísel byly mechanické. U nich nebylo složité zobrazovat čísla se základem deset. Dělal se to pomocí mechanické součástky, která měla deset stabilních stavů, z nichž každý odpovídal jednomu znaku z deseti. Mechanické systémy však byly relativně ložité a rychlost výpočtu byla omezena rychlostí pohybu mechanických součástek. Stupeň rozvoje v podstatě skončil u jednoduchých aritmetických strojů.

Náhrada mechanických prvků elektronickými prvky, zpočátku reléovými obvody, později se používaly elektronky, tranzistory a integrované obvody, si vyžádala pro reprezentaci čísel i změnu základu číselné soustavy. U uvedených elektronických součástek se nejjednodušší realizují dva stavy, relé - *sepnuto* x *rozepnuto*, elektronka či tranzistor – proud *prochází* x *neprochází* apod. Realizovat elektrické obvody se dvěma stabilními stavy je relativně jednoduché. Proto se používá reprezentace čísel (a také všech dalších typů dat, které jsou vlastně z čísel odvozeny) v počítači ve dvojkové soustavě, která přirozeně využívá dvou stabilním stavům elektronických součástek.

1.1.1 Zobrazení čísel se základem 2

Libovolné přirozené číslo N je možné zapsat jako řadu

$$N = a_{n-1} \cdot B^{n-1} + a_{n-2} \cdot B^{n-2} + \dots + a_2 \cdot B^2 + a_1 \cdot B^1 + a_0 \cdot B^0$$

kde B je číslo určující základ soustavy a_i - jsou číselné koeficienty, které se vztahují k váze B^i ($0 < a_i < B$). Základ B může být libovolné přirozené číslo. V běžné praxi používáme desítkový zápis čísel ($B=10$, $a_i = 0 - 9$) a spokojujeme se pouze s vypsáním koeficientů

$$N = a_{n-1}a_{n-2}\dots a_1a_0$$

Hodnota čísla **1302** vznikne tak, že sečteme číslice jednotlivých řádů vynásobené jejich vahami

$$1302 = 1 \cdot 10^3 + 3 \cdot 10^2 + 0 \cdot 10^1 + 2 \cdot 10^0 = 1000 + 300 + 0 + 2$$

*Zápis čísla
v z-adické
soustavě*

V digitálních systémech se většinou používá binární (dvojkové) zobrazení čísel ($B=2$, $a_i = 0,1$).

$$N = a_{n-1} \cdot 2^{n-1} + a_{n-2} \cdot 2^{n-2} + \dots + a_2 \cdot 2^2 + a_1 \cdot 2^1 + a_0 \cdot 2^0$$

Číslo **1302** vyjádřené ve dvojkové soustavě

$$10100010110 =$$

$$1 \cdot 2^{10} + 0 \cdot 2^9 + 1 \cdot 2^8 + 0 \cdot 2^7 + 0 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 =$$

$$1024 + 256 + 16 + 4 + 2 = 1302$$

Příklad počítání v desítkové a dvojkové soustavě:

Desítková soustava	Dvojková soustava
00	0000
01	0001
02	0010
03	0011
04	0100
05	0101
06	0110
07	0111
08	1000
09	1001
10	1010
atd.	atd.

1.1.2 Úvod do logických operátorů – výroková logika

V počítači se data pouze neuchovávají, ale i zpracovávají. Mezi základní operace patří ty, které se nazývají logické a jejich hardwarová realizace tvoří základ většiny počítačových obvodů. Jejich teoretickým formálním základem je výroková logika, původně filozoficky zkoumající pravdivostní hodnotu složených výroků. Základem výrokové logiky je výrok. Výrokem je každá oznamovací věta, u které můžeme určit její pravdivostní hodnotu – je-li pravdivá, nebo nepravdivá.

I když na dvoustavová data nejsou lidé příliš zvyklí, přesto je používají dosti často například při vytváření rozhodnutí podle aktuální pravdivostní hodnoty několika (složených) výroků, které současně uvažují.

Příklad.

Rozhodnutí je funkcí čtyř výroků **V1**, **V2**, **V3**, **V4** definovaných takto:

Rozhodnutí **R**: jít na schůzku

Výrok **V1**: Jsem telefonicky pozván.

Výrok **V2**: Jsem zaneprázdněn v okamžiku telefonického volání.

Výrok **V3**: Auto je pojízdné.

Výrok **V4**: Městská doprava nejezdí.

Jestliže **V1** je pravdivý **A**(zároveň) **V2** je nepravdivý **A**(zároveň) **jestliže** (**V3** je pravdivý **NEBO** **jestliže** **V4** je **NE**pravdivý), **pak** rozhodnutí **R** bude učiněno – půjdu na schůzku ($R=1$). Zvýrazněné spojky v běžném jazyce vedou k definici odvozených logických spojek,

např. konjunkce (A - AND), disjunkce (NEBO - OR), implikace (JESTLIŽ ... PAK - =>), negace (NE - NOT).

Všechny logické výroky mohou mít tvar výrazu, který si později definujeme. V našem případě

$$R = V1 \text{ AND NOT } V2 \text{ AND } (V3 \text{ OR NOT } V4)$$

V počítači si představíme, že výroky jsou nahrazeny přímo logickými hodnotami (pravda – nepravda, True – False, 1 – 0) a vyhodnocované logické výrazy mají obecně specifický význam – např. popisují požadované chování hardwaru – číslicového obvodu. Pro vyhodnocení výrazu je potřeba ovládat matematický prostředek pro práci s logickými výrazy, což umožňuje aparát Boolovy algebry. Fyzicky – pro praktickou realizaci či demonstraci vyhodnocení logických výrazů se používají kombinační logické obvody.

1.2 Základní operátory Boolovy algebry

Teoretickým základem činnosti digitálních zařízení, řídicích jednotek, programovatelných automatů, počítačů, mikroprocesorů atd. je tzv. binární logika (Booleova algebra). Kromě binární (dvoustavové) logiky existují i jiné logické systémy, např. logika tří stavů. Význam binární logiky v porovnání s jinými typy logik je dán její:

- univerzálností a dokonalým teoretickým zvládnutím,
- efektivní realizací fyzikálních logických struktur využívajících binární logiku.

1.2.1 Logické proměnné a logické funkce

Logická proměnná nebo dvojková proměnná, kterou budeme značit x je veličina, která může nabývat pouze dvou hodnot, označovaných 0 a 1 (nepravda a pravda), a nemůže se spojitě měnit. Tato definice může být vyjádřena následujícími axiomy:

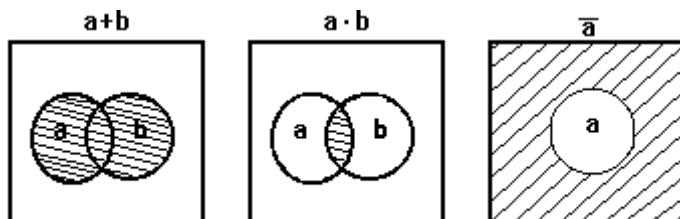
$$x = 1 \text{ jestliže } x \neq 0,$$

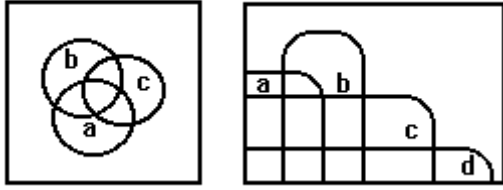
$$x = 0 \text{ jestliže } x \neq 1.$$

Logická funkce n logických proměnných x_1, x_2, \dots, x_n je funkce, která stejně jako všechny její proměnné může nabývat pouze dvou hodnot 0 nebo 1 . Logická funkce může být sama považována za proměnnou vzhledem k jiné logické funkci. Boolova algebra studuje dvojkové proměnné a funkce těchto proměnných. Je to algebra stavů, nikoliv čísel.

1.2.2 Logické operátory

Jsou definovány tři základní logické operátory, které mezi proměnnými provádějí tři základní logické operace – negace ($\bar{}$), konjunkce (\cdot) a disjunkce ($+$). Množinově je možno použít znázornění pomocí Vennových diagramů (v množinách a, b jsou prvky s hodnotou 0 nebo 1 , pokud uvažujeme všechny kombinace a za výsledek považujeme stav ve vyšrafované oblasti, dostaneme pravdivostní tabulky jednotlivých operátorů):





Funkce rovnost

Říkáme, že dvě logické proměnné $A = B$, se sobě rovnají, když $A = I, B = I$ nebo $A = 0, B = 0$, což zapisujeme $A = B$. Dvě veličiny $A = a_0, a_1, \dots, a_n; B = b_0, b_1, \dots, b_n$ se sobě rovnají, když platí $a_i = b_i$ pro všechna i .

Negace

Tato operace, nazývaná také inverze, je definována pravdivostní tabulkou Tab. 1.1. Označuje se přidáním „pruhu“ nad proměnnou x (\bar{x} se čte: x non).

Tato definice vede ke vztahům: $\bar{0} = I; \bar{I} = 0$

Tab. 1.1. Funkce negace

x	\bar{x}
I	0
0	I

Tab. 1.2. Funkce logického součtu

x	y	$x + y$
0	0	0
0	I	I
I	0	I
I	I	I

Negace:

\bar{x}

NOT

\neg

\sim (před operandem)

x se nazývá přímá proměnná

\bar{x} se nazývá negovaná proměnná

Logický součet

Tato operace, nazývaná také sjednocení nebo disjunkce, aplikovaná na dvě proměnné vede k „součtu“ neboli funkci NEBO těchto dvou proměnných. Označuje se znakem \cup mezi dvěma proměnnými, $x \cup y$, což vylučuje záměnu s aritmetickým součtem, ale v praxi je většinou zapisována ve tvaru $x + y$.

Tabulka Tab. 1.2, zvaná pravdivostní tabulka, definuje logický součet. Výsledkem sjednocení dvou proměnných x, y je funkce $f(x, y)$. Rovná se jedničce, když jedna nebo druhá proměnná nebo obě mají hodnotu jedna.

Logický
(Booleův) součet:
OR

\vee

\cup

+

Tato definice může být vyjádřena vztahy:

$$0 + 0 = 0$$

$$0 + I = I + 0 = I$$

$$I + I = I$$

Je třeba si všimnout, že $I + I = I$ se liší od vztahu, na který jsme zvyklí v klasické algebře. Je to příležitost si připomenout, že 0 a I zde znamenají stavy a ne číselné hodnoty.

Logický součin

Tato operace, zvaná také průnik nebo konjunkce, aplikovaná na dvě proměnné, vytváří „součin“ neboli funkci A těchto dvou proměnných. Značí se symbolem \cap mezi dvěma proměnnými ($x \cap y$) nebo jednodušeji $x \cdot y$, popř. xy . Tab. 1.3 ukazuje její pravdivostní tabulku.

Logický (Booleův) součin :

AND

\wedge

\cap

\cdot

(bez značky)

Tab. 1.3. Funkce logického součinu

x	y	$x \cdot y$
0	0	0
0	1	0
1	0	0
1	1	1

Tato definice může být vyjádřena vztahy:

$$0 \cdot 0 = 0$$

$$0 \cdot 1 = 1 \cdot 0 = 0$$

$$1 \cdot 1 = 1$$

Poznámka

Nebyla definována operace odčítání ani dělení. Zatímco je vždy možné přidat výraz na obě strany logické rovnice nebo je vynásobit stejným výrazem, nikdy se rovnice nesmí zjednodušit odečtením téhož výrazu od obou stran nebo dělením na obou stranách tímž členem.

Příklad

$$a + b = a + c \quad \text{neznačená} \quad b = c$$

(například pro $a = I, b = 0, c = I$ rovnice platí, a proto $b = \bar{c}$) a rovněž

$$a \cdot b = a \cdot c \quad \text{neznačená} \quad b = c$$

(například pro $a = I, b = 0, c = I$ rovnice platí, a proto $b = \bar{c}$).

1.2.3 Vlastnosti základních logických operátorů

Pomocí předcházejících postulátů je odvozena skupina zákonů, z nichž ty nejjednodušší tvoří základ metody úprav a zjednodušování logických funkcí. Některé z těchto zákonů lze snadno dokázat pomocí uvedených postulátů. Všechny mohou být dokázány tak, že se proměnným přiřadí všechny možné kombinace hodnot 0 a I , což pro dvě nebo tři proměnné neznamená příliš zdouhavé důkazy. Zároveň je možno při odvozování pozorovat vlastnost, které se říká dualita – duální výraz můžeme odvodit současnou záměnou logických hodnot na opačné, operátorů na duálně opačné – AND na OR, OR na AND ve výchozím výrazu.

Aplikace na jednu proměnnou

Tyto zákony vyplývají přímo z definice základních logických operátorů, dosazujeme-li postupně $x = 0$ a $x = I$

$$\bar{\bar{x}} = x$$

$$x + 0 = x$$

$$x \cdot 0 = 0$$

$$\begin{array}{ll} x + I = I & x \cdot I = x \\ x + x = x & x \cdot x = x \\ x + \bar{x} = I & x \cdot \bar{x} = 0 \end{array}$$

Pro zjednodušování funkcí jsou všechny tyto zákony nezbytné.

Aplikace na více proměnných

Tyto vlastnosti jsou vyjádřeny algebraickými zákony:

$$x + y = y + x$$

$$x \cdot y = y \cdot x$$

Uvedené dva zákony vyjadřují **komutativitu** součtu a součinu

$$x + y + z = (x + y) + z = x + (y + z)$$

$$x \cdot y \cdot z = (x \cdot y) \cdot z = x \cdot (y \cdot z)$$

Tyto dva zákony vyjadřují **asociativitu** součtu a součinu

$$x \cdot (y + z) = x \cdot y + x \cdot z$$

$$x + y \cdot z = (x + y) \cdot (x + z)$$

První zákon vyjadřuje **distributivitu** součinu vzhledem k součtu, druhý zákon vyjadřuje **distributivitu** součtu vzhledem k součinu a nemá svůj ekvivalent v klasické algebře. Tyto zákony umožňují dokázat další zákony velmi důležité pro zjednodušování logických funkcí, které nejsou vždy na první pohled samozřejmé zvláště proto, že každá proměnná je funkcí ostatních proměnných. Například lze uvést čtyři zákony:

$$\begin{array}{ll} x + x \cdot y = x & (x + \bar{y}) \cdot y = x \cdot y \\ x \cdot (x + y) = x & x \cdot \bar{y} + y = x + y \end{array}$$

De Morganovy zákony jsou vyjádřeny dvěma rovnostmi:

$$\overline{x + y + z + \dots} = \bar{x} \cdot \bar{y} \cdot \bar{z} \cdot \dots$$

$$\overline{x \cdot y \cdot z \cdot \dots} = \bar{x} + \bar{y} + \bar{z} + \dots$$

Tento zákon lze vyjádřit takto:

Negaci funkce získáme nahrazením proměnné její negace a záměnou značek součtu a součinu navzájem. Při použití tohoto zákona je třeba věnovat velkou pozornost implicitním závorkám.

Příklad

$$x + y \cdot z = x + (y \cdot z)$$

Tedy $\overline{x + y \cdot z} = \bar{x} \cdot (\bar{y} + \bar{z})$ a neplatí $\overline{x \cdot y + z}$

Přehledně si můžeme znázornit všechna pravidla do tabulky (převzato z literatury)

	Relationship	Dual	Property
Postulates	$AB = BA$	$A + B = B + A$	Commutative
	$A(B + C) = AB + AC$	$A + BC = (A + B)(A + C)$	Distributive
	$1A = A$	$0 + A = A$	Identity
	$A\bar{A} = 0$	$A + \bar{A} = 1$	Complement
Theorems	$0A = 0$	$1 + A = 1$	Zero and one theorems
	$AA = A$	$A + A = A$	Idempotence
	$A(BC) = (AB)C$	$A + (B + C) = (A + B) + C$	Associative
	$\overline{\overline{A}} = A$		Involution
	$\overline{AB} = \overline{A + B}$	$\overline{A + B} = \overline{A} \overline{B}$	DeMorgan's Theorem
	$AB + \overline{AC} + BC = AB + \overline{AC}$	$(A + B)(\overline{A} + C)(B + C) = (A + B)(\overline{A} + C)$	Consensus Theorem
	$A(A + B) = A$	$A + AB = A$	Absorption Theorem

Průvodce studiem

Booleova algebra je množina B o alespoň 2 prvcích nad níž jsou definovány operace sčítání, násobení a negace splňující tyto axiomy:

(předp.: $a, b, c \in B$):

$$a + b \in B$$

$$a \cdot b \in B$$

Existuje prvek 0 , pro který platí: $a + 0 = a$

Existuje prvek 1 , pro který platí: $a \cdot 1 = a$

Komutativní zákon:

$$a + b = b + a$$

$$a \cdot b = b \cdot a$$

$$a + (b \cdot c) = (a + b) \cdot (a + c)$$

$$a \cdot (b + c) = (a \cdot b) + (a \cdot c)$$

Pro každý prvek a existuje prvek $\bar{a} \in B$:

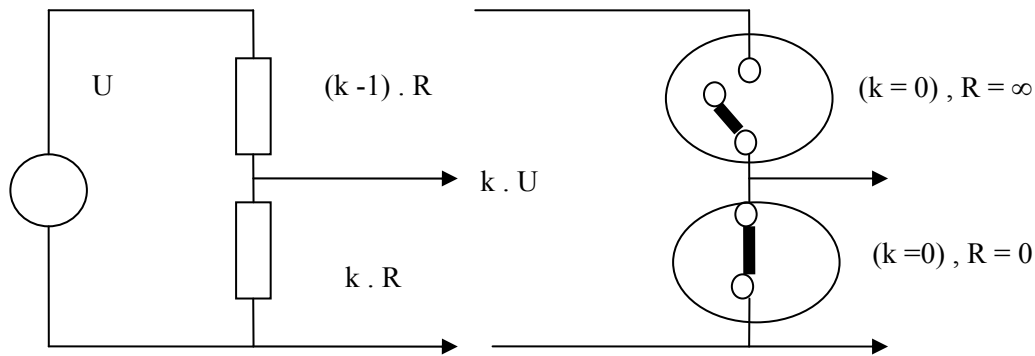
$$a \cdot \bar{a} = 0$$

$$a + \bar{a} = 1$$

1.2.4 Princip fyzické realizace

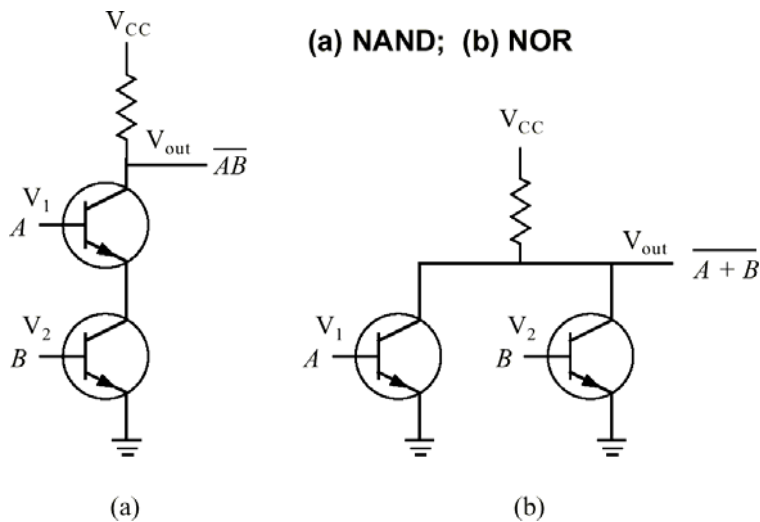
Praktická realizace je založena na principu, jehož základy si zjednodušeně vysvětlíme. Bez hlubších znalostí fyziky, nebo elektrotechniky si připomeneme jednoduchý dělič napětí, zobrazený na obrázku. Napětí na výstupu – na odporovém děliči je podle Ohmova zákona dáno

poměrem hodnot odporu, což je symbolicky popsáno na obrázku. Pokud zvolíme extrémní hodnoty ($k = 1$, nebo $k = 0$), což je pro ilustraci modelováno pomocí spínacích kontaktů, je na výstupu celé napětí U , nebo naopak nulové napětí. Původně v logických obvodech při realizaci skutečně byly spínací kontakty relétek, později spínače nahradily rychleji reagující elektronky, potom tranzistory a integrované obvody.



Obrázek 1 Fyzikální základ kombinačních obvodů

Příklad ilustračního zapojení obvodů, které realizují operátory NAND a NOR, je na následujícím obrázku.



Obrázek 2 Ilustrace zapojení obvodů pro realizaci operací NAND a NOR

1.3 Definice logické funkce

1.3.1 Pravdivostní tabulka

Úplně zadaná funkce

Logická funkce je úplně zadaná, když je známa její hodnota 0 nebo 1 pro všechny možné kombinace proměnných. Těchto kombinací je pro n proměnných 2^n . Lze tak sestavit pravdivostní tabulku.

Příklad: funkce tří proměnných $f_{(x,y,z)}$.

Funkce $f_{(x,y,z)}$ bude úplně zadána, když bude určena její hodnota 0 nebo 1 pro každou z osmi možných kombinací Tab. 1.4.

Tab. 1.4. Pravdivostní tabulka úplně zadané funkce

x	y	z	$f_{(x,y,z)}$
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Tab. 1.5. Pravdivostní tabulka neúplně zadané funkce

x	y	z	$f_{(x,y,z)}$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	×
1	0	0	×
1	0	1	0
1	1	0	1
1	1	1	×

Neúplně zadaná funkce

Logická funkce je neúplně zadaná, když její hodnota pro některé kombinace hodnot proměnných je libovolná nebo není určena. S tímto případem se setkáme, když některé kombinace hodnot nejsou fyzikálně možné. Hodnotu funkce pak značíme \times , Tab. 1.5.

1.3.2 Základní zápis logické funkce

Základní tvar

Je dokázáno, že logická funkce $f_{(x,y,z,\dots)}$ může být zapsána ve dvou tvarech, zvaných základní součtový tvar a základní součinnový tvar. (Někdy se také používá název **úplná disjunktivní normální forma – ÚDNF** a **úplná konjunktivní normální forma – ÚKNF**.)

f *základní součtový tvar* : součet základních součinů přímých nebo negovaných proměnných

f *základní součinnový tvar* : součin základních součtů přímých nebo negovaných proměnných

Proměnná je použita jako přímá, když není negovaná. V prvním případě nabývá každý základní součin (minterm) hodnoty 1 pro jistou kombinaci, kdy funkce má hodnotu 1 , a hodnoty 0 pro všechny ostatní kombinace.

Ve druhém případě nabývá každý součet hodnoty 0 pro jistou kombinaci, kdy funkce má hodnotu 0 , a hodnoty 1 pro všechny ostatní kombinace.

První způsob vyjadřuje funkci jako součet případů, kdy má hodnotu 1 , druhý způsob jako součin případů, kdy má hodnotu 0 .

Příklad zápisu logické funkce

Tab. 1.6. Tabulka majoritní funkce tří proměnných

x	y	z	$f_{(x,y,z)}$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$f_{(x,y,z)} = 1$, když většina proměnných je rovna jedné

$f_{(x,y,z)} = 0$, v ostatních případech

Základní součtový tvar

Případy, kdy funkce se rovná jedné

odpovídající součiny

	0	1	1	$\bar{x} \cdot \bar{y} \cdot \bar{z}$
kombinace	1	0	1	$x \cdot \bar{y} \cdot \bar{z}$
xyz	1	1	0	$x \cdot y \cdot \bar{z}$
	1	1	1	$x \cdot y \cdot z$

Odtud funkce $f_{(x,y,z)} = \bar{x} \cdot \bar{y} \cdot \bar{z} + x \cdot \bar{y} \cdot \bar{z} + x \cdot y \cdot \bar{z} + x \cdot y \cdot z$

Základní součinnový tvar

Případy, kdy funkce se rovná nule

odpovídající součty

	0	0	0	$x + y + z$
kombinace	0	0	1	$x + y + \bar{z}$
xyz	0	1	0	$x + \bar{y} + z$
	1	0	0	$\bar{x} + y + z$

Odtud funkce $f_{(x,y,z)} = (x + y + z) \cdot (x + y + \bar{z}) \cdot (x + \bar{y} + z) \cdot (\bar{x} + y + z)$

1.4 Logické funkce n proměnných

Uspořádání funkcí pomocí pravdivostní tabulky ukazuje, že je počet možných funkcí pro n proměnných je 2^{2^n} . Toto číslo roste velice rychle (pro $n = 3$ je **256**).

1.4.1 Funkce jedné proměnné

Pro jednu proměnnou je počet funkcí $2^{2^1} = 4$. Jsou uvedeny v pravdivostní tabulce Tab. 1.7

Tab. 1.7. Funkce jedné proměnné

x	f_0	f_1	f_2	f_3
0	0	0	1	1
1	0	1	0	1

Platí, že: $f_0 = 0$ a $f_3 = 1$ konstanty
 $f_1 = x$ proměnná sama
 $f_2 = \bar{x}$ negace proměnné

1.4.2 Funkce dvou proměnných

Pro dvě proměnné je počet funkcí $2^{2^2} = 16$. Jsou dány pravdivostní tabulkou Tab. 1.8, kde šestnáct funkcí f_n nabývá všech možných hodnot pro všechny možné kombinace proměnných.

Tab. 1.8. Funkce dvou proměnných

x	y	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Každou funkci uvedenou v tabulce 8 můžeme zapsat v základním součtovém nebo součinném tvaru.

Příklad:

Funkce f_5 vyjádřená v základním součtovém tvaru $f_5 = \bar{x} \cdot y + x \cdot y$

Po zjednodušení funkcí najdeme:

Funkce jedné proměnné

$f_0 = 0$ a $f_{15} = 1$ konstanty
 $f_3 = x$ a $f_5 = y$ proměnná sama
 $f_{12} = \bar{x}$ a $f_{10} = \bar{y}$ negace proměnné

Dvě funkce odpovídající základním operátorům

$f_7 = x + y$ logický součet, OR
 $f_1 = x \cdot y$ logický součin, AND

Osm nových funkcí

$f_6 = x \cdot \bar{y} + \bar{x} \cdot y$ nonekvivalence, XOR, součet modulo dvě, zapisovaný $x \oplus y$

$f_9 = x \cdot y + \bar{x} \cdot \bar{y}$	ekvivalence, zapisovaná $x \equiv y$
$f_8 = \bar{x} \cdot \bar{y} = \overline{x + y}$	negace logického součtu, NOR, Piercova funkce
$f_{14} = \bar{x} + \bar{y} = \overline{x \cdot y}$	negace logického součinu, NAND, Shefferova funkce
$f_2 = x \cdot \bar{y}$	
$f_4 = \bar{x} \cdot y$	
$f_{11} = x + \bar{y}$	
$f_{13} = \bar{x} + y$	

Vidíme, že funkce jsou po dvojicích ve vztahu negace

$$f_8 = \bar{f}_7; \quad f_9 = \bar{f}_6; \quad f_{10} = \bar{f}_5; \quad \text{atd.} \dots$$

1.4.3 Funkce více než dvou proměnných

Existuje 256 funkcí tří proměnných. Můžeme však využít již známých funkcí dvou proměnných, protože

$$f_{(x,y,z)} = x \cdot f_{(1,y,z)} + \bar{x} \cdot f_{(0,y,z)}$$

Pro ověření tohoto vztahu stačí položit $x = 1$ nebo $x = 0$. Z toho je zřejmé, že $f_{(1,y,z)}$ a $f_{(0,y,z)}$ jsou funkce dvou proměnných. Stejná úvaha se může požit pro případ více než tří proměnných a libovolnou logickou funkci n proměnných je možné zapsat pomocí logických funkcí dvou proměnných. Z tohoto faktu vyplývá, že operátory nutné pro realizaci logických funkcí n proměnných jsou stejné jako ty, která slouží k realizaci logických funkcí dvou proměnných.

1.4.4 Základní nutné operátory

Z možnosti zapsat funkci n proměnných pomocí funkcí dvou proměnných vyplývá, že operátory nutné pro funkci n proměnných jsou stejné jako ty, které jsou použity pro funkci dvou proměnných, a to negace, logický součet, logický součin. V praxi to znamená, že každou logickou funkci můžeme fyzikálně realizovat pomocí tří konstrukčních prvků, které vykonávají funkci tří uvedených operátorů.

1.5 Zjednodušování zápisu logické funkce

Pomocí logických funkcí můžeme zapsat řešení různých situací a po technické realizaci takové logické funkce dostaneme logický obvod, který vyhodnotí vstupní informace a určí výsledek logické funkce. Znamená to, že každý logický operátor je realizován technickým prvkem odpovídajících vlastností. Známe-li logickou rovnici odpovídající řešené situaci snažíme se o zjednodušení (optimalizaci) této rovnice tak, aby obsahovala co nejméně logických operátorů. V praxi to znamená menší počet aktivních prvků a z toho vyplývajících výhod – menší zařízení, nižší spotřeba energie, lacinější výroba apod.

1.5.1 Algebraická minimalizace

Jako příklad si ukážeme zjednodušení zápisu majoritní funkce uvedené v Tab. 1.6, zadanou v základním součtovém tvaru

$$f = \bar{x}yz + x\bar{y}z + xy\bar{z} + xyz$$

K funkci přidáme dvakrát součin xyz , to je možné, protože platí $xyz + xyz = xyz$

$$f = (\bar{x}yz + xyz) + (x\bar{y}z + xyz) + (xy\bar{z} + xyz)$$

$$f = yz(\bar{x} + x) + xz(\bar{y} + y) + xy(\bar{z} + z) \qquad \bar{x} + x = 1$$

$$f = xy + yz + xz$$

Původní zápis logické funkce obsahoval 14 logických operátorů (3 x negaci, 3 x logický součet, 8 x logický součin). Po zjednodušení je výsledek této funkce stanoven pomocí 5 logických operátorů (2 x logický součet, 3 x logický součin). Uvedený příklad ukazuje jak lze vhodnou úpravou logické funkce připravit výhodnější podklad pro technickou realizaci.

Tato metoda je pro složitější zápisy logických funkcí s více než třemi proměnnými nepřehledná a používají se další metody.

1.5.2 Karnaughova metoda

Tato metoda nahrazuje algebraické úpravy logických výrazů geometrickými postupy, které při dodržení několika formálních pravidel garantují získání nejjednoduššího konečného tvaru logického výrazu. Logickou funkci v základním součtovém tvaru zjednodušíme takto: Sestaví se tabulka (Karnaughova mapa). Počet buněk této tabulky je roven počtu možných kombinací proměnných v logických součinech základního součtového tvaru. Každému políčku tabulky tak odpovídá jeden logický součin (jedna kombinace proměnných), přitom tabulka musí být sestavena tak, aby logické součiny v sousedních buňkách tabulky (horizontálně i vertikálně) se lišily pouze v jedné proměnné. Při vyplnění tabulky se do políček, která odpovídají logickým součinům realizovaným v dané funkci, zanesou 1 . Do ostatních buněk tabulky (tyto kombinace proměnných se nerealizují v dané funkci) se zanesou 0 . V zaplněné tabulce se hledají smyčky (mintermy) sjednocující 2, 4, 8, 16, ... (mocniny základu 2) sousední buňky tabulky tak, aby z odpovídající skupiny logických součinů byly vyloučeny 1, 2, 3, 4, ... proměnné

Příklad: $xy\bar{z}t + xy\bar{z}\bar{t} = xy\bar{z}(t + \bar{t}) = xy\bar{z}$.

Zjednodušená logická funkce se zapíše jako logický součet logických součinů odpovídajících vyznačeným smyčkám.

Při vyznačování smyček se řídíme následujícími formálními pravidly:

- smyčka musí být čtverec nebo obdélník, uvnitř mohou být jenom políčka vyplněná 1 ,
- počet políček uvnitř smyčky musí být celou mocninou čísla 2,
- jedno políčko může být v několika smyčkách,
- při vytváření smyček se vrchní a spodní řádek počítají jako sousední (totéž platí pro krajní sloupce),
- počet smyček musí být co možná nejmenší,
- rozměry smyček musí být co možná největší (smyčka obsahuje co nejvíce políček),

- každé políčko obsahující 1 musí vejít alespoň do jedné smyčky (políčko, které nemá
- sousední políčko obsahující 1 , tvoří samostatnou smyčku),
- všechny smyčky musí být při sestavení konečného logického součtu použity.

Cvičení

1. Zjednodušíme pomocí Karnaughovy metody logickou funkci $f_{(x,y,z)}$, která je zadána tabulkou 1.4 a jejíž základní součtový tvar je určen výrazem:

$$f_{(x,y,z)} = \bar{x} \cdot \bar{y} \cdot z + \bar{x} \cdot y \cdot \bar{z} + x \cdot \bar{y} \cdot z + x \cdot y \cdot \bar{z} + x \cdot y \cdot z$$

Řešení

1. Tomuto příkladu odpovídá Karnaughova mapa pro tři proměnné (obr. 1.1). Výsledná zjednodušená logická funkce má tvar: $f(x, y, z) = x \cdot y + y \cdot \bar{z} + \bar{y} \cdot z$

Obr. 1.1. Karnaughova mapa

	$\bar{x} \cdot \bar{y}$	$\bar{x} \cdot y$	$x \cdot y$	$x \cdot \bar{y}$
\bar{z}	0	1	1	0
z	1	0	1	1

1.5.3 Algoritmické metody

Ke zjednodušování funkcí více než čtyř proměnných existují různé systematické metody. Příkladem tohoto způsobu je například Quinova-McCluskeyho metoda.

Shrnutí

Pojmy k zapamatování

Kontrolní otázky

1. Jaké číselné soustavy se používají v informatice?
2. Co je Boolova algebra a jaké jsou její základní operace?
3. Co je to logická funkce?
4. Co znamená neúplná definice logické funkce?
5. Jaké jsou normální formy funkčních výrazů a jak se získají?
6. Jaké znáte operátory, odvozené z funkcí se dvěma vstupy?

Úkoly k textu

1. Zadejte vlastní funkci se třemi vstupy ve formě tabulky. Napište ÚDNF a ÚKNF a minimalizujte je.
2. Zapište výraz vlastní logické funkce v ÚDNF a převed'te je na ÚKNF.
3. Zapište výraz vhodné vlastní logické funkce v ÚDNF a popište příklady úprav algebraické minimalizace.

2 Základní logické operátory

Studijní cíle: Cílem kapitoly je poskytnout představu o realizaci kombinačních funkcí pomocí vybraných operátorů a vhodných úprav výrazů.

Klíčová slova: Operátor negace, konjunkce a disjunkce, NAND, NOR,

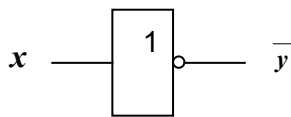
Potřebný čas: 2 hodiny

2.1 Úplný systém logických funkcí

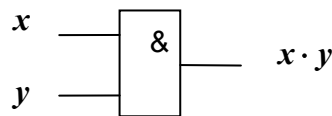
V kapitole 1 (v článku 1.4.4.) jsme viděli, že jakákoliv logická funkce libovolného počtu proměnných může být zapsána pomocí operátorů, negace, logického součinu (konjunkce) a logického součtu (disjunkce). Tato skupina operátorů tvoří tzv. úplný systém logických funkcí. V praxi stačí zkonstruovat tři fyzikální struktury – logické členy (mechanické, pneumatické, elektrické atd.), které realizují tyto tři funkce. Na obr. 2.1. jsou symbolické značky logických členů – negace, konjunkce a disjunkce, které se používají v elektronických schématech.

Obr. 2.1. Logické operátory (symbolické značení)

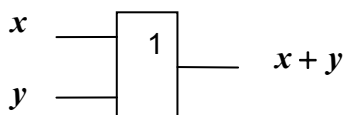
a) negace



b) konjunkce (logický součin)



c) disjunkce (logický součet)



2.2 Minimální úplný systém logických funkcí

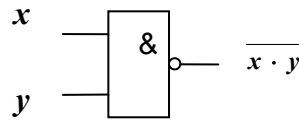
První logické systémy odvozené z obecné teorie byly založeny na třech různých fyzikálních strukturách, pomocí kterých byly realizovány tři základní operátory. Je však možné dokázat, že úplný systém logických funkcí může být tvořen pouze jedinou logickou funkcí (operátorem) **NAND** nebo **NOR** (obr.1.5). Úplný systém logických funkcí tvořený operátorem **NAND** a nebo **NOR** se nazývá minimálním úplným systémem logických funkcí. To znamená, že libovolnou logickou funkcí je možné realizovat pomocí kombinací jednoho operátoru. Pravdivostní tabulky logických operátorů **NAND** a **NOR** a jejich symbolické značky používané v elektronice jsou uvedeny na obr.2.2.

Obr. 2.2. Pravdivostní tabulky a symbolické značky operátorů **NAND** a **NOR**

NAND

Tab. 2.1. Negovaný součin

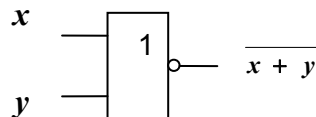
x	y	$\overline{x \cdot y}$
0	0	1
0	1	1
1	0	1
1	1	0



NOR

Tab. 2.2 .Negovaný součet

x	y	$\overline{x + y}$
0	0	1
0	1	0
1	0	0
1	1	0

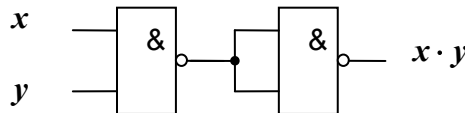
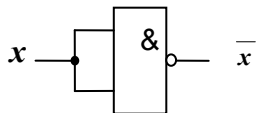


Skutečnost, že operátory *NAND* nebo *NOR* tvoří úplný systém logických funkcí je možno dokázat realizací základních logických operátorů negace, logického součinu a logického součtu pomocí logických operátorů *NAND* nebo *NOR*.

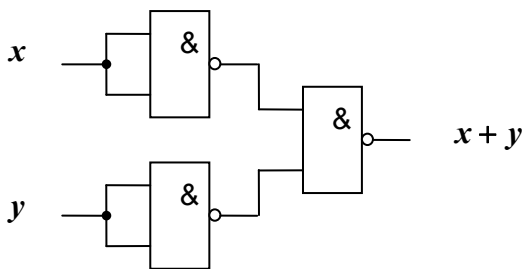
2.2.1 Realizace operátorů NOT, AND a OR pomocí NAND

Negace: $\overline{x} = \overline{x \cdot x}$

Logický součin: $x \cdot y = \overline{\overline{x \cdot y}} = \overline{\overline{x} \cdot \overline{y}}$



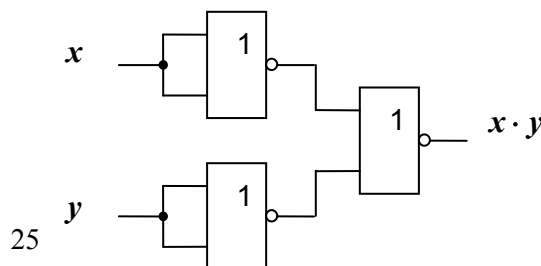
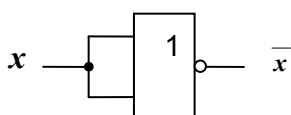
Logický součet: $x + y = \overline{\overline{x + y}} = \overline{\overline{\overline{x} \cdot \overline{y}}}$



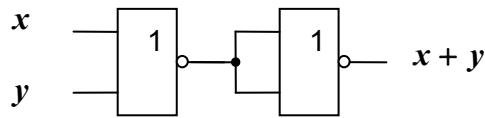
2.2.2 Realizace operátorů NOT, AND a OR pomocí NOR

Negace: $\overline{x} = \overline{x + x}$

Logický součin: $x \cdot y = \overline{\overline{x \cdot y}} = \overline{\overline{\overline{x + x} + \overline{y + y}}}$



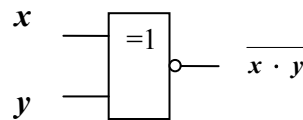
Logický součet: $x + y = \overline{\overline{x + y}} = \overline{\overline{(x + y)} + \overline{(x + y)}}$



2.2.3 Realizace operátoru nonekvivalence pomocí NAND

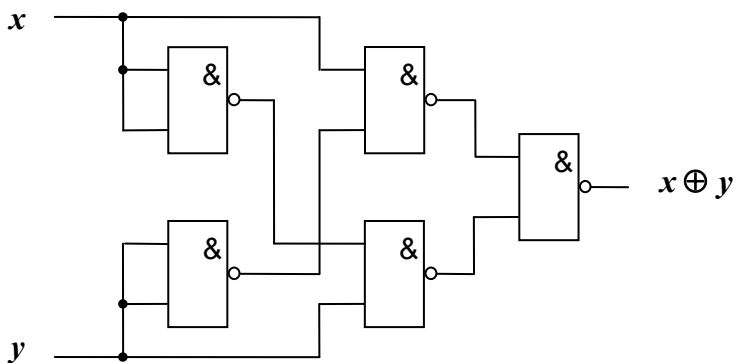
Tab. 2.3. Nonekvivalence (XOR)

x	y	$x \oplus y$
0	0	0
0	1	1
1	0	1
1	1	0



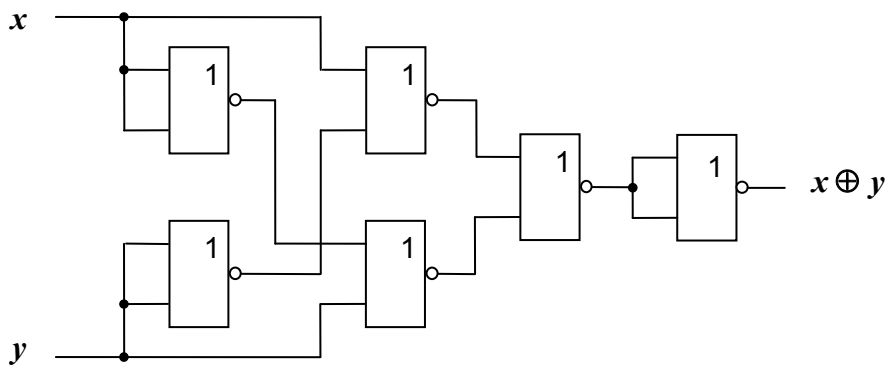
Operátor realizovaný logickými členy NAND

$$x \oplus y = x \cdot \overline{y} + \overline{x} \cdot y = \overline{\overline{x \cdot \overline{y} + \overline{x} \cdot y}} = \overline{\overline{x \cdot \overline{y}} \cdot \overline{\overline{x} \cdot y}} = \overline{\overline{x \cdot \overline{y}} \cdot \overline{\overline{\overline{x} \cdot y}}}$$



Operátor realizovaný logickými členy NOR

$$x \oplus y = x \cdot \overline{y} + \overline{x} \cdot y = \overline{\overline{x \cdot \overline{y} + \overline{x} \cdot y}} = \overline{\overline{(x + \overline{y})} + \overline{(\overline{x} + y)}}$$



2.2.4 Realizace logické funkce pomocí logických operátorů NAND

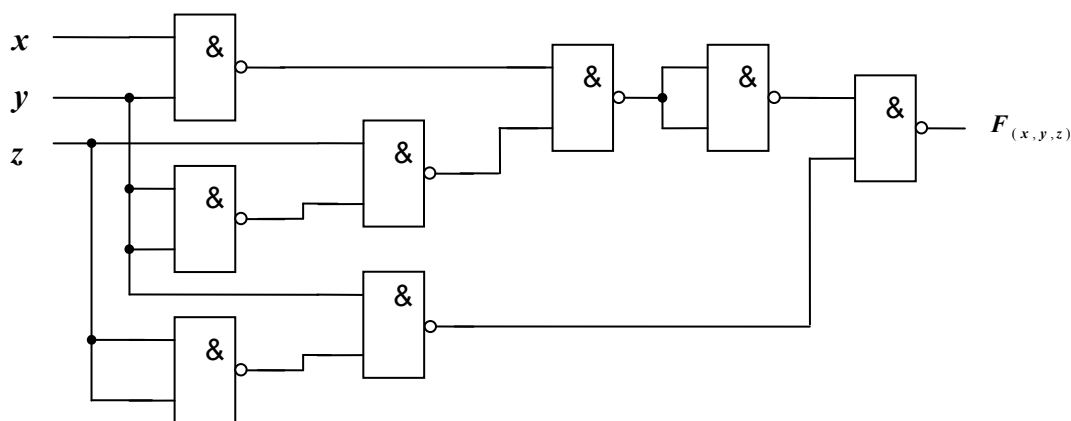
Jako příklad realizace zadané logické funkce pomocí logického operátoru NAND si uvedeme realizaci logické funkce zadané pravdivostní tabulkou (Tab. 1.4).

Při realizaci logických funkcí pomocí logických operátorů NAND a NOR obecně postupujeme takto:

- z pravdivostní tabulky sestavíme základní součtový nebo součinnový tvar logické funkce,
- metodou Karnaughových diagramů zjednodušíme zápis logické funkce,
- aplikací Morganových zákonů transformujeme zjednodušený tvar logické funkce do tvaru, který obsahuje pouze kombinaci operátorů NAND a nebo NOR.

Tak logické funkci zadané tabulkou (Tab. 1.4) odpovídá základní součtový tvar (1.1), který byl zjednodušen do tvaru (1.4) a funkce může být vyjádřena pouze pomocí operátorů NAND takto:

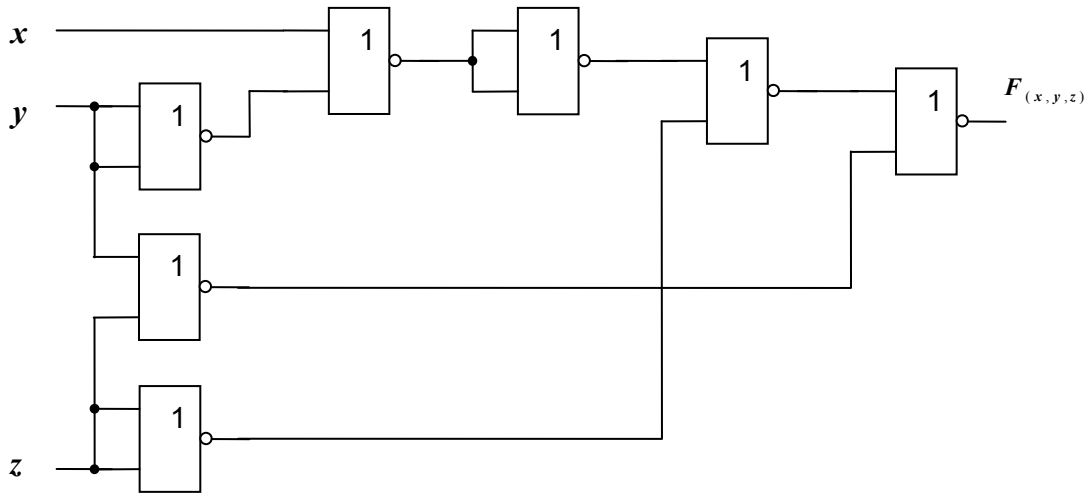
$$\begin{aligned}
 F_{(x,y,z)} &= x \cdot y + \bar{y} \cdot z + y \cdot \bar{z} = x \cdot y + \bar{y} \cdot z + y \cdot \bar{z} = x \cdot y \cdot \bar{y} \cdot z + y \cdot \bar{z} = x \cdot y \cdot \bar{y} \cdot z + y \cdot \bar{z} = \\
 &= x \cdot y \cdot \bar{y} \cdot z + x \cdot y \cdot \bar{y} \cdot z + y \cdot \bar{z} \cdot z
 \end{aligned}$$



Stejná logická funkce (tabulka 1.4) může být vyjádřena základním součinnovým tvarem (1.2), který může být zjednodušen do tvaru (1.6) a logická funkce může být realizována pomocí operátorů NOR

$$F_{(x,y,z)} = (y+z) \cdot (x + \overline{y+z}) = (y+z) \cdot (x + \overline{y+z}) = \overline{\overline{(y+z)} + (x+y)} + z =$$

$$= \overline{\overline{(y+z)} + (x+(y+y))} + (z+z)$$



Kontrolní otázky

7. *Jaké základní operátory se používají v reálných kombinačních obvodech?*
8. *Jak postupovat při návrhu realizace logické funkce?*

Úkoly k textu

4. Navrhněte dvoubitovou sčítačku modulo 2. Pokuste se obvod realizovat doporučeným postupem pomocí členů NAND.

3 Logické obvody

Studijní cíle: Po prostudování kapitoly student zvládne přehled typů, charakteristiku a funkční popis činnosti základních kombinačních obvodů.

Klíčová slova: Komparátor, multiplexor, dekodér, sčítačka,

Potřebný čas: 2 hodiny

3.1 Kombinační logické obvody

Kombinační logické obvody jsou takové logické obvody, ve kterých stavy na výstupech závisí pouze na okamžitých kombinacích vstupních proměnných a nezávisí na jejich předchozích hodnotách, s výjimkou krátkého přechodového děje. Jedné kombinaci vstupních proměnných odpovídá jediná výstupní kombinace funkčních hodnot. Kombinační logické obvody nemají žádnou paměť předchozích stavů.

3.1.1 Komparátor

Komparátor je logické zařízení, které provádí srovnání dvou binárních veličin A a B a určuje, zda je $A = B$, $A < B$, $A > B$. Zařízení má tři výstupy: - "větší", "menší", "rovno". Nejjednodušší je jednobitový komparátor, který může být popsán rovnicemi

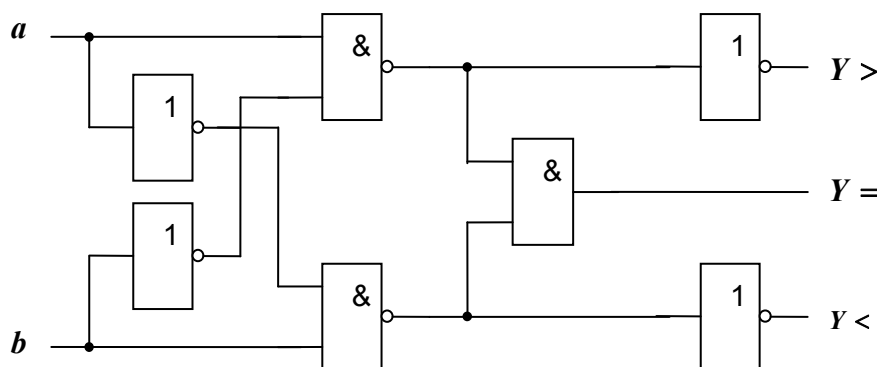
$$Y_{=} = Y(a = b) = a \cdot b + \bar{a} \cdot \bar{b} \quad Y_{>} = Y(a > b) = a \cdot \bar{b} \quad Y_{<} = Y(a < b) = \bar{a} \cdot b$$

Pomocí zákonů Booleovy algebry tyto rovnice mohou být upraveny do následujících tvarů

$$Y_{=} = \overline{(a \cdot \bar{b}) \cdot (\bar{a} \cdot b)} \quad Y_{>} = \overline{\overline{a \cdot \bar{b}}} \quad Y_{<} = \overline{\overline{\bar{a} \cdot b}}$$

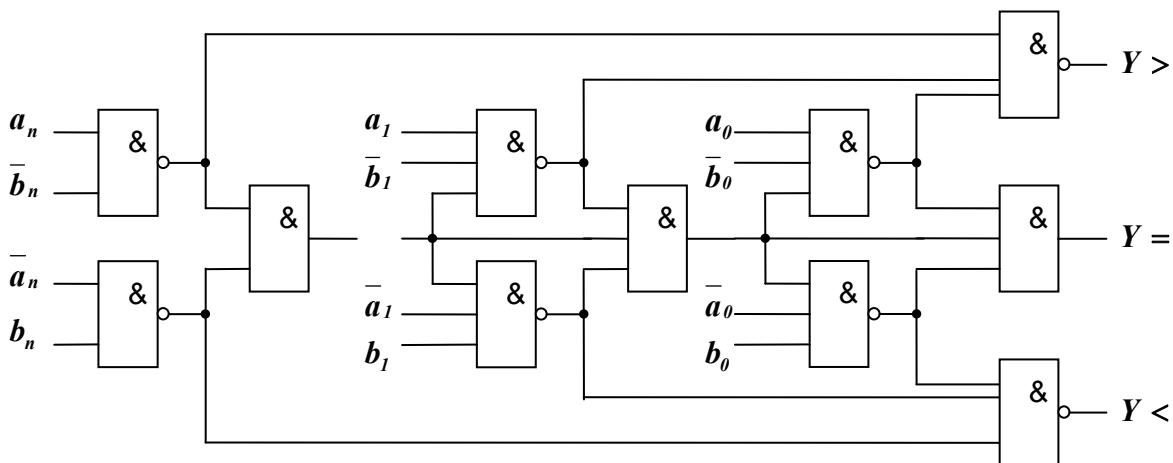
V souladu s těmito výrazy komparátor může být realizován pomocí logických operátorů **AND**, **NAND** a **NOT** (obr.3.1).

Obr.3.1 Jednobitový komparátor



Při použití jednobitových komparátorů je možné realizovat komparátor, který porovnává binární čísla s libovolným počtem řádů $A(a_0, a_1, \dots, a_n)$, $B(b_0, b_1, \dots, b_n)$ (obr.3.2). Zařízení provádí odděleně porovnání každého bitu, počínaje bitem největšího významu. Detekovaná nerovnost ve vyšším řádu nemůže být změněna vlivem bitů řádů nižšího významu, protože tyto jsou blokovány (centrální členy AND na obr.3.2).

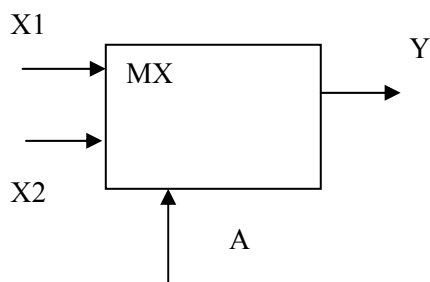
Obr.3.2. Mnohobitový komparátor



3.1.2 Multiplexor (Multiplexer)

Charakteristika :

- multiplexor je kombinační log. síť, která přepíná signál (log. úroveň) z více vstupů na jeden výstup, pracuje jako přepínač. Značí se MX.



Pomocí binární kombinace na vstupním vektoru A se vybírá log. úroveň na jednom z prvků vstupního vektoru X na výstup Y

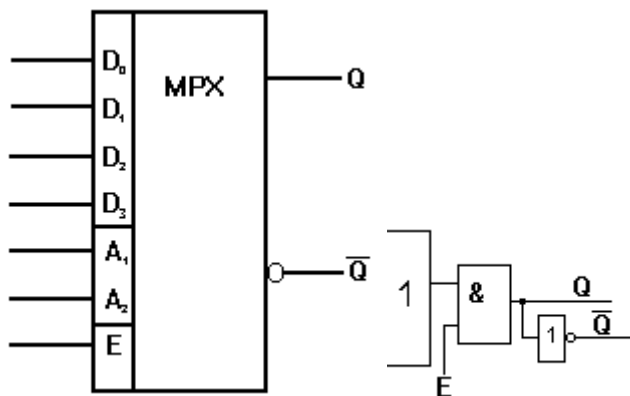
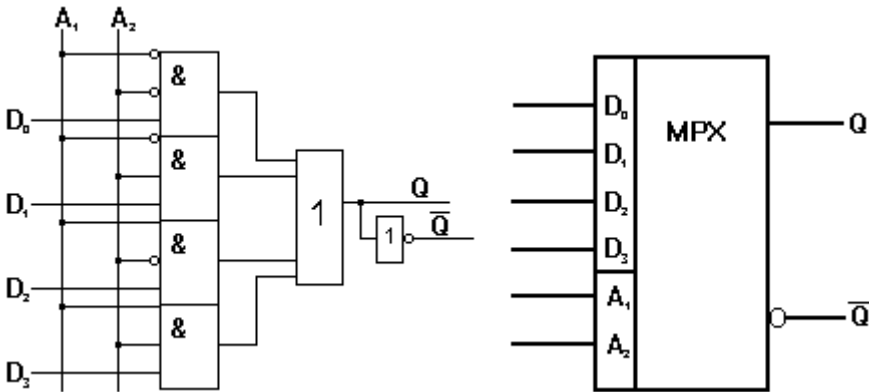
4-vstupý multiplexor

4 datové vstupy

2 adresové vstupy

A ₁	A ₂	Q
0	0	D ₀
0	1	D ₁
1	0	D ₂
1	1	D ₃

$$Q = \bar{A}_1 \cdot \bar{A}_2 \cdot D_0 + \bar{A}_1 \cdot A_2 \cdot D_1 + A_1 \cdot \bar{A}_2 \cdot D_2 + A_1 \cdot A_2 \cdot D_3$$



ENABLE řídicí vstup pro výběr obvodu

- Použití:
- data selector
 - generátor logických funkcí

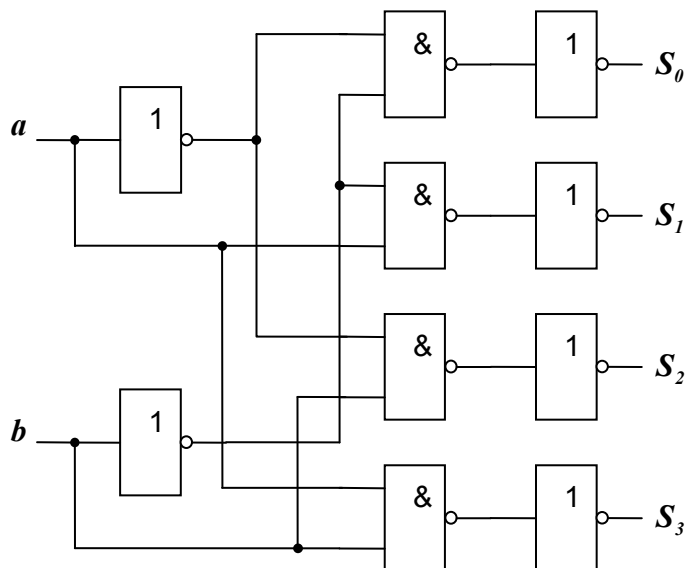
3.1.3 Binární dekodér

Binární dekodér je logický obvod s n tzv. adresovými vstupy a s 2^n výstupy, ze kterých je v daném časovém okamžiku aktivní pouze jeden. Číslo aktivního výstupu odpovídá binární hodnotě kombinace přivedené na vstupy. Obvodu se používá pro výběr jednoho prvku z 2^n prvků. Tento typ kombinačního obvodu může sloužit také jako dekodér místa v paměti. Na vstup dekodéru přivedeme bity adresní sběrnice, podle hodnoty adresy na vstupu dekodéru je aktivní výstup odpovídající místu v paměti. Jako příklad si uvedeme realizaci dekodéru se dvěma vstupy a čtyřmi výstupy (tabulka 3.1, obr.3.3).

Tab. 3.1. Pravdivostní tabulka binárního dekodéru

b	a	S_0	S_1	S_2	S_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

Obr. 3.3. Binární dekodér



3.1.4 Sčítačka

Často je potřebné provádět s binárně kódovanými veličinami aritmetické operace. Aritmetické operace v binárním kódu se řídí stejnými pravidly jako aritmetické operace v soustavě desítkové. Tak pravidla sčítání v binární soustavě můžeme vyjádřit takto:

$$0 + 0 = 0$$

$$0 + 1 = 1$$

$$1 + 1 = 10$$

Sčítání a odečítání čísel je možné realizovat pomocí kombinačních obvodů. Sčítačka je logické zařízení, které v souladu s pravidly sčítání bude v každém dvojkovém řádu (bitu) realizovat součet příslušných dvou bitů, při tom bude akceptovat přenos z nižšího řádu a formovat přenos do řádu vyššího. Funkce jednobitové sčítačky může být popsána pravdivostní tabulkou. Z pravdivostní tabulky mohou být vytvořeny logické funkce, popisující výstup S_i a přenos r_i . Po úpravě můžeme tyto výrazy převést do tvaru

$$S_i = a_i \oplus b_i \oplus r_{i-1} \quad r_i = a_i \cdot b_i + (a_i + b_i) \cdot r_{i-1}$$

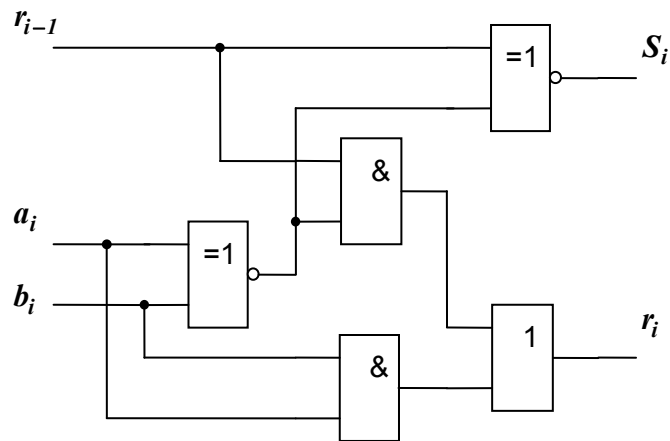
Logické zařízení pracující v souladu s těmito výrazy je ukázáno na obr.3.4.

Tab. 3.2 Pravdivostní tabulka

jednobitové sčítačky

a_i	b_i	r_{i-1}	S_i	r_i
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Obr. 3.4. Jednobitová sčítačka



Sčítačka n -bitová bude vytvořena paralelním spojením n jednobitových sčítaček. V praxi se používají paralelní sčítačky s délkou 8, 16, 32 atd. bitů, podle délky slova, se kterým dané zapojení pracuje.

Kontrolní otázky

9. Jaké typy kombinačních obvodů znáte a jakou mají funkci?

Úkoly k textu

5. Definujte logickou funkci jednoduchého multiplexoru, se vstupy X1, X2 a A, který přepíná vstupy X1 a X2 na výstup Y podle vstupu A (např. pro A=1 je Y=X2). Pokuste se obvod realizovat doporučeným postupem pomocí členů NAND.

4 Zobrazení čísel, kódy

Studijní cíle: Kapitola je zaměřena na kódování informací a zvládnutí kódů pro reprezentaci čísel pro základní operace, kontrolu správnosti přenosu a opravy chyb pomocí samoopravného kódu.

Klíčová slova: Přímý a doplňkový binární kód, BCD, Grayův, p z n kód, parita, Hammingova vzdálenost, rozšířený Hammingův kód, CRC

Potřebný čas: 2 hodiny

Počítač může pracovat nejen s čísly, ale i s písmeny abecedy, zkratkami a grafickými značkami. Souhrnně se jim říká abecedně číslicové (alfanumerické) znaky. Před zpracováním v počítači je třeba znaky převést do tvaru srozumitelného pro počítač, tj. přiřadit jim určité kombinace bitů. Tento převod se nazývá kódování.

Kód je předpis pro jednoznačné přiřazení určité kombinace bitů příslušnému znaku. Kombinaci bitů zobrazující znak se říká kódové slovo nebo jednoduše kód. V počítačích se používají různé druhy kódů, které odpovídají potřebám pro zpracování dat v operačních obvodech, pro záznam dat na záznamová média a pro zabezpečení dat proti chybám.

Kódová slova pro různé kódy se řadí do kódových tabulek. Přiřazování kódových slov určitým znakům se říká kódování. Ke kódování se používá generátor kódu, který může obsahovat logické nebo paměťové obvody. Původní znak se získá z kódového slova dekódováním. K dekódování se používá dekodér, obsahující logické nebo paměťové obvody.

4.1 Přímé kódy

Nejpřirozenější způsob reprezentace čísel v počítači je založen na použití vhodného číselného základu (souvisejícího s dvojkovým) a fixního umístění desetinné čárky. Tak můžeme reprezentovat čísla celá i reálná (přesněji pro konečnou přesnost reprezentace čísla v počítači spíše jistá čísla racionální). Maximální chyba mezi požadovaným reálným číslem a jeho prostorově omezenou reprezentací v počítači je polovina rozsahu nejnižšího řádu reprezentace.

*Přímý kód -
znaménko a
absolutní hodnota
(sign-magnitude)*

Př. (viz. [1]) Mějme pro jednoduchost v desítkové soustavě reprezentaci dva řády a znaménko, tedy interval **(-99, 99)**. Přesnost je tedy **1** (rozdíl mezi sousedními čísly reprezentace), maximální chyba reálného čísla z daného intervalu je **0,5** – je to největší vzdálenost reálného čísla k nejbližšímu číslu reprezentace.

Při reprezentaci čísel používáme nejčastěji vážený poziční kód. Obecně je dán vztahem:

$$\text{Hodnota čísla} = \sum_{i=-m}^{n-1} a_i \cdot z^i$$

V logických systémech se pro práci s čísly používá čtyř číselných základů (z):

- Základ 2 se symboly 0 a 1 (a_i).
- Základ 10 se symboly 0, 1, ..., 8, 9 (a_i).
- Základ 8 (oktálový) se symboly 0, 1, ..., 6, 7 (a_i).
- Základ 16 (hexadecimální) se symboly 0, 1, ..., 8, 9, A, B, C, D, E, F (a_i).

Tab. 4.1. Zápisy čísel

desítkově	dvojkově	osmičkově	šestnáctkově
0	0	0	0
1	I	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A

desítkově	dvojkově	osmičkově	šestnáctkově
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10
17	10001	21	11
18	10010	22	12
19	10011	23	13
20	10100	24	14
21	10101	25	15

Číslo v soustavě o základu z^k (kde z a k jsou přirozená čísla) lze převést do soustavy o základu z jednoduše tak, že každou k -tici číslic nižší soustavy nahradíme číslicí soustavy vyšší. Jednoduše lze převádět mezi soustavou dvojkovou a osmičkovou nebo mezi dvojkovou a šestnáctkovou; jednoduše nelze převádět např. mezi dvojkovou a desítkovou. Uvedme příklad převodu čísla z dvojkové soustavy do šestnáctkové:

$$0011|1011|0100_2 = 3B4_{16}$$

Zpětný převod do dvojkové soustavy probíhá tak, že každou číslici šestnáctkové soustavy převedeme na čtveřici číslic dvojkové soustavy. Nelze-li uplatnit jednoduchý převod, použijeme obecný algoritmus převodu.

Dvojkově desítkový kód (BCD)

Pro kódování desítkových číslic 0 až 9 se používá dvojkově desítkový kód (BCD). V tomto kódu je každá desítková číslice D vyjádřena kódovým slovem se čtyřmi bity ve dvojkové soustavě (Tab. 4.2)

Tab. 4.2. Kódová tabulka kódu BCD

D	b_1	b_2	b_3	b_4
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0

D	b_1	b_2	b_3	b_4
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

4.1.1 Princip změny základu

Metoda postupného odečítání

Metoda se snadno použije pro převod od libovolného základu k k základu 2. Původní číslo se rozkládá odečítáním zmenšujících se mocnin základu 2, přičemž se hledá mocnina čísla 2 rovná převáděnému číslu nebo menší. Například převod čísla 190 (desítkově) k základu 2 vypadá takto:

$$190_{10} \rightarrow N_2$$

$$2^7 \quad 2^6 \quad 2^5 \quad 2^4 \quad 2^3 \quad 2^2$$

$$\begin{array}{cccccc}
 190 - 128 = 62 & 62 - 0 = 62 & 62 - 32 = 30 & 30 - 16 = 14 & 14 - 8 = 6 & 6 - 4 = 2 \\
 \text{I} & \text{0} & \text{I} & \text{I} & \text{I} & \text{I}
 \end{array}$$

$$\begin{array}{cc}
 2^1 & 2^0 \\
 2 - 2 = 0 & 0 - 0 = 0 \\
 \text{I} & \text{0}
 \end{array}$$

Odtud: $190_{10} = 1011110_2$

Metoda postupného dělení

Váha každé číslice v polyadické soustavě se lineárně zvyšuje s pozicí číslice v číselném obraze, reprezentující číslo. Jestliže označíme jednotlivé číslice zprava od 0 do $n-1$, pak pro libovolné číslo x můžeme zapsat výraz reprezentující toto číslo jako

$$\begin{aligned}
 x &= a_{n-1} \cdot r^{n-1} + a_{n-2} \cdot r^{n-2} + \dots + a_0 \cdot r^0 \\
 &= a_0 + r(a_1 + r(a_2 + \dots + r(a_{n-2} + r \cdot a_{n-1}) \dots))
 \end{aligned}$$

Tento způsob zápisu nabízí řešení pro algoritmus konverze mezi dvěma číselnými. Jestliže totéž číslo zapíšeme v soustavě se základem t pak jeho číslice $b_0 \dots b_{n-1}$ a základ budou opět odpovídat rovnici

$$x = b_0 + t \cdot (b_1 + t \cdot (b_2 + \dots + t \cdot (b_{n-2} + t \cdot b_{n-1}) \dots))$$

Při dělení rovnice základem t dostaneme polynom ve tvaru

$$\frac{x}{t} = Q + R$$

kde $Q = b_1 + t(b_2 + t \cdot (b_3 \dots t \cdot b_{n-1}) \dots)$

$$R = b_0$$

Dalším dělením získáme koeficienty b_1, b_2, \dots , což můžeme ukázat na příkladu

Příklad:

Číslo 25_{10} se má vyjádřit v binární soustavě .

	25		
: 2	12	zbytek = 1	b_0
: 2	6	zbytek = 0	b_1
: 2	3	zbytek = 0	b_2
: 2	1	zbytek = 1	b_3
: 2	0	zbytek = 1	b_4

Odtud je číslo $25_{10} = 11001_2$

Příklad:

Číslo 1358_{10} se má vyjádřit v hexadecimální soustavě (základ šestnáct).

$$\begin{array}{r}
 1358 \\
 : 16 \quad 84 \quad \text{zbytek} = 14 \quad b_0 \\
 : 16 \quad 5 \quad \text{zbytek} = 4 \quad b_1 \\
 : 16 \quad 0 \quad \text{zbytek} = 5 \quad b_2
 \end{array}$$

Odtud je číslo $1358_{10} = 54E_{16}$

Pro kontrolu můžeme provést součty decimálních váhových koeficientů

$$\begin{aligned}
 54E_{16} &= (5 \cdot 16^2) + (4 \cdot 16^1) + (E \cdot 16^0) = (5 \cdot 256) + (4 \cdot 16) + (E \cdot 1) = \\
 &1280 + 64 + 14 = 1358
 \end{aligned}$$

Příklad:

Převed'te devítkové číslo 134_9 (základ 9) do sedmičkové soustavy (základ 7).

V tomto případě je tedy $r = 7$ a $t = 9$. Postupné dělení bude vypadat takto:

$$\begin{array}{r}
 (134)_9 = (112)_{10} \\
 : 7 \quad (16)_{10} = (17)_9 \quad \text{zbytek} = 0 \quad b_0 \\
 : 7 \quad 2 \quad \text{zbytek} = 2 \quad b_1 \\
 : 7 \quad 0 \quad \text{zbytek} = 2 \quad b_2
 \end{array}$$

Hledaný sedmičkový obraz čísla je $b_2b_1b_0 = 220_7$

Pro kontrolu můžeme provést součty decimálních váhových koeficientů, tedy:

$$134_9 = (1 \cdot 9^2) + (3 \cdot 9^1) + (4 \cdot 9^0) = 81 + 27 + 4 = 112 \text{ (desítkově)}$$

$$220_7 = (2 \cdot 7^2) + (2 \cdot 7^1) + (0 \cdot 7^0) = 98 + 14 + 0 = 112 \text{ (desítkově)}$$

4.2 Aditivní kód

Používá se pro reprezentaci kladných i záporných čísel takovým způsobem, že výsledná binární reprezentace představuje nezáporné číslo, které vznikne součtem kódovaného čísla a domluvené konstanty. Tato konstanta je nejčastěji definována jako polovina maximálního kladného čísla, které by se do daného paměťového prostoru vešlo při použití přímého kódu bez znaménka – je to tzv. polovina modulu řádové mřížky.

*Aditivní kód -
s posunutou nulou
(biased)*

Příklad:

Pracujeme např. s 1B. To znamená, že modul má velikost $(256)_d$. Pro reprezentaci čísla např. $(-10)_d$ dostaneme výpočtem $(-10)_d + (256)_d / 2 = -10 + 128 = 118$. V paměti je uloženo číslo $(118)_d$.

4.3 Doplnkový kód pro záporná čísla

Doplňkový kód je definován pro nezáporná čísla přímo jako číslo v binární soustavě. Záporná čísla jsou ve formě binárního doplňku do modulu.

Doplňkový kód -
 $= x$ pro
 $x > 0$ nebo $x = 0$
 $= M+x$ pro
 $x < 0$
(2's complement)

Odčítání čísel v přímém kódu je složité, protože vyžaduje odčítačku, kterou není aritmetická jednotka běžně vybavena. Potom je třeba zpracovat znaménkové bity. V doplňkovém kódu je zobrazení záporných čísel výhodnější, protože pak lze odčítání převést na sčítání. Záporné číslo ve dvojkové soustavě se zobrazí v doplňkovém kódu (v dvojkovém doplňku) tak, že se bity invertují, tj. jedničky se změňí na nuly a nuly na jedničky. K takto vzniklému číslu se přičte jednička. Potom po určité úpravě může sčítačka vykonávat operaci odčítání. První operand vstupuje do sčítačky přímo. Druhý operand vstupuje do sčítačky v doplňkovém kódu, který se vytvoří v bloku investorů (po inverzi všech bitů se mezivýsledek nazývá jedničkový doplněk) a přičtením jedničky (dostáváme dvojkový doplněk) na přenosovém vstupu sčítačky. Kladný výsledek operace je v přímém kódu a záporný výsledek je v doplňkovém kódu. U počítačů s délkou slova kratší, než je délka odčítaných dvojkových čísel, se čísla musejí odčítat postupně, počínaje nejnižším dvojkovým řádem. Pro odčítání je potřeba sestavit program.

Princip.

Předpokládejme, že pracujeme s N -bitovými čísly. Hodnota těchto čísel (kladných) leží mezi 0 a $2^N - 1$. Protože sčítačky mají pouze N stupňů, mají výsledná čísla po výpočtu rovněž N bitů, tzn. jsou vyjádřena modulo 2^N . Pak se rozdíl $A - B$ rovněž rovná $A - B + 2^N$, kde váha 2^N se zanedbává.

Zavedeme B' , číslo získané z čísla B negací všech bitů. Toto číslo B' se nazývá logický doplněk (komplement) čísla B nebo též jedničkový doplněk.

Příklad

$$B = 10110100 \qquad B' = 01001011$$

Provedeme aritmetický součet $B + B'$

$$B \quad 10110100$$

$$B' \quad 01001011$$

$$11111111 = 2^N - 1$$

Pro případ N -bitového čísla B máme vztah

$$B + B' = 2^N - 1, \text{ odkud } B = 2^N - 1 - B'$$

Aplikujeme vztah na $A - B$

$$A - B = A - 2^N + 1 + B' = A + B' + 1 - 2^N$$

Zanedbáme 2^N , protože přesahuje rozsah sčítačky, a dostaneme

$$A - B = A + B' + 1$$

Odtud dostáváme jednoduché pravidlo:

Rozdíl dvou čísel A, B získáme tak, že se číslo A sečte s logickým doplňkem čísla B a k výsledku se připočte jednička.

Příklad: Odečtení čísel $125 - 92$ pomocí osmibitové sčítačky.

8	7	6	5	4	3	2	1	0	<i>řád</i>
256	128	64	32	16	8	4	2	1	<i>váhy</i>
	0	1	1	1	1	1	0	1	<i>A</i>
	0	1	0	1	1	1	0	0	<i>B</i>
	0	1	1	1	1	1	0	1	<i>A</i>

$$\begin{array}{c|cccccccc}
 + & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & B' \\
 + & & & & & & & & 1 & +I \\
 \hline
 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & A+B'+I
 \end{array}$$

4.4 Reprezentace čísel s řádovou čárkou

Reálná čísla jsou v počítači reprezentována pomocí čísel, jejichž přesnost je omezena na požadovaný počet platných číslic (opět proti sobě stojí velikost obsazené paměti – počet Bytů - při fyzickém uložení datové reprezentace reálného čísla a jeho přesnost – počet platných číslic). V zásadě existují dvě možnosti reprezentace – čísla uložená s fixní řádovou čárkou a čísla s pohyblivou řádovou čárkou.

4.4.1 Čísla s fixní řádovou čárkou

Pro reprezentaci nejběžnějších reálných (racionálních) čísel (tedy ani extrémně velkých, ani extrémně malých) můžeme použít přirozené rozšíření přímého, nebo doplňkového kódu. Řádová čárka má pevnou pozici ve váženém pozičním kódu.

Př. ve dvojkové soustavě

$$\begin{array}{c}
 \text{---,---} \\
 \mathbf{101,01} \\
 (\mathbf{101,01})_2 = 1 \cdot 4 + 1 \cdot 1 + 1 \cdot 0,25 = (5,25)_{10}
 \end{array}$$

Převod zlomku (necelé části čísla) z jedné číselné soustavy do druhé potom probíhá podle podobných zásad jako je např. metoda dělení pro celá čísla, tak že pro zlomky se použije podobný postup, ale operací je násobení. Místo zbytku po dělení potom registrujeme pro příslušné řády 0, pokud výsledek operace neobsahuje celočíselnou část, nebo 1, pokud výsledek obsahuje celočíselnou část. Při dalším násobení bereme jen neceločíselnou část předchozího výsledku.

Ilustrační příklad:

Je dáno číslo (podobně postupujeme pro neceločíselnou část obecnějšího čísla – např. $23,625_{10}$) $0,625_{10}$ a má se vyjádřit v binární soustavě. Postupně násobíme 2 a upravujeme výsledky násobení odebráním celočíselné části

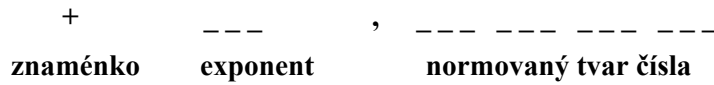
$$\begin{array}{rcll}
 & \mathbf{0,625} & & \\
 \mathbf{0,625} \cdot 2 & \mathbf{1,250} & \text{celočíselná část} = \mathbf{1} & \mathbf{b_{-1}} \\
 \mathbf{0,250} \cdot 2 & \mathbf{0,5} & \text{celočíselná část} = \mathbf{0} & \mathbf{b_{-2}} \\
 \mathbf{0,5} \cdot 2 & \mathbf{1,0} & \text{celočíselná část} = \mathbf{1} & \mathbf{b_{-3}} \\
 \mathbf{0} & & &
 \end{array}$$

Odtud je číslo $0,625_{10} = ,101_2$

4.4.2 Čísla s pohyblivou řádovou čárkou

Pro ilustraci přístupu k řešení problému reprezentace reálných čísel (čísel s pohyblivou desetinnou čárkou) použijeme příkladu z literatury [1]. Typická je šablona fyzického uložení, respektující např. zvolenou přesnost čísla (počet platných číslic), které je

uloženo v semilogaritmickém tvaru. V navrženém případě pro jednoduchost pracujeme se dvěma B, při mapování jednotlivých bitů postupujeme zleva :



- první bit použijeme pro reprezentaci znaménka čísla
- další část (např. 3 bity) jsou využity pro reprezentaci exponentu. V tomto případě jsme kapacitou omezeni na interval exponentů od -4 do +3. S problémem záporného exponentu se vypořádáme tak, že ke skutečnému exponentu připočteme číslo 4.
- poslední část paměťového prostoru je rezervována pro číslo ve znormovaném tvaru – tj. výsledek případného převodu výchozího čísla na semilogaritmický tvar, ve kterém je první nenulová platná číslice v poloze b_{-1} a toto číslo je násobeno základem použité číselné soustavy (8), povýšené patřičným exponentem.

Ilustrační příklad:

Máme číslo $(0,254 \cdot 10^3)_{10}$ uložit v navrženém formátu. Postup:

1. Převod do soustavy se základem 8 a normalizace do semilogaritmického tvaru

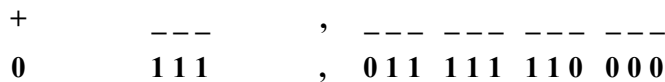
$$(0,254 \cdot 10^3)_{10} = (254)_{10} = (376)_8 = (0,376 \cdot 8^3)_8$$

2. Naplnění bitového pole o velikosti 2 B podle šablony –

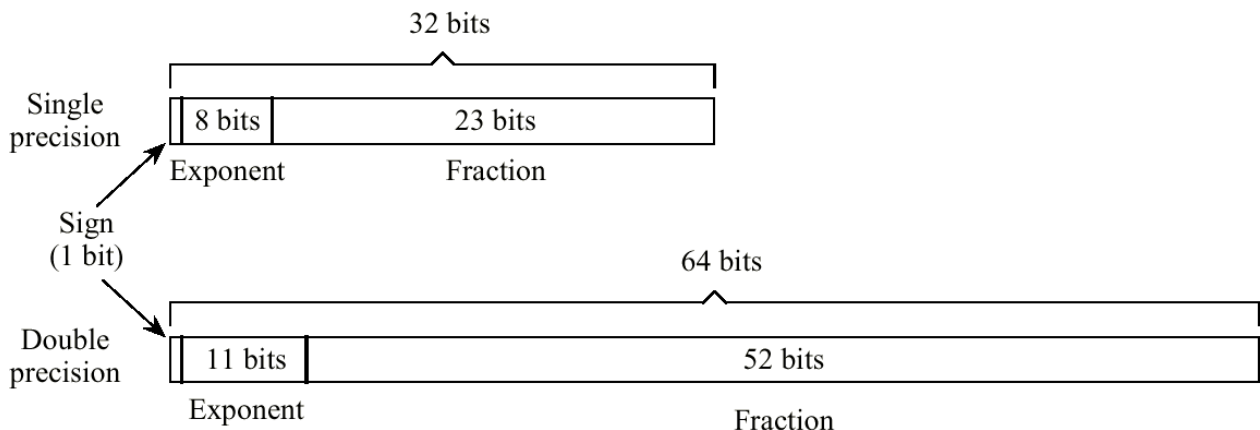
1. bit = 0, protože znaménko je +

Exponent (3), bit 2- 4 je zvětšen o 4 (3+4=7)

Normalizovaná část, bit 5-16, obsahuje čtyři číslice (3760)



Používané formáty jsou samozřejmě rozsáhlejší – viz příklad z [1] pro formáty IEEE- 754



4.5 Kódování textu

Text je typicky v paměti počítače, nebo v souboru uložen jako pole znaků. Historickým standardem je ASCII znaková kódová tabulka, podle které je každý znak textu namapován do jednoho B paměti, tedy do čísel v rozsahu 0 – 255 dekadicky. Původně se jedná o 7-bitový kód (uložený v 1 B), obsahující také řídicí znaky (původně pro ovládání dálnopisu). Rozšíření 8-bitové (druhá polovina tabulky) obsahuje speciální a grafické znaky. Obsah obou polovin tabulky je na obrázku.

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	Null	32	20	Space	64	40	@	96	60	`
1	01	Start of heading	33	21	!	65	41	A	97	61	a
2	02	Start of text	34	22	"	66	42	B	98	62	b
3	03	End of text	35	23	#	67	43	C	99	63	c
4	04	End of transmit	36	24	\$	68	44	D	100	64	d
5	05	Enquiry	37	25	%	69	45	E	101	65	e
6	06	Acknowledge	38	26	&	70	46	F	102	66	f
7	07	Audible bell	39	27	'	71	47	G	103	67	g
8	08	Backspace	40	28	(72	48	H	104	68	h
9	09	Horizontal tab	41	29)	73	49	I	105	69	i
10	0A	Line feed	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage return	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	47	2F	/	79	4F	O	111	6F	o
16	10	Data link escape	48	30	0	80	50	P	112	70	p
17	11	Device control 1	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	50	32	2	82	52	R	114	72	r
19	13	Device control 3	51	33	3	83	53	S	115	73	s
20	14	Device control 4	52	34	4	84	54	T	116	74	t
21	15	Neg. acknowledge	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	54	36	6	86	56	V	118	76	v
23	17	End trans. block	55	37	7	87	57	W	119	77	w
24	18	Cancel	56	38	8	88	58	X	120	78	x
25	19	End of medium	57	39	9	89	59	Y	121	79	y
26	1A	Substitution	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	59	3B	;	91	5B	[123	7B	{
28	1C	File separator	60	3C	<	92	5C	\	124	7C	
29	1D	Group separator	61	3D	=	93	5D]	125	7D	}
30	1E	Record separator	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	63	3F	?	95	5F	_	127	7F	□

Obrázek 3 Tabulka ASCII - 1

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
128	80	Ç	160	A0	á	192	C0	Ł	224	E0	α
129	81	ù	161	A1	í	193	C1	ł	225	E1	β
130	82	é	162	A2	ó	194	C2	Ť	226	E2	Γ
131	83	â	163	A3	ú	195	C3	ł	227	E3	π
132	84	ä	164	A4	ñ	196	C4	—	228	E4	Σ
133	85	à	165	A5	Ñ	197	C5	†	229	E5	σ
134	86	å	166	A6	*	198	C6	‡	230	E6	μ
135	87	ç	167	A7	°	199	C7	‡	231	E7	τ
136	88	è	168	A8	ç	200	C8	Ł	232	E8	Φ
137	89	ë	169	A9	ŕ	201	C9	Ŧ	233	E9	Θ
138	8A	è	170	AA	ŕ	202	CA	Ł	234	EA	Ω
139	8B	ì	171	AB	½	203	CB	Ŧ	235	EB	Θ
140	8C	î	172	AC	¼	204	CC	‡	236	EC	∞
141	8D	ï	173	AD	ı	205	CD	=	237	ED	∞
142	8E	Ë	174	AE	«	206	CE	‡	238	EE	ε
143	8F	ë	175	AF	»	207	CF	Ł	239	EF	∩
144	90	É	176	B0	⋯	208	DO	Ł	240	FO	≡
145	91	æ	177	B1	⋮	209	D1	Ŧ	241	F1	±
146	92	Æ	178	B2	■	210	D2	Ŧ	242	F2	≥
147	93	ô	179	B3		211	D3	Ł	243	F3	≤
148	94	ö	180	B4	†	212	D4	Ł	244	F4	[
149	95	ò	181	B5	†	213	D5	Ŧ	245	F5]
150	96	û	182	B6	‡	214	D6	Ŧ	246	F6	÷
151	97	ù	183	B7	Ŧ	215	D7	‡	247	F7	≈
152	98	ÿ	184	B8	Ŧ	216	D8	‡	248	F8	°
153	99	ÿ	185	B9	‡	217	D9	Ŧ	249	F9	•
154	9A	Û	186	BA	‡	218	DA	Ŧ	250	FA	·
155	9B	◊	187	BB	Ŧ	219	DB	■	251	FB	√
156	9C	£	188	BC	Ł	220	DC	■	252	FC	²
157	9D	¥	189	BD	Ł	221	DD	■	253	FD	z
158	9E	₰	190	BE	Ł	222	DE	■	254	FE	■
159	9F	f	191	BF	Ŧ	223	DF	■	255	FF	□

Obrázek 4 Tabulka ASCII – 2

Dalším kódem je EBCDIC , který je již 8-bitový.

Přehled prakticky používanějších kódů v českém prostředí:

CP852	osmibitové kódování češtiny v systému MS-DOS
ISO 8859-2	osmibitové kódování češtiny v UNIXových systémech
Windows-1250	osmibitové kódování češtiny používané v systémech Microsoft Windows pro reprezentaci textů ve středoevropských jazycích, používajících latinku
Kód Kamenických	osmibitové kódování češtiny částečně kompatibilní s CP437 (zachovává semigrafické znaky)

Unicode	současná celosvětová znaková sada, používaná v současných OS
---------	--

UNICODE

<http://www.unicode.org/>

Unicode je původně šestnáctibitová tabulka znaků všech existujících abeced, později rozšířená na 31 bitů - jeden znak zobrazen dvěma nebo čtyřmi bajty. Standard definuje i název znaku, převodní tabulky malá-velká písmena atp. Nevýhody: násobná délka textu, větší znaková sada.

UCS-2

(Universal coded Character Set)

Je základní způsob zápisu Unicode znaků, 2 bajty na znak.

UTF-8

(Universal Transformation Format)

Nejpoužívanější zobrazení Unicode znaků.

Pokud je znak ve standardním ASCII-7, zobrazí se beze změny v 1 bajtu.

Tzn. nejvyšší bit bajtu je roven nule.

Pokud znak není v ASCII, je zapsán dvěma až šesti bajty:

- 1. bajt: počet jedniček zleva vyjadřuje délku sekvence, nula je oddělovač.
- další bajty: v nejvyšších dvou bitech vždy 10

Pro českou abecedu stačí pro znaky bez diakritiky jeden byte a pro znaky s diakritikou dva byty.

Příklad: 'Příliš'

```
50  c5  99  c3  ad  6c  69  c5  a1
```

```
-----
P      ř      í      l      i      š
```

```
ř:  1100 0101  1001 1001      'ř' v unicode U+0159 (hexa)
    ---      --
```

UTF-16 a UTF-32

rozšíření základní šířky z 8 na 16 a 32 bitů.

UTF-7

pro sedmibitový přenos e-mailem (jako Base64)

4.6 Kontrolní kódy

Přímý dvojkový kód není zabezpečen. Pokud je jeden bit přečten chybně, je nová (chybná) hodnota stejně pravděpodobná jako původní (správná). Z důvodu zajištění správnosti přenosu dat vznikly kódy, které s příslušnou pravděpodobností zajišťují kontrolu přenášených informací.

4.6.1 Grayův kód

U číslicově řízených strojů nebo kreslicích zařízení se používají snímače lineárního posunu či otáčení hřídele. Posunem či pootočením vznikají ve snímači, který pracuje na optickém,

elektromagnetickém, mechanickém apod., impulsy, ty se načítají do registru, který ukazuje aktuální polohu snímače k číslicovému zobrazení polohy se používá Grayův kód. Kód má tu vlastnost, že při přechodu mezi dvěma sousedními hodnotami se změní vždy jen jeden bit.

Tab.4.3. Grayův kód

číslice	Hexadeci- mální kód	Grayův kód	číslice	Hexadeci- mální kód	Grayův kód
0	0 0 0 0	0 0 0 0	8	1 0 0 0	1 1 0 0
1	0 0 0 1	0 0 0 1	9	1 0 0 1	1 1 0 1
2	0 0 1 0	0 0 1 1	10	1 0 1 0	1 1 1 1
3	0 0 1 1	0 0 1 0	11	1 0 1 1	1 1 1 0
4	0 1 0 0	0 1 1 0	12	1 1 0 0	1 0 1 0
5	0 1 0 1	0 1 1 1	13	1 1 0 1	1 0 1 1
6	0 1 1 0	0 1 0 1	14	1 1 1 0	1 0 0 1
7	0 1 1 1	0 1 0 0	15	1 1 1 1	1 0 0 0

4.6.2 Kódy p z n

Kódy p z n mají kódové slovo s délkou n bitů, z nichž p bitů jsou jedničky a je jich konstantní počet, ostatní bity jsou nuly. Tyto kódy jsou kontrolní, protože je v nich možno detekovat jednu chybu.

K zobrazení desítkových číslic se u některých číslicových systému používá kód 2 z 5, detekce chyb je založena na stejném počtu jedniček (dvě) v každém kódovém slově. V tabulce 4.4. jsou uvedeny dva typy kódů 2 z 5, zvané kód 0,1,2,4,7 a kód 0,1,2,4,8.

Tab. 4.4. Kód 2 z 5

Desítková hodnota	Kód 2 z 5	Kód 2 z 5
	7 4 2 1 0	8 4 2 1 0
0	1 1 0 0 0	1 1 0 0 0
1	0 0 0 1 1	0 0 0 1 1
2	0 0 1 0 1	0 0 1 0 1
3	0 0 1 1 0	0 0 1 1 0
4	0 1 0 0 1	0 1 0 0 1
5	0 1 0 1 0	0 1 0 1 0
6	0 1 1 0 0	0 1 1 0 0
7	1 0 0 0 1	1 0 1 0 0
8	1 0 0 1 0	1 0 0 0 1
9	1 0 1 0 0	1 0 0 1 0

4.6.3 Kontrola paritou

Nejpoužívanějším způsobem kontroly správnosti přenášených dat po sběrnici nebo při záznamu a čtení z paměti je kontrola paritou. Při této kontrole se používá detekční kód vytvořený tak, že se k bitům zobrazujícím data připojí další, paritní bit. Když počet jedniček v takto vzniklém kódu je sudý, jde o sudou paritu. Při lichém počtu jedniček jde o lichou paritu. Paritní bit p , označovaný také jako příčná parita (VRC), umožňuje detekovat v každém kódovém slově lichý počet chyb.

Paritní bit sudé parity se vypočte podle vztahu:

$p = b_1 \oplus b_2 \oplus b_3 \oplus b_4 \dots$ a vytváří se v logickém obvodu zvaném generátor parity.

Při přenosu bloku slov je možno na konci bloku vytvořit další kontrolní znak, znak podélné parity (LRC). V LRC je hodnota každého bitu vypočtena jako parita všech bitů v příslušném sloupci bloku.

Kontrola příčnou a podélnou paritou umožňuje lokalizaci a opravu jedné chyby v bloku a detekci všech dvojic a trojic chyb. Libovolná chyba vyskytující se v tab. 4.5. je lokalizována v průsečíku zjištěných chybných bitů příčné a podélné parity. Může být opravena invertováním chybného bitu (Tab.4.6.).

Tabulka 4.5.

číslice	b_4	b_3	b_2	b_1	P
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	1
3	0	0	1	1	0
4	0	1	0	0	1
5	0	1	0	1	0
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	0
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	0
	0	0	0	0	0

bity podélné parity LRC

Tabulka 4.6.

číslice	b_4	b_3	b_2	b_1	P
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	1
3	0	0	1	1	0
4	0	1	0	0	1
5	0	1	0	1	0
6	0	1	0	0	0
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	0
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	0
	0	0	0	0	0

Příklad chyby v bloku dat je v tabulce 4.2. V řádku odpovídajícímu číslici 6 byla zjištěna chyba příčné parity VRC. Ve sloupci zobrazující hodnotu b_2 je lokalizována chyba podélné parity LRC. Chybný je druhý bit čísla 6, opravu provedeme negací tohoto bitu.

4.7 Samoopravný kód

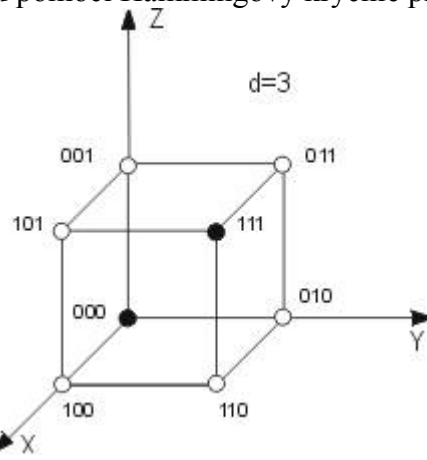
Místo zabezpečení dat jedním paritním bitem se používá rozšířený Hammingův kód obsahující více kontrolních paritních bitů. Kód obsahuje informační a kontrolní bity. Pro různý počet informačních bitů obsahuje kód jiný počet bitů kontrolních, viz tabulka 4.7. Kód je samoopravný, protože umožňuje najít a opravit jednu chybu a detekovat dvojnásobnou chybu.

Tab.4.7.

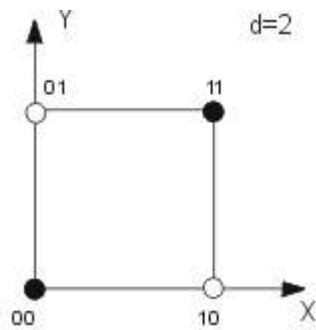
Počet informačních bitů	Počet kontrolních bitů
4	3
8	5
16	6
32	7
64	8

Pro posouzení samoopravných možností kódu je důležitý pojem kódová (Hammingova) vzdálenost. Kódová vzdálenost d představuje počet bitů, v nichž se liší dvě sousední platné kódové kombinace.

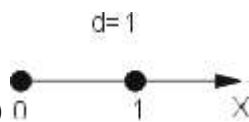
Příklady znázornění d pomocí Hammingovy krychle pro trojmístný kód:



- trojrozměrná (xyz)



- dvojrozměrná (xy)



- jednorozměrná (x)

Platí vztahy:

Detekce k chyb: $d \geq k+1$

Oprava k chyb: $d \geq 2k+1$

d	detekce	oprava
1	0	0
2	1	0
3	2	1
4	3	1
5	4	2

Rozšířený Hammingův kód se vytvoří tak, že se např. k informačním bitům b_1, b_2, b_3, b_4 v kódu BCD připojí kontrolní paritní bity p_1, p_2, p_3 . Kontrolní bity se vypočítají podle rovnic

$$p_1 = b_4 \oplus b_3 \oplus b_1$$

$$p_2 = b_4 \oplus b_2 \oplus b_1$$

$$p_3 = b_3 \oplus b_2 \oplus b_1$$

Čtyři informační a tři paritní bity se seřadí do sedmibitového slova v následujícím pořadí

mnohočlenu	2	3	4	5	6	7
	p_1	p_2	b_4	p_3	b_3	b_2 b_1

Číslo v takto sestaveném kódu se uloží do paměti nebo pošle do jiného zařízení a po novém načtení dat z paměti nebo na straně příjemce informace se znovu vypočítají nové paritní bity. Pokud mají hodnotu 0, jsou data správná. Pokud v jednom bitu došlo k chybě, tato pozice se z nově vypočtených paritních bitů určí a chybný bit opraví.

Cvičení

2. Příklad: Číslice 6 má kódové slovo 1100110 . Předpokládejme, že v něm vznikla chyba na páté pozici a kódové slovo se změní na 1100010 .

Řešení

2. Chyba se lokalizuje takto: Vypočtou se parity podle rovnic

$$p'_3 = p_3 \oplus b_3 \oplus b_2 \oplus b_1 = 1$$

$$p'_2 = p_2 \oplus b_4 \oplus b_2 \oplus b_1 = 0$$

$$p'_1 = p_1 \oplus b_4 \oplus b_3 \oplus b_1 = 1$$

Výsledek výpočtu 101 je příznak chyby zvaný syndrom, který lokalizuje chybu na místě $101 = 5$. Tuto chybu je možno opravit invertováním bitu b_3 .

4.8 Detekce chyb při sériovém záznamu dat

Pro detekci chyb u diskových a páskových pamětí a u dálkového přenosu dat se používá systematický cyklický kód. Vyznačuje se tím, že z kódového slova vznikne kruhovým (cyklickým) posuvem jiné slovo cyklického kódu. Kódové slovo můžeme formálně vyjádřit mnohočlenem $M_{(x)}$, jehož koeficienty se rovnají hodnotám číslic, tj. nulám a jedničkám v kódovém slově.

Příklad: Kódové slovo 1010 lze vyjádřit mnohočlenem třetího stupně, který má tvar

$$M_{(x)} = x^3 + 0 + x^1 + 0 = x^3 + x$$

Mocniny proměnné x jsou při zápisu mnohočlenu pouze pomocné – slouží pro určení polohy jednotlivých koeficientů, tj. umístění jedniček v kódovém slově. Kódování dat v cyklickém kódu se provádí v posuvném registru, v němž je na vstupy některých stupňů zavedena zpětná vazba z výstupu registru přes členy nonekvivalence.

Vlastnosti cyklického kódu určuje generující mnohočlen $G_{(x)}$. Koeficienty generujícího mnohočlenu $G_{(x)}$ totiž určují, na které stupně posuvného registru se má nebo nemá přivést zpětná vazba z výstupu posuvného registru podle toho, zda koeficientem je jednička nebo nula.

Lze dokázat, že v posuvném registru probíhá na kódovém slově výpočet zbytku $R_{(x)}$ po dělení vstupního mnohočlenu $M_{(x)}$ generujícím mnohočlenem $G_{(x)}$ podle vzorce:

$$x^{n-k} \cdot M_{(x)} / G_{(x)} = Q_{(x)} + R_{(x)} / G_{(x)}$$

, kde $n-k$ určuje počet bitů zabezpečujících zprávu $M_{(x)}$

Když se zbytek po dělení $R_{(x)}$ připojí za mnohočlen vstupního kódového slova, vznikne nové kódové slovo $A_{(x)} = x^{n-k} \cdot M_{(x)} + R_{(x)}$. Zbytek $R_{(x)}$ po dělení mnohočlenu se označuje jako kontrolní znak **CRC** a připojuje se k původnímu kódovému slovu $M_{(x)}$.

Nové kódové slovo $A_{(x)}$ je bezzbytku dělitelné původním generujícím mnohočlenem $G_{(x)}$. Proto lze správnost kódového slova $A_{(x)}$ kontrolovat tak, že se dělí mnohočlenem $G_{(x)}$ ve stejném posuvném registru. Pokud nastala v kódovém slově při přenosu nebo záznamu chyba, je hodnota zbytku po dělení $R_{(x)}$ nenulová. Když v kódovém slově není chyba, je $R_{(x)} = 0$.

Systematický cyklický kód pro čtyři bity generovaný mnohočlenem $G_{(x)} = x^3 + x + 1$ je v tabulce 4.8.

Tab. 4.8.

Data	Kontrolní znak	Data	Kontrolní znak
0000	000	1000	100
0001	011	1001	110
0010	110	1010	011
0011	101	1011	000
0100	111	1100	010
0101	100	1101	001
0110	001	1110	100
0111	010	1111	111

Cvičení

3. Příklad: Ověřte CRC na jednom řádku z tab. 4.8 – např. 1010.

Řešení

3. Pro vstupní kódové slovo **1010** z tabulky (sloupec data) je odpovídající mnohočlen $M_{(x)} = x^3 + x$. $R_{(x)}$ určíme dělením modulo 2 :

$$x^{n-k} \cdot M_{(x)} : G_{(x)}$$

Po dosazení a dělení platí :

$$x^3(x^3 + x) : x^3 + x + 1 =$$

$$(x^6 + x^4) : (x^3 + x + 1) = x^3 + 1 \quad \text{a}$$

$$R_{(x)} = x + 1 = 011 \text{ je kontrolní znak. Výsledné slovo je } 1010011.$$

Ilustrace dělení mnohočlenů modulo 2:

x^6	x^5	x^4	x^3	x^2	x^1	x^0		x^3	x^2	x^1	x^0		x^3	x^2	x^1	x^0
1	0	1	0	0	0	0	:	1	0	1	1	=	1	0	0	1
-(1 0 1 1)																
			1	0	0	0										
			-(1 0 1 1)													
					1	1										

V posledním řádku je hledaný zbytek: 1 1 , což odpovídá $R_{(x)} = x + 1$

Kontrolní znak se vytváří současně se záznamem dat na magnetický disk nebo při přenosu dat po telekomunikačním vedení a zapisuje se za blok dat. Při čtení dat z paměti nebo při příjmu dat z vedení se kontrolní znak CRC znovu generuje a porovnává se s původním kontrolním znakem.

Úkoly k textu

6. Převed'te binární číslo 101100011001 do šestnáctkové soustavy.
7. Převed'te desítkové číslo 473 do osmičkové soustavy.
8. Proveďte sčítání a odčítání tří dvojic dvoumístných dekadických čísel v přímém, doplňkovém a BCD kódu.
9. Navrhněte kód pro číslice 0 - 9, jehož Hammingova vzdálenost je tři.
10. Rozšiřte předchozí kód o paritní bit se sudou paritou.
11. Rozšiřte BCD kód o paritní bit s lichou paritou, запиšte několik čísel se zabezpečením příčnou i podélnou paritou.
12. Použijte rozšířený Hammingův kód z příkladu v kapitole a demonstруйте odhalení nějaké chyby v číslici 3.
13. Zkontrolujte CRC pro data 0111 z tabulky 4.8

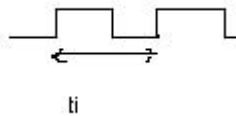
5 Sekvenční obvody

Studijní cíle: Seznámit se s principy, charakteristikou, základním zapojením a řízením sekvenčních obvodů, zvláště klopných obvodů jako paměti jednoho bitu informace.

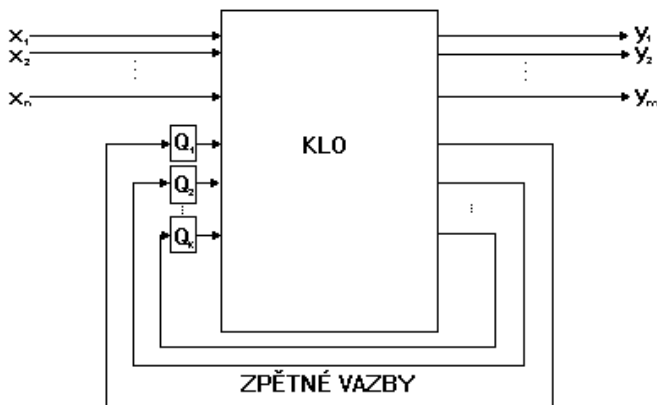
Klíčová slova: astabilní, monostabilní a bistabilní klopný obvod, RS, D, JK klopný obvod, synchronizace, hodinový signál, posuvný registr, sériová sčítačka, čítač, násobička.

Potřebný čas: 2 hodiny

Sekvenční obvody se vyznačují tím, že hodnota výstupu sekvenčního obvodu záleží nejenom na vstupu, ale i na vnitřních stavech obvodu, ve kterých jsou zachyceny informace o předešlých stavech sekvenčního obvodu. Proto je potřeba identifikovat a synchronizovat jednotlivé stavy v čase. Čas vystupuje nejčastěji ve formě speciálního periodického impulsního signálu - „hodin“ (clock), který diskrétně určuje okamžik pro synchronizaci obvodů.



Sekvenční obvod obsahuje zpětné vazby z výstupu dílčích obvodů na vstupy.



Obecně se jedná o obvod, nazývaný konečný automat. Pro popis se používá:

X . . . vstupní abeceda

Y . . . výstupní abeceda

S . . . stavová množina – reprezentuje historii automatu

f_1 . . . výstupní funkce $f_1 : X \times S \rightarrow Y, f_1(x_i, s_i) = y_i$

f_2 . . . přechodová funkce $f_2 : X \times S \rightarrow S, f_2(x_i, s_i) = s_{i+1}$

s_0 . . . iniciální stav

f_1, f_2 . . . kombinační funkce

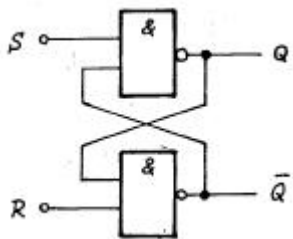
5.1 Klopné obvody

Klopné obvody tvoří nejjednodušší sekvenční obvody a tvoří typicky moduly dříve popsaného obecnějšího automatu. Obecně je můžeme rozdělit na astabilní, monostabilní, bistabilní. Každý z těchto typů hraje důležitou roli v číslicových obvodech. Astabilní obvody nemají žádný stabilní stav, periodicky se překlápí z jednoho do druhého. Používají se jako generátory impulsů. Monostabilní obvody mají trvale stabilní pouze jeden stav a používají se k typicky k prodloužení krátkých impulsů, nebo vztváření impulsů definované délky. Nás bude zajímat hlavně obvod bistabilní, jako základ pro realizaci paměti jednoho bitu v počítači i v ostatních číslicových obvodech. Bistabilní KO je v jednom ze stabilních stavů tak dlouho, dokud není vhodným vnějším řídicím signálem přeplopen do opačného stavu.

5.1.1 Klopný obvod RS

V praxi patří realizací mezi nejjednodušší obvody, tvoří základní paměťový člen, sloužící k zapamatování 1-bitové informace:

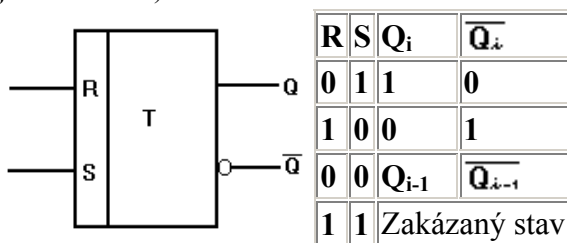
Jednoduchý R-S obvod – nepočítá se mezi klasické sekvenční obvody, ale je základem ostatních. Má pouze asynchronní vstupy- řízené úrovní na vstupech R, S.



S	R	Q
1	1	předchozí stav
0	1	1
1	0	0
0	0	nedefinovatelné

Jednotlivé vstupy jsou označeny: R.....RESET (nulování)
S.....SET (nastavení)

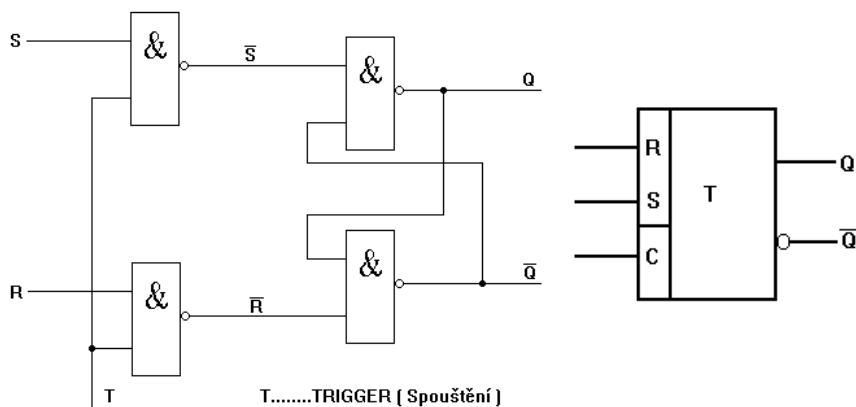
Prakticky se používá notace na následujícím obrázku (pravdivostní tabulka je pro obvod řízený „jedničkami“):



Pro RS řízený „nulami“ platí:

\overline{R}	\overline{S}	Q_i
1	0	1

0	1	0
1	1	Q_{i-1}
0	0	Zakázaný stav

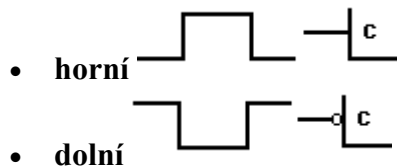


Jednoduchý asynchronní obvod se doplňuje synchronizačním vstupem C pro hodinový signál. Znázorněn je obvod RS řízený „jedničkami“ s časovou synchronizací.

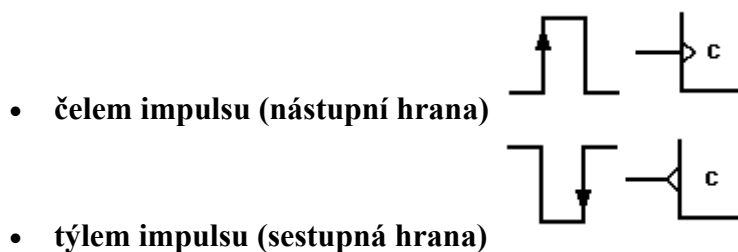
Otázka řízení KO:

Klopné obvody jsou řízené synchronně i asynchronně na vstupech datových i hodinových:

1. HLADINOU:



2. HRANOU (nebo dvěma hranami) HODINOVÉHO IMPULSU:



5.1.2 Klopný obvod D

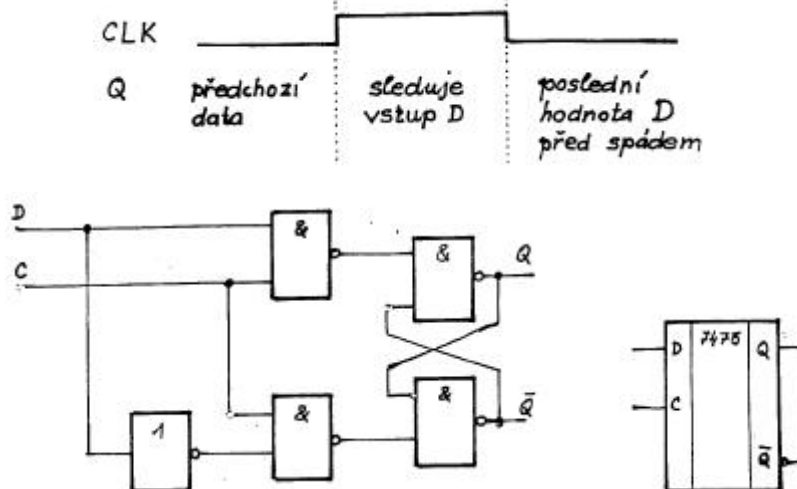
Používanou praktickou variantou (i ve formě integrovaných obvodů) je KO typu D. Výrobci málokdy rozlišují, ale existují tři základní typy obvodu D podle funkce a řízení:

typ Latch

klopný obvod D

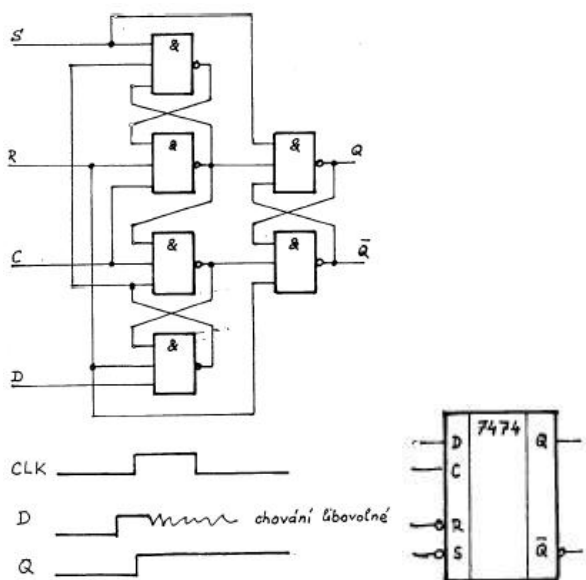
klopný obvod MASTER-SLAVE

a) Latch – asynchronní statické řízení – hladinou vstupního signálu



Např. MH 74 75 ... 4x v pouzdře

b) klopný obvod D – synchronní (Flip-Flop), s dynamickým řízením, ovládané hodinovým signálem (derivační - nástupnou hranou)

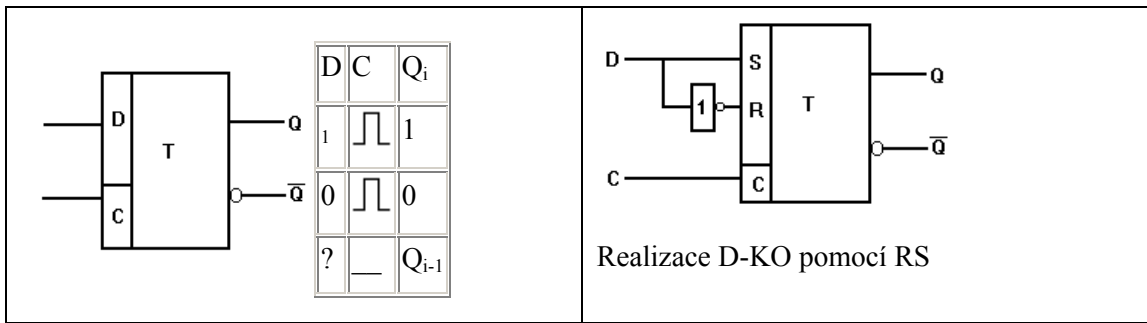


Např. MH 74 74

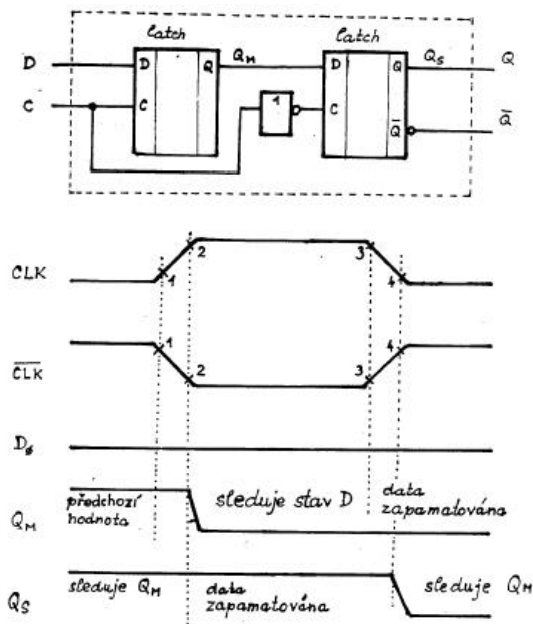
$R=S=1$ – synchronní režim, na Q se přenese D v okamžiku náběhu C

$R=0, S=1 \rightarrow Q=0$ kdykoliv

$R=1, S=0 \rightarrow Q=1$ kdykoliv



c) MASTER-SLAVE – dvouřázkové, ovládané nástupnou i sestupnou hranou hodinového impulsu.

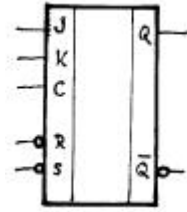


výsledný efekt – obvod přeneše D na Q spádem CLK

5.1.3 Klopný obvod JK

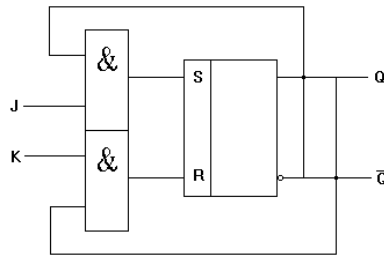
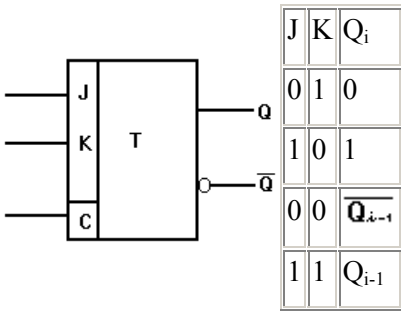
Rozšířený obvod Master-Slave . . . J-K obvod s pravdivostní tabulkou

J	K	S _{i+1}
0	0	S _i
1	0	1
0	1	0
1	1	\overline{S}_i

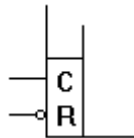


MH 74 72

Asynchronní vstupy R, S, pracují stejně jako u KO D.



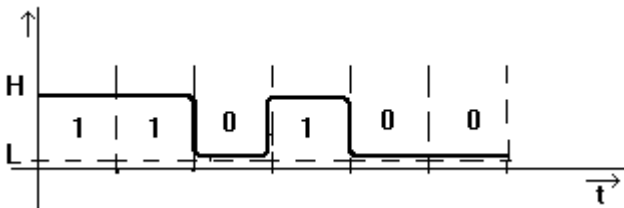
U většiny KO navíc:
R....reset, (Sset)



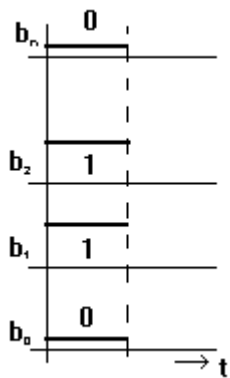
5.1.4 Přenos informací v systému

Rozeznáváme v principu dva základní způsoby přenosu dat v číslicových systémech:

- Sériový (data po jednom vodiči, bity informace se přenáší v čase za sebou):

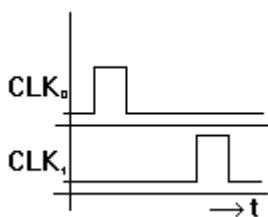
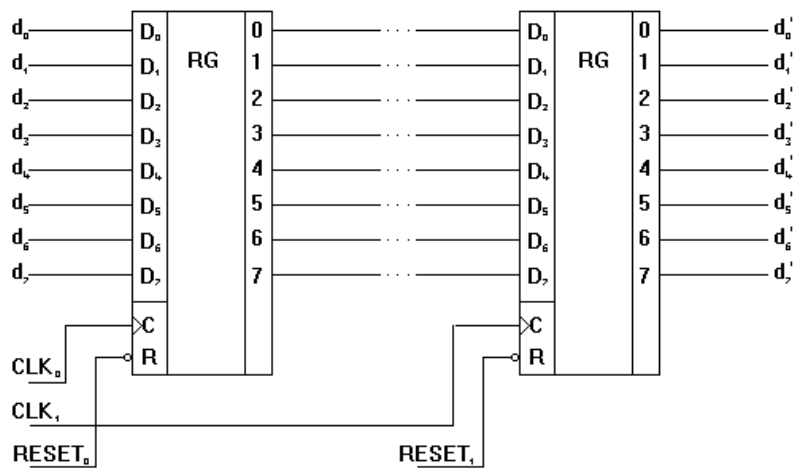


- Paralelní (data po souběžně po 2ⁿ vodičích, aktuální bity informace jsou v jednom okamžiku na příslušných vodičích):



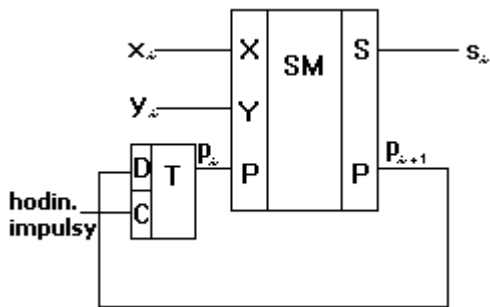
Uvnitř počítače probíhá paralelní přenos například pomocí sběrnic. Častou úlohou je transformace dat ze sériového typu přenosu do paralelního a naopak. Převod sériová informace na paralelní se provádí pomocí posuvného registru.

Využití paralelních registrů pro synchronní paralelní přenos

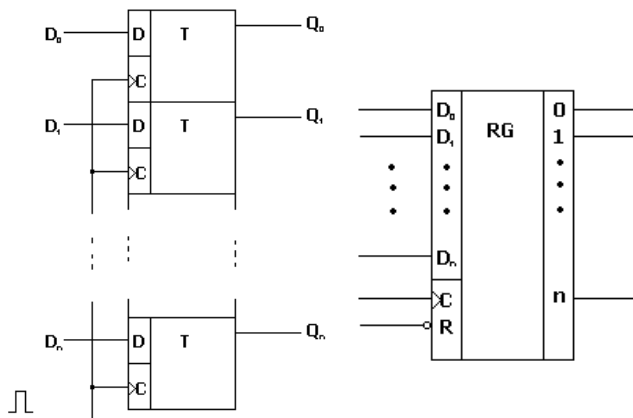


5.1.5 Typické sekvenční obvody v počítačích

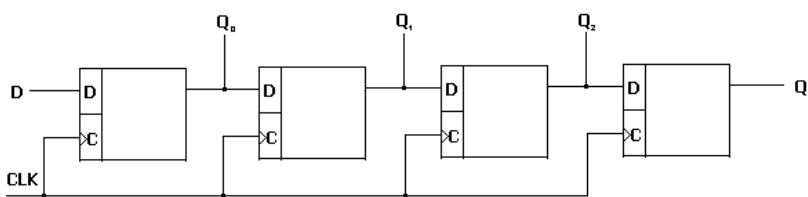
Sériová sčítačka umožňuje sčítat čísla, reprezentovaná a dodávaná na vstup po odpovídajících řádech v sériovém tvaru :



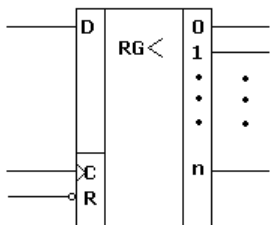
Paralelní registr = střádač umožňuje zachytit a zapamatovat si paralelní informaci (např. na sběrnici):



Sériový registr (posuvný registr) je obvod, který synchronně s hodinovými impulsy posouvá aktuální hodnoty na vstupech (D) na další KO v řadě a umožňuje realizovat transformaci sériových dat na paralelní :



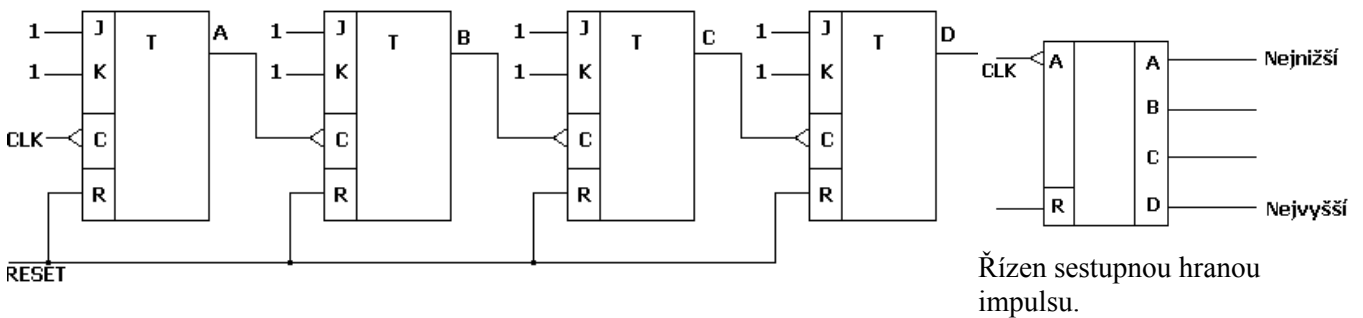
Jedním taktem signálu CLK se informace posune o jeden D-KO.



Čítače:

Čítače jsou obvody, které vhodným zapojením KO dělí (používají se čítače dělicí dvěma – výstup čítače je v binárním kódu a čítače se zkráceným cyklem pro dekadické dělení – výstup

čítače je v BCD kódu) a zároveň registrují počet načtených impulsů. Příklad zapojení čítače pro čtyři binární řády – čísla 0 až 15, výstup představuje stav jednotlivých KO:

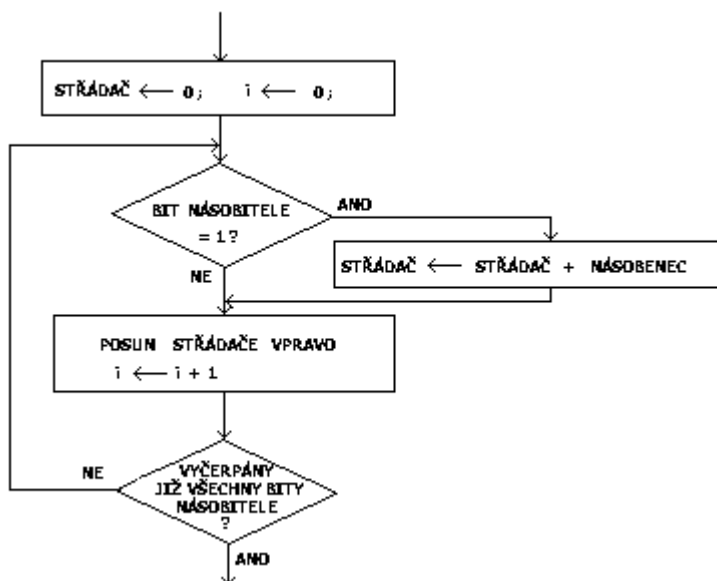
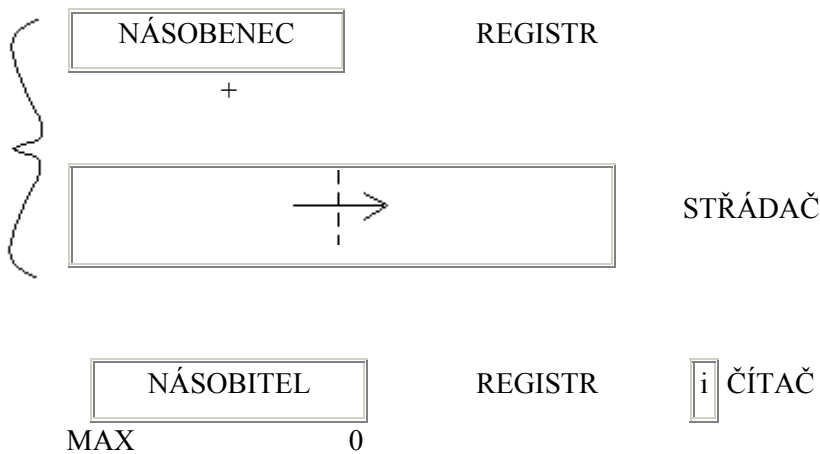


Dvojkový čítač 0...15, 0...15, ...

Takto koncipovaný obvod je možno získat v integrované podobě.

Násobičky

Sekvenční násobení (bez znaménka), používající posuvný registr a střadač, je schématicky znázorněno na následujícím schématu:



Obrázek 5 Algoritmus násobení v jednoduché násobičce

Kontrolní otázky

10. *Co rozlišuje kombinační a sekvenční obvody?*
11. *Jaké typy KO znáte?*
12. *Kolik stabilních stavů má KO RS,D a JK?*
13. *Které typy logických obvodů mají výstupní stav závislý pouze na stavu vstupů?*
14. *Jak se liší způsob řízení, nastavování a synchronizace u klopných obvodů (např. D)?*
15. *Které typické sekvenční obvody znáte a jakou mají funkci?*

6 Hardware počítače

6.1 Generace počítačů

6.1.1 Historie počítačů

- snaha zautomatizovat výpočet – počítačidla
– mechanické stroje
- snaha postavit zařízení pro různé výpočty bez nutnosti fyzické úpravy mezi výpočty
– možnosti elektroniky (relé - elektronka - tranzistor - integrovaný obvod)

Novodobá historie počítačů se datuje od roku 1938, kdy německý stavební inženýr Konrád Zuse sestrojil první mechanický počítač Z1. Pozdější model Z3 měl 2600 telefonních relétek a byl programovatelný. Dříve již v roce 1801 řídí Francouz Maria Jacquard tkalcovský stav pomocí děrné pásky a v roce 1833 vyvíjí Angličan Charles Babage tzv. "Analytical Engine" pracující s děrnými štítky. Na svou dobu vlastně příliš komplikovaný stroj. Počítač ENIAC (Electronic Numerical Integrator and Computer) sestrojený v r. 1946 na University of Pennsylvania byl prvním všeobecně použitelným počítačem na světě. Vážil 30 tun, zabíral 15 m² a měl příkon 174kW pro 17460 elektronek, 1500 relé, 70000 odporů a 10000 kondenzátorů. Pracoval na frekvenci 100 kHz a pro násobení potřeboval 2,8 mikrosekundy. Stroj nepracoval binárně, nýbrž dekadicky. Paměť obsahovala 20 střadačů, každý měl kapacitu 10 dekadických číslic. Každá číslice byla uložena uvnitř kruhu 10 elektronek, z nichž v jednom okamžiku mohla právě jedna reprezentovat "jedničku". Stroj se programoval ručně pomocí přepínačů a kabelových kontaktů. Byl chlazen vzduchem hnaným vrtulemi dvou leteckých motorů. V roce 1955 byl rozebrán. Ideu řízení počítače programem publikoval v roce 1945 pan John von Neumann. Další vývoj v oblasti počítačů byl velmi prudký. Byl uveden do provozu počítač IAS (1952), který se stal vzorem pro první velkosériově vyráběný počítač IBM 701. V Československu byl uveden do provozu první reléový počítač SAPO už v roce 1958. V roce 1948 byl poprvé předveden tranzistor; komerčně byl využit teprve v roce 1952. Integrovaný obvod s více tranzistory byl vyroben koncem 50.let. Složitost integrovaných obvodů se hodnotí podle počtu logických členů (hradel) realizovaných na jedné křemíkové základně.

Generace počítačů

Pokusíme-li se charakterizovat vývoj počítačů, můžeme pro zařazení do generací použít několik rysů. Nejvhodnější bude asi seřazení podle použitých součástek. Shrňme si jej v tabulce na obrázku. Počítače 5. generace jsou charakterizovány kvalitativně odlišnými prostředky komunikace s uživatelem, operační rychlostí a prvky umělé inteligence.

Generace	Rok	Použité součástky
0.	1940	relé
1.	1950	elektronky
2.	1958	tranzistory
3.	1964	integrované obvody
3.,5.	1972	LSI
4.	1981	VLSI

Kategorie počítačů

Další typ klasifikace počítačů vychází z výkonu počítačů a z jejich aplikačního nasazení. Definujme si tyto čtyři kategorie počítačů:

- mikropočítače (jedním z typů je i kategorie označená PC)
- minipočítače
- střediskové počítače
- superpočítače

Všechny tyto kategorie existují v jednom čase (současné době) vedle sebe. Mikropočítače jsou určeny pro každodenní používání jednomu uživateli. Někdy se tato kategorie nazývá také "osobní počítače". Mikropočítače existují díky mikroprocesorům, jejichž nízká cena dovoluje široké použití. Minipočítač sdílí většinou více uživatelů prostřednictvím více terminálů, nebo slouží jako komunikační uzel počítačové sítě apod. Střediskový počítač (Mainframe) svým vysokým výkonem slouží k vědeckotechnickým výpočtům a velkým počtem V/V zařízení pro zpracovávání hromadných dat. Typickými aplikacemi superpočítačů jsou vojenství, meteorologie, seismologie, naftový průmysl, atomová fyzika apod.

Pokud se na poslední období podíváme chronologicky, můžeme charakterizovat:

60. léta – převažují velké sálové počítače, aplikace řeší zpracování dat ve finančnictví, rozsáhlé vědeckotechnické výpočty

70. léta – nástup mikropočítačů, aplikace ve vědeckých laboratořích

80. léta – nástup stolních (osobních PC) počítačů a pracovních stanic s mikroprocesory, pro rozsáhlejší výpočty se používají severy a lokální sítě

90. léta – přináší Internet a webové technologie

Současné trendy se orientují na tři oblasti – přenosné a stolní počítače

- výkonné servery, paralelní a distribuované architektury, superpočítače
- řídicí jednotky v jednoúčelových zařízeních, komunikace, senzory

6.1.2 Koncepce Johna von Neumanna

V roce 1946 (kompletní konečná verze projektu je z roku 1952) John von Neumann a jeho kolegové z Princeton Institute for Advanced Studies vyprojektovali nový počítač postavený na těchto pravidlech (koncept uloženého programu):

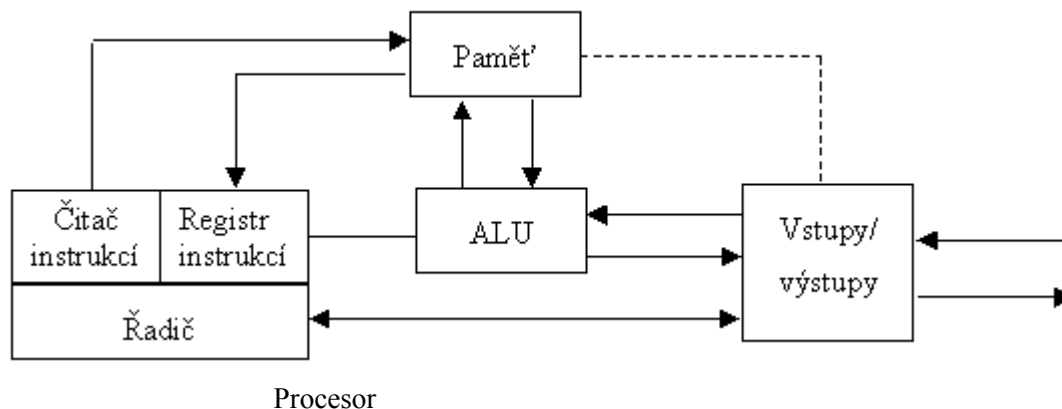
Počítač obsahuje lineárně organizovanou operační paměť, která je rozdělena na stejně velké buňky, procesor s ALU, řadičem a datovými registry, V/V zařízení.

Data a instrukce jsou vyjádřeny binárně, nejsou explicitně označeny (speciálně platí i pro datové typy)

Data a instrukce se uchovávají v **jedné** paměti na místech označených adresami.

*Podle výpisu
paměti nelze
poznat, jestli
jde o data, či
instrukce*

Předpis pro řešení úlohy je převeden do posloupnosti instrukcí. Řídicí jednotka interpretuje instrukce, uložené v paměti, a provádí je jednotlivě v pořadí, ve kterém jsou zapsány v paměti, pokud toto pořadí není změněno speciálními instrukcemi. Ke změně pořadí provádění instrukcí se používají instrukce podmíněného a nepodmíněného skoku. Programem řízené zpracování dat probíhá v počítači samočinně.



Obrázek 6 Von Neumannova koncepce

Struktura von Neumannova IAS počítače je na obrázku. V centru počítače je aritmeticko-logická jednotka (ALU). Činnost celého počítače řídí řadič. Řadič předává povely operační paměti, ALU a V/V zařízením a zpět od nich dostává stavová hlášení. Řadič čte z operační paměti instrukce, tyto dekoduje a převádí na posloupnost signálů. Data čte procesor z operační paměti nebo ze vstupního zařízení; rovněž tak je odkládá do paměti nebo zapisuje na výstupní zařízení. Von Neumannův počítač je významný proto, že až na malé výjimky je jeho schéma základem počítačů dodnes.

Dnešní počítače se od tohoto klasického schématu v některých podrobnostech mohou odlišovat. Je to především možnost pracovat i s více než jedním procesorem. Dále je obvyklé, že počítač zpracovává paralelně více programů zároveň - tzv. multitasking. To vede k efektivnímu využití strojového času. Program se do paměti nemusí zavést celý, ale je možné zavést pouze jeho část a ostatní části zavádět až v případě potřeby. Taktéž existují navíc vstupní / výstupní zařízení (I/O devices), která umožňují jak vstup, tak výstup dat (programu).

6.1.3 Harwardská koncepce

Na rozdíl od von Neumannovy koncepce předpokládá existenci dvou oddělených pamětí. V první paměti jsou uloženy instrukce (programy) a ve druhé jsou uložena proměnná data. Tato koncepce se využívá např. v jednoúčelových programovatelných automatech nebo kapesních kalkulátorech.

V současné době se technické vybavení počítačů rychle vyvíjí. Neustále jsou posouvány hranice možností jednotlivých základních komponent systému. Počítač je však neustále stavebnicí, kterou je možné různě sestavovat a doplňovat. Z jednotlivých komponent je tak možné sestavit vyvážený systém, který za vynaložené finanční prostředky splňuje požadavky v dané třídě aplikací, stejně tak je však možné sestavit systém v některém směru nevyvážený a neodpovídající tak výsledné ceně. Stejně tak je možné navrhnout systém s omezenými možnostmi upravy. Je třeba si také uvědomit, že technické vybavení může plnit požadované funkce pouze v součinnosti s vhodným operačním systémem.

6.1.4 Počítače PC

Kategorie počítačů PC je odvozena od příbuznosti a částečné nebo úplné kompatibility s řadou typů počítačů IBM PC (prvních z roku 1981) s příponou XP(8bitové systémy), později AT(16/32/64bitové systémy). Tím byla zakotvena základní koncepce a řada technických ukazatelů, které splňují i nejnovější verze PC. Firma IBM ztratila vliv na standardizaci PC po zavedení systémů PS/2 (1987). Software PC je v současnosti ovládán převážně firmou Microsoft, hardware pro PC dominantně určuje firma Intel. Společně vydávají dokumenty „PC xx Design Guides“, kde xx je příslušný rok:

(<http://www.microsoft.com/whdc/system/platform/pcdesign/desguide/pcguides.msp> dříve také <http://www.pcdesguide.org>)

6.2 Základní deska PC

Základní deska (Mainboard, Motherboard) ovlivňuje do značné míry kvalitu počítačové sestavy a to i vzhledem k dalšímu rozšiřování a modernizaci. Jedná se o několikavrstvou desku tištěného spoje, která prostřednictvím tzv. vnitřní sběrnice propojuje jednotlivé součásti počítače (procesor, paměť), rozšiřující sloty, t.j. konektory pro další přídavné karty, jako např. grafická karta, síťová karta a pod., konektory integrovaného rozhraní I/O, integrovaného rozhraní IDE pro pevné disky a disketové mechaniky a další. Deska obsahuje procesor, operační paměť, paměť CMOS (obsahuje informace o konfiguraci systému), BIOS uložený v paměti ROM (obvykle Flash-EEPROM), integrovaný řadič pružných a pevných disků (též pro připojení CD-ROM), sloty rozšiřovací (I/O) sběrnice (ISA, PCI, AGP, ...), integrované paralelní a sériové rozhraní, sběrnici USB, konektory pro připojení ostatních částí počítače (klávesnice, malého reproduktoru, tlačítka Reset, indikátoru provozu, ventilátoru procesoru apod.)

Někdy bývá (zejména u levnějších počítačů) na základní desce integrována i grafická karta, zvuková karta, faxmodem apod. – tzv. desky „All-In-One“, což má určité nevýhody v případě poruchy těchto zařízení.

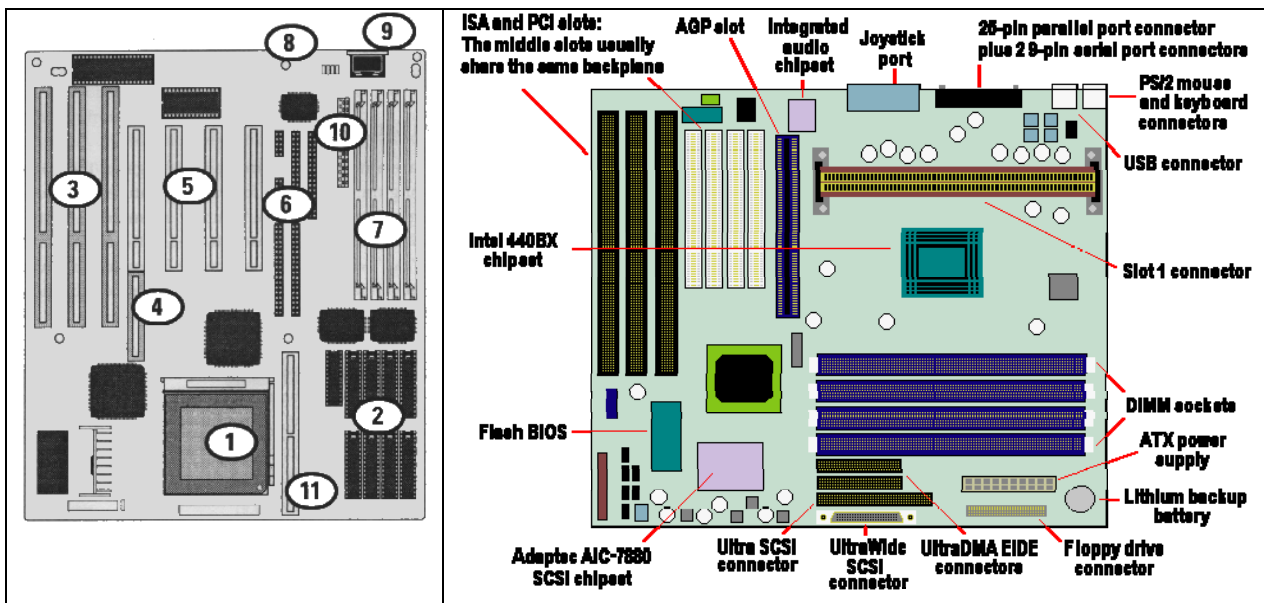
Podle provedení se základní desky dělí na AT (starší) a ATX (novější) – při stavbě počítače musíme vybrat tomu odpovídající základní jednotku („bednu počítače“), rozdíl spočívá především v umístění konektorů na desce a v napájecím zdroji.

Procesor i operační paměť jsou umístěny v patičkách, takže je můžeme v případě poruchy vyměnit nebo modernizovat.

Základní desky nemají vždy stejné rozměry. Klasické desky (tzv. AT) mají rozměr cca 228 x 305 mm. Nový formát (ATX) zavedla firma INTEL a využívá ji většina nových, kvalitních počítačů. Pro desku ATX je však třeba použít jiný typ počítačové skříně. Typická deska ATX a rozložení základních komponent je uvedena na obr. 3.1, kde:

- 1 procesor
- 2 paměť cache L2
- 3 rozšiřující slot ISA
- 4 Slot ASUS MediaBus
- 5 rozšiřující slot PCI
- 6 konektory V/V, FDD a IDE
- 7 paměťové banky SIMM
- 8 konektor PS/2 mouse
- 9 konektor klávesnice

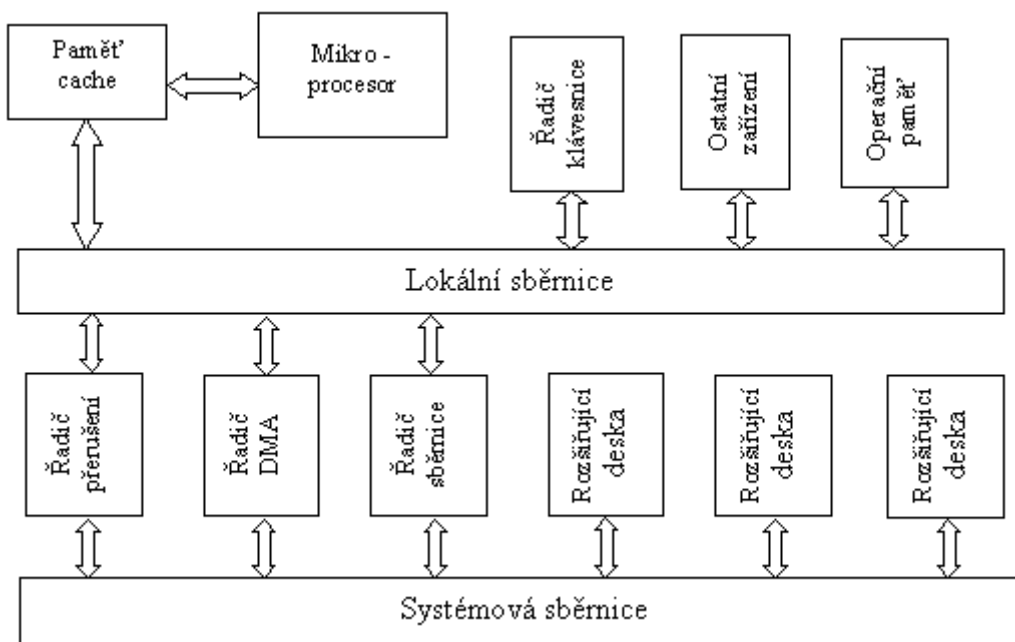
10 konektor napájení
 11 rozšiřující modul L2 cache



Obr.3.1 Typické základní desky z nedávné historie

Při výběru základní desky, příp. při jejím hodnocení je třeba věnovat pozornost především následujícím údajům :

- Počet a typ rozšiřujících slotů
- Typ patice procesoru a počet a typ použitelných procesorů
- Rychlost vnitřní sběrnice
- Integrace dalších obvodů
- Typ Chipsetu
- Typ BIOSu



Obr. 3.2 Blokové schéma počítače

6.2.1 Sběrnice (bus)

Pod pojmem sběrnice obecně rozumíme soustavu vodičů, která umožňuje přenos signálů mezi jednotlivými částmi počítače. Pomocí těchto vodičů mezi sebou jednotlivé části počítače komunikují a přenášejí data.

Zařízení jako jsou procesor, koprocessor, cache paměť, operační paměť, řadič cache paměti a operační paměti a některá další zařízení jsou propojena tzv. **procesorovou sběrnici (CPU bus)**. Rychlost sběrnice se stále zvyšuje, tak jak se vyvíjí propojené komponenty – 133 MHz, 800 MHz,

Rozlišujeme adresovou, datovou a řídicí část sběrnice:

Adresová (část) sběrnice

- slouží k výběru určité paměťové buňky, popř. určitého vstupně výstupního (I/O – Input/Output) zařízení (někdy se říká též periferní zařízení)
- většinou má šířku 16 až 40 bitů
- šířka adresové sběrnice určuje, s jakou velkou pamětí je procesor schopen přímo pracovat, např. při šířce 16 bitů lze adresovat 2^{16} bajtů, tj. 65 536 B = 64 KB, při šířce 40 bitů lze adresovat 2^{40} B = 1 099 511 627 780 B = 1 TB

Datová (část) sběrnice

- její šířka je velice důležitá pro výkon celého počítače, neboť určuje kolik bitů lze najednou přenést mezi procesorem a pamětí (resp. I/O zařízením) a kolik bitů lze najednou zpracovat
- většinou má šířku 8 až 64 bitů
- obvykle (ale ne vždy) má vnitřní datová sběrnice procesoru stejnou šířku jako datová sběrnice vnější
- podle šířky vnitřní datové sběrnice potom mluvíme o procesorech osmibitových, šestnáctibitových apod.

Řídicí (část) sběrnice

- jejím prostřednictvím předává procesor povely ostatním zařízením a přijímá stavové a řídicí informace z okolí
- počet řídicích signálů bývá pro různé procesory různý (10 až 100)

Osobní počítače musí být navrženy tak, aby bylo možné jejich snadné rozšiřování o další zařízení (zvukové karty, síťové karty, řadiče disků apod.). Takovéto rozšiřování je velmi často uskutečňováno pomocí tzv. **rozšiřujících sběrnice** počítače (častěji označované pouze jako sběrnice, nebo vstupně/ výstupní sběrnice), na kterou se jednotlivá zřízení zapojují. Tato rozšiřující sběrnice a zapojovaná zařízení musí tedy splňovat určitá pravidla. Takže ve výpočetní technice je pojem sběrnice také chápán jako standard, dohoda o tom, jak vyrobit zařízení (rozšiřující karty), která mohou pracovat ve standardním počítači.

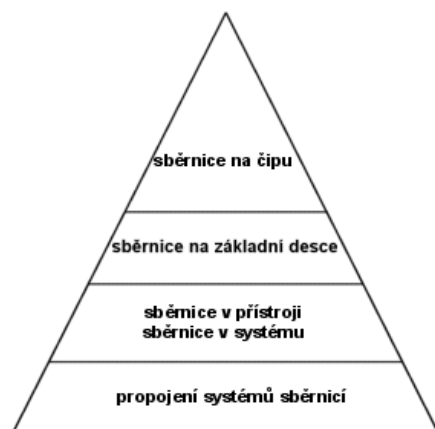
Podle způsobu práce a zapojení rozlišujeme několik základních typů sběrnice:

synchronní sběrnice: sběrnice pracující synchronně s procesorem počítače. Platnost údajů na sběrnici jednoznačně určuje hodinový signál. Tímto způsobem dnes pracuje převážná většina všech sběrnic.

pseudosynchronní sběrnice: dovoluje zpozdít přenos údajů o určitý počet hodinových period.

multimaster sběrnice: dovoluje tzv. **busmastering**, jedná se o sběrnici, která může být řízena několika zařízeními, nejen procesorem.

lokální sběrnice: spočívá ve vytvoření technické podpory toho, že se náročné operace s daty realizují rychlou procesorovou systémovou sběrnici. Tato systémová sběrnice se prodlouží a umožní se tak přístup na ni i ze zásuvných modulů dalších zařízení. O rozvoj lokálních sběrnic se nejvýrazněji zasloužili výrobci videokaret, pro něž byly dosavadní sběrnice pomalé. Nevýhodou lokálních sběrnic je o něco vyšší cena samotné základní desky s lokální sběrnici a také zařízení pro ni určených. Principiální obecná hierarchie sběrnic je na obrázku:



Mezi základní parametry každé sběrnice patří:

Parametr	Význam	Jednotka
Šířka přenosu	Počet bitů, které lze zároveň po sběrnici přenést	bit
Frekvence	Maximální frekvence, se kterou může sběrnice pracovat	Hz
Rychlost (propustnost)	Počet bytů přenesených za jednotku času	B/s

Počet a typ rozšiřujících slotů

Vlastní propojení s rozšiřujícími deskami zajišťuje systémová sběrnice. Ta je zakončena normovanými konektory, nazývanými sloty. Do slotů se zasunují rozšiřující desky, jimiž se připojují k počítači periferní zařízení. S vývojem počítačů se měnily i typy systémových sběrnic. Systémové sběrnice se liší především šířkou datové a adresové části sběrnice a rychlostí přenášených dat.

ISA (Industry Standard Architecture)

Jedná se o nejstarší typ systémové sběrnice, původně vyvinutý pro procesor 80286. Pro dnešní je již pomalá, přesto ji najdeme i na většině dnešních základních desek. Důvod je především v tom, že ne všechna zařízení vyžadují rychlou komunikaci a vystačí s levnou kartou s rozhraním ISA. Tyto karty se zpravidla konfiguruje manuálně za použití propojek na desce.

EISA (Extended ISA)

Jelikož bylo nutné s přechodem na vyšší typy procesorů zvýšit i výkon sběrnice, byl vyvinut nový typ sběrnice zpětně kompatibilní s ISA. Je tak možné desky ISA zasunout i do slotu EISA. Vzhledem k vysoké ceně desek s tímto rozhraním se však nerozšířila.

MCA (Microchannel)

Sběrnice MCA byla vyvinuta firmou IBM pro počítače s procesorem 80386 jako náhrada za ISA. Vzhledem k tomu, že nebyla kompatibilní s ISA, nerozšířila se a dnes ji zpravidla najdeme pouze u počítačů IBM (např. u řady PS/2).

VES AVL-Bus

VESA je jednoduchá a levná sběrnice, vyvinutá pro procesory třídy 80486. Její hlavní nevýhodou je, že její zatížitelnost klesá se stoupající frekvencí. Tak např. při frekvenci 33 MHz je možné použít 3 sloty, při frekvenci 40 MHz už jen jeden. U nových desek už zpravidla nebývá.

PCI (Peripheral Component Interconnect)

Sběrnice PCI je zatím posledním typem lokální sběrnice pro počítače PC. Jedná se o rychlou sběrnici vyrobenou firmou Intel pro počítače s procesory Pentium.

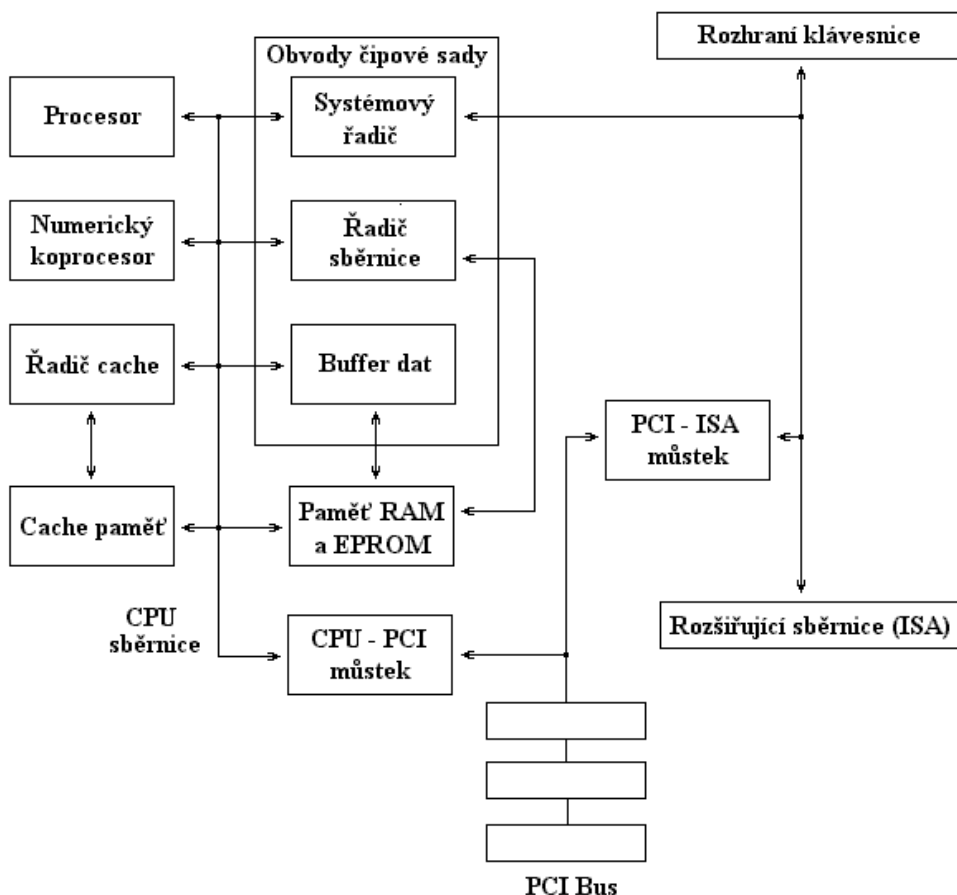
PCI již není klasickou lokální sběrnici jako VL bus, ale je k systémové sběrnici připojena přes tzv. **mezisběrnicový můstek**. Toto řešení s sebou nese následující výhody:

možnost použití sběrnice PCI i v jiných počítačích než jsou PC (např. MacIntosh, DEC)

můstek dovoluje provádět přizpůsobování napětových úrovní

Sběrnice PCI je první sběrnici s šířkou přenosu 64 bitů a využívá tak plně 64bitové datové sběrnice Pentia. Dovoluje však i přenos o šířce 32 bitů pro použití v počítačích s procesorem 80486. Frekvence, se kterou může tato sběrnice pracovat, je obvykle 33 MHz (66 MHz, 133 Mhz, ...), čímž je zajištěna propustnost sběrnice 132 MB/s (32 bitů) nebo 264 MB/s (64 bitů). Novější procesory (Itanium2) podporují šířku 128 bitů, rychlost 100 MHz, čtyři přenosy dat za cyklus a tedy přenosovou rychlost 6 400 MB/s. Dále podobně jako VL bus i PCI umožňuje burst režim, ale na rozdíl od VL busu není shora omezen počtem 4 taktů. Sběrnice PCI je nezávislá na sběrnici ISA, tzn. že nevyužívá žádných jejích signálů jako VL bus. Pro dodržení zpětné kompatibility jsou počítače se sběrnici PCI často osazovány i sběrnici ISA popř. EISA a jejími 16bitovými (ISA) nebo 32bitivými (EISA) sloty.

Podobně jako MCA a EISA a narozdíl od VL busu podporuje PCI busmastering, což dovoluje vyšší výkon počítače, protože přenosy řízené CPU vyžadují spoustu času. PCI dále podporuje standard Plug and Play vyvinutý výrobcí hardwaru v roce 1992, který dovoluje velmi snadnou konfiguraci desek pro PCI, popř. jejich automatickou konfiguraci bez zásahu uživatele. Sběrnice PCI je používána u novějších počítačů s procesorem 80486 a u počítačů s procesory Pentium a vyššími.



Blokové schéma základní desky se sběrnicí PCI

Jedná se o moderní rychlou sběrnici, vyvinutou firmou Intel pro procesory Pentium. Při instalaci rozšiřujících desek PCI dochází k automatické konfiguraci - tzv. systém PnP (Plug and Play). Běžně bývají na desce 3 až 4 sloty PCI, zbylé bývají s rozhraním ISA. Pro odlišení jsou konektory bílé barvy.

AGP (Accelerated Graphic Port) Rozhraní navržené firmou Intel pro optimalizaci práce grafických karet. Umožňuje velmi rychlou komunikaci mezi procesorem a grafickou kartou. Původně pracoval na frekvenci 266 MHz, následně byla frekvence zvyšována (AGP 2x - 533MHz, AGP 4x, AGP 8x). Sběrnice AGP byla navržena pro procesor Pentium II, určena výhradně pro připojení grafických karet. Tím, že umožňuje přímé propojení grafické karty s operační pamětí počítače, se odlehčí systémové sběrnici a současně se zvýší rychlost přenosu. Grafická karta tak může např. přímo přistupovat k texturám uloženým v operační paměti počítače a tím odpadá časově náročná příprava textur pro zobrazení, v provedení AGP 1x (přenosová rychlost 264 MB/s), AGP 2x (528 MB/s) nebo AGP 4x (1056 MB/s)

AMR a CNR

Jedná se o speciální konektory pro připojení modemů, zvukových a síťových karet. V současné době se již nepoužívají.

Typ sběrnice	ISA	EISA	MCA	VLB	PCI
Od procesoru	80286	80386	80386	80486	80486
Typická frekvence [MHz]	8,33	8,33	10-25	25-50	25-130, ...
Šířka datového toku	16	32	32	32	64

Adresovatelný prostor	16 MB	4 GB	4 GB	4GB	17 x 109 TB
Přenosová rychlost [MB/s]	4-5	33	40	40	132/264

Tab. 3.1 Srovnání typických sběrnic

Integrace dalších obvodů - na dražších základních deskách může být integrován především:

- zvuková karta
- grafická karta
- síťová karta
- řadič SCSI
- a další

Tato zařízení bývají zpravidla integrována v jednoduché podobě. Pokud použijeme kvalitnější přídatnou kartu, je třeba integrované zařízení zablokovat, (v programu SETUP nebo pomocí jumperu na základní desce).

Průvodce studiem

rozhraní v PC:

- *sběrnice (systémová sběrnice (CPU bus), ISA,... lokální sběrnice PCI, AGP, ...)*
- *rozhraní pro připojení součástí počítače*
 - *disků (IDE/ATA, SCSI, ...)*
 - *baterií (I2C, ...)*
 - ...
- *rozhraní (pro připojení dalších zařízení)*
 - *průmyslové standardy (GPIB, VME, ...)*
 - *sériová (RS232, USB, ...), paralelní a síťová*

Chipset

Termínem chipset nazýváme řadu integrovaných obvodů na základní desce speciálně zkonstruovaná pro práci s konkrétním typem procesoru. Obvody čipové sady realizují funkce, jako např.

- řízení činnosti paměti DRAM i SRAM
- řízení činnosti jednotlivých sběrnic
- komunikace mezi sběrnicemi

Starší čipové sady jsou založeny na architektuře North / South Bridge. Čip North Bridge propojuje rychlejší procesorovou sběrnici (533/400/266/133/100/66 MHz) s pomalejšími sběrnicemi. North Bridge je "horní polovina" čipové sady. Slouží jako styčný bod pro procesor, RAM paměť, PCI většinou i videoakcelerátor přes AGP sběrnici South Bridge propojuje PCI s ještě pomalejšími rozhraními. South Bridge je "dolní polovina" čipové sady. Integruje rozhraní jako IDE, FDD, klávesnici, paralelní a sériové porty, případně USB, FireWire atd. Mnohdy tento obvod slouží i jako brána pro ISA sběrnici, síťová či zvuková karta atd. Obecně řečeno rozšiřuje možnosti North Bridge. Příklad: South Bridge sady i815 je obvod ICH (i82801AA) South Bridge sady i815E je obvod ICH2 (i82801BA) South Bridge sady VIA Apollo KT133 je obvod VT82C686A

Novější čipové sady jsou založeny na rozbočovačích řadiče paměti a vstupů a výstupů, s rozhraním pracujícím na rychlosti 4x66MHz. Pro podporu činnosti procesoru Pentium byla firmou Intel vyvinuta čipová sada s označením IntelTriton (II, III), a používá se i řada dalších – Apollo, Orion, čipová sada 810, 815, 820, 840, 850, 860, ...

Pojem přerušeni

V době vykonávání úlohy musí být zajištěna i programová obsluha některých dalších událostí (stisk klávesy, pohyb myši, hrozící výpadek napájení, kritická chyba v paměti, . . .). I když by bylo možné testovat stav vybraných zařízení v rámci prováděné úlohy, je efektivnější, jestliže obsluhu důležitých událostí zajistí počítač na úrovni technického vybavení – pomocí přerušeni. Při přerušeni je dočasně přerušena probíhající úloha - sekvenční vykonávání instrukcí programu. Po obsluze se procesor vrací zpět na místo v programu úlohy, ve kterém byl přerušen.

Obsluha přerušeni se prakticky provádí poměrně často. Ilustrační popis procesů v okamžiku přerušeni u procesorů Intel:

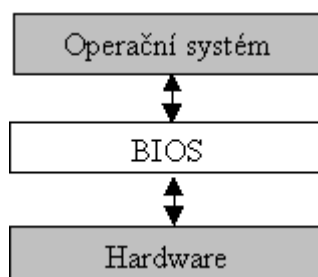
- Do řadiče přerušeni přichází požadavek o přerušeni, ten vyhodnotí jeho prioritu. Jestliže je přerušeni možné, je vyslán do procesoru signál požadavku o přerušeni.
- Mikroprocesor přijal signál požadavku přerušeni. Jestliže je přerušeni možné, po dokončení probíhající instrukce vyšle procesor signál potvrzení přerušeni.

Řadič přerušeni přijal signál povolení přerušeni. Vyšle na datovou sběrnici instrukci přerušeni INT číslo, ta zajistí, že procesor provede následující činnosti:

- do zásobníku se uloží registr příznaků F (po návratu se musí obnovit, určuje stav proc.)
- zakáže se další přerušeni
- do zásobníku se uloží obsahy CS a IP (místo, kde byla přerušovaná úloha)
- registry CS a IP se naplní adresou, přečtenou z tabulky vektorů přerušeni (to je tabulka na začátku paměti, v ní jsou za sebou uloženy celé adresy všech obsluh přerušeni, klíčem pro hledání v této tabulce je právě číslo přerušeni uvedené za instrukcí INT)
- Proběhne obsluha přerušeni (například načtení dat, hláška na obrazovku,...).
- Po obsluze je ze zásobníku obnoven obsah registrů IP, CS, F (procesor se vrátí k původní úloze a stavu).
- Obnovu těchto registrů zajistí instrukce IRET (která je na konci obsluhy přerušeni).

BIOS (Basic Input Output System)

BIOS je program, který zajišťuje vazbu mezi univerzálním operačním systémem a specifických hardwarem. Jen tak je možné zajistit, že tentýž operační systém pracuje bezchybně na počítači třídy 286 i Pentium. BIOS je umístěn v paměti typu ROM (viz kapitola paměti). Fyzicky je to pouzdro DIP, čili integrovaný obvod, zasunutý do patice základní desky. BIOS je možné v jistých mezích i konfigurovat. K tomu slouží program SETUP. Nejznámější tvůrci programů BIOS jsou firmy American Megatrens Incorporated - AMI, Award a Phoenix.



Obr. 3.3 Funkce BIOSu

IO operace

Pokud potřebujeme minimalizovat úplné přímé řízení několika IO operací procesorem, používá se:

- přerušeni - Periferie je obsluhována jen tehdy, požádá-li o to (má připravená data), ale je s ním spojena poměrně velká "systémová režie" a všechny operace stejně zůstanou na procesoru
- přímý přístup do paměti - odpadá jeden přenos dat, místo přenosu periferie - registr procesoru – paměť se provede pouze přenos periferie – paměť, organizovaný řídicím obvodem - řadičem **DMA**. "Programování" DMA definuje parametry přenosu – kolik, odkud, kam. Proces probíhá tak, že když je periferie připravena - řadič požádá o povolení DMA. Procesor dokončí strojový cyklus (mikroinstrukci), uvolní sběrnice a proběhne DMA. Řadič oznámí ukončení operace, uvolní sběrnice a pokračuje procesor
- kanál (periferní procesor)- kanálová architektura byla historicky zavedena u střediskových počítačů. Na systémové sběrnici je připojen procesor, paměť a řadič kanálů. Na řadič kanálů je přes další sběrnici připojeno několik řadičů periferních zařízení. Kanálová architektura je výhodná menší sadou instrukcí, procesor komunikuje pouze s řadičem kanálů a řadič kanálů může provádět větší množství operací bez procesoru (jednoduchá aritmetika s adresami ...)

6.2.2 Technologie Plug & Play (PnP)

Předešlé kapitoly popisovaly postup při instalaci nové karty do počítače. Je pravdou, že v některých případech může být tato instalace poměrně komplikovaná a správné nastavení jednotlivých zdrojů využívaných jednotlivými kartami může působit problémy. Pravděpodobnost výskytu takovýchto problémů je tím vyšší, čím větší je počet karet zapojených do počítače. Takže zapojení další karty do počítače, který již kromě standardních a nezbytných věcí obsahuje ještě např. SCSI rozhraní, síťovou kartu, zvukovou kartu, může způsobit nemalé problémy.

Nové trendy, se snaží problém řešit tím, že počítač by si sám provedl výše zmíněné konfigurace. V ideálním případě by tedy uživatel zasunul kartu do počítače a po jeho zapnutí by počítač sám:

- Rozpoznal zasunutou desku
- Zeptal se desky, které IRQ, DMA, I/O Adresy, RAM a ROM adresy potřebuje
- Zeptal se desky, které IRQ, DMA, I/O Adresy, RAM a ROM adresy může použít
- Nastavil prostředky tak, aby nedošlo k žádnému konfliktu s již existujícími deskami
- Vyhledal a nakonfiguroval potřebné ovladače, které by s nově nainstalovanou deskou spolupracovaly

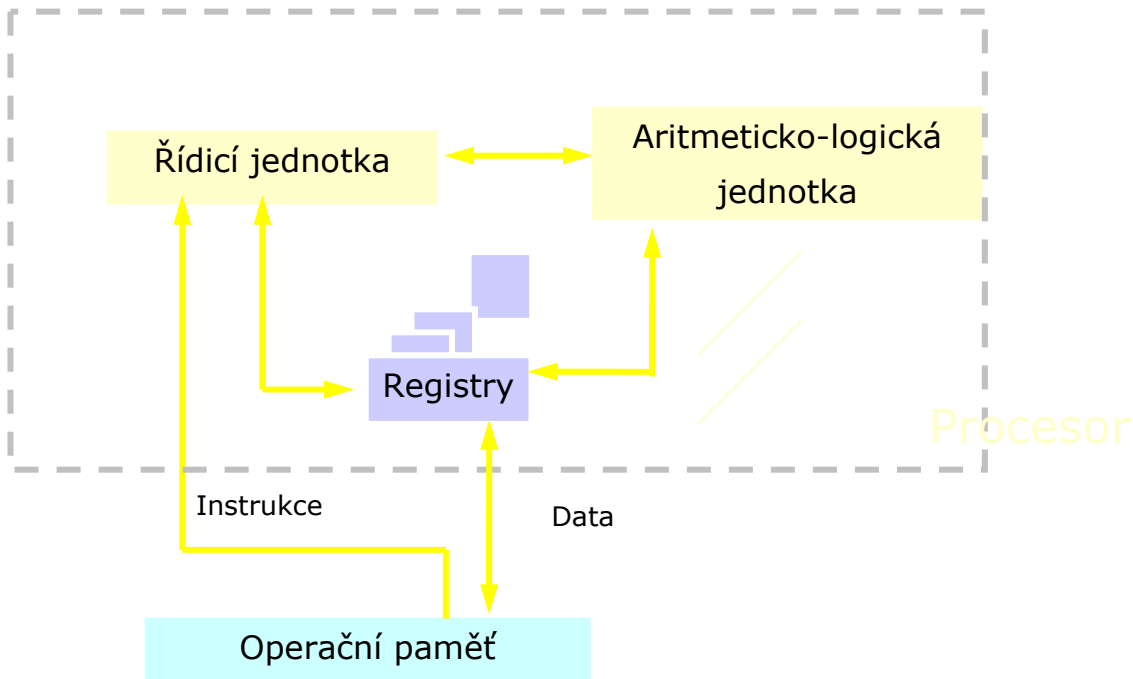
Pro zvládnutí tohoto problému navrhly firmy Intel, Microsoft a Compaq normu nazvanou Plug & Play. Myšlenka Plug & Play spočívá v tom, že výrobci přídatných karet přidávají ke svým kartám takové obvody, aby bylo možné automaticky desky nastavovat a vznášet dotazy na požadované zdroje. Operační systém pak může sám přímo konfigurovat desky a dotazovat se na ně, čímž by se redukovala nutnost otevírání počítače na minimum.

6.3 Procesor

Mikroprocesor (procesor) je integrovaný obvod, který je jádrem celého počítače. Je umístěn v patičce (slotu) na základní desce. Jeho funkce je taková, že vykonává (víceméně) sekvenčně jednotlivé instrukce programu, umístěného v paměti počítače. Seznam všech dostupných instrukcí procesoru se nazývá instrukční sada a je pevně zabudována do procesoru při jeho výrobě. Registry procesoru tvoří velmi rychlá zápisníková paměť. Registry rozlišujeme na:

- Registry univerzální (datové) - slouží k uchování operandů, mezivýsledků a výsledků operací podobně jako operační paměť (registr AX, BX, ...). z hlediska toho, že jsou umístěny přímo v procesoru, informace v nich uložené jsou přístupné nejrychleji ze všech možných typů úložišť

- Registry s pevně stanoveným významem - realizují určitou specifickou funkci a změnou obsahu těchto registrů lze ovlivňovat činnost procesoru



Rozdělení **instrukcí** procesoru:

přesunové (pro přesun dat mezi registry procesoru nebo mezi operační paměti a registry)

aritmetické (sčítání, odčítání, ...)

logické (log. součet, log. součin, rotace a posuvy, ...)

skoku (při rozhodování, ...)

vstupně výstupní (pro práci s periferními zařízeními, ...)

ostatní (řídící, ...)

Dle dostupné instrukční sady existují dvě základní koncepce procesorů:

CISC (Complete Instruction Set Computer) - Procesor obsahuje tzv. úplnou instrukční sadu.

RISC (Reduced Instruction Set Computer) - Procesor obsahuje tzv. redukovanou instrukční sadu. Tato koncepce vychází ze skutečnosti, že pro vykonání 80 % operací je zapotřebí cca 20 % instrukcí. Procesor tedy neobsahuje méně často se vyskytující instrukce, ty jsou nahrazeny programem s využitím základních instrukcí. Tím se procesor zjednoduší a je podstatně rychlejší. U počítačů PC jsou používány procesory CISC, i když moderní typy procesorů začínají využívat i určité prvky RISC a tím rozdíly postupně stírat.

Vnitřní taktovací frekvence procesoru

Tato frekvence se vytváří ve frekvenčním generátoru základní desky, je v určitých mezích nastavitelná přepnutím, taktuje činnost procesoru a tím určuje dobu jeho základního cyklu. Se zvyšující se frekvencí se procesor více zahřívá a je možné jej zničit. Proto při výměně procesoru je třeba dodržet předepsanou frekvenci. Současné procesory dosahují frekvence přes 3 GHz a ve výhledu jsou frekvence až 7 GHz.

Vnější frekvence

Vnější frekvence taktuje činnost periferních modulů a čipových sad (frekvence lokální sběrnice). Je vždy odvozena ze základní (vnitřní) frekvence a bývá zpravidla nižší. Vnější frekvence se pohybuje v rozmezí 50 až 133 MHz u starších a až 200 až 400MHz u novějších počítačů. Výsledná vnitřní taktovací frekvence procesoru je pak dána součinem vnější frekvence a násobitele, který lze u moderních základních desek nastavit softwarově pomocí programu SETUP, který slouží k nastavení základních parametrů počítače (BIOSu).

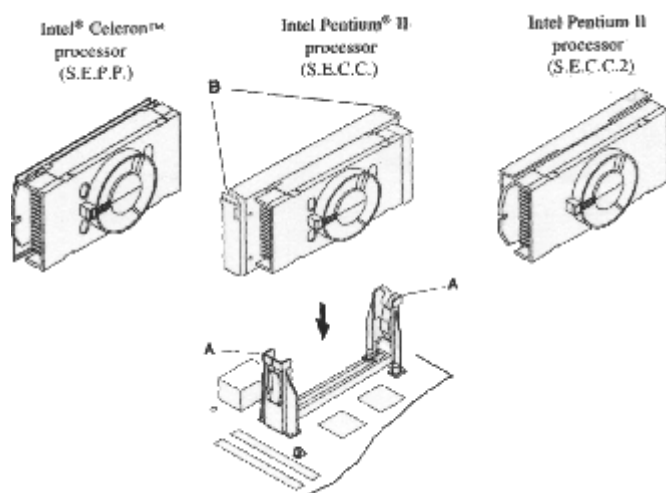
V tabulce je uveden příklad vnější a vnitřní frekvence spolu s příslušnými hodnotami násobitelů pro starší procesory Pentium. U současných základních desek je možné nastavit vnější frekvenci v širokém rozpětí (zpravidla od 66 MHz až 150 MHz) v poměrně malých krocích z důvodů naladění optimálního výkonu procesoru. Základní frekvence bývají 66, 100 a 133 MHz, hodnoty násobitele pak jsou téměř výhradně násobky čísla 0,5.

Vnitřní taktovací frekvence	Vnější frekvence	Násobitel
MHz	MHz	-
75	50	1,5
90	60	1,5
100	66	1,5
120	60	2
133	66	2
150	60	2,5
166	66	2,5
200	66	3
233	66	3,5

Tab. 2.1 Příklad vnitřní a vnější frekvence některých procesorů Pentium.

Socket - patice procesoru

Dřívější způsob připojení - pájením - na základní desku je nahrazen osazením procesoru do zásuvné patice, tzv. socketu. V současné době se používá několik typů patic. Kromě patice typu socket pro procesory s vývody ve formě pinů (samostatné "nožičky - drátky") se vyskytují rovněž patice ve formě tzv. slotů - např. Slot 1 pro procesory Intel Pentium II a III nebo Slot A pro procesory AMD Athon.



Obr. 2.1 - Ukázka montáže procesoru ...

Paměť cache

Paměť cache je rychlá vyrovnávací paměť pro ukládání dat při přesunu mezi různě rychlými částmi počítače. Rozeznáváme cache paměť:

L1 (First Level Cache - vyrovnávací paměť první úrovně) je integrována přímo do procesoru. Slouží pro dočasné ukládání dat ve směru ze sběrnice do procesoru. Funkce je založena na tom, že cache paměť je rychlejší než sběrnice. Do paměti cache se proto načte ve vhodném čase (fázi) práce procesoru větší množství dat, které procesor potřebuje a nemusí zbytečně čekat na pomalejší periferie.

L2 (Second Level Cache) pro zrychlení přesunů mezi mikroprocesorem a operační pamětí, hlavně v případě po špatném cachování L1- např. skok v instrukcích programu. Je zpravidla uložena na základní desce. U moderních typů procesorů výrobci přesouvají tuto paměť do pouzdra procesoru (Intel Pentium Pro) nebo přímo na čip procesoru (např. Athlon 64).

Patice	Podporované typy procesorů	Počet pinů
Socket 1	80486SX, DX, DX2, DX4, OverDrive	169
Socket 2	80486SX, DX, DX2, DX4, OverDrive, Pentium OverDrive 63 a 83	238
Socket 3	80486SX, DX, DX2, DX4, AMD 5x86, Cyrix 5x86, Pentium OverDrive 63 a 83	237
Socket 4	Pentium 60-66, Pentium OverDrive 120 a 133	273
Socket 5	Pentium 75-133, Pentium OverDrive 125-166, Pentium MMX OverDrive 125-166	320
Socket 6	Nepoužívala se	235
Socket 7	Pentium 75-200, Overdrive, MMX, MMX Overdrive, klony Pentia výrobců AMD, Cyrix, IDT atd. (6x86, K5, K6, 6x86MX, WinChip C6, K6-2, K6-III. Pro vyšší vnitřní frekvence se používala patice Super 7.	321
Socket 8	Pentium Pro	387
Socket 370	Pentium III, Celeron, Celeron A a procesory menších výrobců (VIA C3 apod.)	370
Socket 423	Krátce první Pentia 4 (jádro Willamette)	423
Socket 478	Pentium 4, Celeron	478

Socket A (Socket 462)	AMD Athlon, Athlon XP, Duron. Integrovaná procesorová sběrnice EV6.	462
Socket 940	AMD Opteron, Athlon 64 FX	940
Socket 754	AMD Athlon 64	754
Socket 603	Pentium II Xeon	603
Slot 1	Pentium II, Pentium Pro, Celeron, Pentium III,	242
Slot 2	Pentium II Xeon	330
Slot A	AMD Athlon K7, Athlon K75	242

Tab. 2.2 Přehled některých typů patič procesorů.

Mikroprocesory vyrábělo a vyrábí velké množství firem. Některé se specializují na procesory typu RISC, jiné na speciální procesory pro určité aplikace apod. Mezi nejvýznamnější výrobce mikroprocesorů pro PC v současné době patří firmy Intel, AMD, VIA Technologies a Transmeta.

6.4 Procesory Intel

Firma Intel je vedoucí firmou ve vývoji procesorů pro počítače PC. V roce 1971 představila firma Intel první čtyřbitový procesor Intel 4004, který obsahoval přibližně 2300 tranzistorů a byl taktován na frekvenci 108 kHz. V následujícím roce byl představen osmibitový nástupce, procesor 8008. V následujících letech se do výroby a vývoje procesorů zapojila řada dalších výrobců. Významného komerčního úspěchu dosáhl především procesor 8080. V roce 1978 firma Intel představuje první 16bitový procesor Intel 8086 a o rok později jeho levnější variantu Intel 8088. Procesor Intel 8088 použila firma IBM v roce 1981 k vytvoření prvního počítače třídy PC (Personal Computer), tedy osobních počítačů. Měl operační paměť 16 až 256 kB bez pevného disku a využíval již operační systém MS-DOS. O dva roky později byl doplněn i pevným diskem a počítač nesl označení PC/XT (eXTended). V roce 1984 uvedla firma IBM na trh počítač vybavený procesorem Intel 80286 (cca 134 000 tranzistorů, taktovací frekvence až 12 MHz) s větší operační pamětí, jejíž kapacita mohla být až 16 MB a označila tento model IBM PC/AT (Advanced Technology).

Procesory firmy Intel až do uvedení typu Intel Pentium byly označovány číselným kódem, který začínal vždy Intel 80. Od typu Intel 80286 se rozšířil neoficiální zvyk, označovat procesory Intel bez zmíněného počátečního kódu. Procesory ostatních výrobců byly uvozeny zpravidla dalším označením, např. Am386 (firma AMD), 386SLC (IBM) atd.

8088 - byl procesor použitý v počítačích PC XT. Obsahoval pouze 8 bitovou datovou sběrnici.

8086 - byl první 16 bitový procesor, s 16 bitovou datovou i adresovou sběrnici

80286 - vylepšený 16 bitový procesor, obsahoval 24 bitovou adresovou sběrnici. Byl základem počítačů PC AT. Od tohoto typu může procesoru pracovat ve dvou režimech:

- Reálný režim je nastaven po inicializaci procesoru, je slučitelný s procesorem 8086
- Chráněný režim se zapíná programově z reálného režimu, adresuje 16 MB reálné paměti a 1 GB virtuální paměti, poskytuje prostředky 4úrovňové ochrany, nelze se vrátit z chráněného režimu zpět do reálného

80386SX - první procesor s 32 bitovou vnitřní šířkou dat. Byl ale navržen pro montáž do desek 286 s bitovou datovou sběrnici.

80386 (80386DX) - první plně 32 bitový procesor. Přichází s koncepcí, kdy jednotlivé nezávislé programy mohou běžet ve vyhrazených oblastech paměti (tzv.

Protected Mode).

80486SX - je vylepšenou verzí procesoru 80386. Má integrovanou 8 kB paměť cache.

Všechny uvedené procesory mohly být na základní desce doplněny o matematický koprocesor, tedy specializovaný procesor na provádění matematických operací. Jejich označení bylo o 1 vyšší, než procesoru (např. 80387SX).

80486DX, 80486DX/2, 80486DX/4 - jsou 32 bitové procesory s integrovaným matematickým koprocesorem. Lišily se rozdílným poměrem vnitřní a vnější frekvence.

DX - shodná vnitřní a vnější frekvence

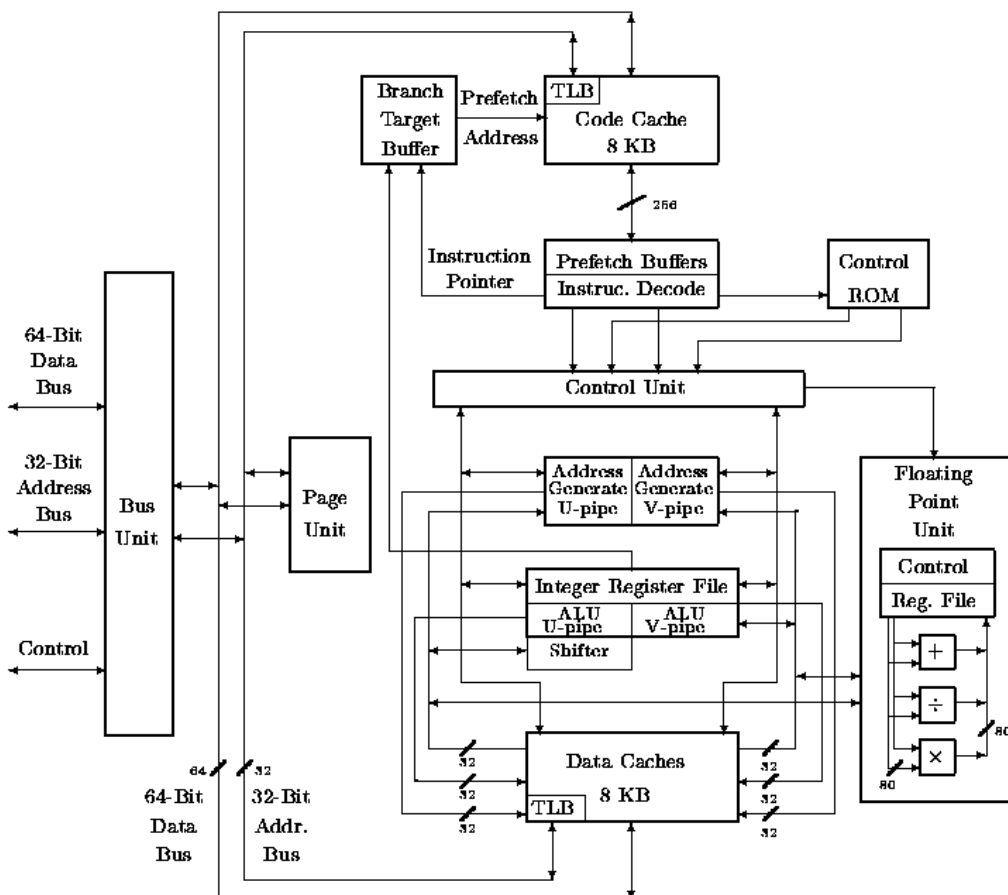
DX/2 - dvojnásobná vnitřní frekvence oproti vnější

DX/4 - trojnásobná vnitřní oproti vnější.

Pentium

V pořadí dalším typem procesoru, který se již v současnosti nevyrábí, je Pentium. Ve své koncepci se dále přibližuje procesorům RISC. Jednou ze základních charakteristik tohoto a následujících typů procesorů je několik vnitřních větví pro vykonávání instrukcí, umožňující provádět více instrukcí současně. Tato schopnost se nazývá superskalární architektura procesoru.

Blokový diagram procesoru Pentium:



Pentium má vnitřní šířku datové sběrnice 32 bitů, vnější datová sběrnice je 64 bitová. Je dále zvětšena i velikost interní paměti cache na 16 kB. Vlivem určitých konstrukčních úprav (např. předvídání cílové adresy podmíněných skoků, zpracování až dvou instrukcí současně ve dvou aritmetických jednotkách) je dosaženo podstatné zvýšení výpočetní rychlosti oproti předcházejícím typům procesorů. Je kompatibilní s AMD K5, Cyrix M1, nekompatibilní s NexGen Nx586.

Pentium MMX

Další variantou procesoru Pentium bylo Pentium MMX (MultiMedia eXtention). Tento typ procesoru byl rozšířen o 57 instrukcí - jsou to rychlé instrukce pro podporu zpracování dat moderních multimediálních formátů. MMX k tomu využívá :

- Malé celočíselné datové typy
- Krátké opakující se smyčky
- Časté násobení a sčítání.
- Výpočetně náročné algoritmy
- Vysoce paralelní instrukce.

Zvýšení výkonu je dosaženo systémem SIMD (Single Instruction Multiple Data - podpora paralelního zpracování dat) a to tak, že jsou data soustředěna do delších registrů po 64 bitech, tedy 8 jednobytových dat, 4 dvoubytových nebo 2 čtyřbytových. Nad touto dvojicí, čtveřicí nebo osmicí lze vykonávat určité operace (např. násobení dvojic a pak je sečíst). Protože jsou užívány registry FPU nedoporučuje se směšovat multimediální aplikace s výpočty ve FPU. Pozor - aplikace napsaná pro MMX proto může zhroutit počítač s procesorem, který nemá MMX.

Pro normální kancelářskou techniku nejsou procesory MMX přínosem z hlediska využití MMX instrukcí. Procesory MMX mají obvykle větší L1 cache a tím i větší průchodnost. Procesory pracují s frekvencí 150, 166 a 200 MHz.

Pentium MMX2

U MMX2 je sada instrukcí MMX je rozšířena o další instrukce (70) včetně instrukcí v pohyblivé čarce, které tak umožňují procesoru rychlé zpracování 3D aplikací, zpracování MPEG2 a zvuků MP3.

Pentium P55c

Jedná se o vylepšené Pentium s těmito vlastnostmi:

- Taktovací frekvence 150 MHz
- využita technologie CMOS (dříve BiCMOS), která umožňuje docílit vyšších rychlostí a menších ztrát
- napájení 2,5 až 2,9V
- pouzdro s 251 vývody

Pentium Pro (pracovní název P6)

Tento typ procesoru byl uveden na trh v r.1995. Zachovává kompatibilitu s řadou 80x86. Stejně jako Pentium je vyroben technologií BiCMOS. Tvar má obdélníkový, protože se skládá ze dvou čipů (sekundární cache je na samostatném čipu v pouzdře procesoru). Pouzdro má 387 vývodů. Jádro procesoru je typu RISC jako celek je však nadále CISC procesorem.

Procesor obsahuje pět paralelních prováděcích jednotek:

- 2 celočíselné
- 1 pro načítání
- 1 pro ukládání
- 1 FPU

Dále je integrována primární instrukční cache 8 kB, primární datová cache 8 kB a sekundární cache 512 (256) kB. Datová sběrnice je 64bitová, adresová sběrnice 36bitová.

Možnosti procesoru se rychle měnily :

- 150 MHz taktovací frekvence /60 MHz (takt sběrnice), 256 kB L2 cache
- 180/60 MHz, 256 kB L2 cache (prosinec 95)
- 200/66 MHz, 256 kB L2 cache (leden 96)
- od března 1996 512 kB L2 cache, od r. 1998 až 1 MB

Processor Intel Pentium II

Processor Pentium II je šestou generací v řadě 80x86. Jeho výroba započala v roce 1997. Po stránce funkční je to procesor s obdobnými vlastnostmi jako procesor Intel Pentium Pro. Obsahuje standardně rozšíření o technologii MMX. Podstatnou změnou je to, že je vyráběn v novém pouzdře S.E.C. (Single Edge Contact Cartridges). Se základní deskou je nové pouzdro spojeno hranovým konektorem přes nový Slot 1. Do tohoto slotu nelze zapojit žádný konkurenční procesor. Další podstatnou změnou je zvýšení taktu vnější sběrnice na 100 MHz.

Processor Intel Celeron

Tento typ procesoru je určen do levnějších typů počítačů. Externí sběrnice pracuje na 66 MHz. Jeho nižší cena je dána tím, že neobsahuje sekundární cache (L2). Tím však procesor ztrácí na výkonu, především u výpočtů v pevné řádové čárce a konkurenční procesory (AMD, Cyrix) jej výkonově překonávají. Intel proto zareagoval a záhy přišel s novou verzí tohoto procesoru (Celeron 300A), který už standardně obsahuje cache 128 kB. Od frekvence 333MHz výše už všechny procesory Celeron obsahují L2 cache o velikosti 128 kB. Procesory Celeron se vyráběly jak v provedení pro Slot 1, tak i v pouzdře PPGA (Plastic Pin Grid Array) pro nově zavedenou patici Socket 370 (obdobu Socket 7). Pro základní desky se Slotem 1 je možné použít redukci Socket 370/Slot 1. Patice Socket 370 si získala velkou oblibu a dodnes ji používá řada výrobců procesorů (VIA C3), možná také proto, že nové patice si Intel patentově chrání. V současné době se již vyrábí procesor **Intel Celeron II**, založený na procesoru Pentium III, oproti kterému má zmenšenou velikost paměti cache L2.

Processor Intel Pentium III

Procesory Intel Pentium III pracují s taktovací frekvencí od 400 MHz. Od frekvence 600MHz dochází ke změně jádra procesoru z jádra Katmai (0,25mikronová technologie) na Coppermine (0,18mikronová technologie). Nová technologie umožňuje zvýšení výkonu a snížení spotřeby. Současně bylo možné integrovat L2 cache přímo na čip procesoru. Pentium III Coppermine využívají 256 kB L2 cache typu *on-die*, která pracuje na stejné rychlosti jako procesor. Procesory se vyráběly ve dvojnásobném provedení - jednak jako SECC2 pro Slot 1 a jako FCPGA pro Socket 370. Procesory v provedení FCPGA však nelze použít ve starších patičkách Socket 370 pro PPGA procesory, protože mají jeden vývod (pin) navíc, který je u Celeronu nezapojen. Odlišná je také velikost napájecího napětí - u FCPGA je to 1,6V a u PPGA procesorů Celeron 2V. Další vlastnosti procesoru :

- 70 nových instrukcí SSE (Streaming SIMD Extensions, někdy označované jako KNI - Katmai New Instructions), které přinášejí zvýšení výkonu zejména ve 3D oblasti a ve zpracování zvuku a videa.

- mikroarchitektura P6
- 100 a 133 MHz systémová sběrnice
- 440BX chipset Intel

Processor Intel Pentium 4

Využívá zcela nové jádro, které není založeno na Pentiu III. Odlišná architektura NetBurst umožňuje dosáhnout vyšších frekvencí. Na čipu jsou integrovány další vyrovnávací paměti, např. Execution Trace Cache (shromažďuje dekodované makroinstrukce procesoru, velikost 12 kB). Kapacita L2 cache zůstala sice na 256 kB, ale zvýšila se její propustnost. Procesor se vyrábí s nejnižší frekvencí od 1.3 GHz. Procesory Pentium 4 vyžadují pochopitelně novou čipovou sadu a nové základní desky (Socket 423), ale také nové zdroje ATX-P4 (ATX 12V) s přídatným konektorem pro napájení. Posledním typem je Pentium 4 HT (HyperThreading) se zdvojenými registry, který obsahuje 55 milionů tranzistorů. Využívá principu simulace několika procesorů v jednom - moderní programy zpracovávají několik úloh současně (lze urychlit použitím několika procesorů), procesor však často musejí čekat na pomalejší komponenty a tak je možné během tohoto čekání vytížit procesor jinou úlohou. Pokud je tedy procesor doplněn dalšími registry a podpůrnými obvody, jejichž výroba není nikterak složitá ani drahá, dosáhne se obdobného efektu jako při použití několika procesorů. Další vlastnosti procesoru Intel Pentium 4:

- podpora paměti DDR RAM a RDRAM

- systémová sběrnice s frekvencí 400 a 533 MHz (technologie DualBus)
- dalších 144 instrukcí SSE2 především pro zrychlení výpočtů v plovoucí desetinné čárce.

Kromě výše uvedených typů vyráběla firma Intel v minulosti i řadu dalších procesorů, které se však z různých důvodů komerčně neprosadily nebo nebyly určeny pro všeobecné použití. Specifickou kategorií tvoří procesory pro servery a profesionální pracovní stanice, kam lze zařadit např. Intel Xeon nebo 64bitový Intel Itanium (IA-64). Další skupinu jsou procesory pro mobilní PC - notebooky, které se vyznačují sníženou spotřebou a nižším tepelným výkonem.

Processor Itanium, Itanium2

Jedná se o první 64bitový procesor firmy Intel. První verze pracovaly na rychlosti 733- 1 GHz, Má tři úrovně Cache paměti, procesorová sběrnice 64 bitů, rychlost 266 MHz, adresace až 16 TB fyzické paměti.

Shrnutí

Mikroprocesory Intel

Name	Date	Trans.	Microns	Clk	Data	MIPS
8080	1974	6,000	6	2 MHz	8 bits	0.64
8088	1979	29,000	3	5 MHz	16 bits, 8-bit bus	0.33
80286	1982	134,000	1.	5 6 MHz	16 bits	1
80386	1985	275,000	1.5	16 MHz	32 bits	5
80486	1989	1,200,000	1	25 MHz	32 bits	20
Pentium	1993	3,100,000	0.8	60 MHz	32 bits, 64-bit bus	100
Pentium II	1997	7,500,000	0.35	233 MHz	32 bits, 64-bit bus	~300
Pentium III	1999	9,500,000	0.25	450 MHz	32 bits, 64-bit bus	~510
Pentium 4	2000	42,000,000	0.18	1.5 GHz	32 bits, 64-bit bus	~1,700

...

6.5 PAMĚTI

Paměť počítače je zařízení, které slouží k ukládání programů a dat, s nimiž počítač pracuje. Paměti lze rozdělit do tří základních skupin:

- registry: paměťová místa na čipu procesoru, která jsou používána pro krátkodobé uchování právě zpracovávaných informací
- vnitřní (interní, operační) paměti: paměti osazené většinou na základní desce. Bývají realizovány pomocí polovodičových součástek. Jsou do nich zaváděny právě spouštěné programy (nebo alespoň jejich části) a data, se kterými pracují.
- vnější (externí) paměti: paměti realizované většinou za pomoci zařízení používajících výměnná média v podobě disků či magnetofonových pásek. Zápis do externích pamětí se provádí většinou na magnetickém nebo optickém principu. Slouží pro dlouhodobé uchování informací a zálohování dat.

Základní parametry pamětí jsou:

- **kapacita:** množství informací, které je možné do paměti uložit

- **přístupová doba:** doba, kterou je nutné čekat od zadání požadavku, než paměť zpřístupní požadovanou informaci
- **přenosová rychlost:** množství dat, které lze z paměti přečíst (do ní zapsat) za jednotku času
- **statičnost / dynamičnost:**
 - **statické paměti:** uchovávají informaci po celou dobu, kdy je paměť připojena ke zdroji elektrického napětí
 - **dynamické paměti:** zapsanou informaci mají tendenci ztrácet i v době, kdy jsou připojeny k napájení. Informace v takových pamětech je nutné tedy neustále periodicky oživovat, aby nedošlo k jejich ztrátě.
- **destruktivnost při čtení:**
 - **destruktivní při čtení:** přečtení informace z paměti vede ke ztrátě této informace. Přečtená informace musí být následně po přečtení opět do paměti zapsána.
 - **nedestruktivní při čtení:** přečtení informace žádným negativním způsobem tuto informaci neovlivní.
- **energetická závislost:**
 - **energeticky závislé:** paměti, které uložené informace po odpojení od zdroje napájení ztrácejí
 - **energeticky nezávislé:** paměti, které uchovávají informace i po dobu, kdy nejsou připojeny ke zdroji elektrického napájení.
- **přístup**
 - **sekvenční:** před zpřístupněním informace z paměti je nutné přečíst všechny předcházející informace
 - **přímý:** je možné zpřístupnit přímo požadovanou informaci
- **spolehlivost:** střední doba mezi dvěma poruchami paměti
- **cena za bit:** cena, kterou je nutno zaplatit za jeden bit paměti

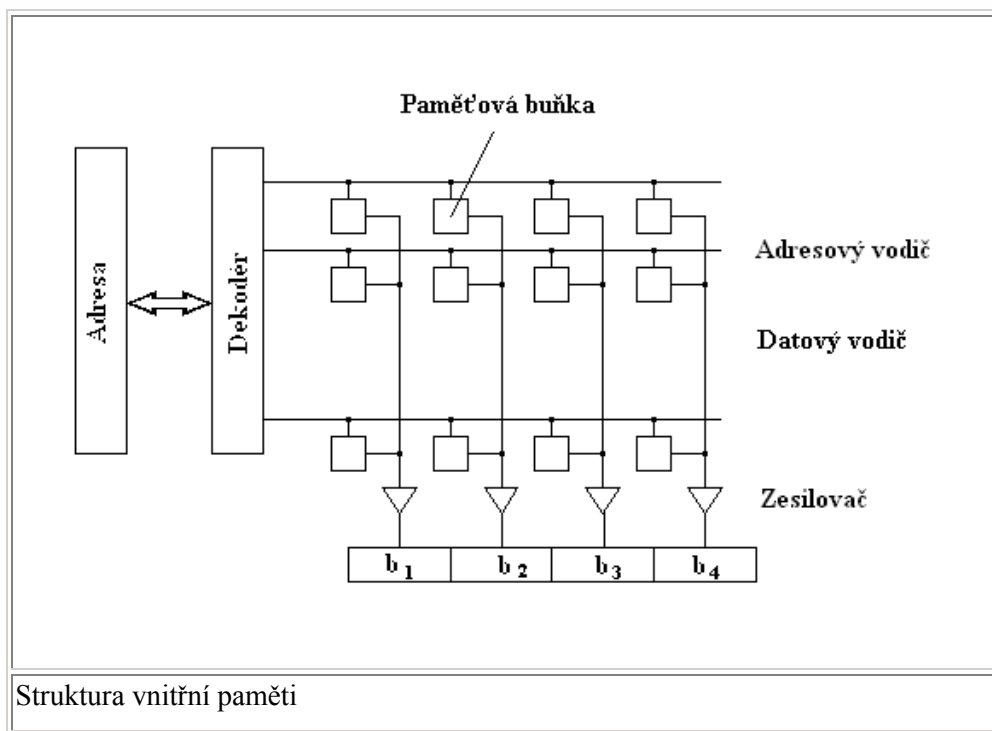
Následující tabulka ukazuje výše popsané tři typy paměti a jejich parametry.

	registry	vnitřní paměti	vnější paměti
kapacita	velmi malá (jednotky bytů)	vyšší (řádově 100 kB - 100MB)	vysoká (řádově 10 MB - 10 GB)
přístupová doba	velmi nízká (velmi rychlá paměťová místa)	vyšší (řádově 10 ns)	vysoká (řádově 10 ms - 10 min)
přenosová rychlost	vzhledem k malé kapacitě se většinou neuvažuje	vysoká (řádově 1 - 10 MB/s)	nižší než u vnitřních pamětí (řádově 10 MB/min - 1 MB/s)
statičnost / dynamičnost	statické	statické i dynamické	statické
destruktivnost při čtení	nedestruktivní	destruktivní i nedestruktivní	nedestruktivní
energetická závislost	závislé	závislé	nezávislé
přístup	přímý	přímý	přímý i sekvenční
spolehlivost	velmi spolehlivé	spolehlivé	méně spolehlivé
cena za bit	vzhledem k nízké kapacitě vysoká	nižší než u registrů a vyšší než u vnějších pamětí	vzhledem k vysoké kapacitě nízká

Vnitřní paměti

Interní paměti jsou zapojeny jako matice paměťových buněk. Každá buňka má kapacitu jeden bit. Takováto buňka tedy může uchovávat pouze hodnotu logická jedna nebo logická 0.

Obecná struktura vnitřní paměti.



Při přístupu do paměti (čtení nebo zápis) je vždy udána adresa paměťového místa, se kterým se bude pracovat. Tato adresa je přivedena na vstup dekodéru. Dekodér pak podle zadané adresy vybere jeden z adresových vodičů a nastaví na něm hodnotu logická 1. Podle toho, jak jsou zapojeny jednotlivé paměťové buňky na příslušném řádku, který byl vybrán dekodérem, projde resp. neprojde hodnota logické jedničky na datové vodiče. Informace je dále na koncích datových vodičů zesílена zesilovačem. V případě, že hodnota logická jedna projde přes paměťovou buňku, obdržíme na výstupu hodnotu bitu 1. V opačném případě je na výstupu hodnota bitu 0.

Zcela analogický je postup i při zápisu hodnoty do paměti. Opět je nejdříve nutné uvést adresu paměťového místa, do kterého se bude zapisovat. Dekodér vybere adresový vodič příslušný zadané adrese a nastaví na něj hodnotu logická 1. Dále se nastaví hodnoty bitů b₁ až b₄ na hodnoty, které se budou do paměti ukládat. Tyto hodnoty jsou potom uloženy do paměťových buněk na řádku odpovídajícím vybranému adresovému vodiči.

Zpravidla základní kritérium pro jejich dělení je to, zda je možné z paměti data pouze číst (paměti ROM), nebo je možné do paměti i data zapisovat (paměti RAM).

6.5.1 Paměti typu ROM

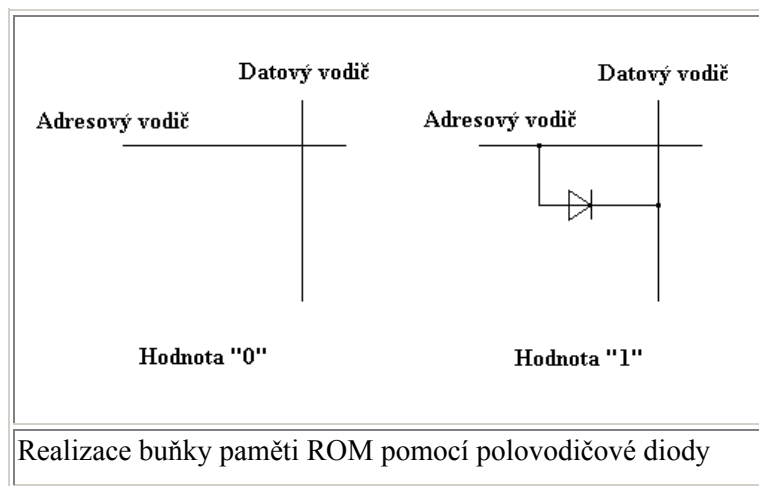
(Read Only Memory - paměti pouze pro čtení)

Data jsou v paměti k dispozici i po vypnutí počítače. V počítači jsou paměti typu ROM využívány především pro uložení BIOSu. Dle způsobu zápisu dat do paměti se dají paměti ROM ještě rozdělit na paměti :

ROM

Tyto paměťové moduly obsahují program (data), vložený do paměti při výrobě. Jednotlivé bity paměti jsou tvořeny prvky (odpory, pojistky, apod.) a zápis log0 (nebo log1) se provede tím, že se odpovídající prvky přepálí a ostatní ponechají.

Paměťová buňka paměti ROM může být realizována jako dvojice nespojených vodičů a vodičů propojených přes polovodičovou diodu.

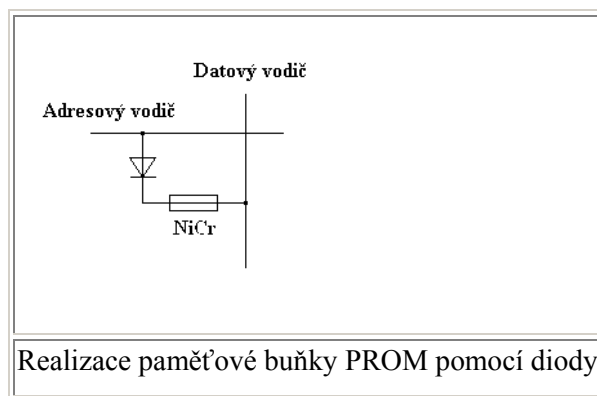


V prvním případě nemůže žádným způsobem hodnota logická jedna přejít z adresového vodiče na vodič datový. Jedná se tedy o buňku, ve které je permanentně uložena hodnota 0. V případě druhém hodnota logická 1 přejde z adresového vodiče přes polovodičovou diodu na vodič datový. Toto zapojení představuje tedy paměťovou buňku s hodnotou 1. Dioda je zapojena tak, aby hodnota logická 1 mohla přejít z adresového vodiče na datový, ale nikoliv v opačném směru, což by vedlo k jejímu šíření po velké části paměti.

PROM (Programmable ROM)

Tyto paměti jsou založeny na stejném principu jako ROM. Z výroby jsou však prázdné a informaci do nich vkládá až uživatel pomocí programátoru paměti PROM. Stejně jako u paměti ROM je jejich obsah neměnný a trvalý.

Buňku paměti je možné realizovat podobně jako u paměti ROM. Při výrobě je vyrobena matice obsahující spojené adresové vodiče s datovými vodiči přes polovodičovou diodu a tavnou pojistku z niklu a chromu (NiCr). Takto vyrobená paměť obsahuje na začátku samé hodnoty 1.



Zápis informace se provádí vyšší hodnotou elektrického proudu (cca 10 mA), která způsobí přepálení tavné pojistky a tím i definitivně zápis hodnoty 0 do příslušné paměťové buňky.

EPROM (Erasable PROM)

U těchto pamětí se binární informace uchovává pomocí elektrického náboje (miniaturní kondenzátor). Vzhledem k značnému svodovému odporu se náboj drží i po odpojení napětí. Tyto paměti se také uživatelsky programují pomocí programátoru. Na rozdíl od předchozích pamětí je možné paměti EPROM vymazat odvedením náboje (např. působením ultrafialového záření cca 0,5 hod) a znovu naprogramovat.

EEPROM (Electrically EPROM)

Paměti EEPROM jsou variantou pamětí EPROM. Rozdíl je ve způsobu mazání. Paměti EEPROM se mažou působením elektrických impulsů (řádově ms).

Flash-PROM

Tyto paměti se svými vlastnostmi velmi podobají pamětem RAM, po odpojení napájení se však nevymažou. Jejich výhodou je, že je možné tyto paměti programovat přímo v PC. Nejčastěji se používají pro uložení programu BIOS.

6.5.2 Paměti typu RAM

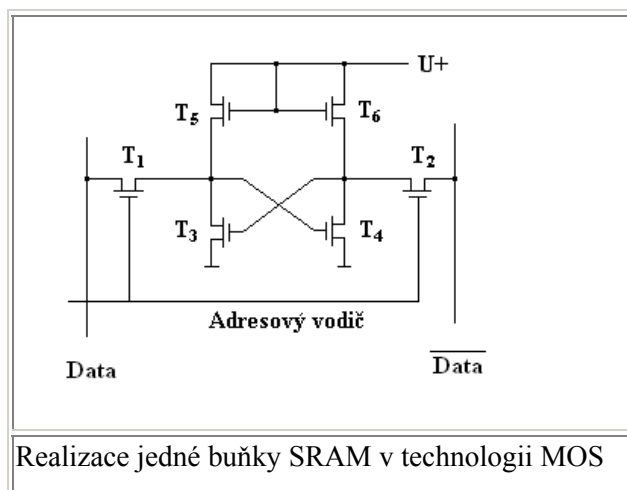
(Random Access Memory - paměti s náhodným přístupem).

Tyto paměti jsou rychlejší než paměti ROM a jsou využívány jako operační paměť počítače. Dají se využít pro čtení i zápis, po vypnutí počítače ztrácí informaci. Dle způsobu realizace je možné rozlišovat :

Statické paměti RAM (SRAM)

Jsou velmi rychlé (přístupová doba cca 20 ns) ale výrobně složité a tedy drahé. Používají se především jako rychlé paměti CACHE kde je potřebná značná rychlost a vystačíme s omezenou kapacitou.

Paměťová buňka SRAM je realizována jako bistabilní klopný obvod, tj. obvod, který se může nacházet vždy v jednom ze dvou stavů, které určují, zda v paměti je uložena 1 nebo 0. Příklad zapojení :



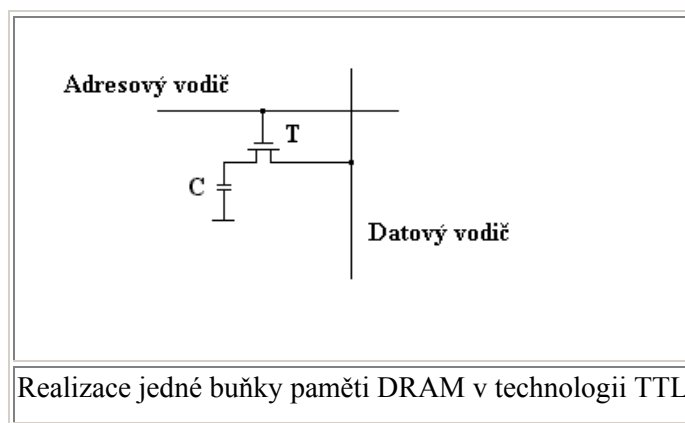
U SRAM paměti se používá dvou datových vodičů. Vodič Data je určený k zápisu do paměti. Vodič označený jako $\overline{\text{Data}}$ (s pruhem) se používá ke čtení. Hodnota na tomto vodiči je vždy opačná než hodnota uložená v paměti. Takže na konci je nutno ji ještě negovat. Při zápisu se na adresový vodič umístí hodnota logická 1. Tranzistory T1 a T2 se otevřou. Na vodič Data se přivede zapisovaná hodnota (např. 1). Tranzistor T1 je otevřen, takže jednička na vodiči Data otevře tranzistor T4 a tímto dojde k uzavření tranzistoru T3. Tento stav obvodu představuje

uložení hodnoty 0 do paměti. Zcela analogicky tato buňka pracuje i při zápisu hodnoty 1. Rozdíl je pouze v tom, že tranzistor T4 zůstane uzavřen a to způsobí otevření tranzistoru T3.

Při čtení je opět na adresový vodič přivedena hodnota logická 1, což opět způsobí otevření tranzistorů T1 a T2. Jestliže byla v paměti zapsána hodnota 1, je tranzistor T4 otevřen (tj. na jeho výstupu je hodnota 0). Tuto hodnotu obdržíme na vodiči \DATA. Opět zcela analogicky v případě uložené hodnoty 0, kdy tranzistor T4 je uzavřen (tj. na jeho výstupu je hodnota 1).

Dynamické paměti RAM (DRAM)

Tyto paměti jsou výrobně levné. Vzhledem k tomu, že jsou v principu realizovány jako miniaturní kondenzátory, ztrácející rychle svoji kapacitu, je třeba náboj rychle obnovovat. Tento proces se nazývá refresh. Tuto funkci plní některý z obvodů čipové sady. Příklad zapojení:



Při zápisu se na adresový vodič přivede hodnota logická 1. Tím se tranzistor T otevře. Na datovém vodiči je umístěna zapisovaná hodnota (např. 1). Tato hodnota projde přes otevřený tranzistor a nabije kondenzátor. V případě zápisu nuly dojde pouze k případnému vybití kondenzátoru (pokud byla dříve v paměti uložena hodnota 1).

Při čtení je na adresový vodič přivedena hodnota logická 1, která způsobí otevření tranzistoru T. Jestliže byl kondenzátor nabitý, zapsaná hodnota přejde na datový vodič. Tímto čtením však dojde k vybití kondenzátoru a zničení uložené informace. Jedná se tedy o buňku, která je destruktivní při čtení a přečtenou hodnotu je nutné opět do paměti zapsat.

Buňka paměti DRAM je velmi jednoduchá a dovoluje vysokou integraci a nízké výrobní náklady. Díky těmto vlastnostem je používána k výrobě operačních pamětí. Její nevýhodou je však vyšší přístupová doba (60 - 70 ns) způsobená nutností provádět refresh a časem potřebným k nabití a vybití kondenzátoru.

V průběhu vývoje vzniklo několik typů dynamických RAM, které se liší svou rychlostí a principem činnosti, například :

- FPM – Fast Page Mode. Využívá se skutečnosti, že požadovaná informace je v souvislé oblasti a zároveň fyzického způsobu přístupu k datům (stránkování) pomocí adresace ve stránkách paměti – nejprve se hledá řádka a potom sloupec paměťové buňky. Při změně adresy se první byte dat čte plným počtem taktů, ale následující na sousedním místě (tři následné buňky) ve stránce menším počtem taktů (typicky 5-3-3-3) bez dalších čekacích stavů. Další možností je technika prokládání čtení střídavě ze dvou bank fyzické paměti, kde se dosáhne částečného překrytí operací.
- EDO – Extended Data Output. Vychází z FPM, ale využívá čipy umožňující překrývání časování suboperací při přístupu do paměti. (typicky 5-2-2-2)
- SDRAM – Synchronous DRAM, jsou synchronní s kmitočtem základní desky, např. 133MHz (typicky 5-1-1-1)

- DDR SDRAM (Double Data Rate) – odvozeny od SDRAM, ale 2 datové přenosy v jednom datovém cyklu. Informace jsou tak přenášeny jak na vzestupné, tak na sestupné hraně hodinového taktu (taktu na sběrnici). Paměť DDR PC2100, která je taktována na 133 MHz pak po vynásobení 2 (t.j. sestup. i vzestup.hrana) pak má 266 MHz. Je tak v každém cyklu přeneseno 64 bitů (8 bytů). Celková propustnost v ideálním případě činí 2 128 MB/s. V případě pomalejších pamětí DDR PC 1600 je frekvence 2 x 100 MHz a šířka pásma (datová propustnost) 1600 MB/s. Na trhu se objevují nové paměti PC 2700 a DDR 333. V jejich případě je frekvence 166 MHz, což představuje teoretickou datovou propustnost 2 656 MB/s.
- RDRAM – Rambus DRAM. Využívá speciální rychlé paměťové sběrnice (paměti s úzkou sběrnici, s rychlostí 800 MHz). Podpůrné čipy pro komunikaci s řadičem paměti jsou umístěny v modulu RIMM(rambus inline memory module)

CMOS RAM (Complementary Metal Oxide Silicon)

V PC slouží pro zápis parametrů programu Setup.Vzhledem k technologii výroby mají nepatrnou spotřebu a po vypnutí počítače jsou napájeny z miniaturní baterie.

Fyzická organizace paměti

Po stránce fyzické realizace pamětí postupně docházelo k vývoji, a to především v důsledku zvětšování kapacity pamětí.

1. Paměti DIP - u prvních typů počítačů (do třídy 286) byly paměti integrovány do paměťového pouzdra DIP

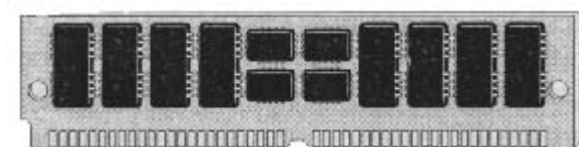
2. Paměťové moduly SIMM (Single Inline Memory Module) umožňovaly modulární rozšiřování pamětí. Původní moduly SIMM byly 30 pinové s datovou šířkou 8 nebo 16 bitů. kapacita 256 KB, 1 MB nebo 4 MB, v počítačích 286 až 486, přístupová doba 70 až 80 ns, dnes se již nepoužívají. Používají 72 pinové moduly o kapacitě 4, 8, 16, 32 a 64 MB. Přístupové doby jsou 60 - 70 ns. Šířka datového toku je 32 bitů. Nově vyráběné SIMM moduly pracují v režimu EDO (Extended Data Output) který se projevuje zvýšením rychlosti.

3. Paměťové moduly DIMM (Dual Inline Memory Module) jsou vylepšením SIMM modulů. Mají 168 pinů, což jim umožňuje 64 bitovou šířku toku dat. Jejich použití musí být podporováno základní deskou, 168 vývodů; kapacita 16, 32, 64, 128, 256

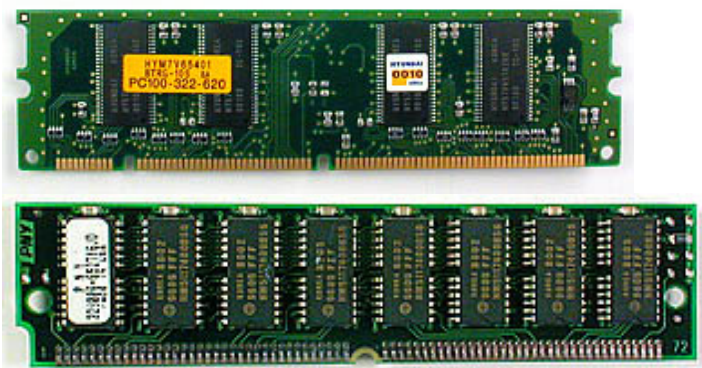
nebo 512 MB; přístupová doba 7 až 10 ns

4. Paměťové moduly RIMM (Rambus In-Line Memory Module), 184 vývodů; kapacita 64, 128, 256 nebo 512 MB, nejnovější

Paměťové moduly jsou uspořádány do tzv. banků, čili řady patič na základní desce. Způsob obsazení a kombinace modulů se liší u různých typů základních desek.



Obr. 4.1 Paměťový modul SIMM (72 pinů)



Obr. 4.2 Paměťové banky pro 72pinové SIMM moduly a DIMM

6.5.3 Cache paměti

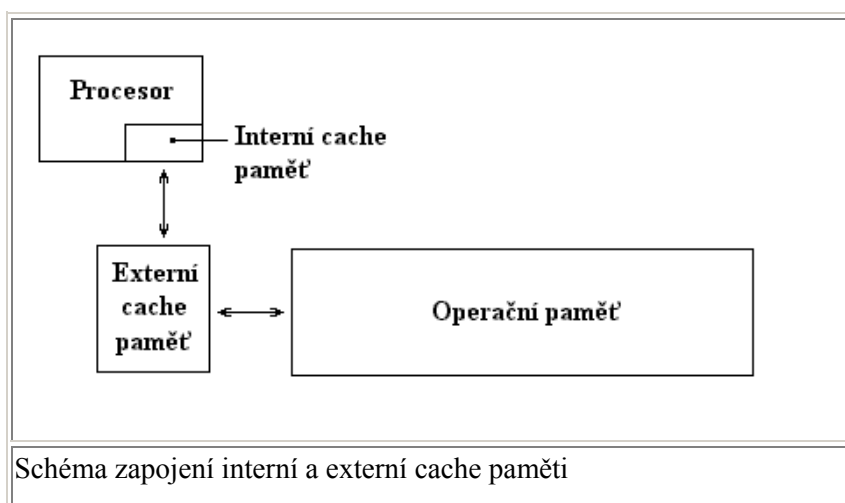
Cache paměť je rychlá vyrovnávací paměť mezi rychlým zařízením (např. procesor) a pomalejším zařízením (např. operační paměť). V dnešních počítačích se běžně používají dva druhy cache paměti:

externí (sekundární, L2) cache:

Externí cache paměť je paměť, která je umístěna mezi pomalejší operační paměti a rychlým procesorem. Tato paměť je vyrobena jako rychlá paměť SRAM a slouží jako vyrovnávací paměť u počítačů s výkonným procesorem, které by byly bez ní operační paměti velmi zpomalovány. První externí cache paměti se objevují u počítačů s procesorem 80386. Jejich kapacita je 32 kB popř. 64 kB. S výkonnějšími procesory se postupně zvyšuje i kapacita externích cache paměti na 128 kB, 256 kB, 512 kB. Externí cache paměť je osazena na základní desce počítače (výjimku tvoří procesory Pentium Pro a Pentium II, které mají externí cache paměť integrovanou v pouzdře procesoru). Její činnost je řízena řadičem cache paměti.

interní (primární, L1) cache:

Interní cache paměť je paměť, která slouží k vyrovnání rychlosti velmi výkonných procesorů a pomalejších paměti. Tento typ cache paměti je integrován přímo na čipu procesoru a je také realizován pomocí paměti SRAM. Interní cache paměť se objevuje poprvé u procesoru 80486 s kapacitou 8 kB. Takovýto procesor musí mít v sobě integrován také řadič interní cache paměti pro řízení její činnosti.



Práce cache paměti vychází ze skutečnosti, že program má tendenci se při své práci určitou dobu zdržovat na určitém místě paměti, a to jak při zpracování instrukcí, tak při načítání

(zapisování) dat z (do) paměti. Je-li požadována nějaká informace z paměti, je nejdříve hledána v cache paměti (interní, pokud existuje, a následně v externí). Pokud požadovaná informace není přítomna v žádné z cache pamětí, je zavedena přímo z operační paměti. Kromě momentálně požadované informace se však do cache paměti zavede celý blok paměti, takže je velká pravděpodobnost, že následně požadované informace již budou v cache paměti přítomny. Pokud dojde k zaplnění cache paměti a je potřeba zavést další blok, je nutné, aby některý z bloků cache paměť opustil. Nejčastěji se k tomuto používá **LRU** (Least Recently Used) algoritmu, tj. algoritmu, který vyřadí nejdéle nepoužívaný blok.

Cache paměti bývají organizovány jako tzv. **asociativní paměti**. Asociativní paměti jsou tvořeny tabulkou (tabulkami), která obsahuje vždy sloupec, v němž jsou umístěny tzv. **tagy** (klíče), podle kterých se v asociativní paměti vyhledává. Dále jsou v tabulce umístěna data, která paměť uchovává, a popř. další informace nutné k zajištění správné funkce paměti. Např.:

informace o platnosti (neplatnosti) uložených dat

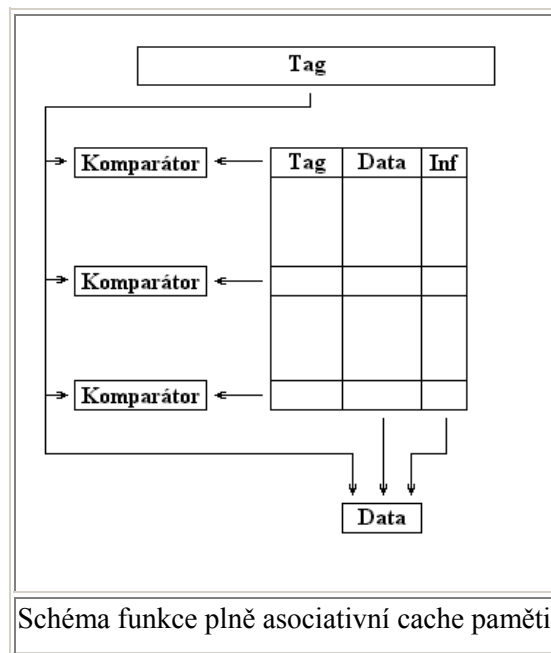
informace pro realizaci LRU algoritmu

informace protokolu **MESI** (Modified Exclusive Shared Invalid), který zajišťuje synchronizaci dat v cache pamětech v případě, že cache paměť je v počítači více (u interních cache pamětí v okamžiku, kdy počítač obsahuje více procesorů).

Při přístupu do cache paměti je nutné zadat adresu, z níž data požadujeme. Tato adresa je buď celá, nebo její část považovaná za tag, který se porovnává s tagy v cache paměti.

Cache paměti jsou konstruovány jedním ze tří způsobů.

plně asociativní:



U plně asociativní cache paměti je celá adresa, ze které se budou číst data (popř. na kterou se budou data zapisovat), brána jako tag. Tento tag je přiveden na vstup komparátorů (zařízení realizující porovnání dvou hodnot) společně s tagem v daném řádku tabulky. Pokud některý z tagů v tabulce je shodný se zadaným tagem na vstupu, ohlásí odpovídající komparátor shodu a znamená to, že požadovaná informace je v cache paměti přítomna a je možné ji použít. Pokud všechny komparátory signalizují neshodu, je to známkou toho, že požadovaná informace v cache paměti není a je nutné ji zavést odjinud (externí cache paměť, operační paměť).

Tento způsob cache paměti má své nevýhody:

Je nutné velké množství komparátorů

Vzhledem k tomu, že se musí v každém řádku tabulky uchovávat celý tag, musí mít cache paměť velkou kapacitu, do které se tyto tagy ukládají a kterou není možné využít k uchování dat.

Z těchto důvodů se plně asociativní paměti prakticky nepoužívají.

n-cestně asociativní:

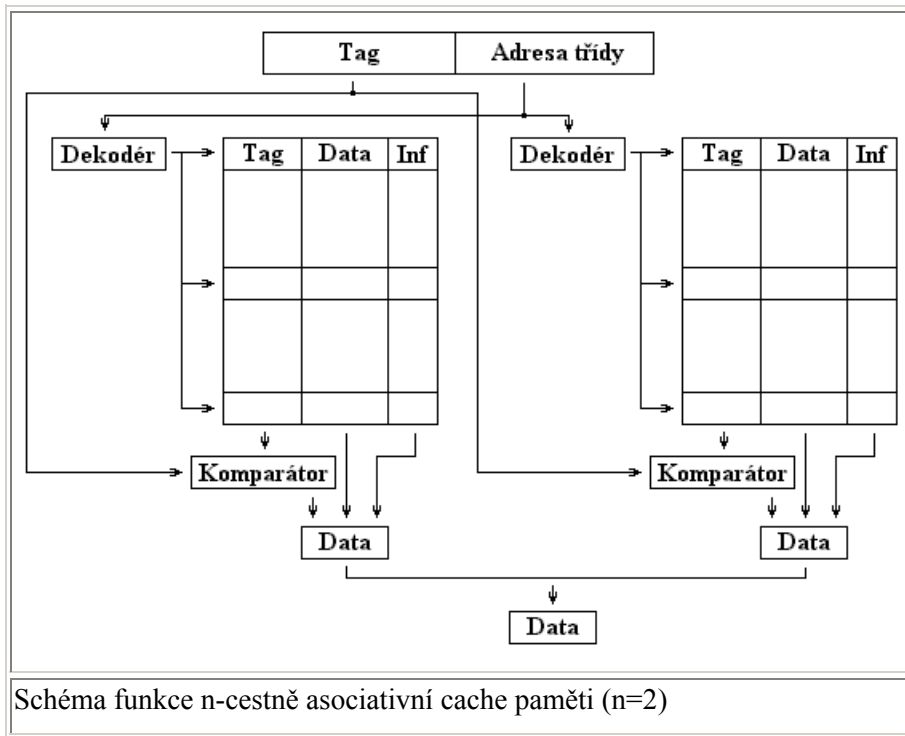


Schéma funkce n-cestně asociativní cache paměti (n=2)

N-cestně asociativní paměti pracují tak, že zadaná adresa se rozdělí na dvě části:

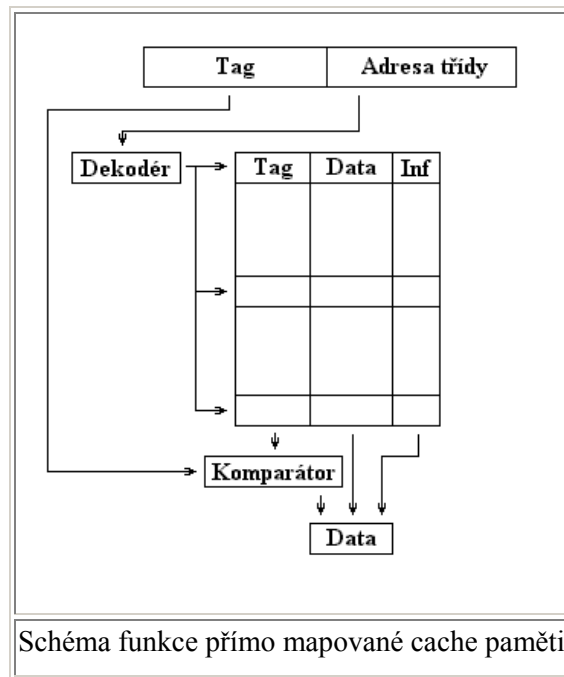
tag

adresa třídy

Adresa třídy je přivedena na n dekodérů (zařízení, které na základě vstupní hodnoty vybere jeden ze svých výstupů, na který umístí hodnotu log. 1, a na ostatní výstupy umístí hodnotu log. 0), které v každé tabulce vyberou jeden řádek. Z těchto řádků se potom vezmou příslušné tagy a komparátorem se porovnají se zadaným tagem. Podobně jako u plně asociativních cache pamětí pokud jeden z komparátorů signalizuje shodu, je informace v cache paměti přítomna. V opačném případě je nezbytné informaci hledat jinde.

N-cestně asociativní paměti částečně eliminují nevýhody plně asociativních cache pamětí a v současnosti jsou nejpoužívanějším typem cache pamětí.

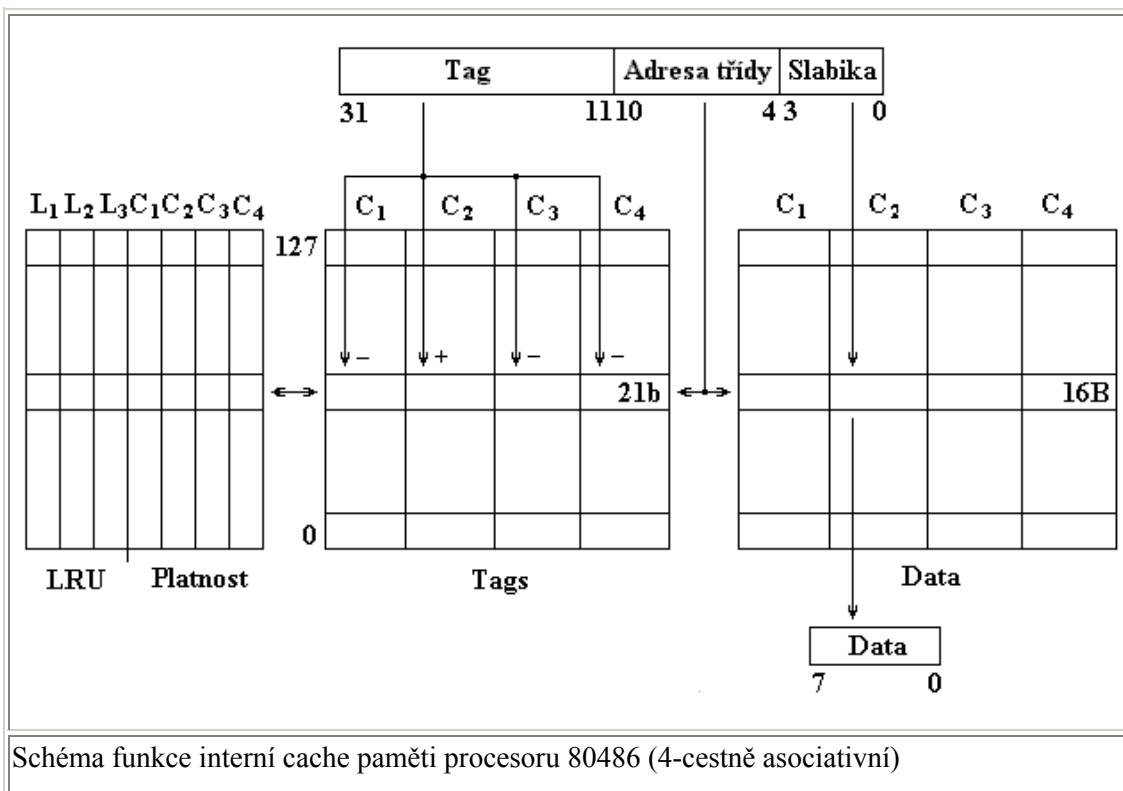
přímo mapovaná:



Přímo mapovaná cache paměť je speciální případ n-cestně asociativní cache paměti pro $n=1$. Zadaná adresa je opět rozdělena na tag a adresu třídy. Adresa třídy je přivedena na vstup dekodéru, který podle ní vybere jeden řádek v tabulce. Tag na tomto řádku je následně porovnán se zadaným tagem, čímž se rozhodne o přítomnosti resp. nepřítomnosti informace v cache paměti.

Přímo mapovaná cache ve srovnání s n-cestně asociativní cache paměti vykazuje nižší výkon, a proto její použití není dnes příliš časté.

Konkrétním příkladem cache paměti může být například interní cache paměť procesoru 80486, která je realizována jako 4-cestně asociativní cache paměť.



Adresa je rozdělena na tři části:

Tag - horních 21 bitů

Adresa třídy - 7 bitů => 128 řádků tabulky

Slabika - dolní 4 bity

Adresa třídy je přivedena na dekodér, který vybere jeden řádek. Zadaný tag je dále komparátorem porovnán proti 4 tagům ve vybraném řádku. Pokud jeden z komparátorů ohlásí shodu, provede se výběr dat v datové části paměti. Datová část obsahuje v každém sloupci 16B, ze kterých je pomocí dolních 4 bitů zadané adresy vybrán jeden požadovaný byte.

Každý řádek cache paměti ještě obsahuje jeden bit, který říká, zda informace v daném sloupci jsou platné, a 3 bity pro realizaci pseudo-LRU algoritmu. Pomocí tří bitů nelze vždy určit nejdéle nepoužívaný blok cache paměti. Tento algoritmus je však jednoduchý a rychlý a díky tomu poskytuje dostatečný výkon.

Podle způsobu práce při zapisování dat lze cache paměti ještě rozdělit do dvou skupin:

write-through: cache paměti, u kterých v případě zápisu procesoru do cache paměti dochází okamžitě i k zápisu do operační paměti. Procesor tak obsluhuje jen zápis a o další osud dat se stará cache paměť.

write-back: cache paměti, u nichž jsou data zapisována do operační paměti až ve chvíli, kdy je to třeba, a nikoliv okamžitě při jejich změně. K zápisu dat do operační paměti tedy dochází např. v okamžiku, kdy je cache zcela zaplněna a je třeba do ní umístit nová data. Tento způsob práce cache paměti vykazuje oproti předešlému způsobu vyšší výkon.

Poznámka: Kromě externích a interních cache pamětí je možné se setkat i se specializovanými cache pamětmi umístěnými mezi operační paměť a některé pomalejší zařízení (pevný disk, apod.).

6.5.4 CMOS paměť

Výrobní technologie CMOS (Complementary Metal Oxide Semiconductor) má v klidovém stavu velmi nízkou spotřebu elektrické energie, a proto se statické paměti s malou kapacitou používá k uchování údajů o nastavení počítače a jeho hardwarové konfiguraci. Tato paměť je energeticky závislá, a proto je nutné ji zálohovat pomocí akumulátoru umístěného většinou na základní desce, aby nedošlo ke ztrátě údajů v ní uložených.

V CMOS paměti bývají většinou uloženy:

- informace o typech a kapacitách jednotek pružných disků
- informace o typech, kapacitách a parametrech pevných disků
- typ používané video karty
- kapacita operační paměti
- nastavení parametrů cache paměti
- pořadí jednotek pro zavádění operačního systému
- povolení / zakázání různých funkcí základní desky:
- využívání interní a externí cache paměti
- antivirová ochrana systémových oblastí disků
- prohození pořadí jednotek pružných disků
- stínování určitých částí paměti (zavádění programového vybavení z pomalejší ROM paměti do rychlejší paměti RAM)
- činnosti rozhraní pružných disků, pevných disků
- činnosti vstup / výstupních portů
- nastavení rychlosti repetice klávesnice
- nastavení parametrů přenosu informací z pevných disků
- nastavení parametrů pro režim s úsporou elektrické energie
- nastavení přiřazení IRQ úrovní
- nastavení hesla k programu SETUP, popř. k celému počítači

Tyto parametry se nastavují většinou pomocí programu zvaného SETUP. SETUP bývá uložen nejčastěji v permanentní paměti počítače, která bývá realizována jako EPROM (u starších počítačů) nebo jako Flash (u novějších počítačů). Špatné nastavení výše zmíněných parametrů může způsobit výrazné snížení výkonu celého počítače, až nefunkčnost některých jeho částí, popř. nefunkčnost celého počítače.

Vzhledem k tomu, že tyto informace jsou pro počítač velmi důležité a jejich špatné hodnoty mohou být příčinou výše zmíněných problémů, není žádoucí, aby k nim měl přístup kdokoliv. Proto při přístupu do programu SETUP a tím i ke změnám parametrů v CMOS paměti je možné požadovat heslo.

6.6 PEVNÝ DISK

Pevný disk patří mezi jednu ze základních součástí počítače. Slouží jako velkokapacitní prostor pro ukládání dat, která zůstávají uchována i po vypnutí počítače. Data jsou na disku uchována prostřednictvím střídavého zmagnetování povrchové vrstvy nemagnetického kotouče resp. více kotoučů soustředně umístěných nad sebou. Nad rotující kotouče jsou vystavovány pomocí vystavovacích cívek (dříve krokových motorů) čtecí/záznamové hlavy, prostřednictvím kterých jsou snímány a zapisovány informace na disk. Vzdálenost hlav od disků je nepatrná a je udržována aerodynamickým vztlakem vznikajícím při rotaci kotoučů. Celý povrch kotoučů je rozdělen na soustředné stopy a ty jsou dále dělena na jednotlivé sektory. Sektor je pak základní jednotkou uložení dat. Celý proces probíhá při tzv. formátování disku, zpravidla před prvním použitím.

Základní charakteristiky disku jsou :

Kapacita disku

Přístupová doba

Typ řadiče (řídící jednotky)

Kapacita disku

Nejstarší disky měly kapacitu několik desítek MB. Současné disky mají běžně kapacitu 40-120 GB, špičkové až 200 GB. Zvyšování kapacity je dosaženo především zvyšováním hustoty záznamu dat a novými způsoby kódování.

Přístupová doba

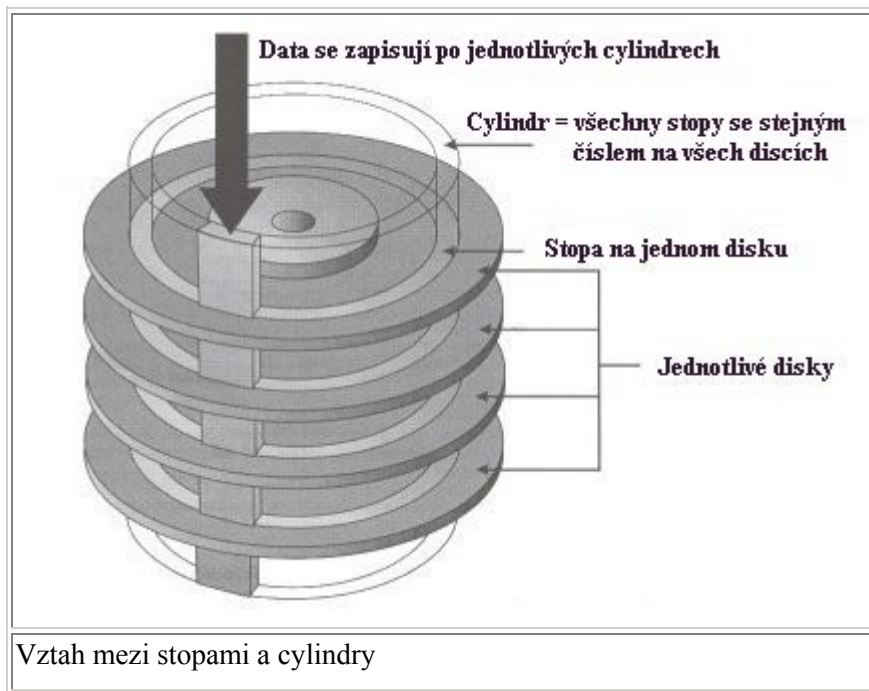
Přístupová doba (access time) udává, jak rychle jsou po požadavku na čtení k dispozici data. Tato doba se pohybuje okolo 10 ms a sestává z tzv. doby vystavení hlaviček (seek time) a doby čekání (rotary latency period) na pootočení disku do potřebné polohy.



Obr. 5.1 Jednotka pevného disku

6.6.1 Geometrie pevných disků

Všechny jednotlivé disky, ze kterých se celý pevný disk skládá, jsou podobně jako u pružného disku rozděleny do soustředných kružnic nazývaných **stopy (tracks)** a každá z těchto stop je rozdělena do **sektorů (sectors)**. Množina všech stop na všech discích se stejným číslem se u pevných disků označuje jako **válec (cylinder)**.



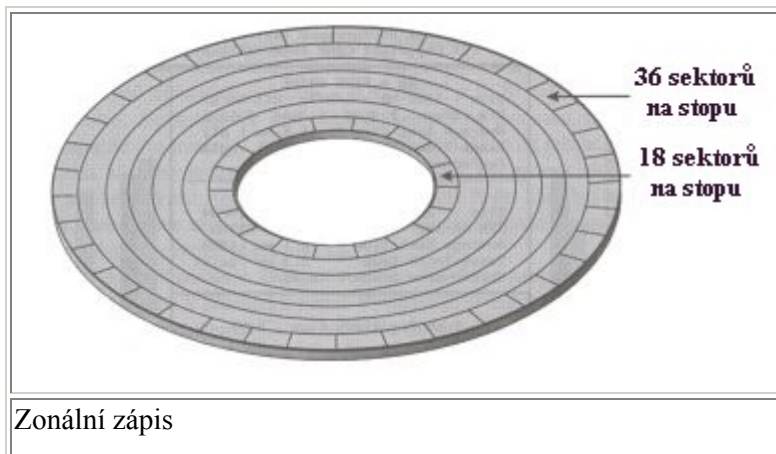
Geometrie disku udává hodnoty následujících parametrů:

Hlavy disku (heads): počet čtecích (zapisovacích) hlav pevného disku. Tento počet je shodný s počtem aktivních ploch, na které se provádí záznam. Většinou každý jednotlivý disk má dvě aktivní plochy a k nim příslušné čtecí (zapisovací) hlavy.

Stopy disku (tracks): počet stop na každé aktivní ploše disku. Stopy disku bývají číslovány od nuly, přičemž číslo nula má vnější stopa disku.

Cylindry disku (cylindry): počet cylindrů pevného disku. Tento počet je shodný s počtem stop. Číslování cylindrů je shodné s číslováním stop.

Sektory (sectors): počet sektorů, na které je rozdělena každá stopa. U většiny pevných disků je podobně jako u pružných disků počet sektorů na všech stopách stejný. Tento způsob do jisté míry plýtvá médiem, protože vnější stopy jsou delší a tudíž by se na ně mohlo umístit více sektorů. Existují však i pevné disky, u nichž se používá tzv. **zonální zápis** označovaný jako **ZBR (Zone Bit Recording)**. Jedná se metodu zápisu na pevný disk, která dovoluje umístit na vnější stopy pevného disku větší počet sektorů než na stopy vnitřní. ZBR tedy lépe využívá záznamové médium, ale způsobuje podstatně složitější přístup k datům. Sektory bývají číslovány od jedničky.



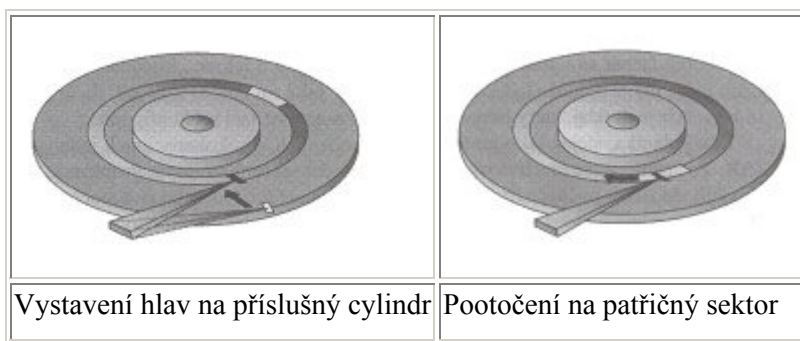
Zápis (čtení) na (z) pevný disk probíhá podobně jako u pružného disku na magnetickou vrstvu ve třech krocích:

vystavení zapisovacích (čtecích) hlav na příslušný cylindr pomocí krokového motorku (dříve) nebo elektromagnetu (dnes)

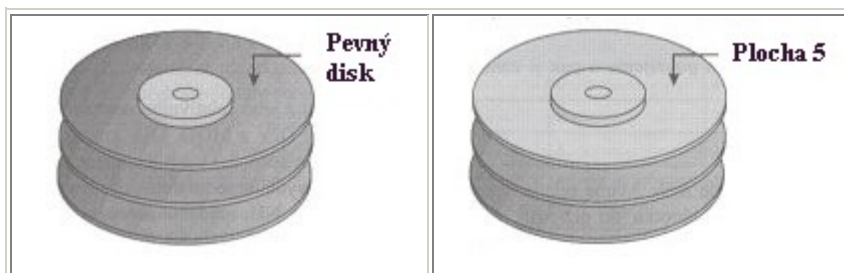
pootočení disků na patřičný sektor

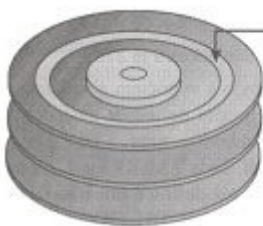
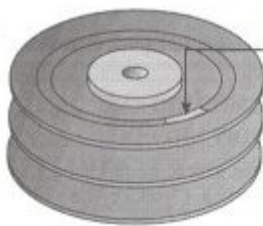
zápis (načtení) dat

Data jsou na pevný disk ukládána tak, že nejdříve je zaplněn celý 1. cylindr, potom 2. cylindr a tak dále až po poslední cylindr. Tento způsob dovoluje, aby se čtecí (zapisovací) hlavy podílely na čtení (zápisu) paralelně. Ukládání dat po jednotlivých discích by bylo podstatně pomalejší, protože v daném okamžiku by vždy mohla pracovat právě jedna hlava. Fáze zápisu (čtení) na (z) pevný disk:



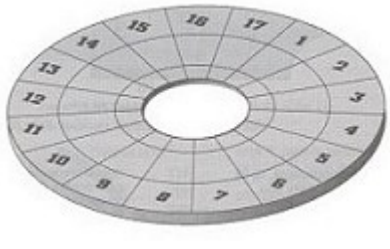
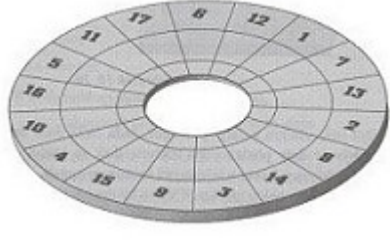
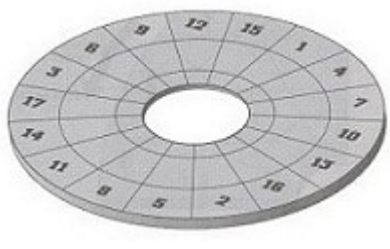
Fáze vyhledání 1. sektoru ve 40. cylindru na 5. povrchu pevného disku



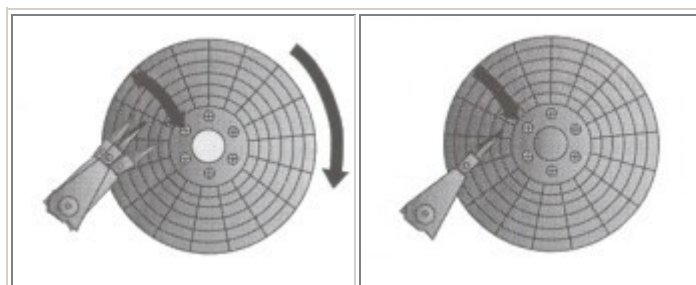
Pevný disk	Vyhledání 5. povrchu
 <p>Plocha 5 Cylindr 40</p>	 <p>Plocha 5 Cylindr 40 Sektor 1</p>
Vyhledání 40. cylindru	Vyhledání 1. sektoru

Protože rychlost otáčení pevného disku je poměrně vysoká, může se stát, že poté, co je přečten (zapsán) jeden sektor a data jsou předána dále, dojde k pootočení disků, takže čtecí (zapisovací) hlavy se nenacházejí nad následujícím sektorem, ale až nad některým z dalších sektorů. Nyní by tedy bylo nutné čekat další otáčku, než čtecí (zapisovací) hlavy budou nad požadovaným sektorem, a pak by se situace znovu opakovala. Protože tento způsob by velmi zpomaloval práci pevného disku, zavádí se tzv. **faktor prokládání pevného disku**. Jedná se o techniku, při které nejsou data zapisována (a posléze čtena) do za sebou následujících sektorů, ale jsou během jedné otáčky disku zapisována vždy do každého n-tého sektoru (faktor prokládání 1:n). Číslo n je voleno tak, aby po přečtení a zpracování dat z jednoho sektoru byla čtecí (zapisovací) hlava nad dalším požadovaným sektorem.

Faktory prokládání

		
Prokládání 1:1	Prokládání 1:3	Prokládání 1:6

Při vypnutí počítače (a tím i pevného disku) se pevný disk přestává otáčet. Tím přestává existovat tenká vrstva, na které se pohybují čtecí (zapisovací) hlavy a vzniká riziko jejich pádu na disky. Tento pád by totiž mohl jednotlivé disky poškodit. Proto v okamžiku, kdy má pevný disk ukončit svou činnost, je nezbytné, aby čtecí (zapisovací) hlavy byly přemístěny do zóny, která je speciálně uzpůsobena k jejich přistání. U starších pevných disků bylo nutné vždy před vypnutím počítače provést pomocí nějakého programu tzv. **zaparkování diskových hlav**, tj. jejich přemístění na patřičné místo. Nové pevné disky již využívají tzv. **autopark**, který je založen na tom, že po vypnutí pevného disku se pevný disk ještě chvíli setrvačností otáčí a tím vyrobí dostatek energie nutné pro přemístění hlav do parkovací zóny. Pro tuto parkovací zónu bývá většinou vyčleněna nejvnitřnější stopa disku, protože je na ní nejnižší rychlost.



Disk v chodu	Zaparkování hlav
--------------	------------------

Řadiče pevných disků

Řadič je řídicí jednotka, prostřednictvím které jsou čtena a zapisována data na pevný disk a která ve spolupráci se sběrnici zajišťuje přenos dat mezi diskem a procesorem. Řadič má podstatný vliv na kvalitu disku. V současné době se používají řadiče IDE, EIDE a SCSI.

IDE (Integrated Drive Electronics)

Řadič typu IDE byl navržen pro sběrnici ISA a patřil do nedávné doby mezi nejrozšířenější. Samotný řadič je integrován přímo na mechanice disku. Spojení se základní deskou je zajištěno prostřednictvím rozhraní, které je provedeno jako rozšiřující karta ISA. K této kartě je možné připojit dvě jednotky. Tyto jednotky musí být nastaveny jako MASTER (hlavní, nadřízená) je jednotka, ze které se načítá operační systém a jednotka SLAVE (podřízená). Nastavení se provádí prostřednictvím propojek (Jumper) umístěných na disku. V praxi však není zajištěna spolupráce všech typů disků různých výrobců, proto při instalaci dvou disků do PC je třeba jejich spolupráci odzkoušet.

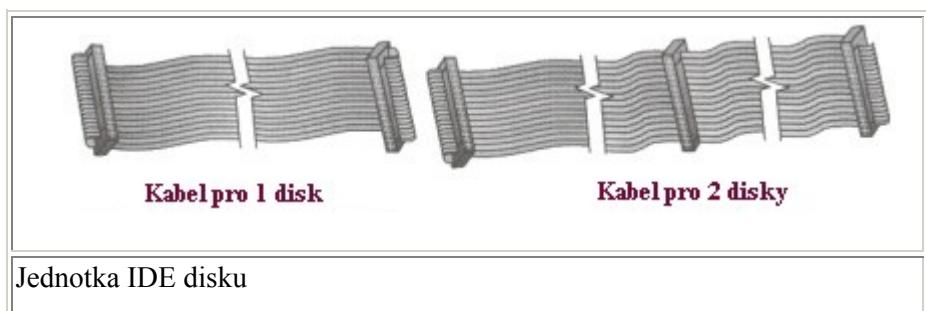
Základní charakteristikou pevných disků je jejich kapacita. Původní řadiče IDE používají k adresování dat na disku metodu CHS (Cylinder/Head/Sector). Maximální hodnoty jsou uvedeny v tabulce. Z nich plyne, že maximální kapacita byla pouhých 512 MB. Existovaly i IDE disky větších kapacit, jejich správná funkce byla zajišťována programově.

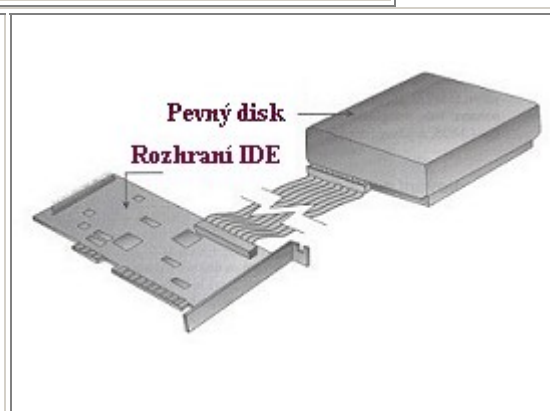
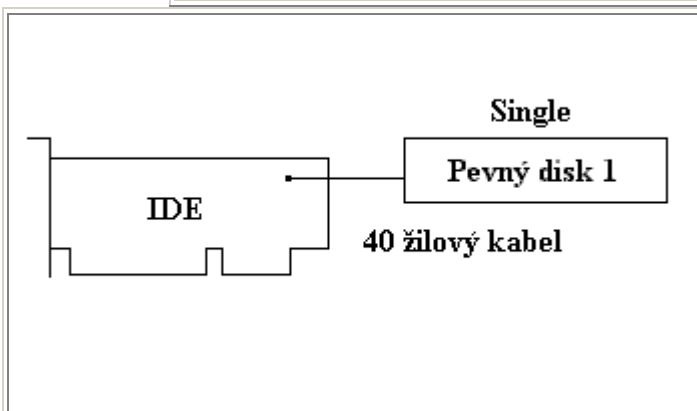
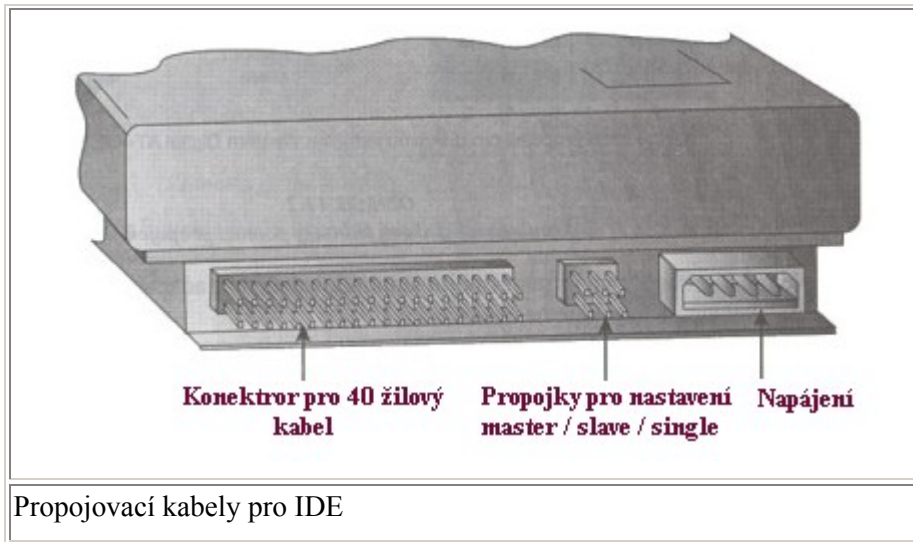
Cylinder	1024
Head	16
Sector	64
Celková kapacita [MB]	512

Tab. 5.1 Maximální hodnoty CHS pro disky IDE

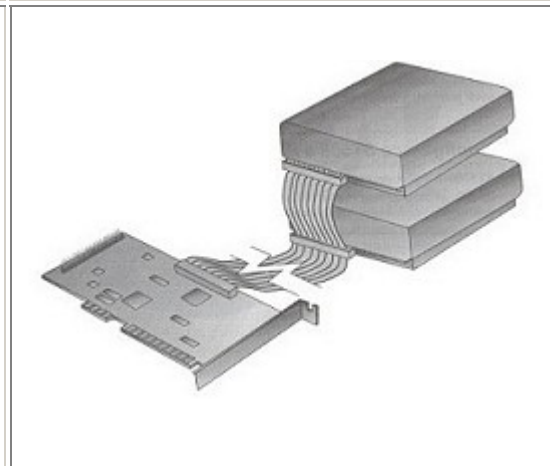
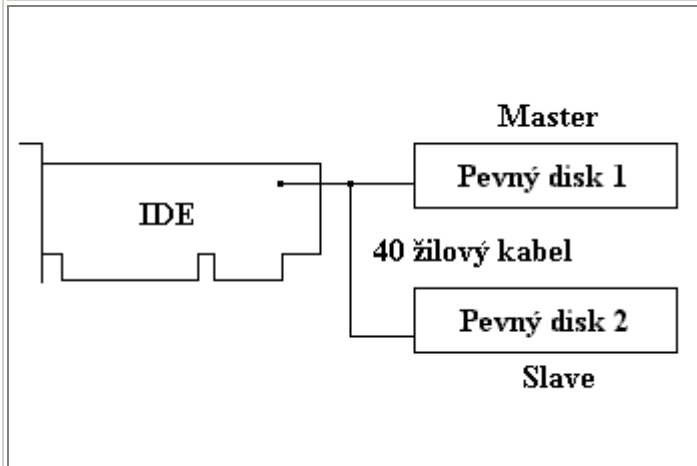
Na rozšiřující desce IDE jsou navíc i konektor pro připojení disketových mechanik, jeden konektor pro připojení LPT portu a dva konektory pro připojení sériových portů.

Teoretická hranice přenosové rychlosti je 8 MB/s a prakticky se pohybuje asi v rozmezí od 700 kB/s do 1400 kB/s. Zapojení diskových jednotek IDE se provádí pomocí jednoho 40 žilového kabelu.





Zapojení 1 disku na rozhraní IDE



Zapojení 2 disků na rozhraní IDE

Rozhraní IDE podobně jako ESDI dovoluje programově zjistit informace o geometrii připojených disků a je možné k němu připojit maximálně dva pevné disky. Protože každý z disků má svou řídicí jednotku umístěnu přímo u sebe, je nutné v případě zapojení dvou disků tyto disky nastavit pomocí propojek (jumperů) tak, aby jeden z nich byl jako master (hlavní) a druhý jako slave (podřízený). Operační systém se pak bude zavádět z disku označeného jako master. Doporučuje se, aby jako master byl nastaven novější disk, protože je možné předpokládat, že jeho elektronika bude lepší než elektronika staršího disku. V případě zapojení jednoho disku je nutné tento disk nastavit jako single (jediný). Toto nastavení bývá někdy

shodné jako nastavení pro master. Připojování jiných zařízení než jsou pevné disky není oficiálně podporováno.

Vzhledem k jednoduchosti rozhraní IDE bývá velmi často toto rozhraní integrováno na jedné desce společně s I/O porty.

Při komunikaci s pevným diskem má rozhraní IDE následující omezení:

4 bity pro adresaci povrchu disku (maximálně 16 povrchů)

10 bitů pro adresaci cylindru (maximálně 1024 cylindrů)

6 bitů pro adresaci sektoru (maximálně 64 sektorů)

Při zápisu 512 B do jednoho sektoru je takto kapacita omezena na 512 MB (0,5 GB).

EIDE (Enhanced IDE)

Tento typ řadiče byl vyvinut především z důvodu možnosti práce s většími disky než 512 MB. Je zpětně kompatibilní s disky IDE, ale na rozdíl od IDE je možné připojení až 4 současně pracujících jednotek. Řadič je integrován přímo do základní desky, není tedy nutná přídatná deska. Zvětšení kapacity disku bylo dosaženo nejčastěji metodou LBA (Logical Block Addressing - logické adresování bloků). Při tomto způsobu jsou sektory číslovány běžným způsobem od nuly do max.hodnoty. Každý sektor má ale ještě 28 bitovou adresu pro logickou adresaci. Při běžné velikosti sektoru 512 B je pak maximální teoretická kapacita disku LBA 128 GB. Jelikož 16 bit. operační systémy používají adresaci sektorů metodou CHS (viz IDE řadiče), musí být adresy přepočítávány. Původně byl přepočet prováděn rezidentním programem zaváděným do paměti při startu počítače, později byl tento přepočet součástí programu BIOS. Rozhraní na základní desce obsahuje dva konektory pro připojení až dvou jednotek, je tedy možné celkem připojit čtyři různá zařízení. Vzhledem k tomu, že do rozhraní EIDE byl zahrnut standard ATAPI (AT Attachment Packet Interface), obsahující příkazy umožňující práci s CD disky, je možné k EIDE libovolně připojovat i CD mechaniky.

Rychlost EIDE

Běžné řadiče EIDE byly mohly dosahovat rychlostí přenosu v rozmezí 3,33 až 16,66 MB/s. Reálné rychlosti se však pohybovaly okolo 6 MB/s. Další zvýšení bylo možné metodou přímého přístupu do paměti (DMA - Direct Memory Access), kdy se vlastního přenosu dat nezúčastní procesor, ale zajišťuje jej řadič DMA. S využitím nového přenosového protokolu Ultra DMA/33 bylo možné dosáhnout až teoretické rychlosti 33 MB/s. Další zrychlení nabízí Ultra DMA/66 – přenosová rychlost 66 MB/s a Ultra DMA/100 – přenosová rychlost 100 MB/s (firma Seagate používá místo Ultra DMA označení UATA).

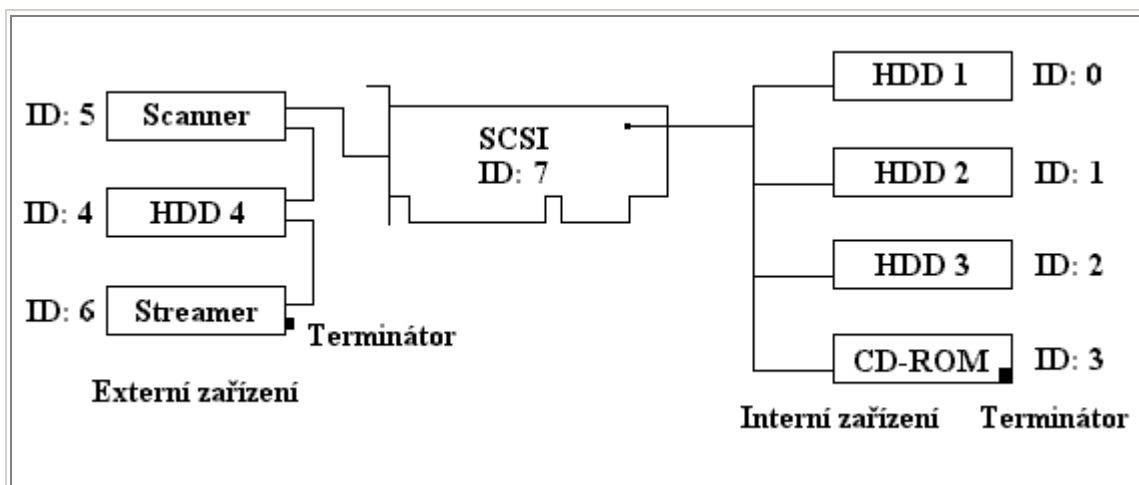
SCSI (Small Computer System Interface)

Řadič SCSI je kvalitnější řešení připojení, než předchozí možnosti. Sestává z rozšiřující desky - tzv. host adaptéru. K němu je připojen datový kabel, připojující jednotlivé periferie. Každá z těchto jednotek musí mít vlastní řadič SCSI, může být interní nebo externí a adaptér pouze řídí provoz na datovém kabelu. K výhodám řadiče SCSI patří především to, že umožňuje připojit mimo pevné disky (bez omezení kapacity) a mechaniky CDROM i skenery, tiskárny, streamery a další zařízení s uvedeným rozhraním.

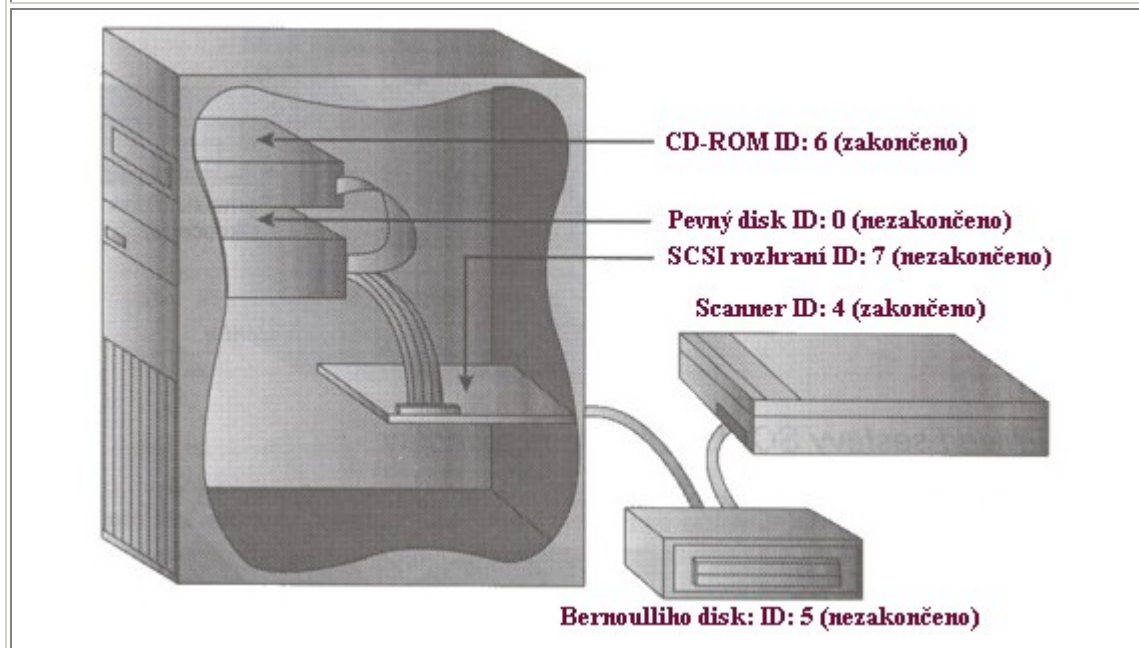
Rozhraní SCSI (Small Computer Systems Interface) bylo vyvíjeno zhruba ve stejné době jako rozhraní ESDI. Cílem SCSI bylo vytvořit standardní rozhraní poskytující sběrnici pro připojení dalších zařízení. SCSI dovoluje připojit ke své sběrnici až 8 různých zařízení, z nichž jedno musí být vlastní SCSI rozhraní. Mezi další velké výhody patří možnost připojení nejen interních zařízení, jako tomu bylo u všech předchozích rozhraní, ale i zařízení externích. SCSI není pevně vázáno na počítač řady PC, ale je možné se s ním setkat i u jiných počítačů (např.: MacIntosh, Sun, Silicon Graphics).

Jednotlivá zařízení jsou propojena pomocí 50 vodičové sběrnice a nesou jednoznačnou identifikaci v podobě **ID čísla** (v rozmezí 0-7). ID 7 bývá většinou nastaveno na SCSI rozhraní

a ID 0 bývá zařízení, ze kterého se zavádí operační systém. Sběrnice musí být na posledních zařízeních ukončena tzv. **terminátory** (zakončovací odpory), které ji impedančně přizpůsobují a zabraňují tak odrazu signálů od konce vedení. Tyto terminátory jsou buď součástí zařízení, nebo lze použít externí terminátory.



Zapojení zařízení na rozhraní SCSI



Zapojení SCSI rozhraní

V prvních verzích (SCSI-1) byla data i příkazy přenášeny po 8 bitové datové sběrnici a rychlost přenosu byla asi 2-4 MB/s. SCSI rozhraní existuje ve formě:

zásuvného modulu - karty

externě připojitelného modulu přes paralelní port

Interně montované karty mají zpravidla dva konektory:

konektor pro připojení interních zařízení

konektor pro připojení externích zařízení

K SCSI rozhraní je možné připojovat celou řadu různých zařízení, jako jsou např. pevné disky, CD-ROM mechaniky, páskové jednotky (streamery), scannery, magnetooptické disky, Bernoulliho disky atd. Externí zařízení mají dva konektory :

vstupní: směrem od řadiče

výstupní: směrem k dalšímu zařízení

Délka celé sběrnice by u SCSI-1 neměla přesáhnout 25 m.

Řadiče SCSI prošly několika vývojovými etapami. Jako rozšíření předchozího SCSI-1 vzniká rozhraní SCSI-2, které je též komerčně nazýváno jako Fast SCSI. SCSI-2 je zdola kompatibilní s původním SCSI-1, má však vyšší přenosovou rychlost (až 10 MB/s) a přísnější nároky na kabeláž (celá délka sběrnice může být maximálně 3 m). Dalším rozšířením rozhraní SCSI je rozhraní označované jako SCSI-3, které dovoluje připojit až 32 zařízení s ID v rozmezí 0-31.

Jejich základní charakteristiky jsou přehledně uvedeny v tabulce.

Standard	Označení	Poč.bitů	Přenosová rychlost	Max.počet periférií
			MB/s	
SCSI 1		8	5	7
SCSI 2	Fast	8	10	7
SCSI 2	Fast+Wide	16	20	7
SCSI 2	Fast+Wide	32	40	7
SCSI 3	Fast	8	10	7
SCSI 3	Fast+Wide	16	20	15
SCSI 3	Fast+Wide	32	40	31

Tab. 5.2 Základní vlastnosti jednotlivých verzí rozhraní SCSI

6.6.2 Disková pole

Diskové pole je soustava pevných disků, určených pro paralelní ukládání dat. Data se přitom zapisují s určitou redundancí. Tím je dosaženo vyššího zabezpečení.

Koncepce použití diskových polí je shrnuta do řady systémů označených jako RAID (Redundant Array of Inexpensive Disks) a to úrovně RAID 0 až RAID 7. Přitom se zvyšujícím se číslem se zvyšuje i úroveň jejich zabezpečení. Nevýhodou diskových polí je jejich vysoká cena.

RAID 0 je systém postupného zápisu a čtení na více disků. Inteligencí řadiče lze výrazně zkrátit vybavovací doby. Nejedná se však ještě o zálohování dat, při poruše musí být obnovena data ze záložní kopie. Metoda pouze zkracuje přístupové doby.

RAID 1 je první úroveň skutečného zabezpečení dat a to formou zrcadlení (mirroring) na více disků. Data při poruše disku lze vnitřně obnovit. Čtení dat se provádí střídavě z různých disků a tím je dosaženo i zkrácení přístupové doby.

RAID 2 (bit stripping with correction code) používá přídavné disky na uložení bitů dat a jejich paritních bitů, nebo Hammingova opravného kódu. Patřičné redundantní bity jsou na všech discích. Zapisovaná data jsou přitom rozdělena po bitech na jednotlivé disky a při čtení jsou opět skládána. Systém je velmi pomalý a je určen pro mainframy.

RAID 3 (bit stripping with correction code) používá obdobný zápis jako RAID 2. Přitom však jen jeden disk obsahuje paritní bity nebo Hammingův opravný kód. Např. na 10 datových disků 4 disky paritní. Zápis a čtení dat je synchronizován. Tento systém je velmi rychlý a je proto používán např. pro aplikace CAD/CAM.

RAID 4 (striping with non synchronised disks) je systém obdobný jako RAID 3, ale data jsou čtena po blocích. Informace je pak složena ze všech bloků.

RAID 5 (Striping with no dedicated parity bit drivers). Tento systém umožňuje zrekonstruovat ztracená data při výpadku jednoho a to libovolného disku. Rozdíl mezi systémy RAID 3 až RAID 5 je v organizaci disků, způsobu řešení konfliktů a pod.

RAID 6 je obdobou RAID 5, ale se dvěma paritními disky.

RAID 7 řídí každý disk nezávisle s vlastními datovými cestami včetně paměti cache. Přenosy do centrální paměti cache jsou nezávislé. Systém pracuje v reálném čase a má jak SCSI sběrnici, tak i interní vysokovýkonnou sběrnici (320 MB/s) a sběrnici pro řízení. Řadič má vlastní o.s. schopný zaručit nezávislost jednotlivých cest a disků. Soustava obsahuje alespoň jeden paritní disk. Umožňuje ochranu dat i při poruše více disků. Na systém lze napojit až 48 disků a 12 hostitelských počítačů. Přitom systém umí detekovat i sudý počet chyb, který byl u předešlých systémů nedetekovatelný. Systém má zdvojený napájecí zdroj a 3 úroňovou paměť cache.

Řadiče RAID

jsou připojitelné na sběrnici PCI, UltraPCI, EISA, SCSI a Microchannel. Desky s řadiči obsahují inteligentní cache o velikostech 2 x 32 nebo 64 MB. Vlastní pole je napojeno na Fast SCSI-2 nebo Ultra SCSI v jednom až třech kanálech (po 7 zařízeních). Rychlost přenosu je až 132 MB/s.

6.7 PRUŽNÝ DISK

Disketové mechaniky prodělaly dlouhý vývoj paralelně s počítači, jelikož byly vždy považovány za standardní součást počítače. Médium pro ukládání dat je pružný plastový kotouč s nanesenou tenkou vrstvou záznamového média. Původní diskety o průměru 8" se dnes využívají pouze ve speciálních zařízeních, 5,25" diskety jsou rovněž historickou záležitostí. Stále se používají jednotky 3,5" se záznamovou kapacitou zpravidla 1,44 MB. Nejnovější typy pro kolmý záznam mají formátovanou kapacitu 2HD 2,88 MB. Jeden z předpokládaných trendů vývoje je kombinace disketové jednotky a pevného disku.



Obr. 6.1 Mechanika FDD 3,5"

Velikost	Hustota	Stopy	Sektory	Kapacita sektoru	Kapacita diskety
5,25"	DD	0-39	1-9	512 B	360 kB
5,25"	HD	0-79	1-15	512 B	1,2 MB
3,5"	DD	0-79	1-9	512 B	720 kB
3,5"	HD	0-79	1-18	512 B	1,44 MB

Tab. 6.1 Parametry disket 3,5" a 5,25"

Životnost

Běžné značkové diskety mají životnost 3,5 mil. otáček, s leštěným povrchem 30 mil. a s teflonovou vrstvou 75 mil. otáček.

6.7.1 Modulace dat při záznamu na magnetická média

Data se na magnetická média ukládají pomocí změn magnetického toku. Tato změna může nastat z kladného toku na záporný nebo naopak ze záporného na kladný. Každá takováto změna

se při čtení projeví jako **impuls**. K reprezentaci dat na magnetickém médiu se tedy používá přítomnosti nebo nepřítomnosti impulsu (mezera).

Například posloupnost bitů 10101110 by byla zakódována jako PNPNPPPN, kde P značí impuls a N značí mezeru.

Bylo by tedy teoreticky možné data zaznamenávat na médium tak, že bit 1 by byl zaznamenán jako impuls a bit 0 jako mezeru. Tato představa je ovšem pouze teoretická a v praxi by nikdy nepracovala. V okamžiku, kdy by následovala delší posloupnost nul, která by byla zaznamenána jako dlouhá posloupnost mezer bez jakýchkoliv impulsů, by došlo ke ztrátě synchronizace řadiče pevného disku a nebylo by možné přesně určit, kolik mezer (nul) bylo přečteno. Impulsy totiž pomáhají vzájemně synchronizovat čtená data a řadič disku.

Z předchozího vyplývá, že data musí být na disk zaznamenávána tak, aby nikdy nedošlo k dlouhé posloupnosti mezer. V rozporu s tímto požadavkem je ovšem požadavek, který říká, že na magnetické médium se vejde větší počet mezer a impulsů, je-li počet impulsů menší. Je tedy nutné zvolit vhodný kompromis, aby při čtení dat nedošlo ke ztrátě synchronizace a zároveň, aby vlivem přehnaně velkého počtu impulsů nedocházelo k plýtvání médium a tím k jeho menší kapacitě.

V praxi je tento problém řešen některým z následujících způsobů:

FM modulace: v případě FM (Frequency Modulation) se jednotlivé bity zakódují následovně:

Bit	Zakódování
0	PN
1	PP

Například: 101101101 se zakóduje jako PPPNPPPPNPPPPPNPP.

To znamená, že jednička je kódována jako dva impulsy a nula jako impuls následovaný mezerou. Při tomto kódování je bezpečně zaručeno, že nikdy nenastane příliš dlouhá posloupnost mezer. Bohužel je zde příliš vysoký počet impulsů, který způsobuje, že na médium je možné zaznamenat menší počet informací. Tento způsob kódování je dnes již poměrně zastaralý a nepoužívá se.

MFM modulace: u MFM (Modified Frequency Modulation) je snaha zmenšit počet impulsů, takže bity jsou kódovány podle těchto pravidel.

Bit	Zakódování
0	PN jestliže je v řetězci 00 NN jestliže je v řetězci 10
1	NP

Například mějme vzorek 101100, který bude pro porovnání zakódován pomocí MFM i FM.

Vzorek	MFM	Počet impulsů	FM	Počet impulsů
101100	NPNNPNPNPN	4	PPNPPPPNPN	9

Je zřejmé, že při MFM modulaci dojde k úspoře média, protože celkový počet impulsů je menší než u FM modulace a také celkový počet po sobě následujících mezer nikdy nebude příliš vysoký, protože po sobě mohou následovat nejvýše 3 mezery. Díky těmto vlastnostem je MFM modulace asi o 20% úspornější než FM modulace.

MFM modulace se používala u prvních pevných disků a dodnes se používá při záznamu na pružné disky.

RLL modulace: modulace 2,7 RLL používá pro kódování následující schéma, pomocí něhož kóduje dvojice až čtveřice bitů:

Vzorek	RLL	Počet impulsů	MFM	Počet impulsů
00	PNNN	1	PNPN	2
01	NPNN	1	PNNP	2
100	NNPNNN	1	NPNNPN	2
101	PNNPNN	2	NPNNNP	2
1100	NNNNPNNN	1	NPNPNNPN	3
1101	NNPNNPNN	2	NPNPNNNP	3
111	NNNPNN	1	NPNPNP	3

Jednotlivé vzorky a jejich zakódování jsou voleny tak, aby mezi dvěma impulsy byly minimálně dvě a maximálně 7 mezer. Toto kódování je asi o 50% úspornější než MFM kódování a bylo používáno u starších pevných disků.

moderní pevné disky používají většinou nějakou modifikaci 2,7 RLL kódování, označovanou např. **ARLL**, **ERLL** apod., která poskytuje ještě větší úsporu a tím umožňuje záznam většího objemu dat na médium.

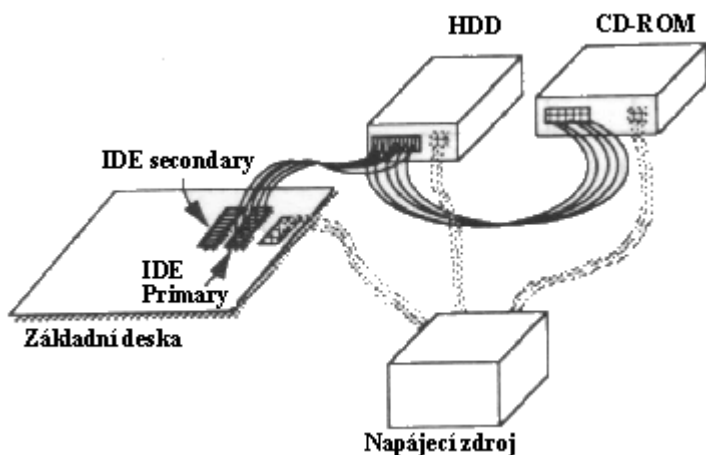
6.8 CD A DVD

6.8.1 Compact Disk - CD

Datové jednotky CD představují řadu technologií, které umožňují ukládat data na nosiče, obdobné nosičům pro ukládání hudebních nahrávek.

Mechanika CD-ROM (Read Only Memory)

Mechaniky CD ROM umožňují ukládat data na stejné kotoučky, jako u hudebních disků - výměnné optické paměťové médium s kapacitou obvykle 650 až 700 MB pouze pro čtení. Klasické CD-ROM se vyrábějí lisováním stejně jako běžné hudební CD (proto na ně nelze zapisovat). Na rozdíl od pevného disku, který má soustředné kruhové stopy rozdělené do sektorů, má CD-ROM záznamovou stopu ve tvaru spirály začínající u středu disku, která je také rozdělena na sektory. Spirálovitá stopa má stoupání 1,6 μm , šířku 0,6 μm a je tvořena různě dlouhými prohlubněmi (tzv. pity). Záznam je snímán laserovým paprskem Existuje celá řada formátů uložení, včetně kombinací s hudbou. Jednotlivé mechaniky se liší rychlostí otáčení kotouče a tím i rychlostí čtení dat. Tato rychlost bývá udávána jako násobek rychlosti běžných zvukových CD. Existovaly tak mechaniky 2x, 4x atd, v současné době jsou zcela běžné mechaniky 32x. Pomalejší jednotky než cca 8x nebyly vhodné pro přehrávání videa v reálném čase. S rozhraním IDE (EIDE, ATAPI) nebo SCSI přístupová doba okolo 100 ms, přenosová rychlost závisí na typu: 1x 150 KB/s, 2x 300 KB/s, 24x 3600 KB/s, ...



Obr. 7.1 Způsob připojení jednotky CD ROM

CD-R (Recordable)

Jedná se o prázdné disky, na které je možné zaznamenat data v odpovídajících mechanikách. Data je možné zaznamenat pouze jednou, je však možné tato data na disk postupně doplňovat (tzv. multisession).

CD-RW (Read Write, ReWriteable)

CD-RW disky jsou přepisovatelné CD disky, které lze přepisovat v mechanikách RW.

Způsob uložení dat na CD

Data jsou uložena v digitální formě jako logická "0" a "1". Na výlisku CD se nachází na spirále směrem od středu drobné dolíky - PITy o velikosti několika tisícín mm. Ze zdroje laser. záření jsou vysílány impulsy záření. Pokud se paprsek při dopadu odrazí od rovné plošky, vrátí se impuls zpět k fotodiode a jejím prostřednictvím vznikne elektrický impuls. Při dopadu laserového paprsku na PIT, je paprsek odražen jiným směrem a impuls nevznikne. Za normálního stavu při plynulém střídání plošek a PITů dochází k plynulé změně impulsů, což je chápáno jako logická "0". Jako logická "1" je chápána odchylka od této posloupnosti. Celková kapacita CD je rozdělena na sektory - velké rámce (Large Frame). Tyto velké rámce obsahují vždy 98 malých rámců (Small Frame). Počet bajtů dat, uložených v malém rámci se může v závislosti na kapacitě měnit. U hudebních CD představuje jednotlivý sektor asi jednu pětasedmdesátinu sekundy.

Datová kapacita	650 MB
Hudební kapacita	74 min
Rychlost otáčení (čtení ze středu CD)	530 x /min
Rychlost otáčení (čtení z obvodu CD)	200 x /min
Tok dat u hudebního CD	176 KB/s
Tok dat u datového CD	n x 176 KB/s (n=n rychlost. mechanika)

Tab. 7.1 Základní technické údaje jednotek CD

DVD - Digital Versatile Disc

Jedná se medium nové generace ve velikosti klasického CD. V překladu DVD značí "digitální univerzální disk". Záznamová kapacita tohoto media je podstatně vyšší (GB). Je určeno především pro záznam filmů ve formátu MPEG-2 a velkých objemů dat. Kapacita se může lišit dle toho, zda je disk oboustranný či vícevrstvý. Podobně jako u klasických mechanik CDROM existují varianty DVD-ROM, DVD-RW se stejným významem. Data jsou snímána bezkontaktně laserem o vlnové délce 635 a 650 nm.



Obr. 7.2 Disk DVD

Základní kapacita jednostranného a jednovrstvého disku je 4,7 GB. Navíc technologie DVD umožňuje v rámci jedné strany použít dvě vrstvy nad sebou. pro přístup do hlubší vrstvy je první vrstva poloprůhledná. Při čtení může laserový paprsek plynule přecházet z jedné vrstvy do druhé. Celkem jsou tedy preferované 4 druhy DVD disků, viz tabulka.

Označení	Počet vrstev	Počet stran	Kapacita [GB]
DVD -5	1	1	4,7
DVD -9	2	1	8,5
DVD -10	1	2	9,4
DVD -18	2	2	17

Tab. 7.2 Preferované druhy DVD disků

6.9 Mechaniky ZIP

Jsou určeny především pro jednoduché zálohování a přenos větších souborů mezi počítači, s kapacitou 100 MB nebo 250 MB. Jako výměnné médium se používá speciální cartridge (podobná 3,5" disketě), přičemž mechaniky ZIP 250 MB dokáží pracovat i se 100 MB cartridge. Podle provedení je lze rozdělit na

- interní ATAPI (připojení na rozhraní IDE)
- externí PP (paralelní port) nebo USB

Cartridge je poměrně drahá, proto jsou v dnešní době mechaniky ZIP nahrazovány mechanikami CD-RW.

6.10 Videokarty

Počítače řady PC používají podobně jako většina počítačů k zobrazení informace vakuovou obrazovku, která je součástí monitoru. Videokarty (grafické karty, grafické adaptéry) jsou zařízení, která zabezpečují výstup dat z počítače na obrazovku monitoru.

Videokarta má vliv na to, jaký software může uživatel na počítači provozovat a jak rychle se data na obrazovku přenášejí. Většina videokaret dovoluje práci ve dvou základních režimech:

textový režim: režim, který umožňuje zobrazovat pouze předem definované znaky, jako jsou písmena (A, a, B, b, C, c, ...), číslice (1, 2, 3, ...), speciální znaky (&, ^, %, ...) a pseudografické

znaky (symboly pro vykreslování tabulek). Tyto znaky jsou přesně definované pomocí matic bodů a je možné je zobrazovat pouze jako celek.

grafický režim: režim, ve kterém jsou informace zobrazovány po jednotlivých obrazových bodech tzv. **pixelech** (Picture Element). Tento režim již nepoužívá předem definované znaky, ale může z jednotlivých pixelů vykreslit prakticky "libovolnou" (závisí na možnostech konkrétní karty) informaci.

Základní parametry každé videokarty jsou

Parametr	Vysvětlení
Rozlišení v textovém režimu	Počet znaků, které je možné v textovém režimu zobrazit na jednom řádku, a počet řádků, které je možné umístit na obrazovku
Matice znaku	Počet bodů (ve vodorovném a ve svislém směru), ze kterých se může skládat jeden znak v textovém režimu
Rozlišení v grafickém režimu	Počet pixelů, které je možné v horizontálním a ve vertikálním směru zobrazit
Počet barev (barevná hloubka)	Počet barev, které je možné zároveň zobrazit. Udává se většinou pouze pro grafický režim.
Rychlost	Počet pixelů, které videokarta dokáže vykreslit za jednotku času. Udává se pouze v grafickém režimu.

Grafické karty, které jsou schopny zobrazit maximálně dvě barvy, jsou označovány jako **monochromatické** (černobílé).

Moderní videokarty se skládají z následujících částí:

procesor

paměť

DAC převodník

ROM BIOS

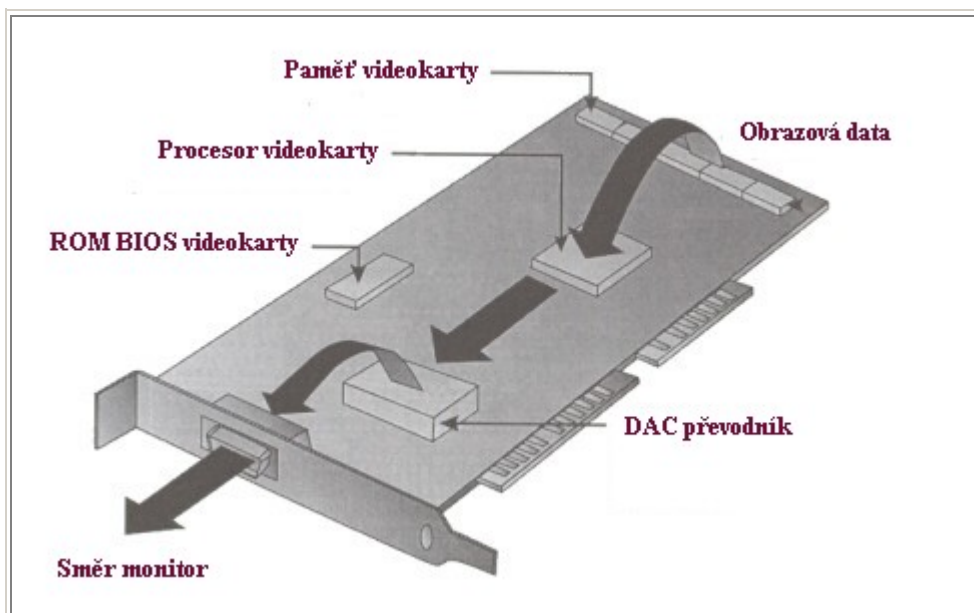


Schéma videokarty

Při práci zapisuje procesor počítače obrazová data do videopaměti. Takto zapsaná data jsou potom čtena procesorem videokarty, který na jejich základě vytváří digitální obraz. Digitální obraz je posílán na vstup DAC (Digital Analog Converter) převodníku, který z něj vytváří analogový obraz nutný pro moderní monitory, řízené spojitě (analogově) měnící se hodnotou signálů tří základních barev (Red - červená, Green - zelená, Blue - modrá)

6.10.1 Typy videokart

Grafická karta MDA

Videokarta MDA (Monochrome Display Adapter) byla první videokartou, která byla dodávána k počítačům řady PC. Byla vyrobena firmou IBM v roce 1981. Tato videokarta pracovala pouze v textovém režimu (mohla zobrazovat jen znaky zadané ASCII kódem).

V textovém režimu mohla zobrazovat 80 znaků na řádek a 25 řádků na obrazovce. Jeden znak byl definován v matici o rozměrech 9 x 14 bodů. Tyto vlastnosti poskytovaly velmi dobře čitelný text. Bohužel zobrazování grafiky u karty MDA nebylo možné. Druhou podstatnou nevýhodou tohoto adaptéru byla schopnost práce pouze v monochromatickém (černobílém) režimu.

Grafická karta CGA

Grafický adaptér CGA (Color Graphics Adapter) vzniká jako nástupce karty MDA opět ve firmě IBM. Karta CGA již dokáže pracovat v textovém i grafickém režimu a dovoluje práci v monochromatickém i barevném režimu.

Nevýhodou této karty oproti kartě MDA jsou její parametry v textovém režimu. Je možné zobrazit 80x25 nebo 40x 25 znaků v 16 nebo 2 barvách, ale matice jednoho znaku se skládá z 8 x 8 bodů, což značí horší čitelnost znaků.

V grafickém režimu dovoluje zobrazit maximálně:

640 x 200 bodů černobíle

320 x 200 bodů ve 4 barvách ze 16

Grafická karta Hercules (HGC)

Hercules (HGC - Hercules Graphics Card) vznikl z adaptéru MDA přidáním možnosti práce i v grafickém režimu. V textovém režimu má Hercules stejné parametry jako karta MDA, to znamená, že dokáže zobrazit 80 x 25 znaků a jeden znak je tvořen 9 x 14 body.

V grafickém režimu je možné dosáhnout rozlišení 720 x 348 bodů ve 2 barvách. Toto rozlišení je však nevýhodné, protože jeho poměr počet bodů v horizontálním směru / počet ve vertikálním směru je přibližně 2/1, zatímco poměr rozměrů obrazovky monitoru je 4/3.

Grafická karta Hercules podobně jako MDA pracuje v monochromatickém režimu. Později vznikla i barevná verze, která ale nezaznamenala většího rozšíření. Hercules byl používán hlavně u počítačů řady PC / XT.

Grafická karta EGA

Videokarta EGA (Enhanced Graphics Adapter) byla vyrobena v roce 1984 a jednalo se o první všestranně využitelnou kartou firmy IBM. V textovém režimu zobrazovala podobně jako předešlé karty 80 x 25 znaků a jeden znak byl vytvořen v matici 8 x 14 bodů. Tyto parametry

zaručovaly poměrně dobrou čitelnost textu. Kromě toho karta EGA pracovala i v grafickém režimu, kde umožňovala maximální rozlišení 640 x 350 bodů v 16 barvách z 64 možných.

Tato karta umožňovala ve své době poměrně solidní využití jak při práci s textem, tak i v grafických aplikacích. Pro profesionální práci s grafikou však nedostačovala.

Jako výkonnější alternativa ke kartě EGA vznikl později grafický adaptér označovaný jako **PGA** (Professional Graphics Adapter). Tento adaptér se však nikdy ve větší míře neujal.

Grafická karta VGA

Grafický adaptér VGA (Video Graphics Array) firmy IBM byl vyroben v roce 1987 původně pro řadu počítačů IBM PS/2. Jedná se o kartu, která je schopna v textovém režimu zobrazovat 80 x 25 znaků a jeden znak je definován v matici 9 x 14 bodů. Znaky v textovém režimu mohou být zobrazovány v 16 barvách.

V grafickém režimu dokáže tato videokarta zobrazit maximálně 640 x 480 bodů v 16 barvách. Tento typ videokarty vyžaduje oproti předchozím kartám nový typ monitoru, který není řízen digitálním sledem signálů, ale spojitě (analogově) mění se hodnotou signálu každé ze základních barev (Red - Červená, Green - Zelená, Blue - Modrá).

Grafická karta MCGA

Grafická karta MCGA (Multi Color Graphics Array) je verze karty VGA pro IBM PS/2 model 25 a 30, má stejné možnosti v textovém režimu jako VGA a dovoluje v grafickém režimu zobrazit:

640 x 480 bodů ve 2 barvách

320 x 200 bodů v 256 barvách

Grafická karta 8514/A

Grafická karta 8514/A označovaná také jako Very High Resolution Graphics Adapter (grafický adaptér s velmi vysokou rozlišovací schopností) byla vyrobena opět firmou IBM a dovolovala rozlišení až 1024 x 768 bodů v 256 barvách.

K této kartě vznikla později jako její nástupce videokarta s označením **XGA** (Extended Graphics Adapter), která byla rychlejší a kompatibilnější s předchozími počítači, ale neznamenala většího rozšíření.

Grafická karta SVGA

Videokarta SVGA (Super Video Graphics Array) je dnes nejpoužívanější typ grafické karty. Jejím nejdůležitějším prvkem je procesor, který do značné míry ovlivňuje její výkon. U modernějších typů videokart je tento procesor schopen realizovat (buď sám nebo za pomoci nějakého dalšího obvodu) některé často používané grafické operace. Takováto videokarta bývá nazývána také jako akcelerátor a umožňuje podstatně vyšší výkon, protože není nutné, aby každý pixel, který se má zobrazit na obrazovce, byl vypočítán procesorem počítače. Procesor počítače tak pouze vydá příkaz kartě, co má vykreslit (linku, kružnici, obdélník), a vlastní výpočet jednotlivých zobrazovaných pixelů provede k tomuto účelu specializovaný procesor videokarty. Kromě těchto jednoduchých operací je možné, aby procesor videokarty prováděl i složitější operace používané při práci s 3D grafikou (např. zakrývání neviditelných hran, stínování apod.) nebo operace spojené s přehráváním videosekvencí. Videokarty tohoto typu se pak nazývají 3D akcelerátory a multimediální akcelerátory. Procesor videokarty je propojen

pomocí sběrnice s videopamětí. Šířka této sběrnice bývá (32b, 64b, 128b). Paměť na videokartě může být následujících druhů:

DRAM (Dynamic RAM) popř. EDO DRAM nebo SDRAM: paměť, do které může v daném okamžiku buď procesor počítače zapisovat, nebo z ní může procesor karty číst. Tato paměť je levnější, ale poskytuje nižší výkon.

VRAM (Video RAM): paměť mající možnost dvou vstupů a výstupů. Tato paměť, která bývá také označována jako dvoubranová (dvouportová), dovoluje, aby v jednom okamžiku do ní procesor počítače zapisoval a zároveň procesor videokarty z ní četl. Tento druh paměti je dražší, ale poskytuje vyšší výkon.

SGRAM (Synchronous Graphic RAM): podobně jako paměť DRAM, ale navíc má podporu blokových operací, tj. má rychlejší operace, jako jsou například přesun bloku dat z jedné části paměti do druhé, naplnění části paměti stejnou hodnotou apod.

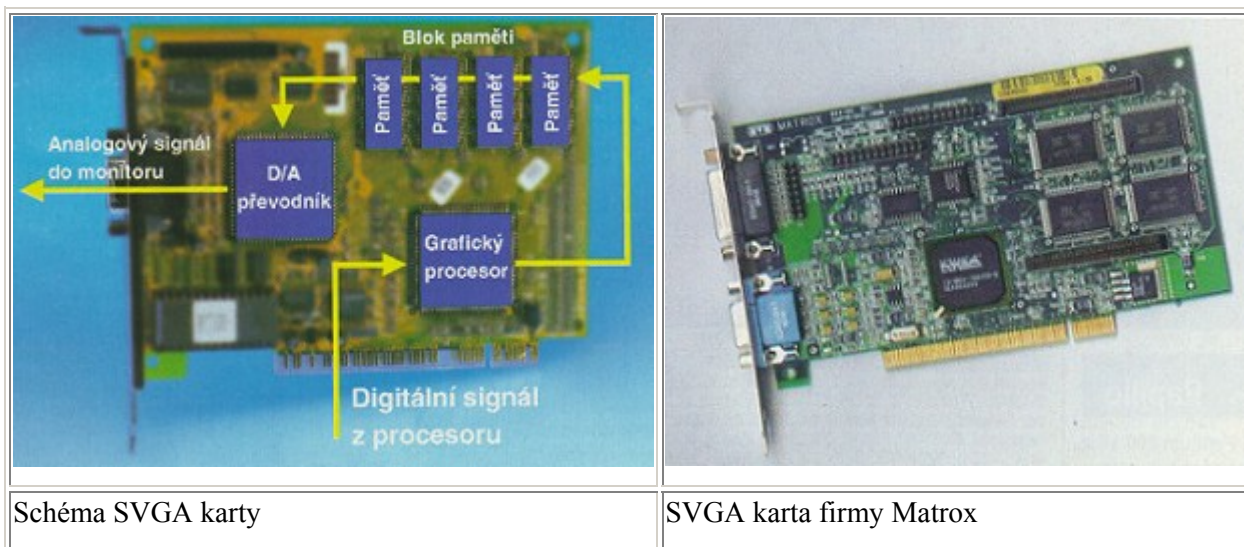
WRAM (WWindowRAM): dvoubranová paměť podobně jako VRAM s podporou blokových operací.

V závislosti na kapacitě této paměti, tzv. video paměti, a procesoru, který tato karta používá, je možné zobrazovat následující režimy:

Kapacita video paměti	Maximální rozlišení	Počet barev
256 kB	640 x 480	16
	800 x 600	16
512 kB	1024 x 768	16
	800 x 600	256
1 MB	1600 x 1200	16
	1280 x 1024	16
	1024 x 768	256
	800 x 600	65536
	640 x 480	16,7 mil.
2 MB	1600 x 1200	256
	1280 x 1024	256
	1024 x 768	65536
	800 x 600	16,7 mil.
3 MB	1600 x 1200	256
	1280 x 1024	65536
	1024 x 768	16,7 mil.
4 MB	1600 x 1200	65536
	1280 x 1024	16,7 mil.
6 MB	1600 x 1200	16,7 mil.

Super VGA vyrábí v dnešní době mnoho výrobců (Diamond, Matrox, ATI a další). Při této výrobě však došlo ke ztrátě vzájemné kompatibility v režimech s vyšším rozlišením. Posledním naprosto standardním režimem, který je na všech SVGA kartách kompatibilní, je režim VGA 640 x 480 v 16 barvách. Režimy s vyšším rozlišením již bývají nekompatibilní a vyžadují speciální programové ovladače určené pro práci s tímto typem videokarty. Z důvodu této nekompatibility byl později zaveden standard, který byl nazván **VESA** (**V**ideo **E**lectronics **S**tandard **A**ssociation). Tento standard dnes většina videokart podporuje přímo svým

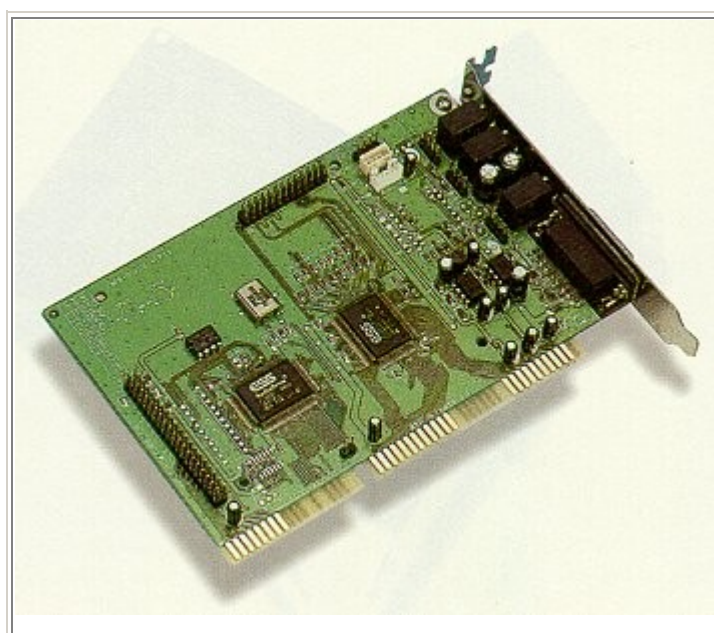
hardwarem, jiné karty jej podporují pouze softwarově pomocí programů dodaných výrobcem videokarty a některé starší karty jej nepodporují vůbec.



Vzhledem k tomu, že dnes většina počítačů pracuje pod operačním systémem provozovaným v grafickém režimu (Windows, OS/2, X Window System), jsou na výkon videokarty kladeny vysoké nároky. Proto je více než vhodné, aby do počítačů s výkonnými procesory byly osazovány výkonné akcelerátory určené pro sběrnici PCI popř. AGP (Accelerated Graphic Port - speciální typ sběrnice vyvinutý firmou Intel určený pouze pro videokarty). V opačném případě je totiž možné, že by videokarta značně degradovala výkon celého počítače.

Zvuková karta

Počítač řady PC je ve své standardní konfiguraci vybaven malým reproduktorem označovaným jako PC speaker. Tento reproduktor je součástí skříně počítače a je připojen přímo na základní desku počítače. Jeho zvukové schopnosti jsou však velmi omezené a slouží většinou pouze k vydávání jednoduchých zvuků, jako jsou např. varovné pípnutí při vzniku chyby apod., popř. k přehrání nějakých jednoduchých zvukových záznamů pro vysloveně amatérské účely. Pokud je požadován kvalitnější zvukový výstup z počítače, je nezbytné tento počítač vybavit zvukovou kartou.



Zvuková karta

Zvuková karta (sound card) je zařízení, které slouží k počítačovému zpracování zvuku. V závislosti na své kvalitě (a tím i ceně) zajišťuje kvalitní zvukový výstup z počítače vhodný i pro profesionální účely. Ke zvukové kartě lze dále připojit následující zařízení:

sluchátka

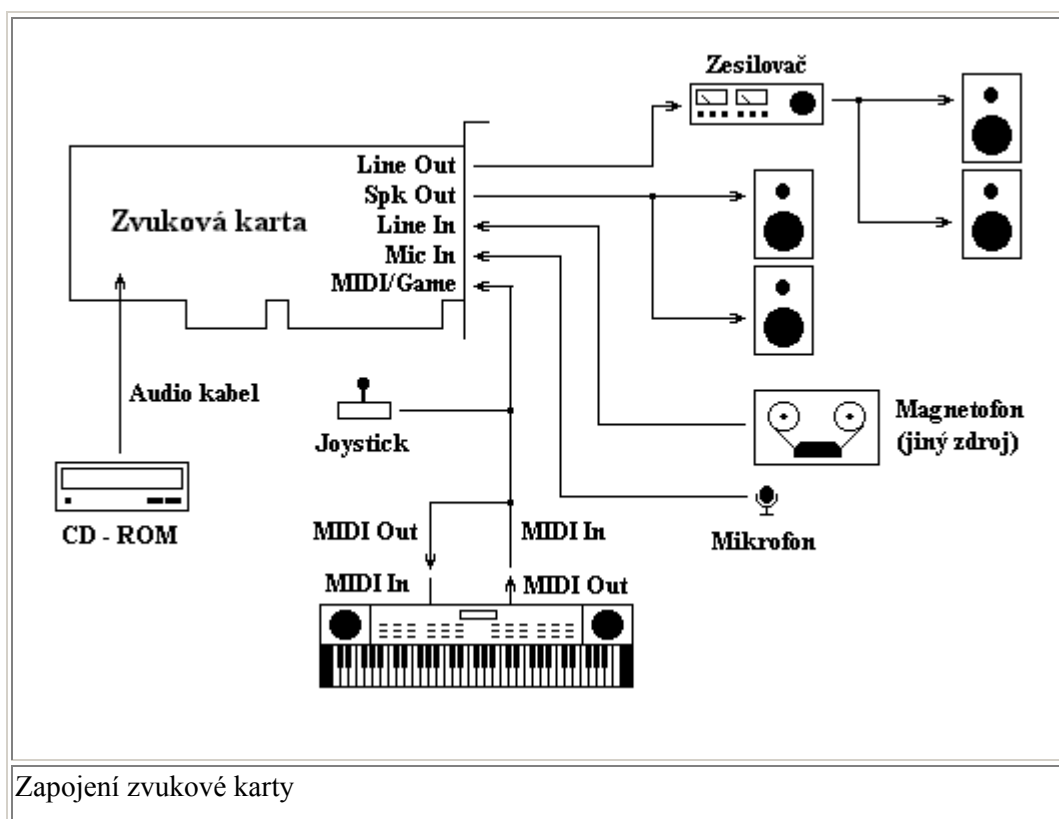
reproduktory

zesilovač

mikrofon

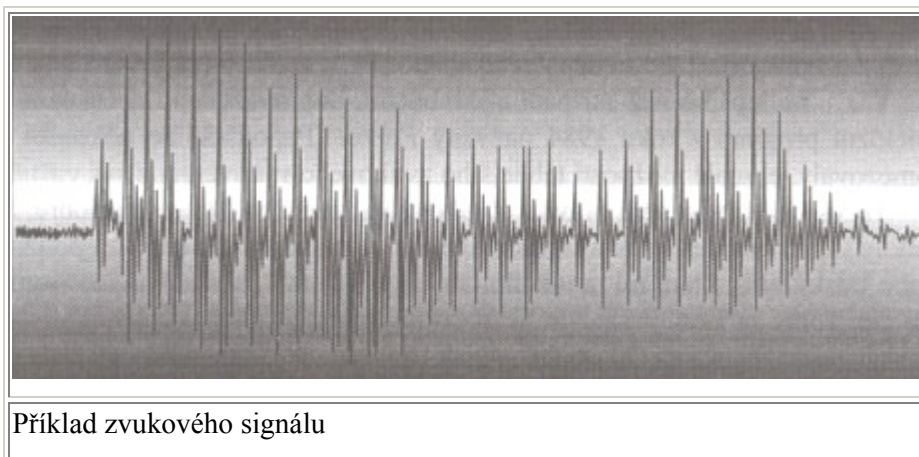
externí zdroje (rádio, magnetofon, ...)

je-li karta vybavena rozhraním **MIDI** (**M**usical **I**nstrument **D**igital **I**nterface), je možné k ní připojit i elektronické hudební nástroje vybavené také tímto rozhraním (např. elektronické varhany, syntetizátory apod.)



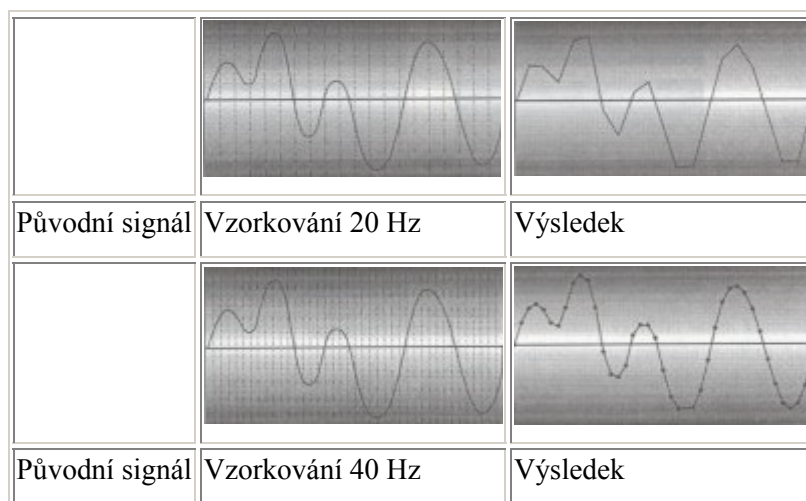
Při záznamu zvuku pomocí zvukové karty je nezbytné rozlišit dva základní případy:

záznam je prováděn z nějakého zdroje poskytujícího analogový signál (mikrofon, rádio, magnetofon, audio CD). Takovýto signál se skládá z vln (kmitů) o nestejném tlaku, který je vytvářen ve vzduchu hlasivkami, hudebními nástroji nebo přírodními silami.

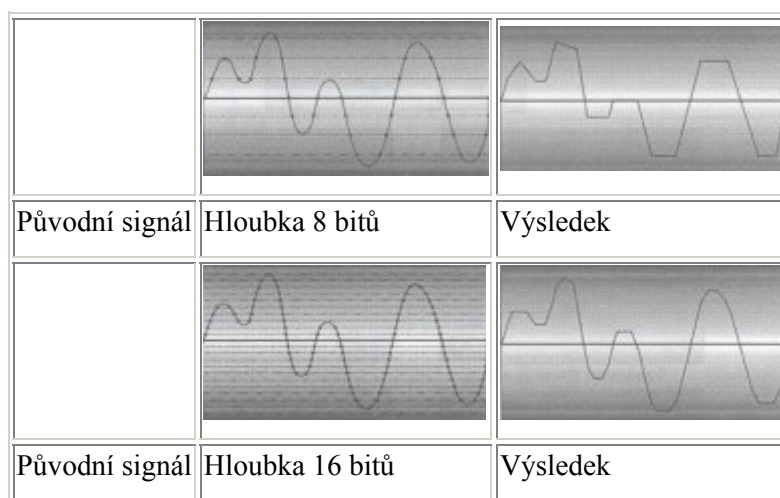


V takovémto případě je nutné tento analogový signál převést na signál digitální. Převod se uskutečňuje pomocí **vzorkování (sampling)**. To znamená, že v každém časovém intervalu je zjištěn a zaznamenán aktuální stav signálu (vzorek). Je zřejmé, že čím kratší je tento interval, tím vyšší je **vzorkovací frekvence**, tím více vzorků bude pořízeno a tím bude výsledný záznam kvalitnější. Kvalitu je možné dále ovlivnit počtem rozlišitelných úrovní v každém vzorku.

Ovlivnění kvality záznamu vzorkovací frekvencí.



Ovlivnění kvality záznamu počtem rozlišitelných úrovní na každý vzorek



Při takovémto záznamu se běžně rozlišují následující úrovně kvality záznamu:

Kvalita	Vzorkovací	Počet bitů	Počet	Délka dig.
----------------	-------------------	-------------------	--------------	-------------------

	frekvence	na vzorek	vzorků	záznamu (B/s)
Telephone Quality	11025 Hz	8	1 - Mono	11 kB/s
Radio Quality	22050 Hz	8	1 - Mono	22 kB/s
CD Quality	44100 Hz	16	2 - Stereo	172 kB/s

Při záznamu tímto způsobem se využívá **Shannonovy vzorkovací věty**, která říká: Signál spojité v čase je plně určen posloupností vzorků odebíraných ve stejných intervalech, je-li jejich frekvence větší než dvojnásobek nejvyšší frekvence v signálu. Uvážíme-li, že lidské ucho vnímá zvuky od frekvencí 16 Hz - 20 Hz až do frekvencí 16 kHz - 20 kHz, je zřejmé, že frekvence 44 kHz použitá pro CD kvalitu je dostačující. Z výše uvedené věty také vyplývá, že pokud dojde ke snížení vzorkovací frekvence, budou ve výsledném záznamu chybět vyšší frekvence, což se při přehrání projeví jako ztráta výšek.

Protože záznam tímto způsobem vede při vyšší kvalitě záznamu ke vzniku velmi dlouhých souborů, existují algoritmy dovolující provést ztrátové komprese, které podstatným způsobem kvalitu výsledného záznamu neovlivní. Tyto komprese je nutné provádět buď po provedení záznamu, nebo kvalitnější karty vybavené speciálními procesory umožňují jejich provedení v reálném čase přímo při záznamu.

záznam je prováděn z nějakého zdroje poskytujícího již digitální signál (např. elektronické varhany připojené prostřednictvím MIDI rozhraní). V takovém případě se již neprovádí vzorkování, ale zaznamenávají se přímo jednotlivé byty zasílané tímto rozhraním. Tyto byty obsahují informace, jako jsou:

nástroj, který tón hraje (piano, housle, varhany, ...)

výška tónu

délka tónu

dynamika úhozu na klávesu

další

Pro přehrání takového záznamu je nezbytné, aby zvuková karta (nebo jiné zařízení) byla schopna podle těchto informací sama vytvářet jednotlivé tóny.

Při vytváření zvuků pomocí zvukových karet je nutné vyjít ze skutečnosti, že u každého hudebního nástroje mají jeho zvuky podobu cyklu, který se skládá ze čtyř částí:

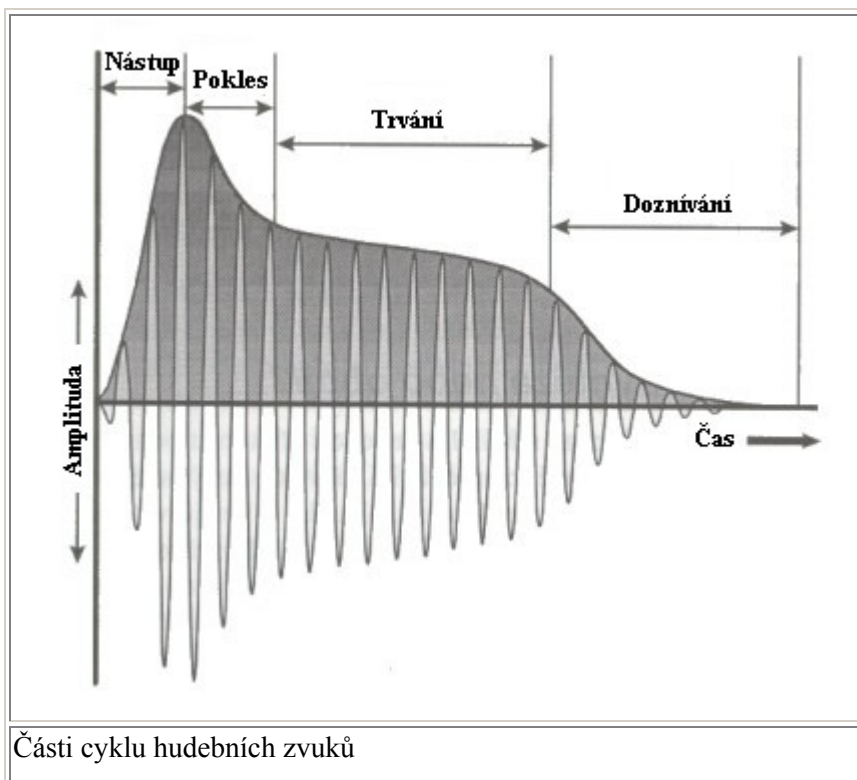
Nástup (Attack)

Pokles (Decay)

Trvání (Sustain)

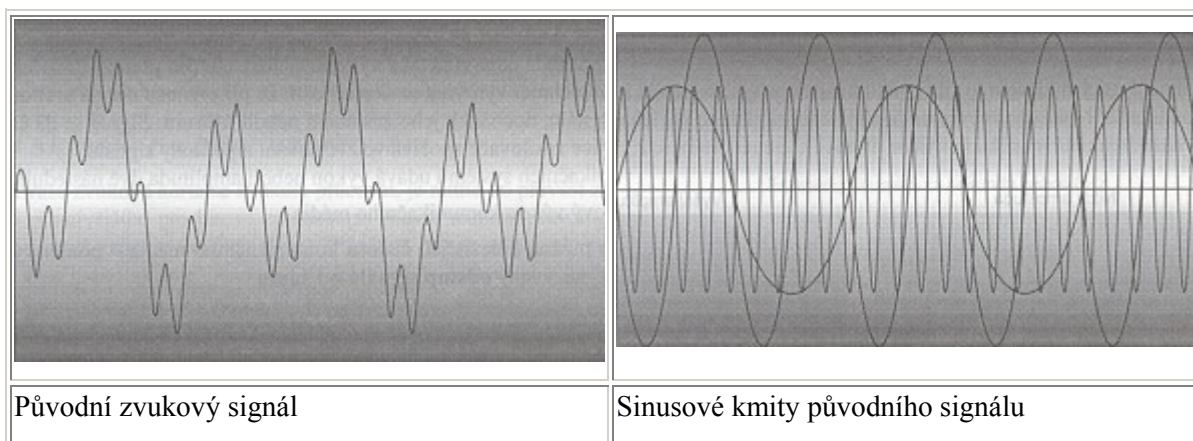
Doznívání (Release)

Konkrétní hodnoty jednotlivých fází cyklu jsou charakteristické pro každý hudební nástroj a je potřeba, aby zvuková karta tyto hodnoty co možná nejpřesněji dodržovala. V opačném případě by zvuky ztrácely na věrnosti.



Pro vlastní vytvoření zvuku se využívá dvou rozličných mechanismů:

FM syntéza: realizovaná tzv. FM syntetizátorem (obvod OPL 2, OPL 3 nebo OPL 4). Tato metoda vychází z faktu, že každé vlnění lze sestavit složením vybrané série sinusových kmitů o patřičné frekvenci a amplitudě.



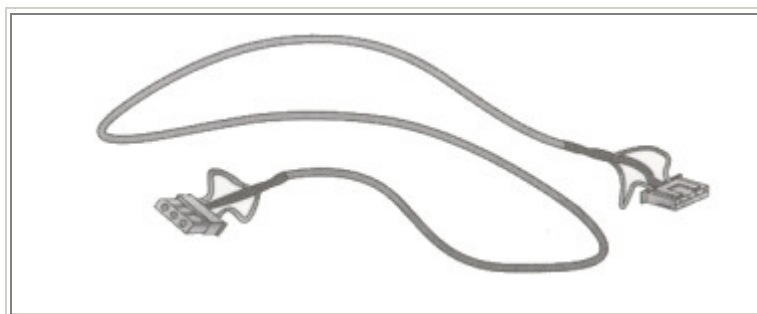
FM syntéza tedy vychází z popisu příslušného hudebního nástroje na základě Fourierova rozvoje, s jehož pomocí se potom zvuk těchto nástrojů emuluje jako superpozice několika sinusových signálů. Takto získaný signál se může ještě dále upravit různými efekty. Jedná se o levnější realizaci, která se svými výsledky zvukům reálných nástrojů pouze blíží a nikdy jich nemůže dosáhnout. Zvukové karty, které používají pouze tento způsob pro vytváření zvuků, jsou vhodné jen pro amatérské použití (ozvučení her apod.).

Wave Table syntéza: používaná u dražších zvukových karet. Tato metoda používá přímo navzorkovaný signál skutečného nástroje uložený ve své vlastní paměti (ROM nebo RAM). Protože je nemožné, aby v paměti byly uchovány vzorky všech výšek tónů od všech nástrojů, je v paměti vždy uložen jeden tón od každého nástroje. Různých výšek tohoto tónu se pak dosahuje různou rychlostí přehrání tohoto vzorku.



Reproduktory pro připojení ke zvukovým kartám

Zvukové karty bývaly obzvláště dříve vybavovány ještě IDE rozhraním, které sloužilo k připojení CD-ROM disku nebo speciálním rozhraním pro první CD-ROM mechaniky. Dnes toto řešení nemá velké opodstatnění, protože počítače jsou standardně vybaveny EIDE rozhraním, které dovoluje pohodlnější a rychlejší zapojení mechaniky CD-ROM. V případě, že v počítači je osazena CD-ROM mechanika a zároveň i zvuková karta, je velmi vhodné, aby obě tato zařízení byla propojena pomocí tzv. audio kabelu. Díky tomuto propojení je pak možné na CD-ROM přehrávat zvukové CD a poslouchat je z reproduktorů připojených ke zvukové kartě.



Audio kabel

Kromě uvedených vlastností mohou být ještě zvukové karty vybaveny pozicemi pro paměťové moduly RAM, do kterých si uživatel může ukládat vlastní vzorky různých nástrojů vytvořené buď elektronickým syntetizátorem nebo vzniklé nějakou úpravou již existujících vzorků. Dále je možné na zvukových kartách vidět i různé specializované obvody pro vytváření různých efektů v reálném čase (např. prostorový zvuk apod.)

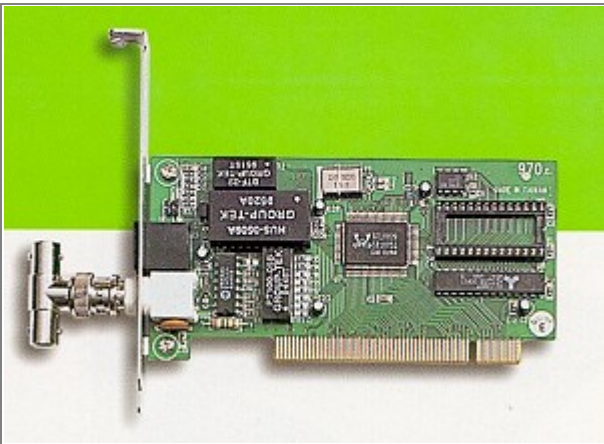
6.11 Síťová karta

Síťová karta je zařízení, které umožňuje připojení počítače do počítačové sítě.

Mezi základní parametry každé síťové karty patří:

Parametr	Vysvětlení	Rozsah
Typ sítě	Typ sítě, pro který je daná karta určena	Ethernet, Fast Ethernet, Arcnet, Tokenring
Rychlost	Množství dat, které je karta do sítě schopna vyslat (ze sítě přijmout) za jednotku času	100 kb/s – 1Gb/s
Typ	Typ síťového média (kabelu), které je možné k síťové	Tenký koaxiální kabel,

média	kartě připojit.	silný koaxiální kabel, kroucená dvojlinka
-------	-----------------	--



Síťová karta

Každé síťové médium se k síťové kartě připojuje pomocí specifického konektoru, který karta musí obsahovat. Síťová média:

tenký koaxiální kabel: určený zejména pro vnitřní rozvody uvnitř budovy. Pro jeho připojení se používá konektoru BNC. V dnešní době bývá častěji nahrazován kroucenou dvojlinkou.

silný koaxiální kabel: používaný dříve k venkovním rozvodům, k jeho připojení se používá konektoru Canon, který zde bývá označován jako AUI. Tento AUI konektor může sloužit také k připojení tzv. transcieveru, pomocí něhož je potom možné připojit jiný typ média (transciever AUI - BNC, transciever AUI - RJ45). Dnes je silný koaxiální kabel používán jen zřídka, protože je nahrazován kvalitnějším optickým kabelem.

kroucená dvojlinka: používaná pro vnitřní rozvody. Kroucená dvojlinka se připojuje pomocí konektoru RJ-45)

Při realizaci sítě v rámci budovy se dnes poměrně často používá tzv. strukturovaná kabeláž, u které se pro horizontální rozvody (v rámci patra) používá kroucená dvojlinka a pro vertikální rozvody je použito optického vlákna.

Některé síťové karty jsou vybaveny patičkou pro obvod zvaný Boot ROM. Boot ROM je paměť typu EPROM (EEPROM), která obsahuje programové vybavení nezbytné pro zavádění operačního systému z počítačové sítě místo jeho zavádění z lokálního disku.

6.12 Karty pro příjem rádia a televize

Jsou to karty sloužící k příjmu televizního a rádiového signálu na počítači.

Radiokarty

Radiopřijímač je možné získat jako samostatnou rozšiřující kartu, nebo bývá součástí některých typů zvukových karet. Umožňují kvalitní příjem rozhlasových stanic v pásmu FM.

Televizní karty

Většina těchto karet pouze zobrazuje televizní obraz na monitoru. Zaznamenávat v reálném čase jej buď nedovedou, nebo jej zaznamenávají v poměrně nízké kvalitě, která pro profesionální účely nedostačuje.

Karty pro zpracování videa: zařízení určená ke zpracování videosekvencí v reálném čase.

Karty pro příjem teletextu: karty umožňující na počítači přijímat teletext.



Obrázek 7 Televizní karta

6.13 Monitor

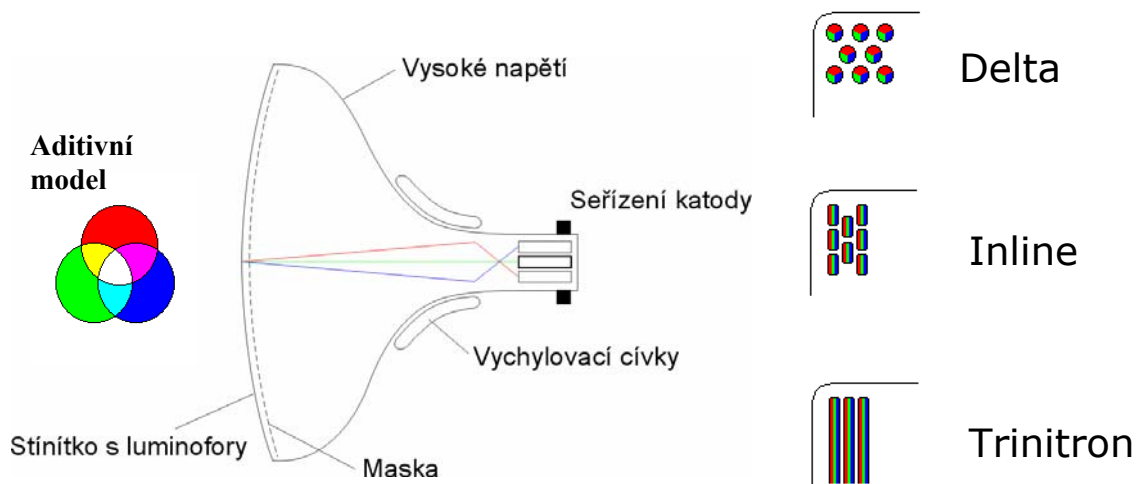
Nejdůležitější parametry:

- úhlopříčka – 14", 15", 17", 19", 21"
- max. rozlišení – 640 × 480, 800 × 600, 1024 × 768, 1152 × 864, 1280 × 1024, 1600 × 1200
- max. obnovovací frekvence (refresh frequency) – 60 až 160 Hz (optimální 85 až 100 Hz), udává počet snímků zobrazených za jednu sekundu. Obnovovací frekvence není závislá jen na monitoru, ale i na grafické kartě a jejím ovladači. Pokud grafická karta zvládá při zvoleném rozlišení např. 75 Hz a monitor 100 Hz, bude monitor pracovat s obnovovací frekvencí 75 Hz. Pokud však nastavíme na videokartě obnovovací frekvenci např. 100 Hz a monitor bude zvládat pouze 85 Hz, zobrazí se na monitoru několik přes sebe překrytých obrazů. U novějších monitorů zůstane černá obrazovka s nápisem např. *Frequency out of range* (Frekvence je mimo rozsah).
- splňované normy – MPR II, TCO92, TCO95, TCO99
- typ obrazovky
- možnosti – OSD resp. OSM (On Screen Display resp. On Screen Menu), vestavěné reproduktory, USB, ...

Monitory jsou základní výstupní zařízení počítače. Slouží k zobrazování textových i grafických informací. Klasické monitory mají obrazovku, pracující na principu katodové trubice (CRT - Cathode Ray Tube). Na jejím stínítku se zobrazují jednotlivé body obrazu – pixely tak, jak jsou generovány připojenou videokartou. Při práci barevné obrazovky jsou ze tří katod emitovány elektronové svazky, které jsou i pomocí jednotlivých mřížek (viz obrázek řez barevnou obrazovkou) modulovány až na stínítko obrazovky. Na zadní stěně stínítka obrazovky jsou nanášeny vrstvy tzv. luminoforů (luminofor = látka přeměňující kinetickou energii na energii světelnou). Tyto luminofory jsou ve třech základních barvách - Red (červená), Green (zelená), Blue (modrá) - pro aditivní model skládání barev. Vlastní elektronové svazky jsou bezbarvé, ale po dopadu na příslušné luminofory dojde k rozsvícení bodu odpovídající barvy.

Protože elektronový svazek je vlastně svazek částic stejného náboje (záporného), mají tyto částice tendenci se odpuzovat a vlivem toho dochází k rozostřování svazku. Proto těsně před stínítkem obrazovky se nachází maska obrazovky. Je to v podstatě mříž, která má za úkol propustit jen úzký svazek elektronů. Maska obrazovky musí být vyrobena z materiálu, který co

nejméně podléhá tepelné roztažnosti a působení magnetického pole. Oba dva tyto jevy by totiž způsobily, že elektronové svazky nedopadnou přesně na svůj luminofor, což by se projevilo nečistotou barev. Elektronové svazky jsou vychylovány pomocí vychylovacích cívek tak, aby postupně opisovaly zleva doprava a shora dolů jednotlivé řádky obrazovky.



Obrázek 8 Řez barevnou obrazovkou

Displeje s kapalnými krystaly (LCD)

Ploché zobrazovací displeje

Obrazovkové monitory mají sice kvalitní obraz, ale pro svou váhu, velikost a spotřebu jsou nevhodné pro přenosné počítače. Proto se u těchto počítačů, ale nejenom u nich, využívá téměř výhradně displejů na bázi kapalných krystalů (LCD-Liquid Crystal Display). Základem těchto displejů je tzv. TN buňka (Twisted Nematic) pracující na principu technologie nematických kapalných krystalů. Kapalnými krystaly rozumíme látku v kapalném stavu s krystalickou strukturou, která otáčí polarizační rovinu procházející světla o 90 stupňů. Pokud vložíme takovýto materiál mezi dvě elektrody, pak při zvyšování elektrického pole velikost polarizačního úhlu klesá k nule. Existují i tzv. matické kapalně krystaly, které bez působení elektrického pole polarizační rovinu nestáčíjí a při zvětšování elektrického pole ji otáčejí od 0 do 90 stupňů.

Běžné denní světlo je nepolarizované - kmitá ve všech směrech. Pokud takovéto světlo projde polarizačním filtrem, dojde k jeho polarizaci - kmitá jen v jedné rovině. Pokud takovéto světlo narazí na další filtr, pak jím buď projde - filtr má stejnou rovinu polarizace jako světlo, nebo neprojde - filtr má otočenou rovinu polarizace otočenou o 90 stupňů proti světlu. Pokud je mezi těmito filtry umístěn materiál otáčející rovinu polarizace světla o 90 stupňů, je tomu s průchodem světla naopak.

Takto je tomu i u skutečných displejů. Mezi dvojicí polarizačních filtrů je vložen materiál na bázi nematických kapalných krystalů, jejichž výroba je levnější na rozdíl od kapalných krystalů matických. Na stěnách polarizačních filtrů jsou napařeny průhledné elektrody ve tvaru rovnoběžek. Na jednom filtru jsou napařeny vodorovně - určují linky displeje a na druhém svisle - určují sloupce displeje. Přivedeme-li napětí na jednu elektrodu linky a jednu sloupce, vytvoří se v místě křížení elektrické pole a na displeji se zobrazí bod. Pokud postupně přivádíme konstantní napětí na linky a v době aktivování určité linky přivedeme napětí opačné polarity modulované úrovní jasu do elektrod určujících body v lince, můžeme takto vykreslit celý

obraz na displeji. Při tomto zobrazení je nutné zajistit určitou setrvačnost pootočení polarizační roviny kapalného krystalu (obdobně jako u monitoru dosvit jednotlivého bodu obrazovky). Protože takto zkonstruovaná zobrazovací jednotka dosahuje malého kontrastu a malého úhlu v němž je obraz zřetelný, používá se jí u levných kalkulaček a digitálních hodin. Další nevýhodou této jednotky je, že se velmi špatně hledá materiál s přesně stanovenou dobou dosvitu, takže obraz buď bliká nebo při změně obrazu vznikají duchové. Proto byla technologie TNLCD postupně zdokonalována.

První zdokonalením byla technologie STNLCD (Sup Twisted Nematic Liquid Crystal Display). Zdokonalení spočívá ve složitějším uspořádání elektroa polarizačních filtrů. Pro definici jednoho bodu se používá větší množství buněk. Jeho nevýhodou je zvýšená absorpce světla, proto musíme zajistit spodní prosvícení displeje. Z důvodu minimální spotřeby se využívají elektroluminiscenční folie nebo fluorescenceční zářivky se studenou katodou. Tím, že je použit zdroj světla, dojde ke zbarvení světla. Omezí se tím použitelnost technologie při výrobě barevných obrazovek.

Dalším zdokonalením jsou technologie DSTNLCD (Double STNLCD) a technologie TSTNLCD (Triple STNLCD). Pro zobrazení jednoho bodu používají dvojici nebo trojici buněk STN. Nevýhodou tohoto displeje je jemnější struktura elektrod. Výhodou je zvýšení počtu úrovní šedé. Pomocí technologie TSTNLCD byla vyrobena i barevná zobrazovací jednotka. Každá buňka STN v trojici, která tvoří zobrazovaný bod, odpovídá jedné základní barvě (červená, modrá, zelená). Ke každé buňce vedou tedy tři sloupcové elektrody určující barvu bodu. Pokud napájíme určitou buňku modulovaným napětím, které odpovídá velikosti jedné barevné složky pro daný bod, pak buňka propustí bílé světlo ze zdroje, které prošlo přes barevný filtr. Tento filtr odpovídá svou barvou barvě zobrazované barevné složky pro daný bod. Výroba těchto filtrů je náročná (např. pro zobrazení 640 x 480 bodů na zobrazovací ploše s úhlopříčkou 10" je velikost bodu 0,3mm a velikost barevného filtru je tedy 0,1mm). Protože tyto filtry zvyšují ztrátu jasu při průchodu světla zobrazovací jednotkou, musí mít barevné LCD zobrazovací jednotky mnohem výkonnější zdroj světla než jednotky monochromatické.

Všechny doposud uváděné technologie nedokáží dosáhnout kontrastního a přitom jasného obrazu, při rychlé změně obrazu se objevují duchové atd. Těmto displejům se říká pasivní LCD displeje. Ty aktivují zobrazovací buňku po krátkou dobu a zbytek času využívají setrvačnosti zobrazovací buňky. Tyto nepříjemné vlastnosti pasivních LCD displejů odstraňují tzv. aktivní LCD displeje. U těchto displejů se využívá kapalných krystalů s nízkou setrvačností. Je tedy nutné zabezpečit uchování elektrického pole pro každý zobrazovaný bod do té doby než bude bod znovu aktivován. K tomuto byla vyvinuta technologie TFT (Thin Film Transistor), která obdobně jako u dynamických pamětí využívá tranzistor ve funkci kondenzátoru. Technologie je díky výrobě dynamických pamětí sice poměrně dobře zvládnuta, ale zatím ne na tak velké ploše odpovídající celému displeji. Protože je tranzistorů zapotřebí velká spousta (pro VGA jednotku s plochou 20 x 15cm je potřeba 921600 tranzistorů - tedy té měř tolik jako na jeden MB DRAM) a všechny musí být plně funkční jsou tranzistory zdvojeny. Tím se snížila nebezpečí vyrobení jednotky s vadnými body. Nevýhodou při výrobě ale je, že jen jeden tranzistor z dvojice musí být funkční. Proto se při výrobě musí zničit vysokým napětím vždy jeden tranzistor z plně funkční dvojice.

Z tohoto důvodu se hledaly cesty ke zlevnění této technologie. Jednou z těchto cest je nahrazení tranzistoru diodou. Tato technologie dostala název technologie MIM (Metal Isolator Metal). Nevýhodou této technologie ale je, že poskytuje méně kvalitní obraz než technologie TFT.

Další z technologií, která se začíná v dnešní době používat, je technologie FLC (Ferroelectric LCD). Principem této technologie je, že materiál po připojení a odpojení elektrického napětí zůstane ve stavu, do něhož byl elektrickým polem přiveden. Do stavu původního se materiál dostane po přivedení napětí opačné polarizace. Materiál tedy rozeznává jen dva stavy a je tedy vhodný pouze pro monochromatické displeje. Další nevýhodou těchto displejů je náročnost výroby, protože materiál je používán v podstatně menších vrstvách.

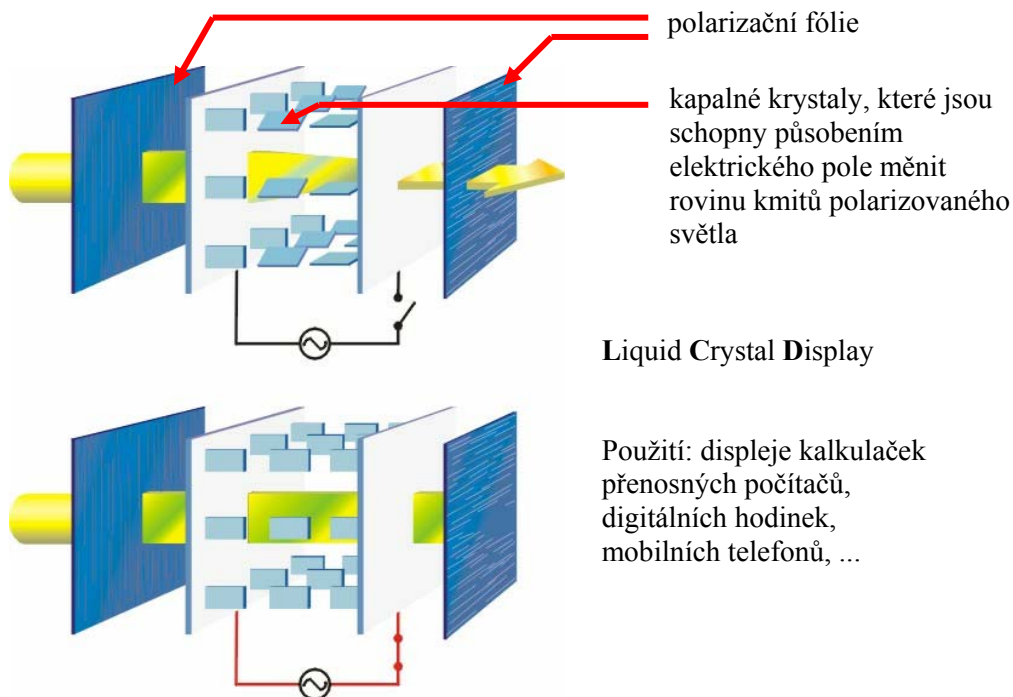
Vzdálenost skleněných desek od sebe je jen 1 až 2um narozdíl od 5 až 7um u TFT technologie. Velikým kladem této technologie je to, že odstraňuje složité a nespolehlivé pole prvků TFT nebo MIM a dosahuje přitom velmi dobrého jasu a kontrastu. Displeje se vyrábí o úhlopříčce 15" a výhledově i 20" a 30" s rozlišením 1280 x 1024 bodů.

Jednou z dalších technologií používaných při výrobě displejů je tzv. technologie plazmově adresovatelného displeje (PALCD - Plasma Addressed LCD). Je obdobou TFT technologie, ale narozdíl od ní není elektrické pole udržováno pomocí tranzistoru, ale pomocí inertního plynu. Ten se působením elektrického pole ionizuje a udržuje elektrické pole působící na LCD do doby, než je ionizace plynu zrušena. Je zřejmé, že technologie je jednodušší než TFT, nevýhodou ale je zbarvení procházejícího světla do oranžova důsledkem ionizovaného plynu.

Výše uvedené technologie pro přenosné počítače jsou na bázi kapalných krystalů. Existují však technologie, které kapalné krystaly nevyužívají. První z nich je technologie plazmového displeje (nezaměňovat s PALCD). U ní je každý bod naplněn inertním plynem, který se po přivedení napětí na příslušné elektrody ionizuje až dojde k monochromatickému výboji. Ten se udržuje i po odepnutí napětí. Je zrušen až po přivedení napětí opačné polaroty. Obraz tedy neblíká, je dostatečně jasný a kontrastní. Rychlá odezva buněk umožňuje dosáhnout snímkovou frekvenci blízkou klasickým monitorům (to ostatní technologie neumožňují). Rozšíření této technologie brání skutečnost, že obraz je monochromatický a žlutý a výroba je finančně náročná. Vývojově jsou už vyráběny i displeje, které zářením jednotlivých buněk (zářením inertního plynu) aktivují různobarevné luminofory.

Další dvě technologie využívají luminoforů a jsou založeny na podobném principu jako televizní obrazovka. První z nich je technologie ACTFEL. U ní jsou luminofory umístěné mezi dielektrickými vrstvami napájenými řádkovými a sloupcovými elektrodami. Elektrony aktivující luminofor jsou uváděny do pohybu změnami napětí na elektrodách obdobně jako u kondenzátoru. Tím se luminofor rozžárí. Výroba barevných displejů je omezena malým jasnem luminoforu modré barvy, proto se zatím vyrábí touto technologií pouze displeje v monochromatickém provedení.

Druhou technologií je technologie FED, která se svým principem blíží klasické obrazovce. Stejně jako u ní je luminofor aktivován elektronovým dělem. Elektronové dělo má každý bod obrazovky, čímž je dosaženo nepatrné hloubky zobrazovací jednotky. Každé elektronové dělo je vybaveno paměťovým prvkem. Dělo tedy působí na luminofor po celou dobu zobrazování obrazu. Tím je dosaženo toho, že obraz neblíká jako u klasického monitoru. Technologie tím umožňuje dosáhnout kvalitnějšího obrazu než má klasický monitor menšího množství vyzařování elektromagnetického a rentgenového záření. Rozšíření technologie však brání cena, která je díky statisícům elektronových děl vysoká.



6.14 VSTUPNÍ ZAŘÍZENÍ

Mezi vstupní zařízení počítače je možné zahrnout klávesnice, polohovací zařízení (myš, trackball, touchpad, tablet, světelné pero, joystick, atd...), dotykové obrazovky, atd. Nejvýznamnějšími však stále zůstávají klávesnice a myši.

Klávesnice

Klávesnice (keyboard) je základní vstupní zařízení počítačů PC pro komunikaci s obsluhou.

Existují tyto typy klávesnic:

XT klávesnice obsahuje mikroprocesor zabudovaný přímo v klávesnici. Klávesnice má 83 kláves. Klávesy této klávesnice lze rozdělit do 3 základních skupin:

- Abecední pole: obsahuje klávesy abecedy, číslice, speciální znaky (!,@,#,...) a některé speciální klávesy (SHIFT, CTRL, ALT, ENTER, ...)
- Funkční klávesy: klávesy označené F1 až F10, jejichž význam závisí na konkrétním programu, se kterým uživatel pracuje
- Kurzorové a numerické klávesy: obsahují klávesy pro číslice a ovládání kurzoru

Klávesnice XT je nekompatibilní s klávesnicí AT (není ji možné použít u počítače AT).

AT klávesnice. Tento typ je určen pro řady PC - PC/AT. Obsahuje 101 (US standard) nebo 102 (European standard) kláves. Tyto klávesy lze rozdělit do 4 bloků:

- Abecední pole je obdobné jako u XT klávesnice
- Funkční klávesy: obsahuje klávesy F1 - F12, jejichž význam je opět podobný jako u XT klávesnice.
- Kurzorové klávesy: klávesy pro ovládání kurzoru

- Kurzorové a numerické klávesy: podobné jako u XT klávesnice

Klávesnice AT předpokládá procesor pro klávesnici na základní desce počítače. Většina AT klávesnic je vybavena přepínačem, který ji dovozuje přepnout do režimu XT a používat ji tak u počítače XT.

Win95 Natural. Klávesnice vznikla s příchodem operačního systému MS Windows 95. Původní AT klávesnice byla doplněna o speciální klávesy pro ovládání tohoto systému (klávesa pro vyvolání Start menu a pro vyvolání kontextového menu). Speciální ergonomické klávesnice se speciálními tvary, jejichž cílem je zajistit co možná nejpřirozenější polohu rukou při práci.



Obr. 8.1 Ergonomická klávesnice

Dle principu snímání stisku je možné rozdělit klávesnice :

- s mikrospínači - pro každou klávesu je použit mikrospínač
- kapacitní: stisk klávesy je převeden na změnu kapacity snímače. Ta je vyhodnocena procesorem 8048 umístěným v klávesnici, a příslušný kód je pak vyslán do počítače.

Způsob připojení klávesnice:

Klávesnice bývá k počítači připojena většinou 5 kolíkovým konektorem DIN, popř. pomocí PS/2 konektoru nebo konektoru USB. Existují však i klávesnice se speciálními konektory (servery).

Polohovací zařízení - Myši

Myši lze rozdělit podle způsobu generování signálu na:

- mechanické, kde je odvalování kovové pogumované kuličky přenášeno na kolmé kladky s kotouči s nanosenou vodivou vrstvou, která spojovala kontakty spínačů (dnes už neexistuje),
- mechanicko-optické, kde je pohyb zmíněných kladek snímán opticky pomocí kotoučků s otvory. Tato myš dnes ustupuje do pozadí ve prospěch myši optických,
- optické, kde je pohyb myši snímán odrazem infračerveného paprsku od plochy, po které se myš pohybuje. Kulička tedy zcela chybí.

Z hlediska připojení se vyskytují obdobně jako u klávesnic tři typy konektorů:

- sériový (COM) - zásuvka s 9 nebo 25 piny pro připojení k sériovému rozhraní, dnes se už téměř nepoužívá, standard u XT/AT
- PS/2 - 6pinová zástrčka stejného tvaru jako u klávesnice, ale s odlišnými hodnotami napájecího napětí, standard u ATX

- USB - konektor umožňuje připojení za chodu bez rizika poškození I/O obvodů PC, rychlejší než PS/2, často jsou myši s USB konektorem dodávány také s redukcí na PS/2 z důvodů zpětné kompatibility.

6.15 Připojování vstup/výstupních zařízení

Počítač je vybaven celou řadou konektorů, které lze v prvním kroku rozdělit podle mechanického uspořádání na:

- zásuvky
- zástrčky (konektor s kolíky, které se zasunují do zásuvky)

Z hlediska funkce je zde zpravidla možné nalézt některé z následujících konektorů:

- napájecí konektory
- zásuvka grafické karty
- zásuvka DIN pro připojení klávesnice (u starších PC AT)
- zásuvka PS/2 pro připojení klávesnice nebo polohovacího zařízení (myši) u modernějších PC
- zástrčka 9 nebo 25 pinů pro sériový port (starší konektor pro polohovací zařízení, modemy, ...)
- zásuvka s 25 piny pro paralelní porty (pro tiskárny, skenery, záznamová zařízení, ...)
- zásuvka gameportu (pro připojení joysticku, kláves, ...)
- zvukové konektory (jacky pro reproduktory, mikrofon, ...)
- zástrčky USB
- zásuvka SCSI (25, 50 nebo 68 pinů)
- zásuvka FireWire pro rychlá zařízení
- konektor pro připojení síťové karty nebo telefonní konektor, zvuková karta a řadu dalších...

Sériové rozhraní

Data jsou posílána po jednom vodiči za sebou tak, že teprve až dojde vyslaný bit je zaslán další. Rychlost komunikace je závislá na použitém médiu a technologii, je obecně maximálně jednoduchá.

Paralelní rozhraní

Pro přenos se využívá více vodičů současně - naráz je tak možné přenést nejméně celý byte. Jistou nevýhodou je možnost tzv. přeslechů, tj. ovlivňování napěťových signálů mezi jednotlivými vodiči. Toto rozhraní si vynutila především potřeba zrychlení komunikace počítače s tiskárnami.

V počítači zajišťuje komunikaci IO karta (Input/Output) obsahující porty pro připojení periferních zařízení, častěji je součástí základní desky. Port je místo spojení procesorové jednotky s komunikačním kanálem a slouží k připojení dalších periferních zařízení. Standardní I/O karta obsahuje:

1 paralelní port: bývá označován jako LPT1 a slouží např. pro připojení tiskárny, ZIP disku, propojení dvou počítačů.

2 sériové porty: bývají označovány jako COM1, COM2 a slouží pro připojení počítačové myši, tiskárny, propojení dvou počítačů

1 game port: slouží k připojení křížového ovladače pro hry (joystick)

6.15.1 Rozhraní Centronics (EPSON)

Paralelní rozhraní určené převážně pro výstup informace. dnes bývá přímo integrováno na základní desce, dříve bylo na rozšiřující kartě společně s řadičem pevných disků a disketových mechanik, označení LPT či PRN, dříve byl přenos dat pouze jednosměrný (z počítače do tiskárny). Norma IEEE 1284 zajišťuje obousměrnost a sdružuje následující režimy přenosu dat (možno nastavit v setupu): Compatible Mode (Centronics), Nibble Mode, Byte Mode, EPP (Enhanced Parallel Port) Mode, ECP (Extended Capabilities Port) Mode. Nejpoužívanější je režim EPP.

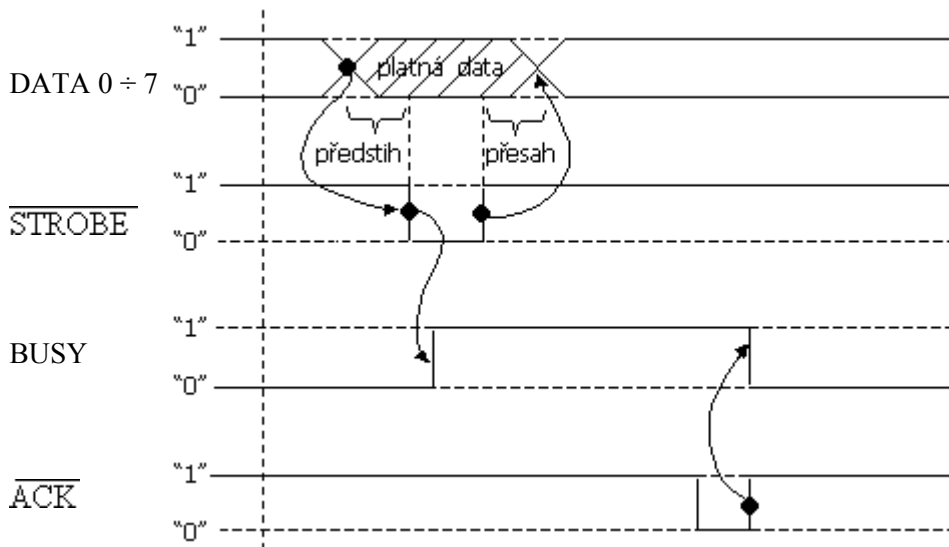
1. Mechanická úroveň: Korektor Cannon 25-kolíkový na počítači je zásuvka

2. Elektronická úroveň: "0" ... 0V až 0.4 V } úrovně TTL
 "1" ... 2.4V až 5V

Zapojení:

Špička	Signál	Zdroj	Význam
1	$\overline{\text{STROBE}}$	Poč.	platnost dat
2	DATA 0	Poč.	DATA
.	.	.	DATA
.	.	.	DATA
.	.	.	DATA
9	DATA 7	Poč.	DATA
10	$\overline{\text{ACK}}$	Tisk.	konec tisku znaků
11	BUSY	Tisk.	tiskárna obsazena
12	PE	Tisk.	paper empty
13	SLCT	Tisk.	připravenost tisku
14	AUTO	poč.	automat. LF po CR
15	$\overline{\text{ERROR}}$	Tisk.	chyba tiskárny
16	$\overline{\text{INIT}}$	Poč.	inicializace tiskárny
17	SLCT IN	Poč.	žádost o přípravu
18-35	GND	-	zem

3. Logická úroveň



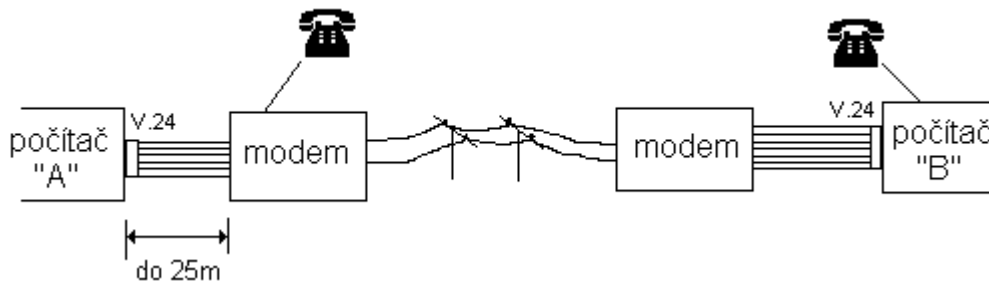
6.15.2 Rozhraní V.24 I. (RS-232C)

Dnes bývá přímo integrováno na základní desce, dříve bylo na rozšiřující kartě společně s řadičem pevných disků a disketových mechanik. Používá se konektor Canon 9 nebo 25 pinů samec, označení COM nebo RS 232. Používá setaké k připojení modemu atd.

V současné době se myš (a klávesnice) připojuje prostřednictvím konektoru PS/2 nebo USB

Připojení:

a)



b)



Obrázek 9 Propojení počítačů pomocí V.24

V.24 je rozhraní přispůsobené pro telefonní linky

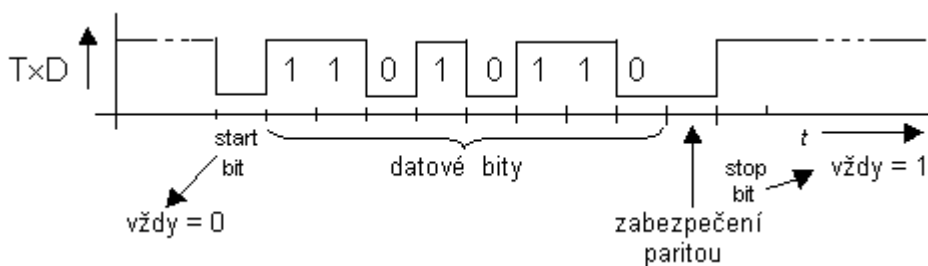
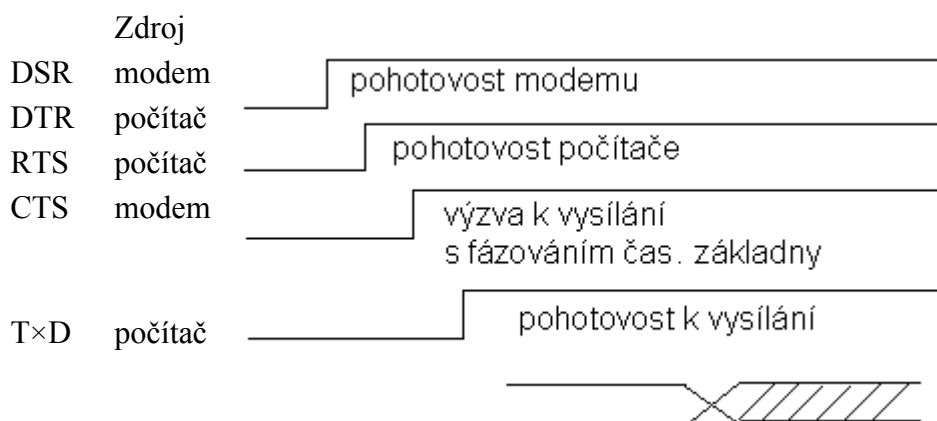
1. Mechanická úroveň: Konektor cannon 25-kolíkový
na počítači je zástrčka
2. Elektronická úroveň: "1" ... -15V ÷ -3V

"0" ... 3V ÷ 15V

Zapojení:

špička	Číslo obvodu	Označení	Zdroj	Význam
1	101	PG		ochr. zem
2	103	T×D	počítač	vysílaná data
3	104	R×D	modem	přijímaná data
4	105	RTS	počítač	výzva k vysílání
5	106	CTR	modem	pohotovost k vysílání
6	107	DSR	modem	pohotovost modemu
7	102	SG		signálová zem
8	109	DCD	modem	detekce nosné
20	108	DTR	počítač	pohotovost počítače
22	125	RI	modem	zvonek
.

3. Logická úroveň



Obrázek 10 Formát přenosu dat

Parametry přenosu dat:

rychlost: v bitech za sekundu (Baud Rate: Bd)

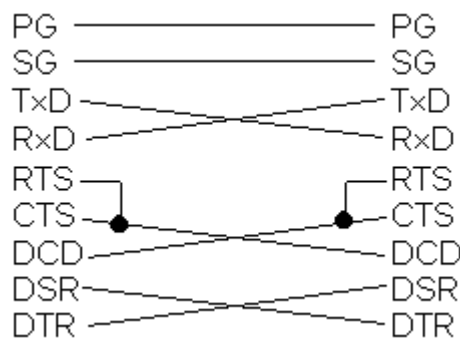
110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200

počet datových bitů: 5, 6, 7, 8

zabezpečení: sudá parita (Even)
lichá parita (Odd)

délka stop bitu: 1, 1.5, 2

Nulmodem: propojení dvou počítačů bez modemu, propojení vodičů kabelu je na následujícím obrázku:



Obrázek 11 Zapojení nulmodemu

6.15.3 USB (Universal Serial Bus)

USB - jedná se o perspektivní rozhraní, sloužící pro připojování externích zařízení a periférií. Jeho vznik si vynutil zrychlující se vývoj ve světě výpočetní techniky a stoupající nabídka výrobců elektronických zařízení, která lze připojovat k PC. Vedle myši, tiskárny, scanneru, modemu je dnes také možno připojit digitální fotoaparát, kameru, databanky, tablety, externí jednotky ZIP a samozřejmě i PC do sítě s ostatními počítači. Sběrnici společně navrhly společnosti Intel, Compaq, IBM, Microsoft, NEC, Digital Equipment a Northern Telecom v roce 1995.

Vlastnosti:

- Všechna typická zařízení se stejně připojují na společnou sběrnici
- hvězdicová topologie sběrnice
- maximální délka kabelu mezi dvěma zařízeními 5 metrů
- Náhrada připojení klávesnice, myši, RS232 zařízení, Centronics
- Snadnost použití , podpora Plug and Play

- distribuce napájení pro zařízení jako myš či klávesnice
- Možnost připojování/odpojování bez vypnutí

USB 1.x (1996)

v.1.0 1994-1996: Compaq, Intel, Microsoft a NEC, v.1.1 1998
Rychlost 1,5 Mb/s nebo 12 Mb/s

USB 2.0 (1999)

Rychlost 480 Mb/s (teoretická rychlost)

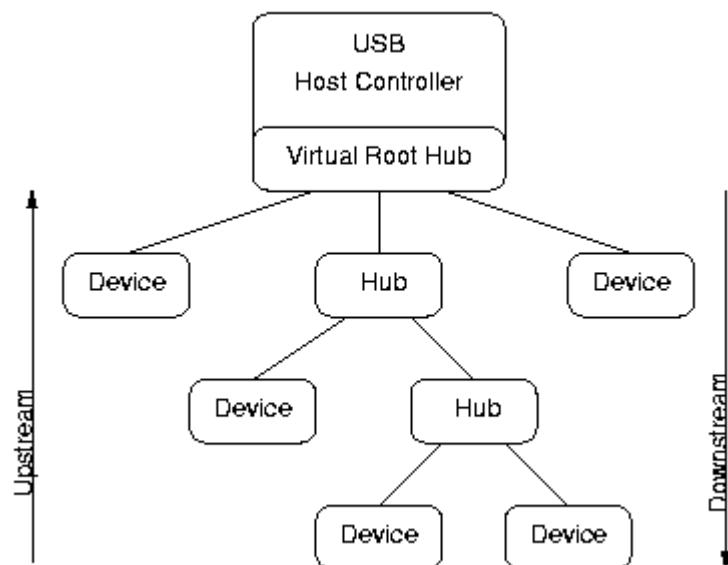
Master/Slave protokol

Komunikace je řízena/vyvolávána počítačem (host), max. 127 zařízení.

Pro přenos dat mezi aplikací a zařízením vytváří řadič virtuální komunikační kanál - rouru. V jednom časovém okamžiku může existovat roura více, aby všechna zařízení na sběrnici mohla pracovat zároveň. Podle počtu komunikačních portů daného zařízení je možné vytvořit i více rour s jediným zařízením. Roury se od sebe odlišují datovou propustností, směrem toku dat, velikostí vyrovnávací paměti a typem přenosového protokolu, kterým může být:

- **Kontrolní přenos** - slouží k instalaci zařízení při jeho připojení či dodatečné konfiguraci
- **Sekvenční přenos** - hromadné poslání velkého množství dat najednou, například při tisku dokumentu, snímání obrázku nebo kopírování souboru
- **Přenos přerušení** - přenáší se pouze pár bajtů, které jsou odpovědí zařízení na nějakou událost (např. "dokument vytištěn") nebo náhodně vyvolané uživatelem. Typický přerušovaný přenos je psaní na klávesnici či rejdní myši.
- **Isochronní přenos** - neboli pravidelný přenos, přenos v reálném čase. Používá se zejména k přenosu zvuku, kdy je třeba nevelké množství dat neustále zasílat v pravidelných intervalech, abychom si mohli kupříkladu poslechnout písničku z rádia.

Sběrnice se stará o stálou kvalitu signálu a nezávislost rour mezi sebou, kontroluje data pomocí kontrolního součtu CRC, poškozená data je někdy schopna automaticky opravit.



Obrázek 12 Zapojení sítě zařízení USB

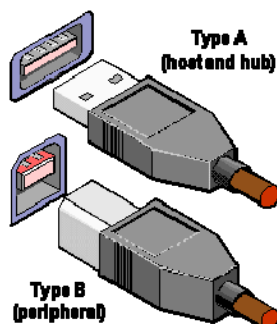
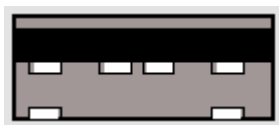
Ochrana proti zkratu a přepětí

Dovoluje se připojení/odpojení zařízení bez vypnutí počítače. Nutnost ochrany proti elektrostatickému výboji (člověk až 15 kV na koberci).

Zapojení:

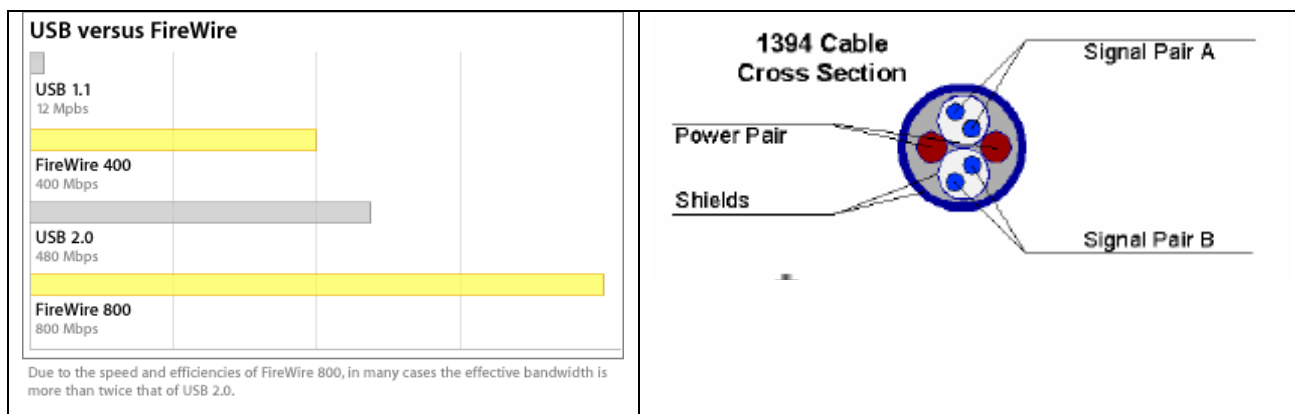
Čtyřdrátová sběrnice:

- +5V
- data -
- data +
- zem



6.15.4 IEEE 1394 (FireWire)

druh vysokorychlostní sériové sběrnice, založené na technologii Apple (také třeba “Sony Link” vyvinuté pro připojení kamer), od roku 1995 standard (IEEE 1394). Umožňuje isochronní datové služby v reálném čase (64 adresových bitů), s rychlostmi mezi 100 až 800 MB/s (vyvíjí se 3.2 Gb/s). Je to “konkurent” USB 2.0, podporuje technologii plug and play. V případě použití optického kabelu je FireWire 800 schopný přenášet data na vzdálenost až 100 metrů. Mezi výhody se počítá i možnost napájení připojeného zařízení výkonem až 45 W.



Kontrolní otázky

16. *Podle jakých kritérií je možné rozdělit počítače?*
17. *Jaká je základní struktura a činnost procesoru?*
18. *Co charakterizuje jednotlivé typy řady procesorů Intel?*
19. *Co obsahuje základní deska PC?*
20. *Jaké jsou funkce, rozdělení a charakteristiky jednotlivých typů sběrnice?*
21. *Jaké jsou funkce, rozdělení vlastností jednotlivých typů vnitřních pamětí?*
22. *Jaké jsou funkce, rozdělení vlastností jednotlivých typů vnějších pamětí?*
23. *Jaké jsou funkce a charakteristiky dalších typických karet v PC?*
24. *Jak pracují počítačové monitory?*
25. *Jak se typicky připojují vstup/výstupní zařízení?*

Úkoly k textu

14. Popište co nejpodrobněji technické parametry svého, nebo dostupného PC – základní desku, procesor, paměť, disky, ...
15. Vyhledejte a vyberte na internetu, nebo v dostupné literatuře aktuálně nejnovější nabídku výkonného PC a popište technické parametry jednotlivých hardwarových komponent – např. procesoru, pamětí, sběrnic, ...

7 Závěr

V tomto textu jsou popsány základní partie z oblasti organizace a architektury počítačů. Text je zpracován tak, aby uvedl čitatele do základů problematiky jednotlivých témat, ale nemůže v daném rozsahu textu detailně rozebrat celou šířku se všemi souvislostmi a prakticky využitelnými dovednostmi. Také vývoj nových technologií v oblasti architektury, hardwaru a softwaru je velmi rychlý. Předpokládám proto, že po nastudování tohoto textu sáhne studující po dalších aktuálních materiálech, ve kterých si doplní potřebné detaily a zdokonalí své teoretické znalosti.

8 Seznam literatury

[Murdocca-Heuring07] MURDOCCA, M., HEURING, V. *Computer Architecture and Organization: An Integrated Approach*. John Wiley & Sons, Inc., 2007. ISBN 978-0-471-73388-1.

[Scott-Mueler03] SCOTT, MUELLER *Osobní počítač: Hardware, upgrade, opravy*. Brno, Computer Press, 2003., ISBN: 80-7226-796-5

9 Seznam obrázků

Obrázek 1 Fyzikální základ kombinačních obvodů	16
Obrázek 2 Ilustrace zapojení obvodů pro realizaci operací NAND a NOR	16
Obrázek 3 Tabulka ASCII - 1	41
Obrázek 4 Tabulka ASCII – 2	42
Obrázek 5 Algoritmus násobení v jednoduché násobičce	58
Obrázek 6 Televizní karta	116
Obrázek 7 Řez barevnou obrazovkou.....	117
Obrázek 8 Propojení počítačů pomocí V.24	124
Obrázek 9 Formát přenosu dat	126
Obrázek 10 Zapojení nulmodemu	126
Obrázek 11 Zapojení sítě zařízení USB	128