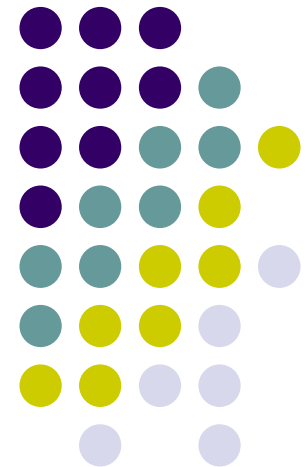


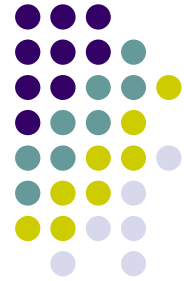
PPI

02.11.2011



Prepojovací podsystém počítača

Zbernice (prepojený každý podsystém počítača s každým)



Rozdelenie zberníc

1. Podľa spôsobu riadenia

Single master iba 1 nadriadený podsystém-master

Multi-master – každé zariadenie môže riadiť zbernicu, ale v danom okamžiku iba jedno

2. Podľa synchronizácie prenosu

Synchrónne zbernice synchronizované synchronizačným impulzom

Asynchrónne zbernice-prenos synchronizovaný odpoveďou podriadeného, pomalšie čaká sa na potvrdenie prenosu, vhodné na spojenie zariadení s rôznou prenosovou rýchlosťou

3. Podľa časového multiplexu

Multiplexované zbernice-druh informácie sa mení s časom (adresa, inštrukcia, údaj)

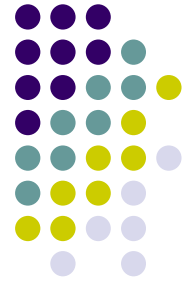
Nemultiplexované zbernice – význam a druh informácie sa s časom nemení

4. Podľa tvaru prenášaných údajov

Paralelné zbernice

Sériové zbernice (prenos bit po bite)

Rozdelenie zberníc



Dnešné počítačové zbernice sú paralelné, asynchrónne, nemultiplexované a skladajú sa z nasledujúcich sekcií :

adresová (adresa pamäte, v/v zariadenia),
údajová (inštrukcie, údaje),
riadiaca (povely, žiadosti)

Na zbernici sú definované signálové sledy-časové priebehy signálov, ktoré sa musia dodržať



Processor

interpretuje inštrukcie programu

- Výber inštrukcie z pamäte
- Vykonanie operácie s operandami
- Realizuje sa prenos informácií medzi časťami počítača

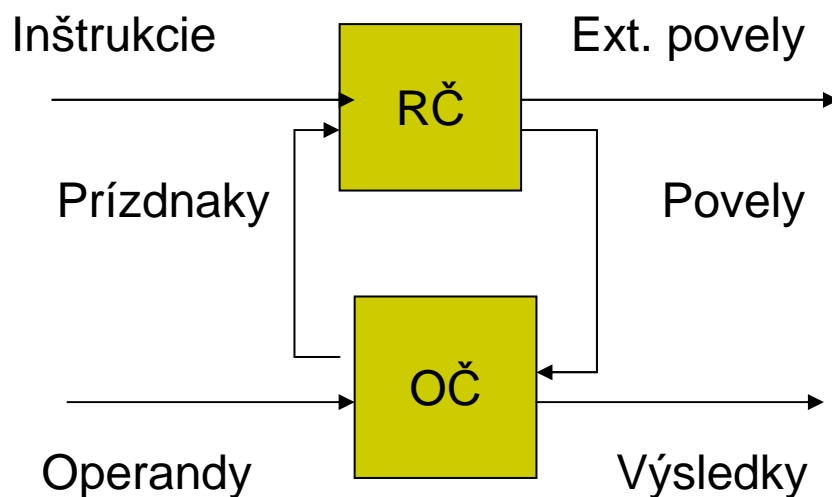
Univerzálne procesory (bohatý, úplný inštrukčný súbor)

Problémovo-orientované procesory špecializované funkcie napr. numerické koprocessory, grafický procesor, v/v procesory

Hlavné časti procesora



- Riadiaca časť
Výber inštrukcii, dekodovanie, zabezpečenie ich vykonania
- Operačná časť
Operácie s operandami



Operačná časť procesora



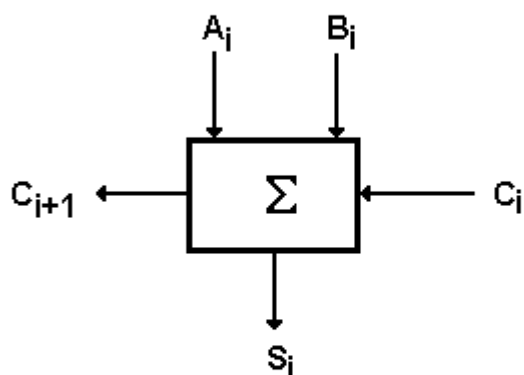
- Aritmeticko logická jednotka (ALU) určená na vykonanie inštrukcií
- Registre-prechodné uloženie operandov vstupujúcich do operácií a na uloženie výsledkov
- Komunikačné obvody na vykonávanie medziregistrových prenosov

ALJ-ALU

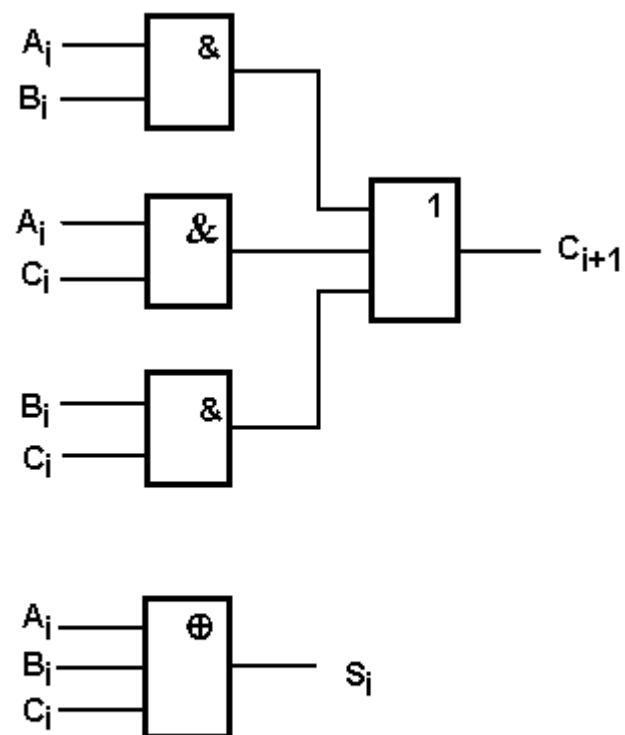
- Paralelná dvojková sčítačka dvoch n-bitových operandov sa používa na realizáciu základných aritmetických operácií, na výstupe n-bitový operand a prenos
- Sériové sčítačky, dvojková sčítačka a preklápací obvod
Sčítačku možno realizovať i pomocou pevnej pamäte ROM
- Funkčné jednotky pre logické operácie
- Posúvacie obvody, logický posun, aritmetický posun, kruhový posun



i-ty rád paralelnej dvojkovej sčítačky



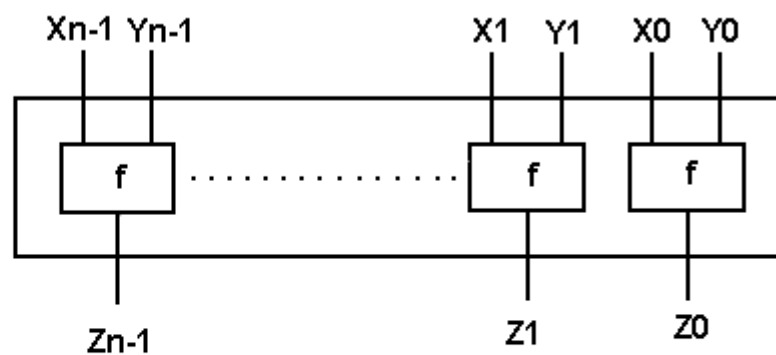
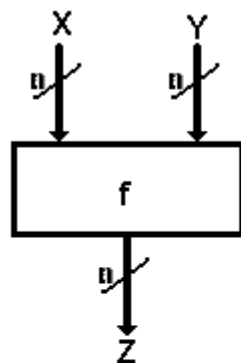
Bloková schéma



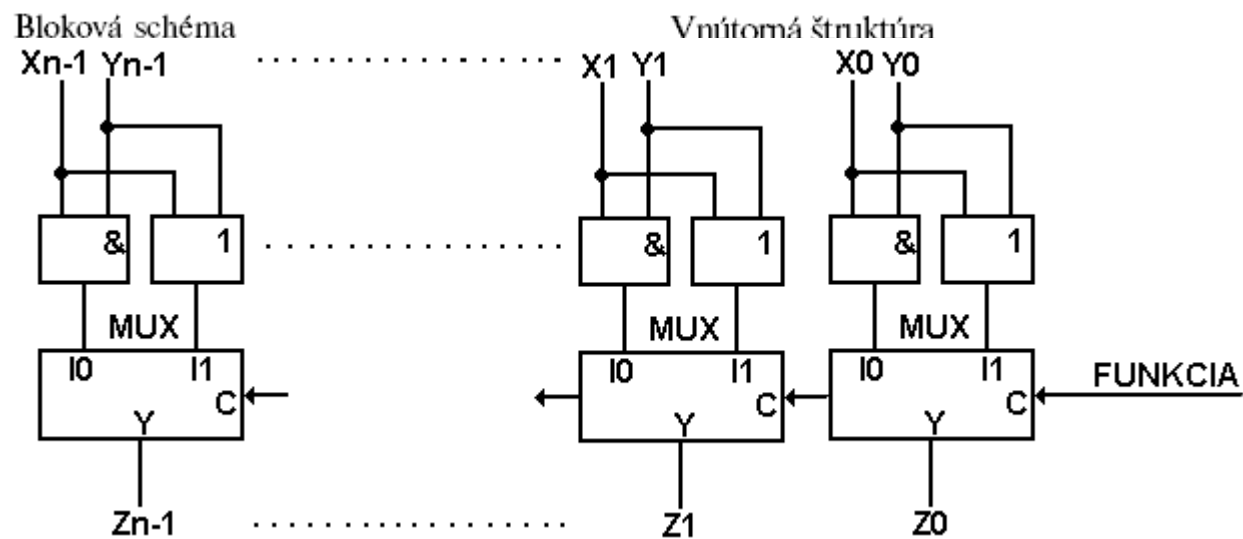
Vnútna štruktúra

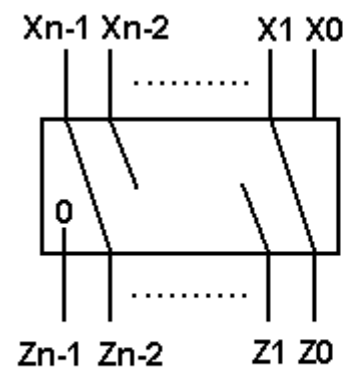
$$S_i = (A_i \oplus B_i) \oplus C_i$$

Logické operácie

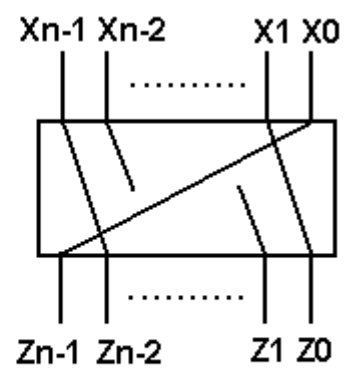


$f \in \{\text{OR, NOR, AND, ...}\}$

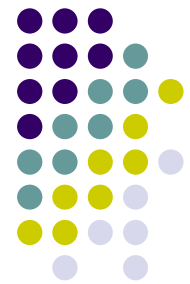




Logický posuv



Kruhový posuv



Operačná časť procesora



- **Registre** (napr. ACC, príznakový register, ...)
- **Predikáty, príznaky** = dvojhodnotové funkcie nad hodnotami premenných, v procesore sa uchovávajú na ďalšie spracovanie a vyhodnotenie v špeciálnom **príznakovom registri**.
- **Prepojovacie obvody**.
 - Spájajú jednotlivé prvky operačnej časti.
 - Realizujú medziregistrové prenosy
 - Dva spôsoby realizácie
 - multiplexory a demultiplexory - prepojenie registrov s ALJ.
 - zbernica.

Riadiaca časť procesora



- Uskutočňuje výber inštrukcií
- Dekódovanie inštrukcií
- Vykonanie inštrukcií
- Riadi spoluprácu procesora s okolím
- Inštrukčný cyklus sa spravidla skladá zo 6 fáz
 - Výber inštrukcie z pamäte - instruction fetch IF
 - Dekódovanie inštrukcie - decode D
 - Výpočet adresy operandov – operand address OA
 - Výber operandov z pamäte alebo zo vstupného zariadenia operand fetch OF, odpadne pokiaľ sú operandy v registroch
 - Vykonanie požadovanej operácie s operandami – execute EX
 - Zápis výsledku do pamäte alebo výstupného zariadenia – S
- V počítačoch von Neumana (SISD) sa jednotlivé fázy vykonávajú postupne-sériovo **IF D OA OF EX S IF D OA OF EX S IF**, jediný paralelizmus , paralelná sčítacia

Riadiaca časť procesora



Registre

(napr. PC, IR, príznakový register, ...) aj v časti pamäte

počítadlo inštrukcií (program counter - PC)
inštrukčný register (instruction register - IR).

PC je register v ktorom je adresa inštrukcie v operačnej pamäti, ktorá sa má práve vykonať.

IR je register do ktorého sa načíta inštrukcia, ktorá sa bude vykonávať z operačnej pamäti a ktorej adresa je v PC. Po načítaní inštrukcie do IR sa obsah PC zväčší o jednu, čím sa určí adresa nasledujúcej inštrukcie. Pokiaľ je načítana inštrukcia inštrukcia skoku, obsah PC sa prepočíta podľa adresy v inštrukcii skoku.

Riadiaca časť procesora

FORMÁT INŠTRUKCIE

KÓD ADRESA (code address)

0,1, 2, 3 - adresové inštrukcie

Typy inštrukcií

Presunové inštrukcie

Výpočtové inštrukcie

Skokové inštrukcie

Riadiace inštrukcie



Riadiaca časť procesora

Typy inštrukcií



- Presunové inštrukcie
($R \rightarrow R$, $R \rightarrow M$, $M \rightarrow R$, $M \rightarrow M$, $R \rightarrow VV$, $VV \rightarrow R$, $M \rightarrow VV$, $VV \rightarrow M$)
- Výpočtové inštrukcie (ADD, MUL, TEST, AND,...)
- Skokové inštrukcie
(podmienené JNZ, nepodmienené skoky JMP, skok do podprogramu, návrat z podprogramu, prerušenia),
zásobník, ...
- Riadiace inštrukcie (špeciálne operácie jako nastavovanie, nulovanie príznakov...)

Riadiaca časť procesora



Adresovanie	V inštrukcii	V registri	V pamäti	Príklad
Implicitné : <i>Registrové</i> <i>Nepriame registrové</i> <i>Zásobníkové</i>		Operand Adresa Adresa	Operand Operand	SCASW MOVSB POPF
Bezprostredné	Operand			MOV CX, 10H
Registrové	Register	Operand		MOV BX, DI
Priame	Adresa		Operand	MOV ALFA, DX
Nepriame	Adresa 1		Adresa2	JMP WORD PTR DST
Nepriame registrové	Register	Adresa	Operand	JMP WORD PTR [BX]
Indexové	Indexový reg. *Posunutie	Index	Operand	MOV QQQ[DI], AX
Bázovo-indexové	Bázový reg. Indexový reg. *Posunutie	Bázová adresa Index	Operand	MOV AX, QQ[BX][SI]

Krajčovič, T. "Počítače, STU, 2000, 157 str.



Riadiaca časť

- **Riadiaca časť s pevnou logikou (sekvenčný synchrónny obvod)**
 - ***Vstupný vektor*** tohto sekvenčného obvodu je tvorený *inštrukciami, signálmi z externého okolia* (napr. žiadosť o prerušenie, signál pripravenosti periférie atď.) a *príznakmi z operačnej časti*.
 - ***Výstupný vektor*** sa skladá z *povelov pre operačnú časť* (napr. signál zápisu do registra, nastavenie funkcie aritmeticko-logickej jednotky atď.) a *pre externé okolie* (napr. signál čítania z pamäte, zápisu do výstupného zariadenia, potvrdenia prerušenia atď.).



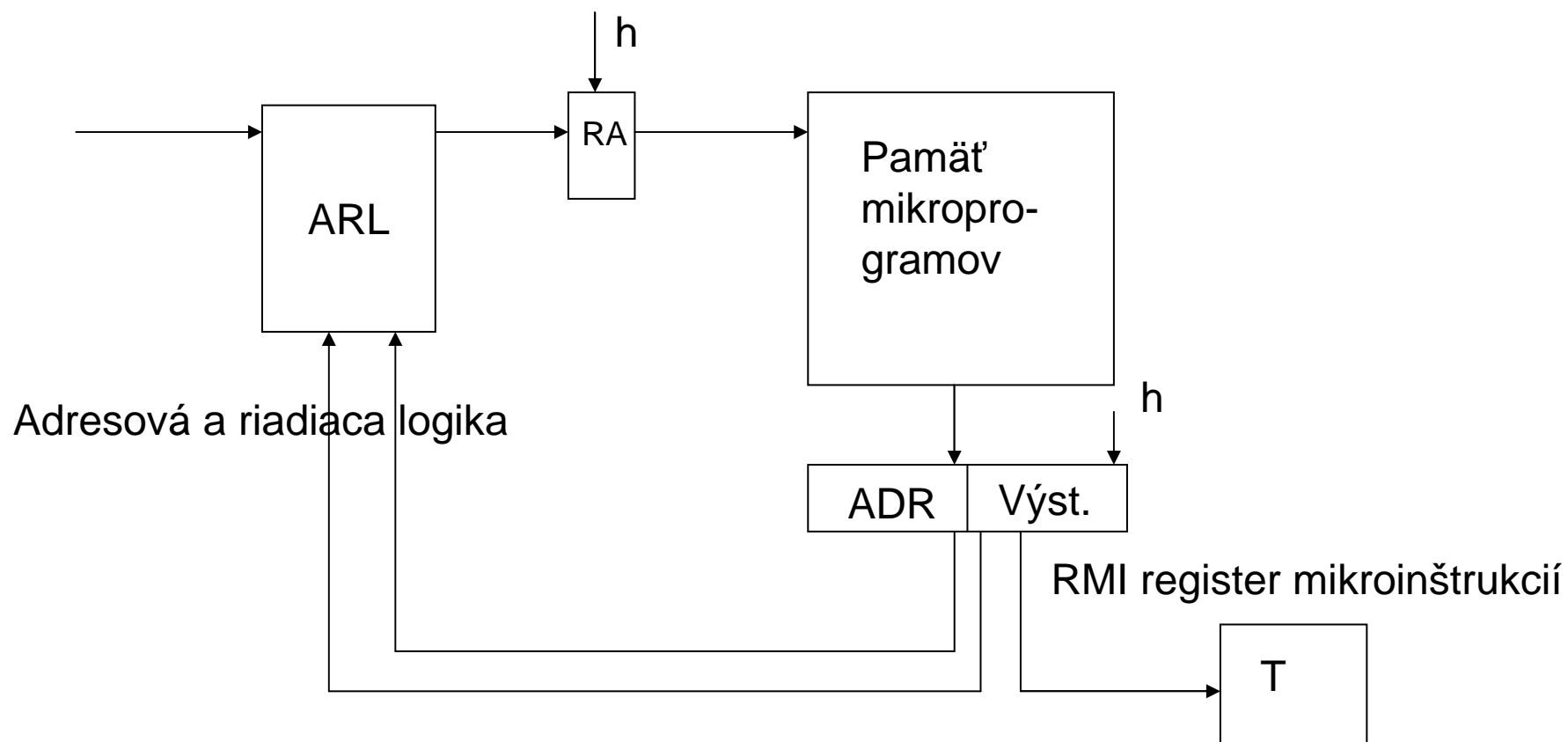
Riadiaca časť

- **Riadiaca časť s programovateľnou logikou**
 - Synchronný systém, v ktorom je inštrukcia realizovaná vykonaním mikroprogramu
 - **Mikroprogram** je postupnosť *mikroinštrukcií*, ktoré sú uložené v *pamäti mikroinštrukcií*. Mikroprogramovú riadiacu jednotku môžeme teda charakterizovať ako špecializovaný procesor, ktorý priamo interpretuje mikroprogramy, uložené v svojej pamäti mikroprogramov a vykonáva ich na danej operačnej časti.
 - Zmenou obsahu pamäte mikroinštrukcií je možné dosiahnuť zmenu inštrukčného súboru procesora. Takýmto spôsobom je potom možné na jednom procesore vykonávať programy pre iný typ procesora. Vtedy hovoríme o **mikroprogramovej emulácii**.



Riadiaca časť

Register adresy mikroinštrukcie



Transformačný obvod

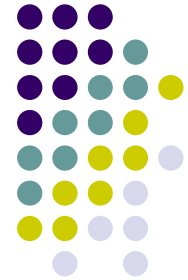
Zvyšovanie výkonnosti procesorov

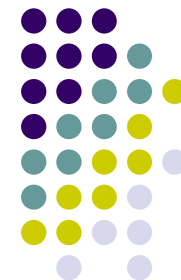
- **Zdokonalovanie technológie**

- Zmenšovanie fyzických rozmerov

- **Zdokonalovanie organizácie spracovania údajov**

- Pohyblivá rádová čiarka-numerický koprocessor
 - Predvýber a predspracovanie inštrukcií - rozdelenie výberovej a výkonnej fázy inštrukcií, jednotka predvýberu inštrukcií nevyužitie zbernice, rozdiel medzi rýchlosťou procesora a pamäte \Rightarrow predspracovanie inštrukcií, výber z pamäte, dekódovanie, výber operandov do Cache
 - Efektívne využitie zbernice
 - Problém skoky, dva prúdy inštrukcií





- Prúdové spracovanie inštrukcií – jednotlivé fázy (IF, D, OA ...0 sa realizujú nezávislou funkčnou jednotkou)

IF, D, OA, OF, EX, S

IF, D, OA, OF, EX, S

IF, D, OA, OF, EX, S

Údajová (data) a riadiaca nezávislosť inštrukcií, hazardy, konflikty

- Paralelné spracovanie inštrukcií
 - Aj v SISD
 - údajová a riadiaca nezávislosť inštrukcií
 - Párovanie inštrukcií (Pentium)

- **Procesory**

- CISC
- RISC
- NISC