

• **POŘÍSTE ZÁKLADNÍ ČÁSTI PŘEPÍNAČE A JEJICH FUNKCE**

- vstupní rozhraní - propojuje vstupní modul a přepínací logiku
 - rozděluje příchozí pakety na buňky podle určitého dělíky pro přepínání
- vstupní buffer - ukládá vstupní data, pokud je není možné poslat na přepínací logiku
- přepínací logika - dynamicky propojuje síťové rozhraní a přenáší mezi nimi data
 - vytváří propojení pomocí přepínacích obvodů
 - implementování na základní desce
- plánovač - plánuje přepínání dat ze vstup na výstup
- výstupní rozhraní a výstupní buffer

• **POŘÍSTE VLASTNOSTI A CHARAKTERISTIKY PŘEPÍNAČE SE SÍŤOVÝCH PAMĚTÍ**

- fronty pro vstupní porty - oblasti pevné/proměnné velikosti
- rychlost přístupu do paměti $bw = 2 \times N \times R [b/s]$, R - rychlost rozhraní, N - počet síťovek
- doba přenosu dat o velikosti C: $t = \frac{C}{bw}$
- potřeba doby zápisu? - 32 portů, 1 Gb/s, velikost buňky 40 byte
- jak se změní pro 10 Gb/s a 40 Gb/s

$$t_1 = \frac{C}{bw} = \frac{C}{2 \times N \times R} = \frac{40 \cdot 8}{2 \cdot 32 \cdot 1 \cdot 10^9} = 5 \cdot 10^{-9} = 5 ns$$

$$t_{40} = \frac{2 \cdot 32 \cdot 10 \cdot 8}{40 \cdot 8} = 0,16 ns$$

• **VÝHODY A NEVÝHODY KŘÍŽOVÉHO PŘEPÍNAČE**

- + interní neblokující architektura
 - vysoká rozšiřitelnost
 - možnost kolizí
 - obřížná zátěž kvalita služeb při současných požadavcích
 - chybí redundance
- + kvalitní podpora multiplexu
- + jednodušší implementace
- + centrální plánovač

• **CO JE TO BLOKOVÁNÍ HOL A JAK HO LZE ELIMINOVAT**

- HOL - blokováání na základě fronty COBLOKOVÁNÍ DOLEVA BE SPEC. PRÁHU
- PAKET čeká na vstupním portu ve FIFO frontě
- první buňka blokuje zbyte požadavků, blokové buňky čekají i když se odvolají port celých
- ŘEŠENÍ: - virtuální fronty podle cílových portů

• **PRÍKLAD A POUŽITÍ ALG. P1H**

- hledá maximální početů pomocí náhodné volby požadavků ze vstupních portů
- 1. žádost o port - vstupní port P1 má žádost že všech front na vstupní porty náhodně
- 2. udělení portu - výstupní port Q1 má právo přenosu, při více žádostech volí jednu náhodně a přenese
- 3. přijetí a přenos - vstupní port P2 má udělení k přenosu, v případě více požadavků, v případě více požadavků

• **PRÍKLAD A POUŽITÍ ISLIP**

- iterativní alg. pro hledání maximálního přenosu
- Soupeření o port - Rychleji ukazatele: 1 - ukazatel na vstupním portu i (přijetí)
- Ukazatel na výstupním portu j (udělení povolání)
- Po každé iteraci se ukazatele inkrementují, ukazatel se inkrementuje až do žádosť o port potrvá
- 1. žádost o port - každý port poše žádost na výstupní port
- 2. udělení portu: výstupní port y vybere žádost, která má číslo portu větší nebo rovno ukazateli y. když
- 3. přijetí a přenos: vstupní port x vybere povolání od portu, které je větší nebo rovno ukazateli x. v případě více požadavků, v případě více požadavků

• **ČÍTYKOPROSTÝ KŘÍŽOVÝ PŘEPÍNAČ - VSTUP A-D, VÝSTUP 1-4. NA VSTUPU JE: A → (1,3,3), B → (2,3,4), C → (3,2,1), D → (4,3,4). JAK BUDE VYPADET PŘEKOV V ČAS. SLOTECH 1-6 PŘI 1. PŘIDĚLOVÁNÍ LÍSTKŮ**

1. PŘIDĚLOVÁNÍ LÍSTKŮ

	1	2	3	4	5	6		1	2	3	4	5	6
A	1	3	-	-	3	-	A	3	1	-	-	-	-
B	2	-	3	4	-	-	B	2	4	5	-	-	-
C	3	2	1	-	-	-	C	1	2	-	3	-	-
D	4	-	-	3	1	-	D	4	3	1	-	-	-

2. P1H

	1	2	3	4	5	6
A	3	1	-	-	-	-
B	2	4	5	-	-	-
C	1	2	-	3	-	-
D	4	3	1	-	-	-

více, tak nejmenší hodnotu
tak náhodně