

Sistem Operasi

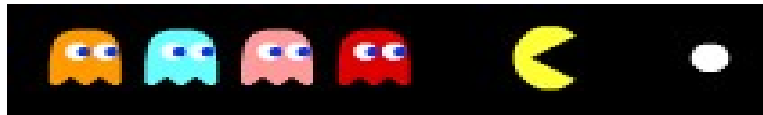
Thread
2016

Thread

- Bagian terkecil dalam sebuah instruksi yang ter-program yang dapat diatur secara independen oleh sistem operasi
- Jika dibandingkan dengan proses, kinerja thread lebih ringan (light weight process)
- Implementasi thread dan proses dapat berbeda pada setiap OS, tapi pada umumnya thread selalu berada di dalam sebuah proses

Dimana thread digunakan?

- Task independen
 - contoh: debugger memerlukan GUI, program, perf monitor...
 - terutama saat blocking untuk I/O!
- Program tunggal, operasi concurrent
 - Server
 - contoh: file server, Web server
 - Kernel OS kernels
 - concurrent system request oleh banyak pemakai
 - Game



Proses vs. Thread

- Proses lebih lambat
 - 30 kali lebih lambat untuk di create di Solaris
 - Destroy lebih lambat
 - Pertukaran kontext lebih lambat
- Proses meminta banyak memori
 - beberapa ribu proses → tidak ok
 - beberapa ribu thread → ok

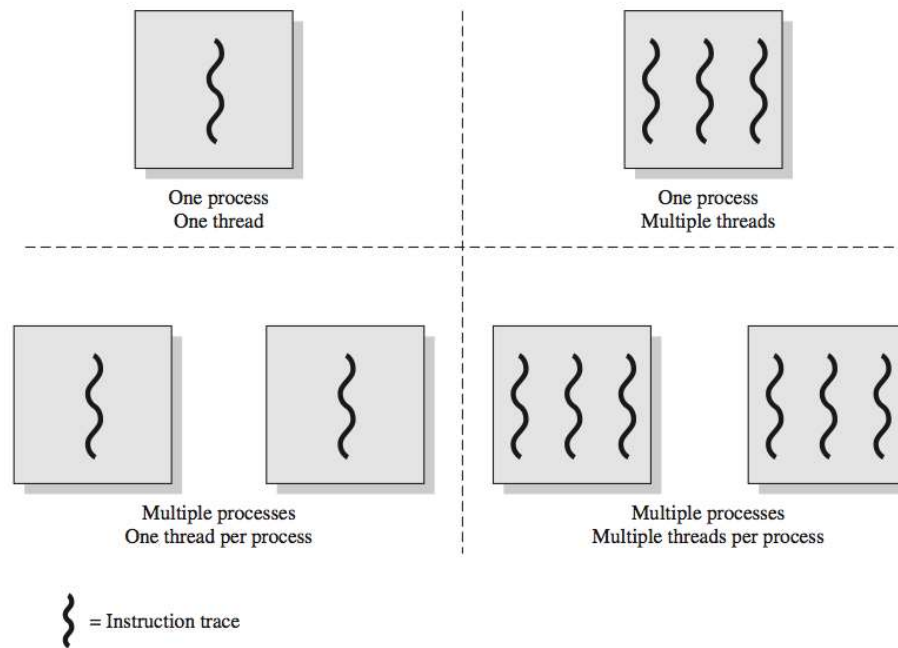
Multithreading

- Multithreading mengacu kepada kemampuan sebuah OS untuk mendukung beberapa eksekusi yang dapat dijalankan secara serentak yang berada dalam 1 proses

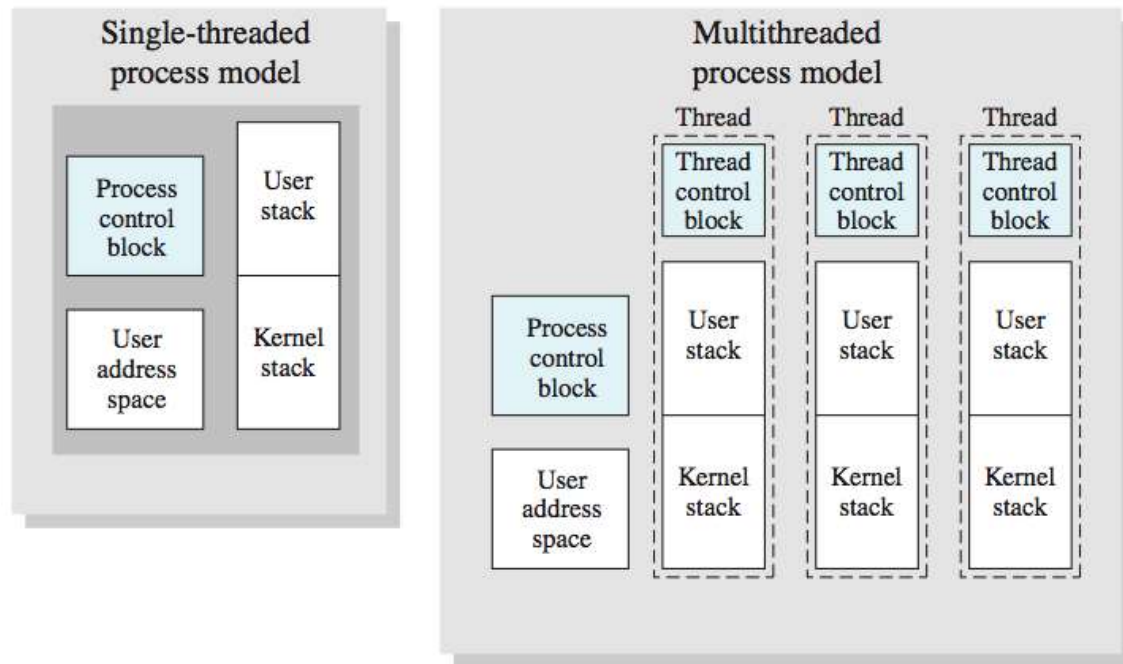
Multithreading

- MS DOS → single user process, single thread
- Some variants of UNIX → multiple process, single thread
- Java run-time environment → single process, multiple thread
- Modern OS (UNIX, Win, etc) → multiple process, multiple thread

Multithreading?



Single thread vs Multithreading



Single thread vs Multithreading

- Secara konsep hampir sama dengan single thread, tetap memiliki PCB, dan user address space.
- Tetapi user & kernel stack sudah diatur oleh thread
- Dalam thread tersebut berisikan nilai-nilai register, prioritas, dan informasi state dari thread yang lainnya
- Pada single thread, sebuah proses direpresentasikan dalam sebuah kesatuan process control block, user address space, user & kernel stack.
- Ketika sebuah proses berjalan, proses mengontrol processor register
- Isi dari register disimpan ketika prosesnya sedang tidak berjalan

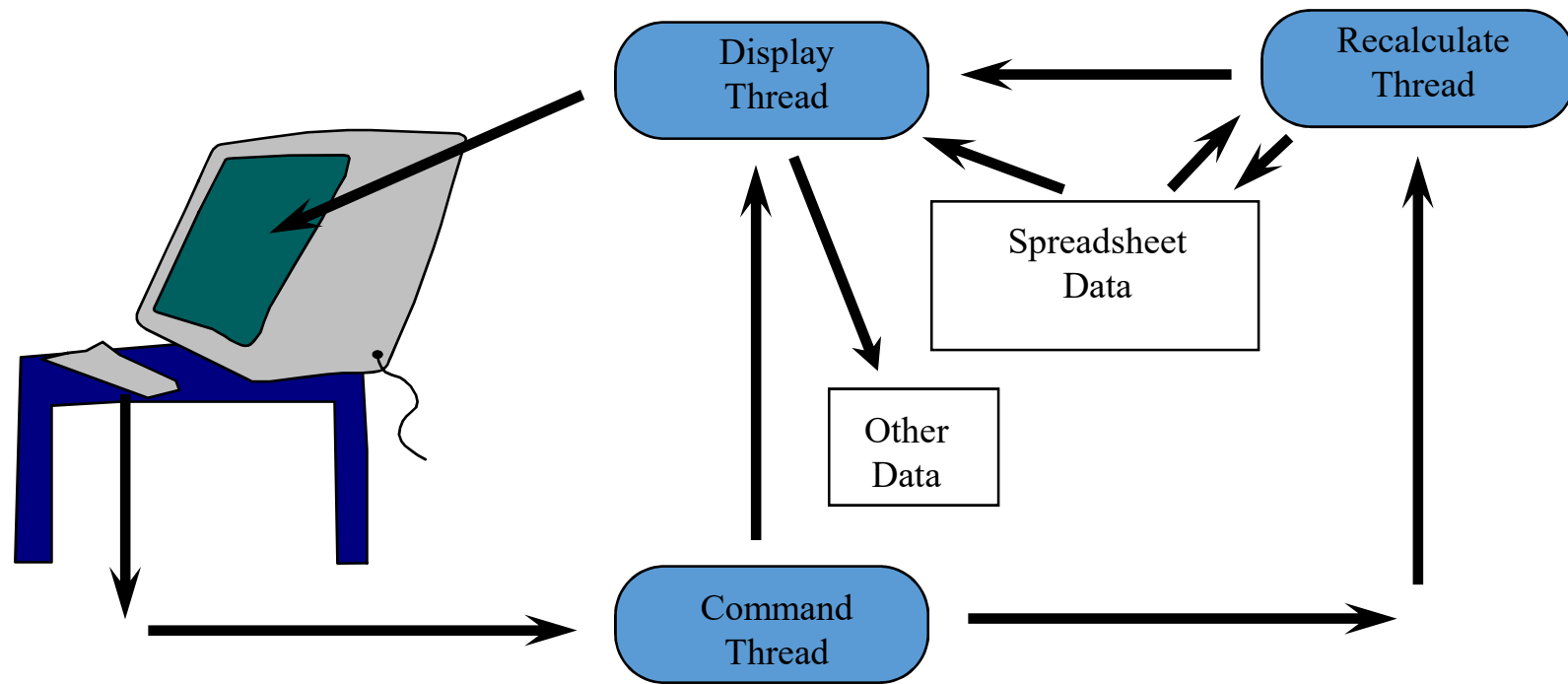
Keuntungan Threading?

- Lebih cepat membuat thread daripada membuat proses. Studi membuktikan membuat thread 10x lebih cepat daripada membuat proses
- Lebih cepat mematikan (terminate) thread
- Lebih cepat untuk berpindah antar thread dalam proses yang sama
- Thread menggunakan resource(memory/files) yang sama dengan proses, thread dapat berkomunikasi secara langsung tanpa mengganggu kernel yang ada

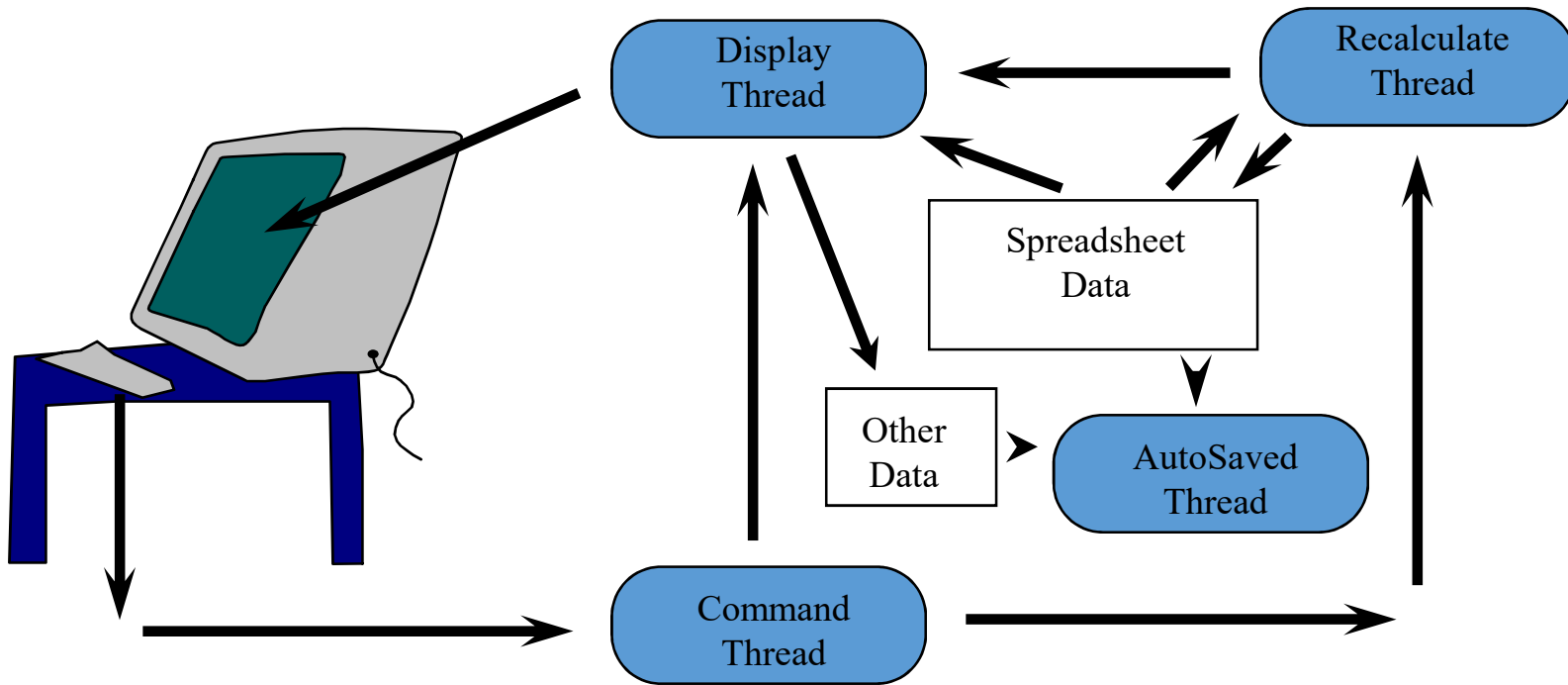
Penggunaan Thread

- Pada Single-User Multiprocessing System□
 - Foreground to background work
 - Asynchronous processing
 - Speed execution
 - Modular program structure

Foreground to Background work



Asynchronous processing



Speed execution

- Multithread proses dapat mem-batch sebuah data pada saat membaca batch berikutnya dari perangkat.
- Pada sistem multiprocessor, thread-thread yang berasal dari proses yang sama, dapat dikerjakan secara simultan
 - Jika ada thread yang di block karena operas I/O, thread yang lain masih dapat dieksekusi.

Modular program structure

- Penggunaan thread akan lebih mudah untuk sebuah program yang membutuhkan banyak variasi I/O

Kekurangan Thread?

- Melakukan suspend terhadap sebuah proses, berarti semua thread ikut di-suspend
- Jika sebuah proses dimatikan (terminate), semua thread juga ikut mati.

State?

Process State

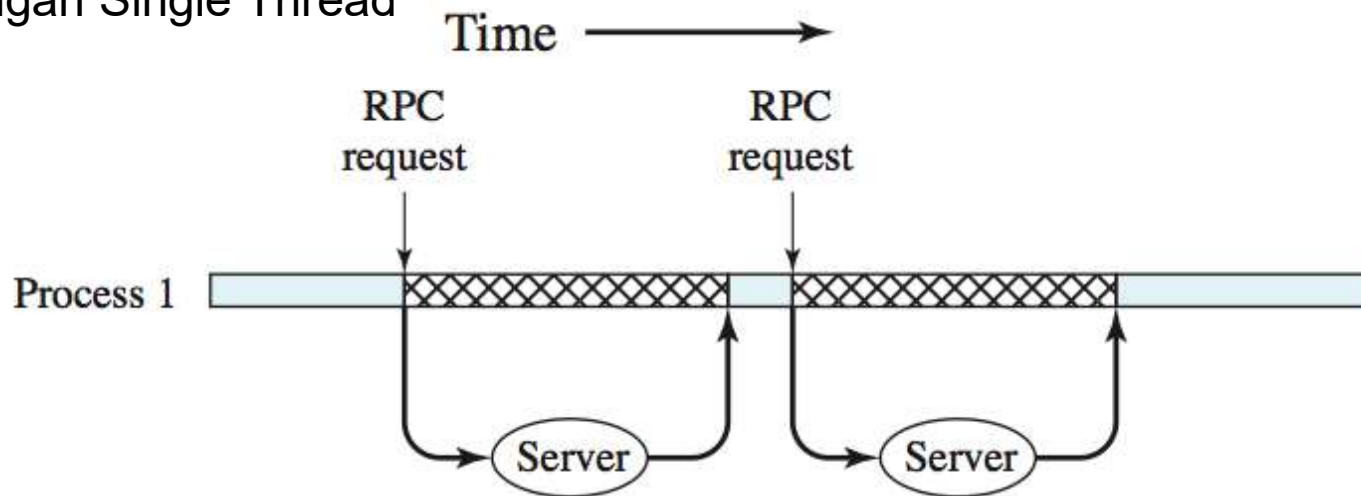
- Ready
- Running
- Block

Thread State

- Spawn
 - Spawn thread yang lain
- Block
- Unblock
- Finish
 - Dealokasikan register

Contoh : Threading di RPC

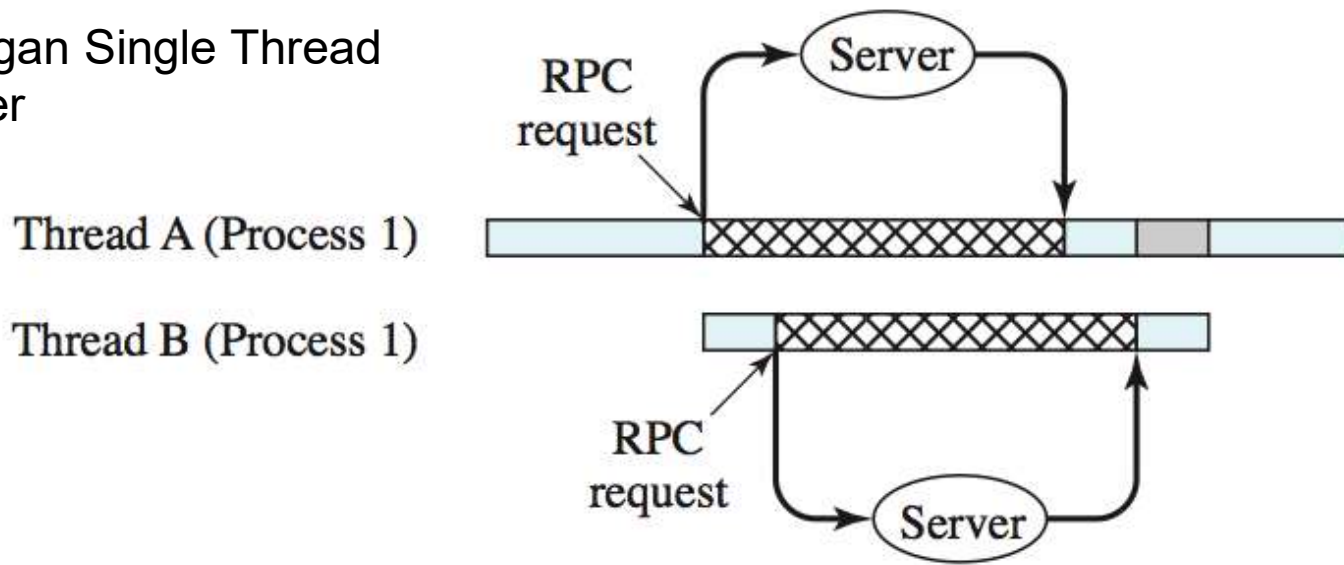
RPC dengan Single Thread



- ⊠ Blocked, waiting for response to RPC
- Blocked, waiting for processor, which is in use by Thread B
- Running

Contoh : Threading di RPC

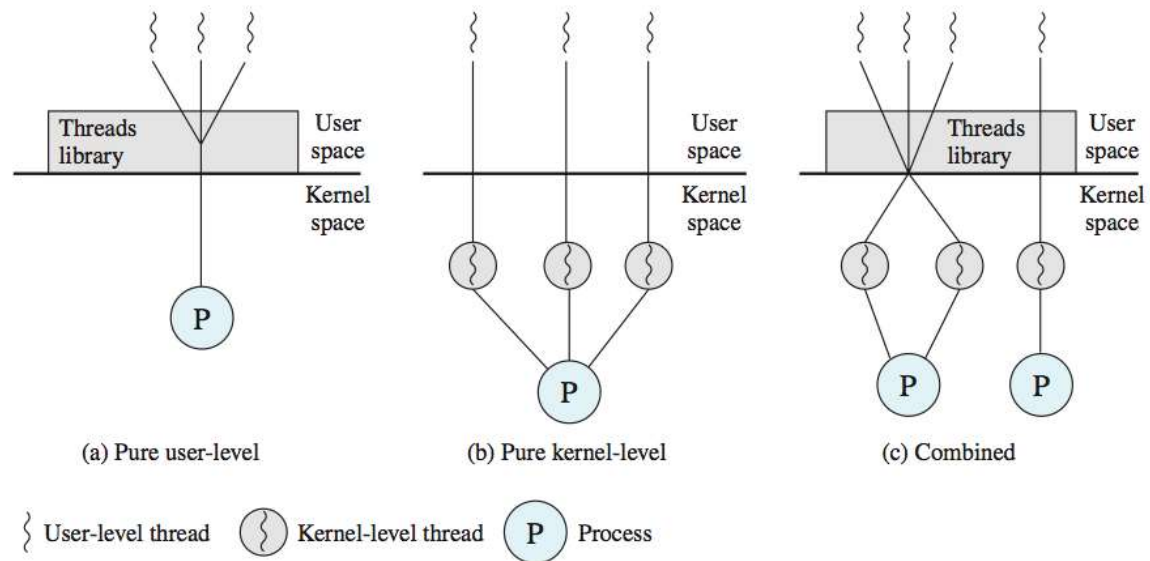
RPC dengan Single Thread Per Server



- ⊠ Blocked, waiting for response to RPC
- Blocked, waiting for processor, which is in use by Thread B
- Running

Implementasi Thread

- User-Level Thread
- Kernel-Level Thread
- Kombinasi



User-Level Thread (ULT)

- Semua pekerjaan (Thread) dilakukan oleh aplikasi
- Kernel tidak tahu akan keberadaan thread ini

Kernel-Level Thread (KLT)

- Implementasi threading diatur oleh kernel
- Kernel menjaga informasi untuk proses dan thread
- Pengaturan penjadwalan dilakukan berbasis thread

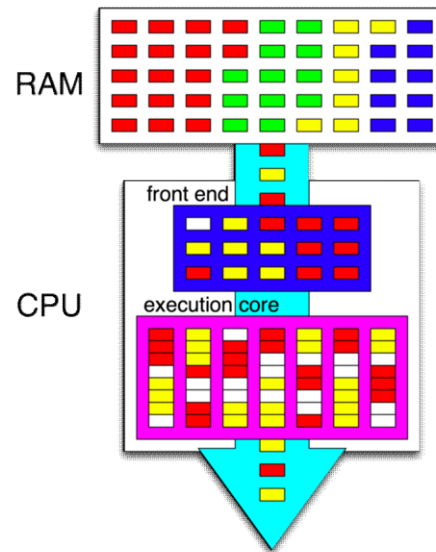
Combined

- Example : Solaris
- Pembuatan thread terletak di area user
- Penjadwalan dan sinkronisasi dikerjakan di area user

Perbandingan Thread:Process

Threads:Processes	Description
1:1	Each thread of execution is a unique process with its own address space and resources.
M:1	A process defines an address space and dynamic resource ownership. Multiple threads may be created and executed within that process.
1:M	A thread may migrate from one process environment to another. This allows a thread to be easily moved among distinct systems.
M:N	Combines attributes of M:1 and 1:M cases.

Hyperthreading




CPU-Z

CPU | Caches | Mainboard | Memory | SPD | Graphics | About

Processor

Name	Intel Xeon W3520		
Code Name	Bloomfield	Brand ID	
Package	Socket 1366 LGA		
Technology	45 nm	Core Voltage	1.200 V



Specification: Intel(R) Xeon(R) CPU W3520 @ 2.67GHz

Family	6	Model	A	Stepping	5
Ext. Family	6	Ext. Model	1A	Revision	D0

Instructions: MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, EM64T

Clocks (Core #0)		Cache	
Core Speed	4000.1 MHz	L1 Data	4 x 32 KBytes 8-way
Multiplier	x 20.0	L1 Inst.	4 x 32 KBytes 4-way
Bus Speed	200.0 MHz	Level 2	4 x 256 KBytes 8-way
QPI Link	3600.1 MHz	Level 3	8 MBytes 16-way

Selection: Processor #1 | Cores: 4 | Threads: 8


CPU-Z Version 1.52 | Validate | OK

CPU-Z **Atom N270**

CPU Cache Mainboard Memory SPD About

Processor

Name	Intel Atom N270		
Code Name	Diamondville	Brand ID	
Package	Socket 437 FCBGA8		
Technology	45 nm	Core VID	1.275 V



Specification

Intel(R) Atom(TM) CPU N270 @ 1.60GHz			
Family	6	Model	C
		Stepping	2
Ext. Family	6	Ext. Model	1C
		Revision	C0
Instructions	MMX, SSE, SSE2, SSE3, SSSE3		

Clocks (Core #0)		Cache	
Core Speed	798.1 MHz	L1 Data	24 KBytes
Multiplier	x 6.0	L1 Inst.	32 KBytes
Bus Speed	133.0 MHz	Level 2	512 KBytes
Rated FSB	532.1 MHz	Level 3	

Selection Processor #1 Cores 1 Threads 2

Version 1.50

CPU-Z **OK**