

PB153

Mgr. Jiří Činčura

jiri@x2develop.com

blog.cincura.net

Mooreův zákon

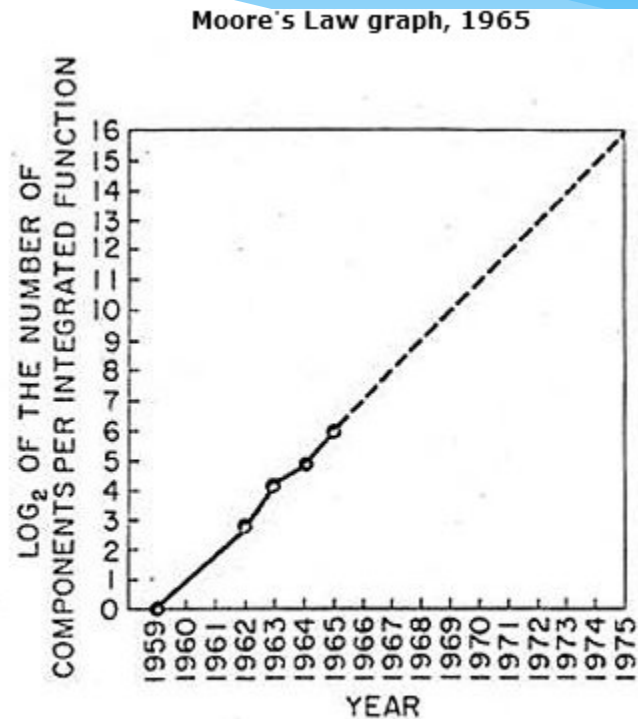


Fig. 2 Number of components per Integrated function for minimum cost per component extrapolated vs time.

„free lunch“

- * Hrubý výkon CPU již neroste jako dříve
 - * teplo
- * Přidávají se další sofistikované jednotky
 - * Hyperthreading
- * Především celá jádra
 - * Vícejádrové procesory
- * Stroje s více procesory

Aktuální trendy

- * Využití více výpočetních jednotek těžké
 - * Efektivní
- * Mnoho procesů, ale pracuje jich jen málo a neefektivně
- * Potřeba využít celý výpočetní výkon
 - * V budoucnu 80+ jader/CPU

Vlákna (thready)

- * Spolehlivost systému
- * Bezpečnost (izolace)
- * Škálovatelnost
- * Robustnost
- * Virtualizace CPU

Vlákna (thready)

- * Vlákna jsou drahé objekty
 - * Vytvoření, zrušení
 - * 200 000 cyklů vytvoření, 100 000 zrušení
- * Thread kernel object
 - * Vlastnosti vlákna
- * Thread environment block (TEB, 4KB)
 - * Exception-handling chain, TLS, GDI/OpenGL
- * User-mode stack
 - * 1MB, CLR committed
- * Kernel-mode stack (12KB/24KB)
- * DLL attach/detach notifikace

Vlákna (thready)

- * V systému mnoho vláken
 - * Nejlepší vlákna = počet CPU
 - * Většina vláken nic nedělá
- * Context switching
 - * 6000-8000 cyklů
- * CLR vlákna mapována na Windows vlákna

Vlákna (thready)

- * Windows scheduler
 - * Preemptive multitasking
 - * Běží vlákna, ne procesy
- * Time slice/quantum
- * Switching
- * Priorita

Vlákna (thready)

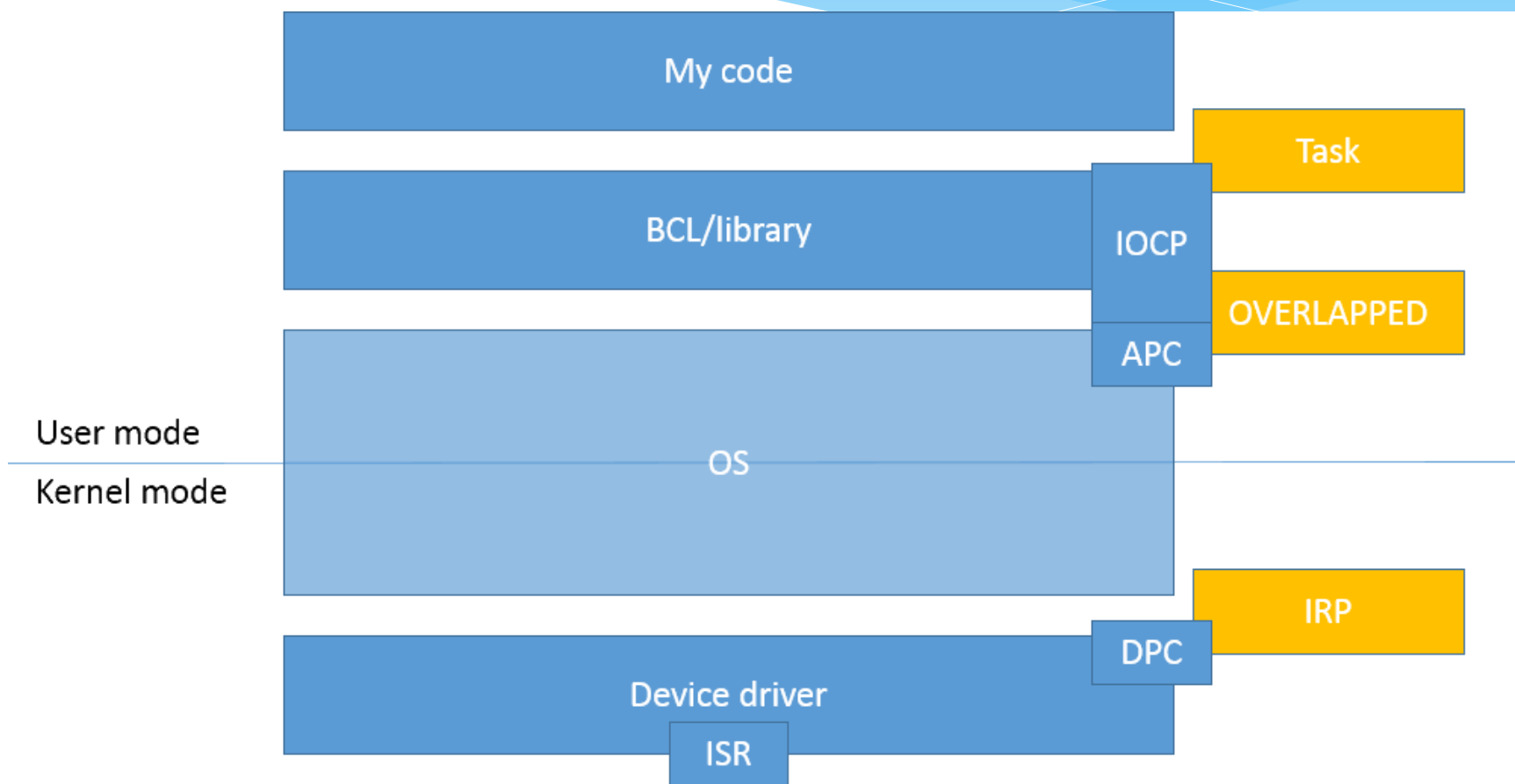
- * Priority
 - * 0 až 31
- * Stárnutí
- * Priority boost
- * Priorita procesu (6)
- * Priorita vlákna (7)
- * <http://msdn.microsoft.com/en-us/library/ms685100.aspx>

DEMO

I/O bound operace

- * I/O operaci provádí HW ne CPU
- * Systém je informován o výsledku
- * Thread je blokován a nic nedělá
 - * Plýtvání zdroji
- * Asynchronní operace jsou efektivnější

I/O bound operace



DEMO

Zamykání, koordinace

- * Nejzajímavější část vícevláknového/paralelního programování
- * Ovlivňuje výslednou efektivitu řešení
- * Izolace více vláken od konkurenční změny sdílených dat a jejich poškození
- * Zamykat co nejméně
- * Na co nejkratší + rozumnou dobu
- * Zamykání je drahá operace

Zamykání, koordinace

- * User-mode
 - * Rychlejší než kernel-mode
 - * Speciální CPU instrukce
 - * Systém neví o blokování (plýtvání CPU)
 - * Vhodné pro krátké operace
- * Kernel-mode
 - * Poskytuje jádro systému
 - * Přechod user-mode → kernel-mode
 - * Systém blokuje vlákno
 - * Při dostupnosti zdroje je vlákno systémem probuzeno
 - * Cross process
- * Hybridní

Zamykání, koordinace

- * Deadlock
- * Livelock
- * Starvation
- * Fairness
- * Convoys

DEMO

Q & A

