Datová úložiště

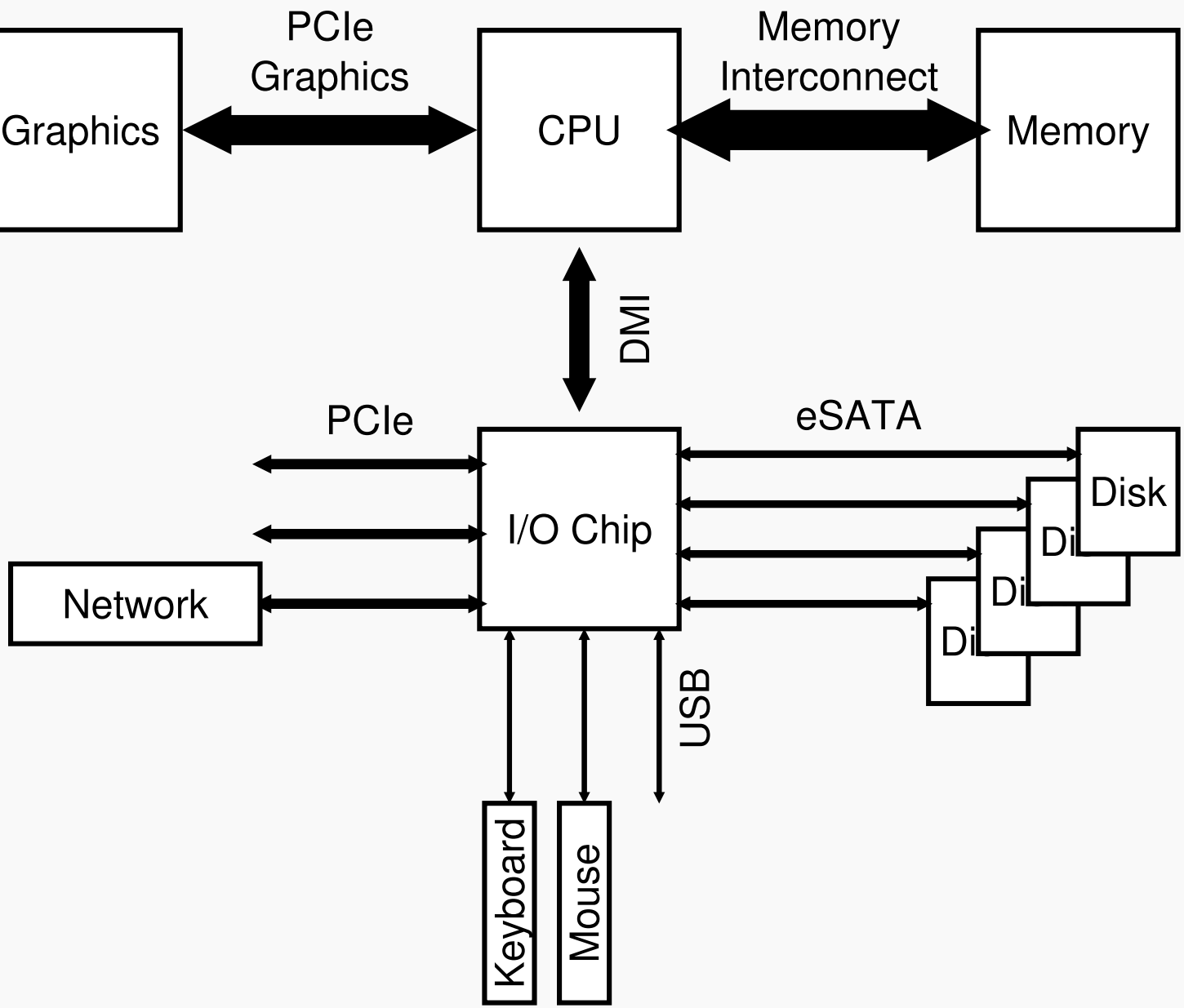
Komunikace s I/O, HDD, RAID

Milan Radojčić

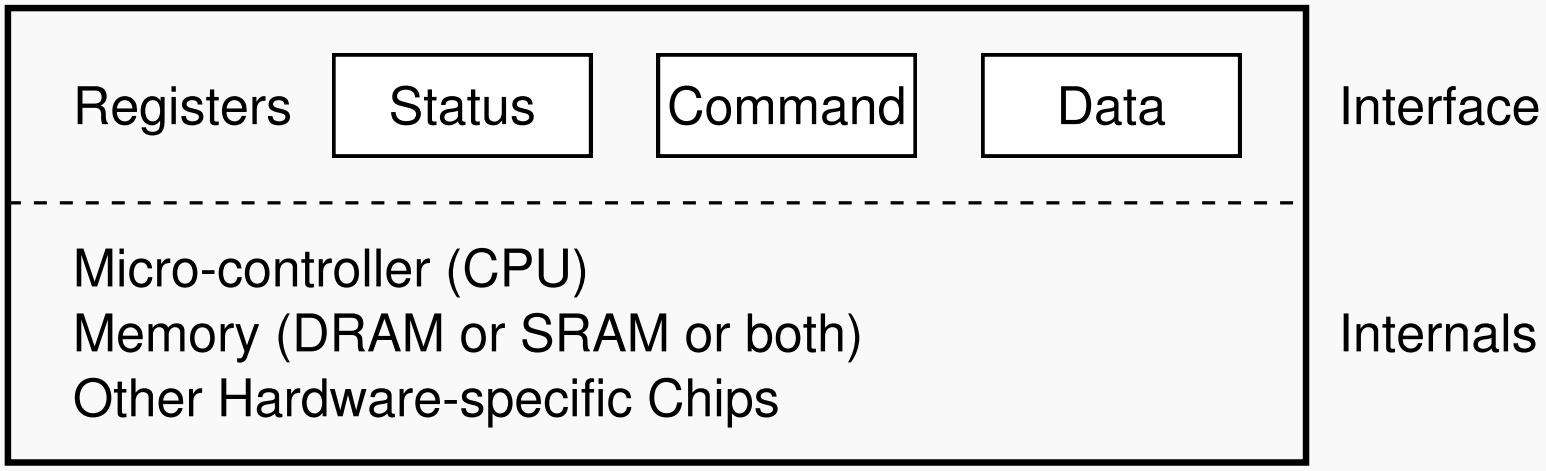
SSPŠaG – Operační systémy

30. 3. 2026

Komunikace s periferiemi



Model zařízení



Nějaké zařízení: vnější rozhraní a vnitřek

- CPU a periferie si informace předávají pomocí **speciálních registrů**.
  - Více info o způsobu zápisu bude později.
- Např. pro zápis do HDD může proběhnout:
  1. Zápis daného bloku a adresy do datového registru.
  2. Zápis požadavku o zápisku do registru pro příkaz.
  3. Periodicky kontrolujeme stavový registr.

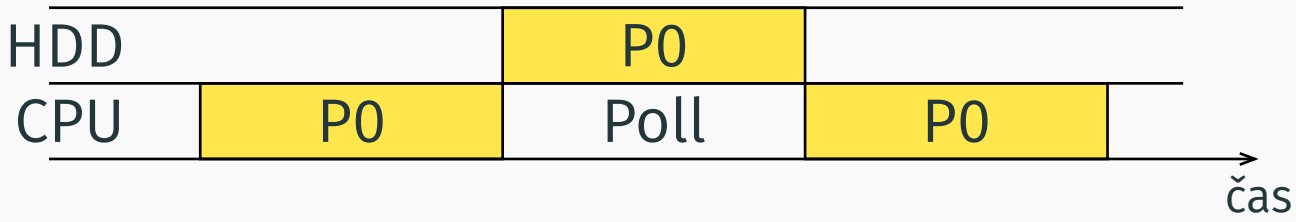
Komunikace s periferiemi

- Zapisovat a číst registry periferií můžeme číst pomocí speciálních instrukcí – **port-mapped I/O**.
- Jiný způsob jak to provést je *namapováním registrů periferií do paměti* – **memory-mapped I/O**.
  - Přístup k periferiím je privilegovaná operace – **uživatelské programy nemůžou spouštět tyto instrukce ani přistupovat do této paměti**.

Jaký z přístupů myslíte, že volí architektura x86?

Komunikace s periferiemi

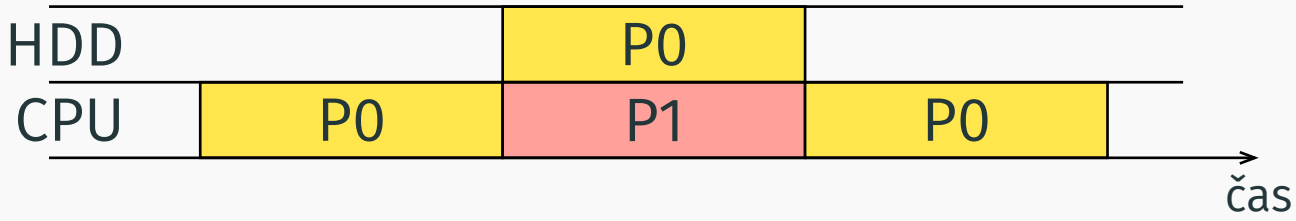
- Představte si, že chceme zapsat něco na disk.
  1. Zapíšeme všechny správné data do registrů.
  2. Jak zjistit, že zápis skončil? (Úspěšně či neúspěšně.)
- Jedna možná technika je **číst stavový registr dokola ve smyčce** (polling).



- Z diagramu vidíme, že část času CPU nevyužíváme.

Komunikace s periferiemi

- Můžeme využít přerušení z předchozí hodiny!
- HDD pošle CPU přerušení, až skončí.
  - My víme, že v tu chvíli HDD dokončil zápis a my můžeme přečíst stav z registru



- CPU může v mezičase přeplánovat na jiný proces.

Komunikace s periferiemi

- Kdy pollovat a kdy používat přerušení? Záleží na rychlosti zařízení...
  - Pokud je pomalejší tak se vyplatí přerušení.[1]
  - Když je zařízení rychlejší tak se může vyplatit pollovat. (Pro přeplánování se musí provést změna kontextu!)

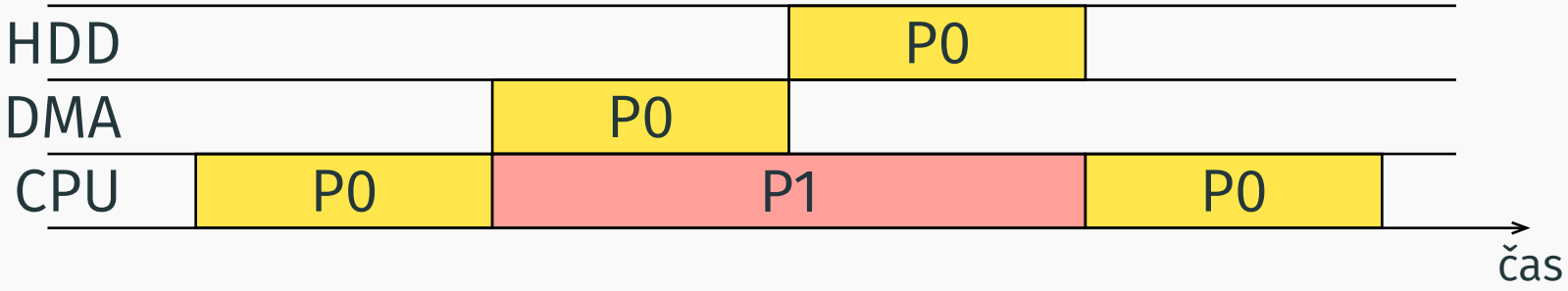
[1] Např. pokud zápis na disk trvá v několik ms tak přicházíme o tisíce cyklů na CPU v případě frekvence v řádech GHz.

Komunikace s periferiemi

- Bloky jsou ale typické velké (4KB) a do zařízení se musí kopírovat po slovech (typicky 32 nebo 64 bitů).
- Takže náš průběh vypadá spíše takto:

Komunikace s periferiemi

- Tento problém řeší **Direct Memory Access** (*DMA*).
- DMA controller je zařízení, které dokáže kopírovat data mezi hlavní pamětí a zařízeními.
- CPU tedy stačí naprogramovat DMA controller, aby přesunul dané data na HDD.

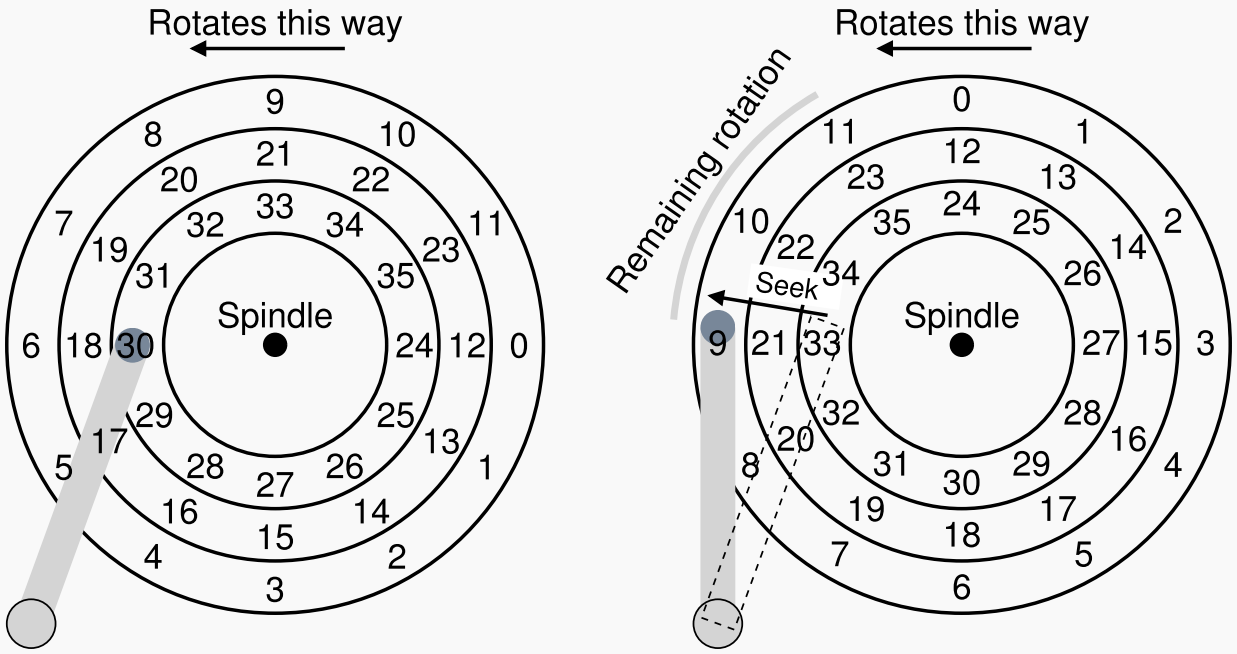


Poznámky: Komunikace s periferiemi

- Stačí nám chápat obecně jak funguje komunikace s periferiemi.
- Pro zájemce o jednoduchou opravdovou implementaci je v Operating Systems: Three Easy Pieces[<2>] v kapitole I/O Devices na konci rozebrána ukázka IDE driveru v xv6 kernelu.

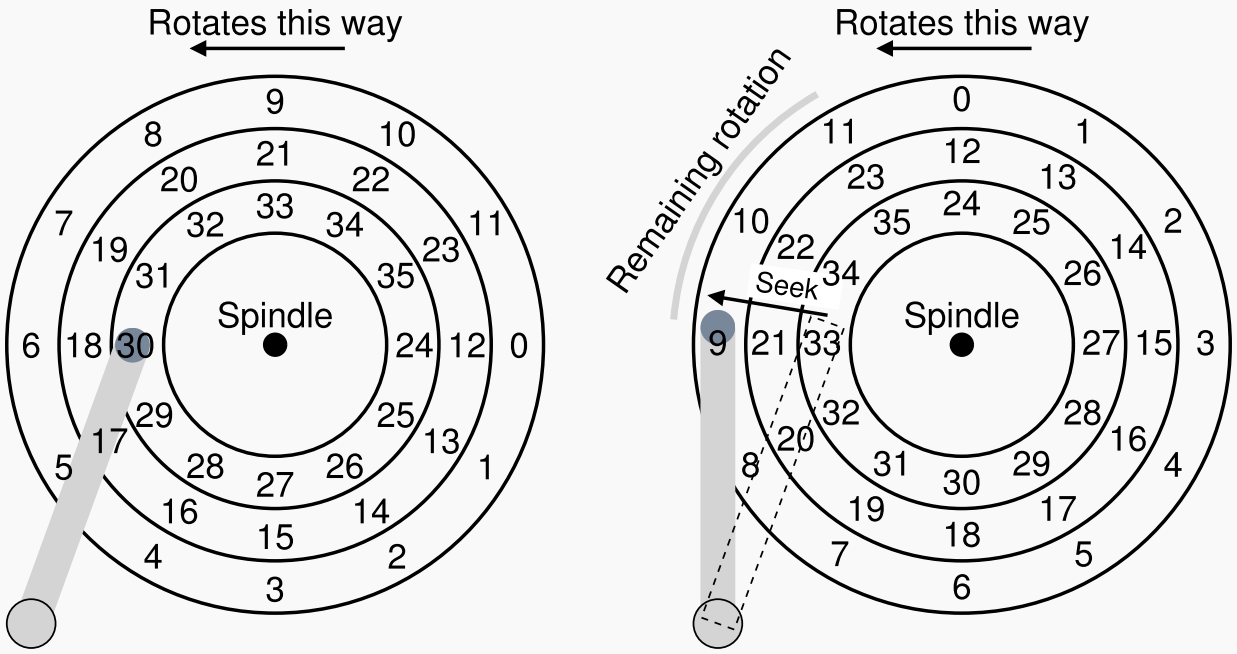
[<2>] https://pages.cs.wisc.edu/~remzi/OSTEP/

Geometrie HDD



- HDD se skládá z několika **ploten**, na jejich povrchu je speciální materiál do kterého se pomocí směru magnetizace ukládá informace.
- Ke čtení se používají speciální **hlavičky**, ty se můžou pohybovat směrem od středu.
- Celý disk se točí a proto se mohou hlavičky dostat všude.

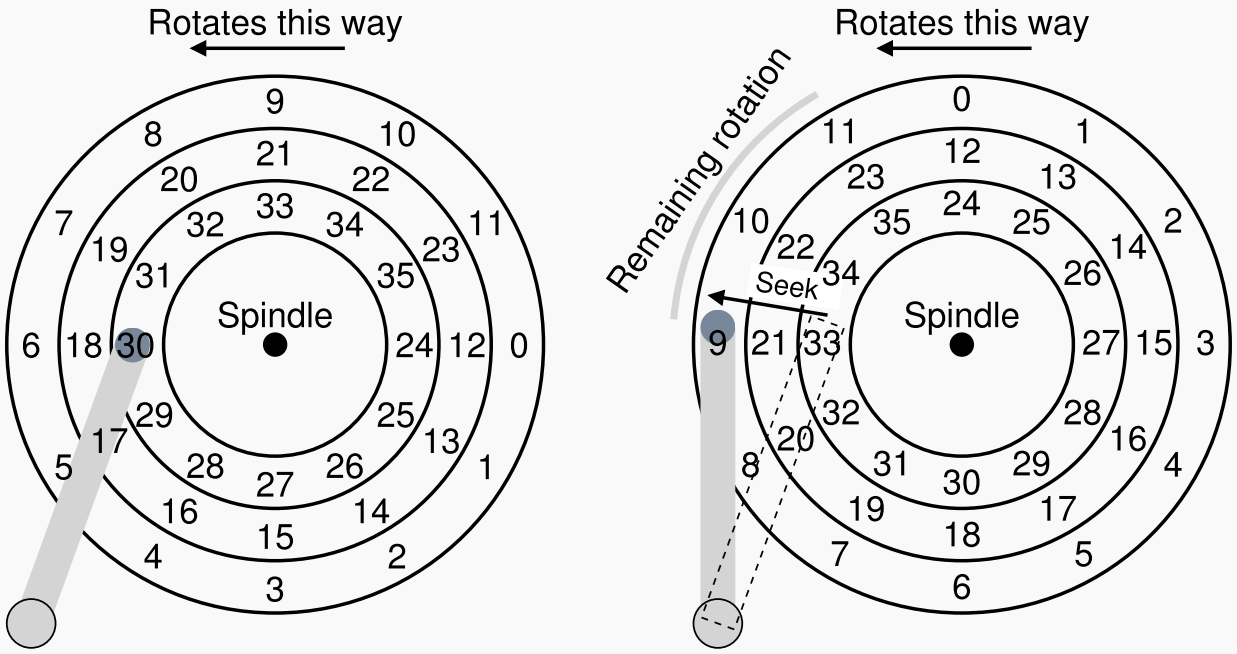
Geometrie HDD



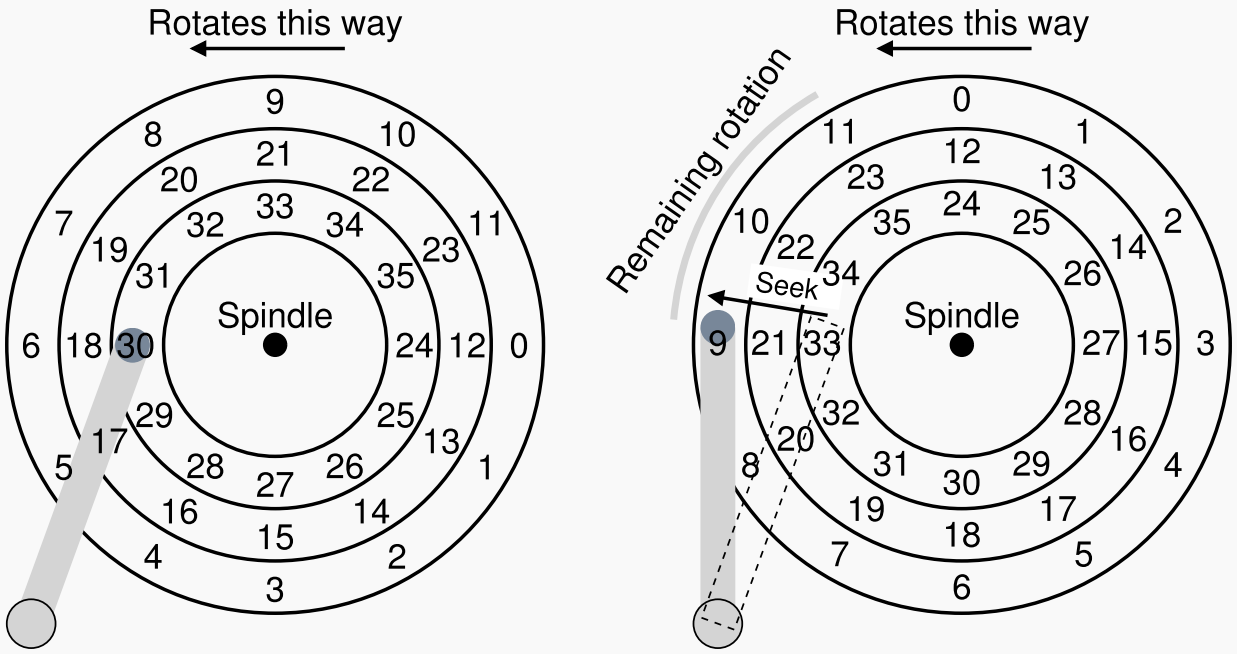
- Jednomu kruhu na povrchu plotny říkáme **stopa** (*track*).
  - Protože můžeme mít více ploten, tak všem stopám na všech plotnách se stejným poloměrem říkáme **cylindr**.
- Nejmenší adresovatelná jednotka se nazývá **sektor** nebo **blok** a má (většinou) 4KB.

Geometrie HDD

Na čem závisí rychlost zápisu?



Geometrie HDD



1. **Doba vystavení** (*seek time*)
   - Čtecí hlavička se musí dostat nad správnou stopu.
2. **Rotační zpoždění** (*rotational delay*)
   - Čtecí hlavička musí počkat, než se disk otočí a dostane se nad správný sektor.

Vsuvka: Testování rychlosti úložiště

```
fio --name=read_test \
    --filename=/dev/sdX \
    --rw=read \
    --bs=4k \
    --size=1G \
    --numjobs=1 \
    --iodepth=16 \
    --direct=1 \
    --runtime=60 \
    --time_based \
    --group_reporting
```

- Provádí sekvenční čtení, pro náhodné čtení lze použít `--rw=randread`.

Rychlost HDD

- Mějme HDD s těmito parametry: 15000 RPM, průměrná doba vystavení 4ms, rychlost přenosu: 125 MB/s.
- Jak rychlý bude tento HDD když budeme uvažovat náhodný přístup k sektorům?
- Celkový čas na provedení I/O můžeme rozepsat jako

$$T_{\text{I/O}} = T_{\text{seek}} + T_{\text{rotation}} + T_{\text{transfer}}.$$

- Potřebujeme získat z RPM čas na jednu rotaci v ms:

$$\frac{1\,\cancel{\text{min.}}}{15000\,\text{rot.}} \cdot \frac{60\,\cancel{\text{sek.}}}{1\,\cancel{\text{min.}}} \cdot \frac{1000\text{ms}}{1\,\cancel{\text{sek.}}} = 4\frac{\text{ms}}{\text{rot.}}$$

- Máme tedy $T_{\text{seek}} = 4\text{ms}$, $T_{\text{rotation}} = 2\text{ms}$.

Rychlost HDD

- Mějme HDD s těmito parametry: 15000 RPM, průměrná doba vystavení 4ms, rychlost přenosu: 125 MB/s.
- Jak rychlý bude tento HDD když budeme uvažovat náhodný přístup k sektorům?
- Čas na přenos můžeme získat jako

$$T_{\text{transfer}} = \frac{\text{velikost přenosu}}{\text{rychlost přenosu}}.$$

- Pro náš případ je to $T_{\text{transfer}} = \frac{4\text{KB}}{125\text{MB/s}} \approx 30\mu s$ – zanedbáme.

Rychlost HDD

- Mějme HDD s těmito parametry: 15000 RPM, průměrná doba vystavení 4ms, rychlost přenosu: 125 MB/s.
- Jak rychlý bude tento HDD když budeme uvažovat náhodný přístup k sektorům?
- Celkem tedy $T_{I/O} = 6\text{ms}$, propustnost získáme jako

$$R_{I/O} = \frac{\text{velikost přenosu}}{T_{I/O}}.$$

- Tedy pro náš případ

$$R_{I/O} = \frac{4\text{KB}}{6\text{ms}} = \frac{1\text{MB}}{1024\cancel{\text{KB}}} \cdot \frac{4\cancel{\text{KB}}}{6\cancel{\text{ms}}} \cdot \frac{1000\cancel{\text{ms}}}{1\text{s}} \approx 0,65\text{MB/s}!$$

Rychlost HDD

- Mějme HDD s těmito parametry: 15000 RPM, průměrná doba vystavení 4ms, rychlost přenosu: 125 MB/s.
- Jak se změní situace když budeme uvažovat sekvenční přístup např. ke 100 MB?
- T_{seek} a T_{rotation} budou stejné, potřebujeme přepočítat T_{transfer} :

$$T_{\text{transfer}} = \frac{100\text{MB}}{125\text{MB/s}} = 800\text{ms}.$$

- Tedy $T_{\text{I/O}} = 806\text{ms}$ a $R_{\text{I/O}} = \frac{100\text{MB}}{806\cancel{\text{ms}}} \cdot \frac{1000\cancel{\text{ms}}}{1\text{s}} \approx 124.$