

## BAB II

### DASAR TEORI

#### 2.1 Komputasi Komputer

Dengan menilik kembali sejarah ke belakang, komputasi sangat berkaitan erat dengan ilmuwan Galileo yang mengenalkan ide bahwa perhitungan matematis dapat digunakan untuk memprediksi pola dalam menjelaskan fenomena yang terjadi di dunia secara ilmiah. Dari ide inilah kemudian muncul penelitian yang lebih baik tentang cara menghitung trayektori suatu peluru meriam, pergerakan planet, dan lain sebagainya. Pada saat itu mungkin belum terpikirkan tentang pemanfaatan komputasi dalam merintis teknologi komputer, karena investasi waktu dan uang masih didominasi tentang pengembangan persenjataan.

Pengembangan komputer pada masa itu mulai terdorong ketika orang mulai berkomunikasi dengan orang lain melalui cara yang tidak dapat dipahami oleh kebanyakan masyarakat umum (kode rahasia, enkripsi). Tak lama kemudian, sebagian orang lain berkeinginan untuk memecahkan kode rahasia maupun enkripsi yang digunakan dalam berkomunikasi. Seiring perkembangan komputer yang tidak hanya berfungsi sebagai alat hitung, para pemecah kode rahasia menyadari bahwa komputer memiliki kemampuan untuk membuat deretan kode rahasia yang lebih aman sekaligus melakukan pemecahan kode sandi dengan lebih cepat. Ini dibuktikan pada sejarah pemecahan kode yang umum digunakan pada Perang Dunia II. Komunitas pemecah sandi inilah yang kemudian mendorong pengembangan komputer hingga mengarah seperti kondisi saat ini, sebagai perangkat tercepat untuk melaksanakan instruksi dengan penggunaan dalam jumlah besar.

##### 2.1.1 Definisi

Komputasi (Shackelford, Russell. 2006) dapat didefinisikan sebagai berikut : “Secara umum, komputasi dapat didefinisikan sebagai usaha untuk mencapai tujuan yang membutuhkan, memanfaatkan, atau membuat perangkat komputer. Oleh karena itu, komputasi dapat terdiri atas perancangan dan pembuatan sistem perangkat keras maupun lunak untuk tujuan yang luas; pemrosesan, penyusunan, dan pengelolaan beragam informasi; melakukan pembelajaran dengan komputer; membuat sistem komputer pintar; membuat dan menggunakan komunikasi dan media hiburan;

menemukan dan mengumpulkan informasi yang relevan dengan tujuan tertentu, dan lain-lain. Pemanfaatan komputasi memiliki kemungkinan sangat luas dengan metode penggunaan yang tidak terbatas.”

Namun terminologi komputasi masih bisa dipersempit kembali, seperti yang tertulis dalam (Comer, D.E. ,1989) menjadi : “Bidang ilmu komputasi adalah sistem pembelajaran mengenai proses algoritma yang menggambarkan dan mengubah suatu informasi : teori yang digunakan, analisis, perancangan, efisiensi, pelaksanaan, dan penerapan.”

### 2.1.2 Kinerja Komputer

Secara singkat, sebuah komputer terdiri dari atas beberapa elemen pokok sebagai berikut :

- a. Memory (*Storage*), yang dibagi kembali menjadi 2 yaitu memori internal dan memori eksternal. Memori internal mencakup RAM dan ROM, sedangkan memori eksternal adalah perangkat keras untuk melakukan operasi penulisan, pembacaan, dan penyimpanan data, di luar komponen RAM dan ROM.
  - 1) RAM (Random Access Memory) : Berfungsi untuk menyimpan informasi yang kita olah untuk sementara waktu. Contoh : EDORAM, SDRAM, DDRAM, dan RDRAM.
  - 2) ROM (Read Only Memory) : Memori yang hanya bisa dibaca dan berguna sebagai penyedia informasi pada saat komputer pertama kali dinyalakan. Pada komputer, ROM terdapat BIOS yang terletak pada *motherboard* yang berfungsi mensetting *peripheral* yang ada pada sistem. Informasi yang dapat tersimpan pada memori dapat berupa data seperti *character*, *integer*, *floating point number*, dan sebagainya, ataupun dapat berupa instruksi program untuk memanipulasi data.
- b. CPU merupakan otak sistem komputer yang memproses instruksi program yang tersimpan untuk memindah, menambah, mengurangi, mengalikan, membagi, dan membandingkan data. Elemen pokok yang terlibat di dalamnya adalah :
  - 1) *Control unit* untuk memproses instruksi program

- 2) *Logic unit* untuk membandingkan data. Hasil perbandingan bisa menghasilkan nilai benar dan salah, atau mengembalikan data ke *control unit* untuk diproses kembali dengan menjalankan instruksi tertentu jika kondisinya terpenuhi
  - 3) *Arithmetic unit* untuk melakukan operasi penambahan, perkalian, dan pembagian
  - 4) *Register* untuk menyimpan instruksi dan data yang sedang digunakan. Kelebihannya, instruksi dan data yang dimuat di dalamnya selalu tersedia untuk segera diakses oleh unit logika dan aritmatika. Jika menggunakan memori utama, membutuhkan banyak siklus clock untuk mencapai CPU
  - 5) *Cache* untuk menyimpan instruksi dan data yang segera akan digunakan. Instruksi dan data yang dimuat didalamnya juga memiliki kelebihan yang hampir sama dengan *register*
- c. *Peripheral Processor* untuk menangani operasi input dan output. Perangkat input dan output terhubung melalui *I/O port* untuk menerima ataupun mengirim data keluar sistem komputer
  - d. Bus untuk memindahkan data dan instruksi antar 3 elemen pokok di atas.
  - e. Clock adalah pendorong utama operasi yang melibatkan perangkat lainnya dalam CPU. Pulsa clock akan memicu terjadinya penterjemahan dan eksekusi instruksi. Kecepatan clock dihitung dalam satuan Hz, sehingga semakin besar nilainya, semakin banyak instruksi yang mampu diselesaikan dalam waktu singkat. Secara teori, komputer dengan kecepatan clock 1 GHz akan menghasilkan pulsa *clock* sebanyak 1.000.000.000 kali per detik. Idealnya jika mesin tersebut dirancang dan diprogram dengan tepat, akan ada 1.000.000.000 proses instruksi yang diselesaikan per detiknya.

## 2.1 Power Dissipation pada CPU

Power dissipation pada CPU adalah proses dimana CPU melakukan konsumsi daya listrik, mengubah bentuk energinya baik melalui aktivitas komponen switch di dalamnya (seperti transistor atau tabung vakum), maupun hilangnya energi dalam bentuk panas akibat impedansi komponen listrik.

*Power dissipation* biasanya terjadi pada komponen yang mengkonsumsi daya besar dimana sebagian energi akan dilepaskan dalam bentuk panas.

### 2.2.1 Sumber Konsumsi Daya

Ada beberapa faktor yang berperan dalam disipasi daya pada CPU ( $P_{cpu}$ ) (Chang, Naehyuck, 2006) yaitu : disipasi daya dinamis ( $P_{dyn}$ ), disipasi daya listrik *short-circuit* ( $P_{sc}$ ), dan hilangnya daya akibat kebocoran arus pada transistor ( $P_{leak}$ ) dan diukur dalam satuan Watt. Jika dirumuskan akan menjadi :

$$P_{cpu} = P_{dyn} + P_{sc} + P_{leak}$$

Konsumsi daya listrik dinamis ( $P_{dyn}$ ) terjadi pada saat kapasitansi total pada gerbang menyimpan dan melepaskan muatan (Geiger, 1990:597). Jika kapasitansi total gerbang termasuk kapasitansi parasitik dan kapasitor beban adalah C, dan nilai tegangan Vdd adalah V, maka energi yang diberikan pada kapasitor adalah :

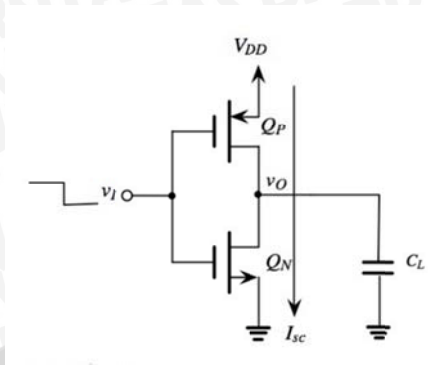
$$E = \int_0^{\infty} Vi(t)dt = CV^2$$

Karena energi total yang dapat diserap dan disimpan oleh kapasitor adalah  $\frac{1}{2}CV^2$  atau setengah dari jumlah energi harus dilepaskan. Ketika kapasitor dalam siklus melepas muatan, jumlah energi yang disimpan sebelumnya akan dilepaskan pula. Sehingga pada satu siklus, energi keseluruhan yang dilepas kapasitor adalah  $CV^2$ . Bila frekuensi operasi adalah f, maka disipasi daya rata-ratanya adalah :

$$P = CV^2 f$$

Sehingga disipasi daya listrik dinamis akan sebanding dengan kapasitansi total (F), kuadrat dari tegangan catu daya (V), dan frekuensi operasi (Hz atau  $s^{-1}$ ).

Tiap peralihan nilai logika pada gerbang logika, juga menyebabkan keadaan transistor di dalam CPU ikut berubah. Perubahan kondisi tersebut terjadi dalam waktu yang terbatas, sehingga pada suatu saat yang singkat akan ada beberapa transistor yang bekerja secara bersamaan. Pada saat kondisi tersebut dicapai, sumber tegangan  $V_{dd}$  dan  $V_{ss}$  komponen akan terhubung sehingga terbentuk jalur di antara keduanya. Inilah yang menyebabkan disipasi daya *short-circuit* ( $P_{sc}$ ).



**Gambar 2.1** Kondisi rangkaian disipasi daya *short-circuit*

Sumber : (Naehyuck Chang, 2006)

Hilangnya daya akibat kebocoran arus ( $P_{\text{leak}}$ ) terjadi karena sejumlah arus jejak yang terus mengalir antara bagian sumber dan pembuangan pada transistor pada saat tegangan dihilangkan. Arus bocor yang tidak diinginkan ini akan menghabiskan daya dan bahkan dapat menjadi masalah karena transistor mengecil dan banyak. Oleh karena lapisan inversi (konduktor dalam sebuah transistor) mengecil, jumlah arus yang dapat dihantarkan juga berkurang hingga ke tingkat arus bocor, sehingga akan sulit membedakan kondisi transistor “hidup” dan “mati”, yang berarti tujuan utama transistor, mengirimkan input dan output biner menjadi hilang.

## 2.2 Kemampuan Komputasi

FLOPS adalah singkatan dari istilah dalam bahasa Inggris yaitu *floating point operations per second* yang merujuk pada satuan untuk jumlah operasi perhitungan yang dapat dilakukan oleh sebuah perangkat komputasi (dalam hal ini adalah komputer) terhadap bilangan nyata (*floating point*) tiap satu satuan waktu. FLOPS merupakan satuan pengukuran kecepatan kinerja suatu mikroprosesor biasanya dalam suatu perhitungan ilmiah (*scientific calculation*), yang sangat berguna dalam menyatakan hasil perhitungan yang sangat besar atau sangat kecil.

Terkadang masih banyak orang yang salah mengartikan bahwa FLOPS dapat mewakili bentuk *singular* (satu) maupun *plural* (banyak). Untuk membedakannya, disepakati bahwa singkatan FLOP dapat mewakili bentuk *singular* (satu), sedangkan FLOPS dapat diartikan sebagai  $f_{\text{flop}} = 1 \text{ s}^{-1} \Leftrightarrow n_{\text{flops}} = 1$  yang lebih bisa diartikan sebagai bentuk *plural* (banyak).

Secara teoritis, FLOPS sebuah CPU dapat dihitung dengan menggunakan rumus :

$$\text{FLOPS} = \text{cores} \times \text{clock} \times \frac{\text{FLOPs}}{\text{cycle}}$$

dimana total jumlah flop per detik yang dapat dihasilkan prosesor adalah FLOPS, jumlah *core* inti prosesor adalah *cores*, nilai frekuensi CPU adalah *clock*, jumlah operasi *floating point* adalah FLOPs, dan siklus clock adalah *cycle*.

Jika pada saat ini kebanyakan prosesor dapat menghasilkan 4 FLOPs per *cycle*, maka prosesor dengan *clock* sebesar 2,5 GHz akan dapat dihasilkan 10 giga FLOPS = 10 GFLOPS.

Dengan mengetahui kemampuan puncak sebuah prosesor dalam menghasilkan FLOPS dan membandingkan FLOPS yang dihasilkan dari pengguna aplikasi/perangkat lunak yang sering digunakan, maka akan diperoleh suatu informasi berharga untuk menentukan apakah kinerja yang dicapai oleh komputer tersebut sudah optimal.

### 2.3.1 Daya Komputasi

Kinerja dan konsumsi daya listrik suatu komputer dapat diukur dari berbagai hal tergantung tujuannya. Pengukuran kinerja umumnya melibatkan pengukuran pada nilai FLOPS, MIPS, atau hasil dari *benchmark software*. Untuk pengukuran konsumsi daya listrik, ada yang melakukan pengukuran pada suatu komponen komputer, sebagian yang lain mempertimbangkan keseluruhan daya yang diperlukan untuk menjalankan komputer seperti pendinginan dan sistem pemantauan.

**Tabel 2.1 Tabel performansi prosesor Intel**

Sumber : (Intel Corporation)

Jenis Prosesor	Clock (Ghz)	PEAK GFLOPS	IDLE POWER (W)	PEAK POWER (W)	TDP (W)	Jenis Prosesor
i7-975	3,33	53,328	83	205	130	CORE i7
i7-980X	3,33	79,992	79	194	130	CORE i7
i7-3960X	3,3	158,4	57	210	130	CORE i7
i7-870	2,93	46,928	67	181	95	CORE i7
i7-875K	2,93	46,928	68	176	95	CORE i7
i7-880	3,06	49,056	68	183	95	CORE i7
i7-920	2,66	42,656	78	174	130	CORE i7
i7-930	2,8	44,8	78	188	130	CORE i7

Jenis Prosesor	Clock (Ghz)	PEAK GFLOPS	IDLE POWER (W)	PEAK POWER (W)	TDP (W)	Jenis Prosesor
i7-960	3,2	51,2	78	198	130	CORE i7
i-970	3,2	76,8	79	198	130	CORE i7
i7-3770K	3,5	112	47	144	77	CORE i7
i7-2700K	3,5	112	47	135	95	CORE i7
i7-2600K	3,4	108,8	46	129	95	CORE i7
i5-661	3,33	26,664	68	122	87	CORE i5
i5-670	3,46	27,728	68	121	73	CORE i5
i5-750S	2,4	38,4	68	135	82	CORE i5
i5-760	2,8	44,8	67	153	95	CORE i5
i3-530	2,93	23,464	67	110	73	CORE i3
E8400	3	24	131	158	65	CORE 2
540	3	6	150	223	84	PENTIUM 4
840	3,2	12,8	165	240	130	PENTIUM D
940	3,2	12,8	168	253	130/95	PENTIUM D
E6600	2,4	19,2	117	156	65	PENTIUM DUAL CORE
E6700	2,66	21,328	123	163	65	PENTIUM DUAL CORE

Tabel 2.2 Tabel Performansi Prosesor AMD

Sumber : (AMD Corporation)

Jenis Prosesor	Clock GHz	PEAK GFLOPS	IDLE POWER (W)	PEAK POWER (W)	TDP	JENIS PROSESOR
A8-3870K	3	35,45	38	134	100	AMD A-SERIES
A10-5800K	3,8	30,4	44	125	100	AMD A-SERIES
Phenom II X6 1100T	3,3	62,67	69	182	125	AMD PHENOM
Phenom II X6 1055T	2,8	59,25	69	164	95	AMD PHENOM
Phenom II X4 965	3,4	24,1	64	171	125	AMD PHENOM
Phenom II X4 980	3,7	11,7	63	170	125	AMD PHENOM
Phenom II X4 840	3,2	10,9	53	142	95	AMD PHENOM
Phenom II X3 720	2,8	7,99	62	123	106,5	AMD PHENOM
Phenom II X2 550	3,1	9,95	63	166	94	AMD PHENOM

Jenis Prosesor	Clock GHz	PEAK GFLOPS	IDLE POWER (W)	PEAK POWER (W)	TDP	JENIS PROSESOR
Phenom II X2 555	3,2	5,84	63	121	80	AMD PHENOM
Phenom II 955	3,8	49,247	148	236	125	AMD PHENOM
Phenom II 720	3,6	34,336	155	213	95	AMD PHENOM
Athlon II X4 620	2,6	34,466	59	138	95	AMD ATHLON
Athlon II X4 645	3,1	41	54	136	95	AMD ATHLON
Athlon II X4 630	2,8	18	61	138	95	AMD ATHLON
Athlon II X3 445	3,1	8,23	55	123	95	AMD ATHLON
Athlon II X3 435	2,9	13,9	59	120	95	AMD ATHLON
Athlon II X2 250	3	10,1	56	101	65	AMD ATHLON
Athlon II X2 260	3,2	6,39	52	106	65	AMD ATHLON
Athlon II X2 240e	2,8	7,17	55	96	45	AMD ATHLON
FX 8120	3,1	37,5	64	188	125	AMD FX SERIES
FX 8150	3,6	32,4	63	193	125	AMD FX SERIES
FX 8350	4	28,3	61	210	125	AMD FX SERIES
FX 6100	3,3	26,9	62	145	95	AMD FX SERIES
AMD Athlon 3700+	2,21	2,9802	120	148	89	AMD ATHLON

Keterangan :

- GFLOPS = *flops* dalam orde giga
- Nilai IDLE POWER dihasilkan dari pengukuran langsung pada colokan saat komputer dalam kondisi idle.
- Nilai PEAK POWER dihasilkan melalui hasil pengujian software benchmarking Prime95.



- Nilai TDP adalah nilai energi panas yang dilepaskan oleh CPU tanpa merusak komponen yang ada di dalamnya. TDP dapat menjadi acuan daya yang dibutuhkan dalam kondisi penggunaan secara normal (sehari-hari).

Daya komputasi adalah jumlah konsumsi daya yang dibutuhkan oleh prosesor untuk menghasilkan sejumlah nilai komputasi, dan diukur dalam satuan watt/FLOPS.

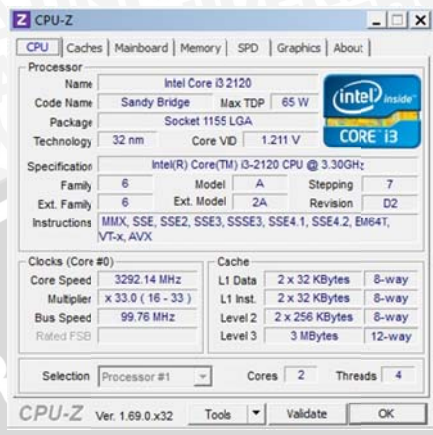
Berdasarkan tabel 2.1 dan 2.2 di atas yang diperoleh dari [www.intel.com](http://www.intel.com) dan berbagai sumber hasil benchmarking pada bulan Maret 2014, dapat dilihat adanya kecenderungan bahwa prosesor dengan versi yang lebih baru akan menghasilkan nilai komputasi (FLOPS) yang lebih tinggi daripada versi yang dirilis sebelumnya untuk masing-masing prosesor Intel dan AMD. Namun tentunya, komputasi yang semakin hebat dalam menghasilkan nilai FLOPS yang lebih tinggi, akan menyebabkan meningkatnya konsumsi daya listrik yang dibutuhkan oleh prosesor. Namun di sisi lain, peningkatan konsumsi yang semakin besar juga diiringi dengan efisiensi komputasi yang dapat dihitung dari jumlah daya listrik yang dibutuhkan untuk menghasilkan tiap FLOPS (watt/FLOPS). Sebagai contoh pada tabel tersebut, meskipun Intel i7-905 membutuhkan daya listrik yang lebih daripada i5-760, namun jika dihitung nilai efisiensi daya komputasinya maka Intel i7-905 akan menghasilkan nilai yang lebih tinggi daripada Intel i5-760.

Pada tabel 2.1 dan 2.2 dapat diamati pula bahwa komputer dalam kondisi tidak bekerja (*idle*), tetap membutuhkan konsumsi daya listrik yang cukup besar. Jika diasumsikan bahwa ketika komputer tidak bekerja, nilai FLOPS yang dihasilkan sama dengan nol karena prosesor tidak melakukan proses komputasi, maka akan terjadi inefisiensi daya komputasi yang besar karena nilai efisiensi yang tidak terbatas.

Terlebih lagi, penggunaan komputer saat ini masih belum mengarah untuk menghasilkan kinerja komputer yang optimal dimana nilai efisiensi komputasinya mendekati nilai maksimal. Nilai efisiensi komputasi yang optimal dapat dicapai dengan difungsikannya komputer hingga nilai FLOPS yang dihasilkan mendekati kemampuan maksimal yang dapat dicapai prosesor tersebut, sekaligus mendayagunakan *resource* secara maksimal.

Untuk mendukung hipotesis di atas, dilakukan pengumpulan data pada sebuah lingkungan perkantoran dimana pegawai yang bekerja di dalamnya merupakan pengguna komputer dan kesehariannya bekerja dengan menggunakan aplikasi perkantoran. Pengumpulan data tersebut dilakukan dengan cara *software benchmarking*

untuk mendapatkan nilai kemampuan komputasi yang dimiliki, spesifikasi prosesor yang digunakan, dan penggunaan *resource* memori dari penggunaan aplikasi perkantoran yang umum dipakai berupa Ms Office dan Open Office.



**Gambar 2.2** Tampilan spesifikasi prosesor pada software benchmarking CPU-Z

Sumber : Pengujian

**Tabel 2.3** Tabel kebutuhan memori pada penggunaan aplikasi perkantoran

Sumber : (George Ou, 2005)

	Memory usage				CPU time		
	Private bytes (MB)	Private peak bytes (MB)	Working set (MB)	Working set peak (MB)	Kernel time milliseconds	User time milliseconds	Total time milliseconds
Word	4.85	5.35	10.42	10.84	62	109	171
Writer	19.20	19.20	35.64	35.64	765	1156	1921
Excel	3.71	3.71	7.61	7.61	93	47	140
Calc	18.89	18.89	35.02	35.02	484	1172	1656
PowerPoint	2.59	2.59	7.52	7.52	125	78	203
Impress	24.65	24.65	42.39	42.39	890	1422	2312
Access	2.66	2.66	7.28	7.28	171	125	296
Base	14.51	14.51	27.95	27.95	546	860	1406

**Tabel 2.4** Hipotesis komputasi komputer yang dihasilkan pada penggunaan aplikasi perkantoran

Nama Aplikasi	Memori (MB)	GFLOPS LinX			
		GFLOPS terkecil	Waktu (s)	GFLOPS terbesar	Waktu (s)
MS Word (private)	5	5,7978	0,036	22,4417	0,009
MS Word (working)	11	13,5425	0,065	22,4836	0,039
Open Writer (private)	20	19,4008	0,113	22,2227	0,099
Open Writer (working)	35	25,0668	0,234	25,8834	0,226
MS Excel (private)	4	2,6984	0,048	22,4284	0,006
MS Excel (working)	8	8,9923	0,056	22,3027	0,023
Open Calc (private)	20	15,9347	0,138	22,2660	0,099
Open Calc (working)	35	20,8294	0,281	26,0248	0,225
MS Power Point (private)	4	1,9090	0,044	21,5067	0,004
MS Power Point (working)	7	13,5002	0,029	22,5560	0,018
Open Impress (private)	25	23,5446	0,146	24,4852	0,140
Open Impress (working)	42	25,1816	0,310	26,6517	0,293
MS Access (private)	4	4,1846	0,020	21,6889	0,004
MS Access (working)	7	13,2415	0,030	22,1044	0,018
Open Base (private)	14	21,5726	0,062	22,2911	0,06
Open Base (working)	28	19,9522	0,206	24,8222	0,166
<b>Maximum Stress Level (1)</b>	<b>1270</b>	<b>31,9363</b>	<b>44,283</b>	<b>33,6667</b>	<b>42,007</b>
<b>Maximum Stress Level (2)</b>	<b>1434</b>	<b>30,7197</b>	<b>55,279</b>	<b>31,7811</b>	<b>53,433</b>

Sumber : pengujian

Dari gambar 2.2 yang merupakan keluaran dari CPU-Z, terlihat bahwa software benchmarking dilakukan pada komputer dengan jenis prosesor Intel Core i3 2120 dengan kecepatan clock 3,3 GHz. Spesifikasi prosesor tersebut adalah yang umum digunakan di lingkungan perkantoran tersebut.

Dengan mengamati tabel 2.3 yang didapat dari pengamatan kebutuhan resource software perkantoran yaitu MS Office/Open Office dengan menggunakan Process Explorer milik SysInternals, kebutuhan memori pada penggunaan aplikasi perkantoran masih sangat kecil jika dibandingkan dengan resource memori maksimal yang terpasang pada CPU yaitu sebesar 2 GB.

Kemudian pada tabel 2.4 yang dihasilkan dengan cara menguji nilai komputasi pada tabel 2.3 dengan software benchmarking LinX, kemampuan komputasi yang dihasilkan pada penggunaan aplikasi perkantoran tersebut masih belum mencapai kemampuan maksimal jika dibandingkan pada kondisi *maximum stress level*.

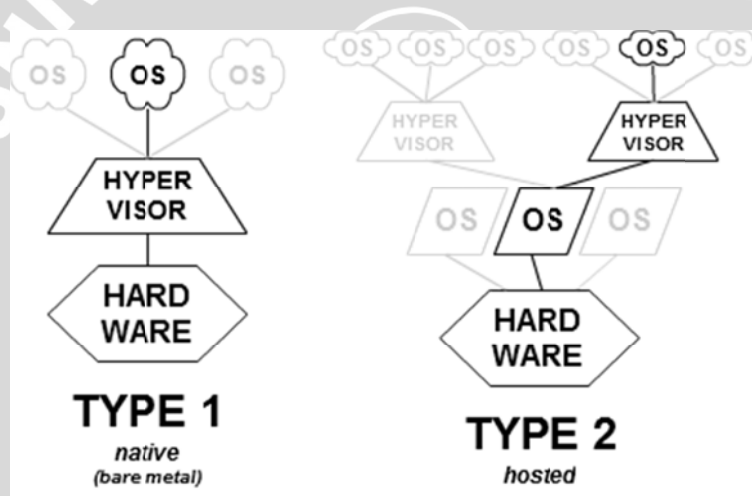
Dengan mengacu pada data pendukung hipotesis yang telah disusun diatas, dimana dihasilkan jumlah penggunaan *resource* maupun kemampuan komputasi yang masih di bawah kemampuan maksimal yang dapat dikerjakan oleh prosesor tersebut, maka optimalisasi proses komputasi dalam prosesor sangat mungkin dilakukan.



## 2.3 Hypervisor

Hypervisor adalah teknik virtualisasi yang memiliki kemampuan untuk membuat dan menjalankan mesin virtual. Komputer yang menjalankan hypervisor dimana di dalamnya terdapat mesin virtual, disebut dengan *host machine*. *Guest machine* adalah mesin virtual yang telah dibuat melalui hypervisor. Hypervisor dapat memungkinkan beberapa *guest machine* dengan sistem operasi yang berbeda untuk beroperasi dalam waktu yang sama pada sebuah *host machine*. Hypervisor juga memiliki tugas untuk mengalokasikan *memory*, *processor*, dan sumber daya yang digunakan oleh masing-masing mesin virtual agar kinerjanya tidak saling terganggu. Oleh karena itu, hypervisor juga sering disebut sebagai *virtual machine management (VMM)*.

Hypervisor dapat diklasifikasikan ke dalam dua tipe yaitu :



**Gambar 2.5** Tipe-tipe hypervisor

Sumber : (Gerald J. Popek.1974)

### a. Hypervisor tipe 1

Hypervisor ini biasa disebut dengan *bare metal OS*, karena hypervisor diinstall secara langsung pada perangkat keras untuk menangani mesin virtual, bukan pada sistem operasi *host machine*. Contoh teknologi yang menerapkan hypervisor *bare metal OS* adalah VMWare ESXi, Citrix XenServer, dan Microsoft Hyper-V hypervisor. Hypervisor ini memiliki kelebihan dalam akses *resources* perangkat keras dan efisiensi yang memungkinkan ketahanan, kinerja, dan kemampuan menangani proses dalam skala yang lebih besar daripada hypervisor tipe 2.

## b. Hypervisor tipe 2

Hypervisor tipe 2 adalah kebalikan dari hypervisor tipe 1 dimana hypervisor yang memuat mesin virtual dijalankan secara konvensional pada sistem operasi *host machine*. Contoh hypervisor ini adalah VirtualBox, VMWare Workstation, VMWare Server atau KVM, dan Xen Hypervisor yang berjalan diatas sistem operasi Linux seperti Ubuntu, CentOS, SLES (SUSE Linux Enterprise Server), RHEL (Red Hat Enterprise Linux), dan lain-lain.

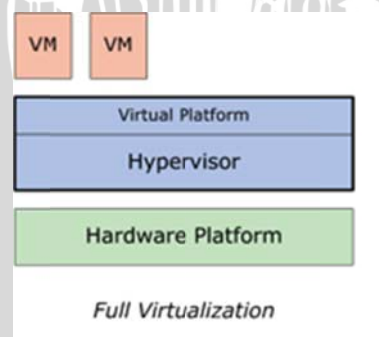
Hypervisor ini memiliki kelebihan dalam hal kemudahan pengelolaan baik duplikasi, perawatan, dan simulasi pada mesin virtual.

## 2.4 Virtualisasi

Virtualisasi (dalam komputer) merujuk pada teknik untuk membuat sesuatu ke dalam bentuk maya, tidak seperti kenyataan yang ada. Umumnya virtualisasi dapat digunakan untuk mengemulasikan perangkat fisik, baik itu *hardware* komputer, sistem operasi, perangkat penyimpanan, dan sumber daya jaringan komputer, dengan cara membuatnya seolah-olah perangkat tersebut tidak ada (disembunyikan).

Virtualisasi memiliki cakupan yang luas, sehingga pendekatan virtualisasi yang akan dibahas hanya seputar metode virtualisasi pada perangkat keras yaitu pada pendekatan *full virtualization* dan *paravirtualization*.

### 2.5.1 Full Virtualization



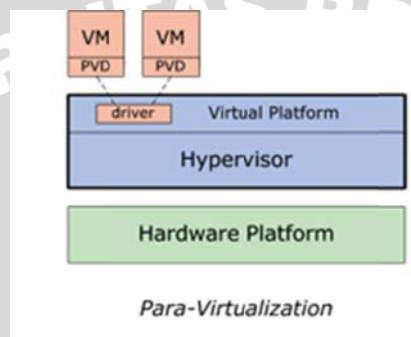
**Gambar 2.6** Metode full virtualization

Sumber : (George Heeter, 2014)

Seperti yang terlihat pada gambar 2.6 , metode *full virtualization* memberikan kemampuan kepada *hypervisor* untuk menyimulasikan hampir keseluruhan kinerja

sebuah perangkat keras pada nyatanya. Sistem operasi yang berjalan pada mesin virtual yang telah dibuat *hypervisor* tidak perlu mengalami modifikasi, sehingga hampir keseluruhan aplikasi dapat dijalankan selama sistem operasi yang disimulasikan memiliki kompatibilitas terhadap aplikasi tersebut. Mesin virtual yang telah dibuat sifatnya independen terhadap mesin virtual lain dan tidak menyadari keberadaan satu sama lain. Mesin virtual ini juga berjalan dalam sistem operasi masing-masing yang telah ditentukan, sehingga dimungkinkan antar mesin virtual berjalan dalam sistem operasi yang berbeda. (*White Paper*, 2006)

### 2.5.2 Paravirtualization



**Gambar 2.5** Metode *paravirtualization*

Sumber : (George Heeter, 2014)

Metode *paravirtualization* seperti yang terlihat pada gambar 2.5, tidak menyimulasikan keseluruhan perangkat keras dan sehingga *hypervisor* memiliki keterbatasan untuk menyimulasikan sistem operasi dan aplikasi yang berjalan di dalamnya. Agar aplikasi di dalam mesin virtual dapat berjalan, sistem operasi *guest* yang disimulasikan harus dimodifikasi terlebih dahulu untuk mendukung kompatibilitasnya. Antara mesin virtual yang satu dengan yang lain, memiliki kesadaran akan keberadaan masing-masing bahwa sedang divirtualisasikan dan terhubung melalui *paravirtualization driver*. Driver tersebut yang menjembatani komunikasi mesin virtual dalam mengakses *resources* perangkat fisik melalui *hypervisor*.

Pada beberapa penggunaan aplikasi, tingkat efisiensi kinerja metode ini lebih tinggi daripada metode *full virtualization* karena akses ke perangkat fisik utama (*host machine*) telah dialokasikan secara tepat oleh *hypervisor*. (*White Paper*, 2006)

