



Department of Digital Business

Journal of Artificial Intelligence and Digital Business (RIGGS)

Homepage: <https://journal.ilmudata.co.id/index.php/RIGGS>

Vol. 5 No. 2 (2026) pp: 13239-13246

P-ISSN: 2963-9298, e-ISSN: 2963-914X

Analisis Peran Arsitektur Komputer terhadap Kinerja Sistem Komputasi Modern

Saffana Qolbi Atma Widi, Muhammad Daniar Firdaus, Muhammad Raiska Fahrezi, Much Aziz Muslim

Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Semarang, Indonesia

saffanaqolbi2007@gmail.com, mdaniarf@gmail.com, raiskafahrezi@gmail.com, a212muslim@mail.unnes.ac.id

Abstrak

Arsitektur komputer merupakan fondasi utama dalam sistem komputasi modern yang menentukan bagaimana komponen perangkat keras saling berinteraksi untuk mengeksekusi instruksi secara efisien. Saat ini, perkembangan teknologi seperti prosesor multi-core, komputasi paralel, dan optimasi hierarki memori telah meningkatkan kinerja sistem secara signifikan dalam menghadapi keterbatasan fisik perangkat keras. Penelitian ini bertujuan untuk menganalisis pengaruh arsitektur komputer terhadap kinerja sistem komputasi modern secara komprehensif. Metode yang digunakan adalah studi literatur dengan pendekatan deskriptif-komparatif yang mengkaji berbagai jurnal ilmiah dan buku teks akademik pada rentang tahun 2021–2026. Hasil penelitian menunjukkan bahwa arsitektur komputer berpengaruh besar terhadap kinerja sistem melalui desain prosesor, memori, dan teknik paralel. Optimalisasi pada desain internal CPU melalui mekanisme pipelining dan out-of-order execution berhasil mendongkrak throughput instruksi. Pada sektor penyimpanan, manajemen hierarki memori cache yang adaptif terbukti efektif mereduksi latensi akses data sekaligus mengurai problem kesenjangan kecepatan yang dikenal sebagai memory wall. Lebih lanjut, pemanfaatan komputasi paralel heterogen yang mengintegrasikan unit CPU, GPU, dan akselerator kecerdasan buatan mampu memberikan lompatan performa yang masif serta efisiensi energi yang tinggi demi menyokong ekosistem cloud computing dan big data processing. Dapat disimpulkan bahwa inovasi rancang bangun arsitektur perangkat keras yang adaptif terhadap karakteristik beban kerja modern menjadi parameter penentu utama bagi efektivitas, skalabilitas, dan keberlanjutan performa sistem komputasi masa depan.

Kata kunci: Arsitektur Komputer, Kinerja Sistem Komputasi, Prosesor Multi-Core, Hierarki Memori, Komputasi Paralel, Optimasi Sistem, Perangkat Keras Komputer, Efisiensi Komputasi

1. Pendahuluan

Perkembangan teknologi komputer modern dalam satu dekade terakhir telah mengubah lanskap industri digital dan cara sistem komputasi dirancang, dioptimalkan, serta diimplementasikan. Pada era awal komputasi, peningkatan performa sebuah sistem sangat bergantung pada peningkatan frekuensi *clock speed* prosesor secara linear. Namun, saat ini industri perangkat keras telah membentur dinding pembatas fisik yang dikenal sebagai *Power Wall* dan *Thermal Wall*, di mana peningkatan *clock speed* secara terus-menerus tidak lagi memungkinkan karena konsumsi daya yang eksponensial dan disipasi panas yang ekstrem [1]. Oleh karena itu, arsitektur komputer bergeser menjadi fondasi paling krusial yang mengatur bagaimana hubungan struktural dan fungsional antara CPU, hierarki memori, dan perangkat input-output dikelola demi mencapai efisiensi maksimum [1, 3].

Selain kendala fisik berupa *Power Wall*, arsitektur komputer klasik juga terbentur oleh batasan teoritis yang dikenal sebagai *Von Neumann Bottleneck*. Hambatan ini terjadi karena pemisahan fisik antara unit pemroses (CPU) dan unit penyimpanan memori utama, di mana data dan instruksi harus melewati jalur bus interkoneksi yang sama secara bergantian. Seiring dengan peningkatan performa CPU yang jauh lebih pesat daripada kecepatan bus memori, saluran komunikasi ini menjadi sangat padat, memaksa prosesor modern berada dalam kondisi *stall* (menunggu suplai data) selama jutaan siklus kerja. Oleh karena itu, perombakan arsitektur melalui pendekatan *Non-Von Neumann*, seperti teknik komputasi di dalam memori (*Processing-In-Memory* atau PIM), mulai diadopsi guna meminimalkan perpindahan data jarak jauh dan mengoptimalkan efisiensi pemrosesan secara radikal.

Secara lebih spesifik, fenomena *Power Wall* dan *Thermal Wall* muncul akibat ketidakmampuan material semikonduktor berbasis silikon untuk mengatasi arus bocor (*leakage current*) ketika ukuran transistor diperkecil hingga skala beberapa nanometer. Ketika frekuensi kerja prosesor dipaksakan melampaui batas optimalnya,

konsumsi daya dinamis akan meningkat secara kuadratik mengikuti rumus dasar disipasi daya fisik. Akibatnya, panas yang dihasilkan oleh kepadatan komponen di atas cip silikon melonjak drastis hingga mencapai titik kritis yang dapat merusak sirkuit itu sendiri. Batasan fisik inilah yang memaksa para arsitek komputer untuk mengalihkan fokus mereka dari peningkatan kecepatan mentah (*raw speed*) menuju perancangan struktur arsitektur mikro yang jauh lebih cerdas, efisien, dan adaptif terhadap variasi beban kerja modern.

Seiring dengan meredupnya hukum Moore (*Moore's Law*), fokus utama dari pengembangan arsitektur modern tidak lagi terletak pada seberapa banyak transistor yang dapat dipadatkan dalam satu inti tunggal (*single-core*). Paradigma baru kini berpusat pada efisiensi energi (*energy efficiency*), eksploitasi paralelisme pada tingkat instruksi maupun data (*parallelism*), serta optimasi arsitektur memori yang adaptif [3]. Salah satu tantangan terbesar yang dihadapi oleh perancang sistem komputasi modern adalah *Memory Wall*, sebuah kondisi di mana kesenjangan kecepatan operasi antara prosesor yang sangat cepat dan memori utama (DRAM) yang relatif lambat menjadi hambatan utama (*bottleneck*) dalam pemrosesan data [5]. Tanpa adanya desain arsitektur memori cache yang cerdas, performa prosesor mutakhir akan terbuang sia-sia karena terlalu lama menunggu instruksi disuplai dari memori utama.

Dalam konteks aplikasi kontemporer, peran arsitektur komputer menjadi jauh lebih vital. Teknologi-teknologi berbasis data intensif seperti komputasi awan (*cloud computing*), kecerdasan buatan (*artificial intelligence / AI*), pembelajaran mesin (*machine learning*), serta pemrosesan data skala besar (*big data processing*) membutuhkan infrastruktur perangkat keras yang tidak hanya cepat, tetapi juga fleksibel [7]. Sebagai contoh, pemrosesan model AI dengan parameter berskala miliaran tidak akan efisien jika hanya mengandalkan arsitektur CPU tradisional, melainkan membutuhkan akselerator khusus. Oleh karena itu, pemahaman, analisis, dan inovasi pada sektor arsitektur komputer mutlak diperlukan untuk memastikan keberlanjutan performa sistem komputasi modern di masa depan. Berdasarkan latar belakang tersebut, penelitian ini bertujuan untuk menguraikan secara komprehensif bagaimana komponen-komponen arsitektur modern berinteraksi dan mengidentifikasi strategi optimal dalam mendesain sistem komputasi yang efisien dan berdaya guna tinggi.

Selain masalah kecepatan proses, optimalisasi arsitektur komputer saat ini juga berdampak langsung pada efisiensi biaya operasional pusat data modern. Dengan kebutuhan komputasi yang semakin masif, desain arsitektur yang hemat energi dapat menekan pengeluaran daya tanpa mengorbankan kemampuan pemrosesan data. Oleh karena itu, analisis dalam penelitian ini tidak hanya melihat aspek performa teknis semata, melainkan juga bagaimana rancangan arsitektur komputer modern menjadi solusi praktis dalam menghadapi tantangan efisiensi di era digital yang kian berkembang pesat.

2. Metode Penelitian

Penelitian ini dilaksanakan dengan menggunakan metode studi literatur secara sistematis (*systematic literature review*) untuk mengkaji, menganalisis, dan menyarikan berbagai publikasi ilmiah, buku teks standar, serta jurnal bereputasi yang berkaitan erat dengan arsitektur komputer dan performa sistem [4]. Proses pencarian literatur difokuskan secara ketat pada artikel-artikel yang diterbitkan dalam rentang waktu tahun 2021 hingga 2026. Pembatasan waktu ini sengaja dilakukan agar seluruh data, tren teknologi, dan analisis yang disajikan dalam artikel ini tetap relevan dengan perkembangan arsitektur perangkat keras modern paling mutakhir.

Strategi pengumpulan data dilakukan melalui penelusuran pangkalan data ilmiah digital terkemuka, seperti IEEE Xplore, Google Scholar, ACM Digital Library, dan Scopus. Kata kunci (*keywords*) yang digunakan dalam proses pencarian meliputi "*Computer Architecture*", "*Processor Performance*", "*Memory Wall Optimization*", "*Parallel Computing*", dan "*Heterogeneous Computing*". Dari hasil pencarian awal, dilakukan seleksi ketat menggunakan kriteria inklusi dan eksklusi. Kriteria inklusi mengharuskan artikel membahas dampak langsung arsitektur terhadap performa sistem komputasi, sementara kriteria eksklusi memfilter artikel yang hanya membahas perangkat lunak murni tanpa keterkaitan dengan karakteristik *hardware*.

Tahapan penelitian dibagi menjadi empat:

1. Identifikasi Literatur: Mengumpulkan draf jurnal dan pustaka awal yang relevan dengan ekosistem arsitektur komputer modern.

2. Seleksi Sumber: Menyaring literatur berdasarkan tingkat kredibilitas penerbit, metodologi yang digunakan oleh peneliti terdahulu, serta relevansinya terhadap isu *bottleneck* performa [4].
3. Analisis Komponen Utama: Membedah elemen-elemen kritis dalam arsitektur komputer, yang meliputi desain internal CPU, struktur hierarki memori, mekanisme interkoneksi, dan modul akselerasi paralel.
4. Sintesis Hasil Penelitian: Menghubungkan berbagai temuan dari literatur yang berbeda guna memetakan pola hubungan yang logis dan kausal antara variasi desain arsitektur dan efisiensi kinerja komputasi yang dihasilkan.

Metode analisis data yang diterapkan adalah pendekatan deskriptif-komparatif. Melalui pendekatan ini, peneliti membandingkan data performa dari berbagai model arsitektur—misalnya arsitektur berbasis *multi-core* konvensional dengan arsitektur heterogen (*heterogeneous computing*)—untuk menarik sebuah kesimpulan umum yang objektif mengenai arah evolusi arsitektur komputer.

Dalam proses seleksi sumber, draf atau artikel yang tidak memberikan data hasil pengujian perangkat keras yang jelas atau hanya membahas perangkat lunak murni sengaja dieliminasi. Hal ini dilakukan untuk memastikan bahwa seluruh referensi yang dipilih benar-benar fokus pada hubungan langsung antara komponen fisik komputer dan dampaknya terhadap kecepatan sistem. Melalui langkah penapisan yang sederhana namun ketat ini, data yang terkumpul diharapkan dapat memberikan kesimpulan yang valid, objektif, dan tepercaya.

3. Hasil dan Pembahasan

Pada bagian ini dilakukan pembahasan mendalam mengenai hasil analisis pengaruh nyata arsitektur komputer terhadap kinerja sistem komputasi modern. Pembahasan difokuskan pada peran fungsional dan struktural dari komponen-komponen utama arsitektur komputer, seperti unit pemroses sentral (prosesor), hierarki memori, sistem interkoneksi, serta teknologi komputasi paralel. Analisis ini juga menguraikan bagaimana sinergi antar-elemen tersebut mampu mendorong efisiensi dan kecepatan sistem, sekaligus mengevaluasi implementasi arsitektur modern dalam menjawab kebutuhan komputasi skala besar pada sektor *cloud computing*, kecerdasan buatan (AI), dan analisis data raya (*big data analytics*).

Kebutuhan spesifik dari ketiga sektor kontemporer tersebut berpusat pada penanganan volume data yang sangat masif (*high data volume*) disertai tuntutan kecepatan pemrosesan tingkat tinggi (*high velocity processing*). Sebagai contoh, pada implementasi AI dan pembelajaran mesin, pelatihan model dengan parameter berskala miliaran memerlukan kalkulasi matriks simultan yang akan memicu *bottleneck* parah pada jalur komunikasi memori jika hanya dipaksakan menggunakan arsitektur CPU klasik. Sementara itu, dalam ekosistem *cloud computing* dan analisis data raya, efisiensi energi serta kepadatan komputasi menjadi parameter yang paling kritis demi meminimalkan disipasi panas pada pusat data operasional. Oleh karena itu, analisis pada bab ini akan menguraikan bagaimana pergeseran ke arah unit pemrosesan khusus (*domain-specific accelerators*) dan pembenahan jalur interkoneksi menjadi solusi mutlak dalam memikul beban kerja digital masa kini..

3.1. Evolusi Arsitektur Komputer Modern

Dalam beberapa dekade terakhir, arsitektur komputer telah mengalami pergeseran paradigma yang sangat radikal. Paradigma lama yang mengandalkan satu inti prosesor (*single-core*) dengan frekuensi tinggi telah ditinggalkan akibat kendala disipasi panas yang destruktif. Sebagai gantinya, arsitektur komputer berevolusi menuju pemrosesan multi-inti (*multi-core*), banyak-inti (*many-core*), hingga yang paling mutakhir adalah arsitektur komputasi heterogen (*heterogeneous computing*) [3].

Tabel 1. Evolusi Arsitektur Komputer dari Single-Core hingga Heterogeneous Computing dan Dampaknya terhadap Kinerja Sistem

Generasi	Karakteristik Utama	Teknologi	Dampak pada Performa
Single-core	Satu inti prosesor	CPU tunggal	Performa terbatas
Multi-core	Beberapa inti CPU	Dual-core, Quad-core	Paralelisme meningkat

DOI: <https://doi.org/10.31004/riggs.v5i2.10537>

Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

Many-core	Banyak inti	GPU, Xeon Phi	Komputasi besar lebih cepat
Heterogeneous	CPU + GPU + TPU	AI accelerator	Efisiensi tinggi & AI-ready

Berdasarkan data dan klasifikasi yang disajikan pada Tabel 1, terlihat jelas bahwa evolusi arsitektur komputer digerakkan oleh kebutuhan untuk melompati batasan-batasan fisik perangkat keras. Transisi dari *single-core* ke *multi-core* berhasil menyelesaikan masalah efisiensi daya dengan membagi beban kerja ke beberapa inti yang berjalan pada frekuensi lebih rendah namun menghasilkan *throughput* yang sama atau lebih tinggi [8]. Selanjutnya, munculnya komputasi heterogen menandai era baru di mana tugas-tugas komputasi tidak lagi dipaksakan untuk diselesaikan oleh CPU general-purpose saja. Integrasi antara CPU yang andal dalam eksekusi serial cepat, bersama GPU yang superior dalam pemrosesan paralel masif, serta TPU (Tensor Processing Unit) yang dioptimalkan untuk operasi matematika AI, terbukti mampu menghasilkan efisiensi energi dan performa sistem yang jauh lebih superior [9].

Perkembangan evolusi ini membuktikan bahwa masa depan komputasi tidak lagi bergantung pada pemaksaan kinerja satu inti tunggal, melainkan pada kerja sama banyak inti yang efisien. Dengan pembagian beban kerja yang lebih cerdas pada arsitektur heterogen, perangkat keras saat ini mampu menyelesaikan tugas-tugas berat dengan konsumsi daya yang jauh lebih hemat. Hal ini membuat perangkat komputasi modern menjadi lebih responsif saat digunakan untuk menjalankan berbagai program yang rumit secara bersamaan.

Lebih jauh lagi, transisi menuju komputasi heterogen didorong oleh munculnya fenomena *Dark Silicon*. Fenomena ini merujuk pada kondisi di mana sebagian besar area dari sebuah cip prosesor terpaksa dimatikan atau tidak dialiri daya penuh saat beroperasi guna mencegah terjadinya penumpukan panas ekstrem (*thermal throttling*). Dalam arsitektur heterogen, tantangan ini dimitigasi dengan mengalokasikan unit pemrosesan yang sangat spesifik untuk tugas yang spesifik pula. Sebagai contoh, alih-alih membiarkan CPU mengeksekusi kalkulasi matriks bervolume tinggi yang tidak efisien, tugas tersebut dialihkan sepenuhnya kepada unit *Neural Processing Unit* (NPU) atau *Tensor Processing Unit* (TPU). Pendekatan modular ini memastikan bahwa pemanfaatan ruang silikon pada cip modern tetap optimal tanpa melanggar batasan termal yang diizinkan oleh sistem perangkat keras.

3.2. Peran CPU dalam Kinerja Sistem

Unit Pemroses Sentral (CPU) tetap memegang kendali utama sebagai eksekutor instruksi dan pengontrol koordinasi dalam sistem komputer [1]. Namun, pada arsitektur modern, peningkatan performa CPU tidak lagi bertumpu pada peningkatan frekuensi *clock*, melainkan pada kecerdasan desain arsitektur internalnya [3]. Beberapa teknik arsitektur mikro mutakhir yang diimplementasikan pada CPU modern meliputi *pipelining*, *superscalar execution*, *out-of-order execution* (OoO), dan *speculative execution* melalui *branch prediction* [13].

Teknik *pipelining* memungkinkan CPU untuk membagi proses eksekusi satu instruksi menjadi beberapa tahapan kecil (seperti *fetch*, *decode*, *execute*, *write-back*), sehingga beberapa instruksi berbeda dapat diproses secara simultan pada tahapan yang berbeda dalam satu siklus *clock*. Sementara itu, *out-of-order execution* memungkinkan prosesor untuk mengeksekusi instruksi yang sudah siap terlebih dahulu tanpa harus mengantri secara kaku sesuai urutan program asli, selama tidak ada ketergantungan data (*data dependency*) [13]. Hal ini secara signifikan meminimalkan waktu menganggur (*idle time*) dari unit-unit eksekusi di dalam CPU. Kombinasi dari teknik-teknik ini, didukung oleh manajemen pembagian beban kerja pada multi-core processing, mampu melipatgandakan *Instruction Per Cycle* (IPC), meningkatkan *throughput*, dan mengoptimalkan efisiensi sistem secara keseluruhan tanpa memicu panas berlebih.

Namun, lompatan performa yang dicapai melalui arsitektur mikro *speculative execution* dan *out-of-order execution* ini tidak luput dari konsekuensi keamanan yang serius pada tingkat perangkat keras (*hardware-level security vulnerabilities*). Fenomena serangan siber modern seperti *Spectre* dan *Meltdown* memanfaatkan celah dari cara kerja eksekusi spekulatif di dalam CPU. Ketika prosesor menebak jalur instruksi dan melakukan komputasi di area memori sensitif sebelum validasi hak akses selesai, jejak-jejak data tersebut tertinggal di dalam *cache memory* (efek samping berupa *side-channel*). Meskipun instruksi spekulatif yang salah akhirnya dibatalkan oleh CPU, peretas dapat mengukur perbedaan latensi akses memori untuk membaca data rahasia tersebut. Penemuan celah keamanan ini memaksa para arsitek komputer modern untuk mendesain ulang mekanisme isolasi memori kernel pada tingkat sirkuit, membuktikan bahwa parameter keberhasilan arsitektur modern kini wajib menyeimbangkan antara kecepatan ekstrem dan ketahanan keamanan (*hardware security*).

Penerapan teknik-teknik internal tersebut sangat krusial karena memungkinkan CPU untuk tetap meningkatkan produktivitasnya tanpa harus memicu panas berlebih pada perangkat kerasnya. Ketika aliran instruksi dapat diatur secara paralel dan fleksibel di dalam sirkuit CPU, waktu tunggu prosesor menjadi jauh berkurang. Hasilnya, pengguna dapat merasakan performa komputer yang lebih lancar dan stabil, bahkan saat mengeksekusi instruksi-instruksi program yang kompleks.

Di samping mekanisme *pipelining* dan *out-of-order execution*, CPU modern juga sangat bergantung pada keandalan sistem *Branch Prediction* (prediksi percabangan) dan *Speculative Execution* (eksekusi spekulatif) [13]. Dalam sebuah kode program, instruksi percabangan seperti kondisi *if-else* sering kali menciptakan ketidakpastian jalur eksekusi yang dapat menghentikan aliran data di dalam jalur pipa (*pipeline stall*). Untuk mencegah hal tersebut, komponen pembantu di dalam CPU akan menebak jalur mana yang kemungkinan besar akan diambil berdasarkan riwayat eksekusi sebelumnya. Prosesor kemudian akan mengeksekusi instruksi pada jalur tebakan tersebut terlebih dahulu secara spekulatif. Jika tebakan itu benar, kinerja sistem akan meningkat secara masif karena tidak ada waktu siklus yang terbuang; namun, jika tebakan tersebut salah, prosesor akan membuang hasil kerja spekulatif tersebut dan kembali ke jalur yang benar dengan konsekuensi penalti waktu yang minimal.

Selain memengaruhi kecepatan pemrosesan data secara langsung, optimalisasi desain internal CPU juga berkontribusi pada kestabilan sistem operasional dalam jangka panjang. Ketika instruksi program dapat dikelola dengan efisien melalui mekanisme internal yang matang, risiko kegagalan sistem akibat penumpukan beban pemrosesan (*overhead*) dapat diminimalkan. Hal ini memastikan bahwa komputer tidak hanya bekerja dengan cepat dalam durasi yang singkat, melainkan mampu mempertahankan performa puncaknya secara konsisten meskipun dihadapkan pada tugas-tugas berat yang berjalan terus-menerus.

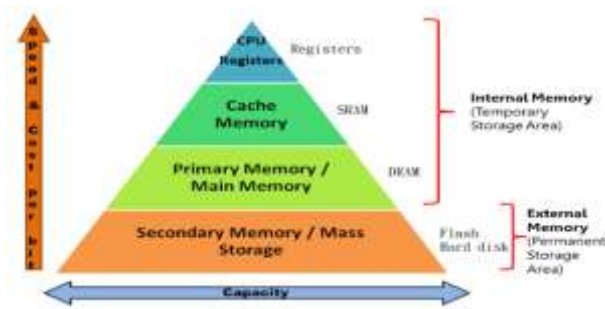
3.3. Hierarki Memori dan Efisiensi Akses Data

Performa prosesor yang luar biasa cepat tidak akan memiliki arti jika sistem interkoneksi dan penyedia data lambat. Hierarki memori yang terdiri dari register, cache (L1, L2, L3), memori utama (RAM), dan penyimpanan sekunder (*storage*) dirancang secara berlapis untuk menjembatani jurang pemisah antara kecepatan eksekusi prosesor dan kapasitas penyimpanan [1]. Lapisan ini diatur sedemikian rupa sehingga semakin dekat posisi memori ke CPU, maka kecepatan aksesnya akan semakin tinggi namun dengan kapasitas yang semakin kecil dan biaya per bit yang semakin mahal.

Cache memory yang berbasis static RAM (SRAM) memegang peran vital dalam mereduksi latensi komunikasi data antara CPU dan memori utama yang berbasis dynamic RAM (DRAM) [5]. Cara kerja cache sangat bergantung pada prinsip *locality of reference*, yang terbagi menjadi dua, yaitu *temporal locality* (kecenderungan data yang baru saja diakses akan diakses kembali dalam waktu dekat) dan *spatial locality* (kecenderungan data yang berada di dekat alamat data yang baru saja diakses akan ikut diakses segera) [10].

Optimalisasi pada desain hierarki memori—seperti penambahan kapasitas cache L3 yang besar serta penerapan teknologi memori baru seperti *High Bandwidth Memory* (HBM)—terbukti secara empiris mampu menekan angka kegagalan pencarian data di cache (*cache miss*). Penurunan *cache miss* berarti berkurangnya frekuensi CPU harus mengakses RAM utama secara langsung, yang pada akhirnya secara signifikan mengurangi problem ketimpangan performa *Memory Wall* dan menstabilkan efisiensi sistem komputasi modern [5, 10].

Masalah interkoneksi pada hierarki memori juga diperparah oleh tantangan koherensi cache (*cache coherence*) pada arsitektur prosesor multi-core [5]. Ketika beberapa inti prosesor memiliki salinan data yang sama dari memori utama di dalam cache L1 atau L2 masing-masing, perubahan data oleh satu inti dapat menyebabkan data pada inti lain menjadi tidak valid (*stale data*). Oleh karena itu, arsitektur komputer modern mengimplementasikan protokol manajemen yang ketat, seperti protokol MESI (*Modified, Exclusive, Shared, Invalid*) atau MOESI, untuk memastikan sinkronisasi data real-time antar-core tetap terjaga. Selain itu, adopsi arsitektur memori modern berbasis NUMA (*Non-Uniform Memory Access*) pada sistem multi-processor memastikan bahwa setiap kelompok inti memiliki akses langsung ke pengontrol memori lokalnya sendiri, sehingga meminimalkan kemacetan lalu lintas data pada bus interkoneksi sistem utama.



Gambar 2. Hierarki Memori

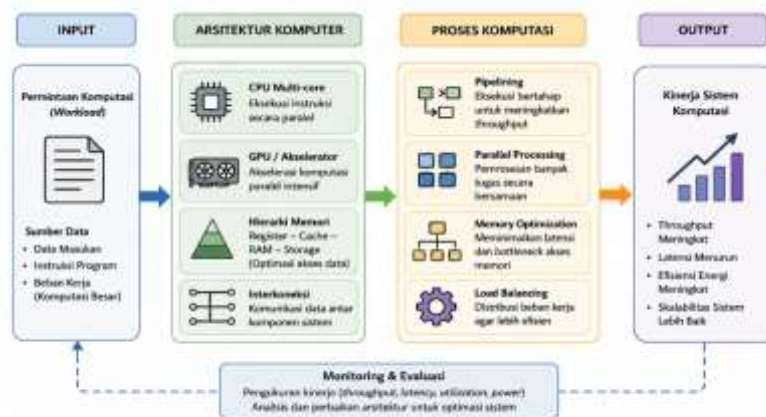
Berdasarkan Gambar 2, hierarki memori digambarkan dalam bentuk piramida untuk menunjukkan hubungan antara kecepatan akses, kapasitas, dan biaya per bit. Pada tingkatan teratas, terdapat CPU Registers dan Cache Memory (SRAM) yang memiliki kecepatan operasional paling tinggi namun dengan kapasitas yang sangat terbatas. Sebaliknya, semakin ke bawah menuju Main Memory (DRAM) dan Secondary Memory (Flash/Hard disk), kapasitas penyimpanan data akan semakin besar dan murah, meskipun kecepatan aksesnya lebih lambat. Pembagian lapisan ini sangat penting agar sistem komputer dapat menyeimbangkan kebutuhan data prosesor secara efektif dan ekonomis.

Pengaturan hierarki yang tepat ini memastikan bahwa data yang paling sering dibutuhkan oleh prosesor selalu berada di tempat yang paling cepat diakses. Jika koordinasi antara cache dan memori utama berjalan dengan optimal, hambatan akibat kecepatan RAM yang tertinggal dapat diminimalkan. Oleh karena itu, efisiensi akses data dalam hierarki memori memegang peran yang sangat menentukan agar performa tinggi dari CPU modern tidak terhambat oleh lambatnya penyaluran data.

3.4. Komputasi Paralel dan Akselerasi Sistem

Komputasi paralel adalah pilar utama yang memungkinkan eksekusi banyak instruksi atau proses secara bersamaan (*simultaneously*) dengan memanfaatkan infrastruktur CPU multi-core, arsitektur GPU masif, maupun kluster komputer terdistribusi [3]. Pendekatan ini terbukti menjadi satu-satunya solusi yang paling efektif untuk menangani beban kerja ekstra berat yang tidak dapat diselesaikan oleh komputasi sekuensial tradisional. Implementasi nyata dari teknologi ini jamak ditemukan pada sistem kecerdasan buatan, visualisasi grafis 3D real-time, simulasi saintifik cuaca, hingga pemrosesan data besar pada platform cloud [7, 14].

Untuk memahami bagaimana variabel arsitektur komputer berinteraksi secara sistematis dalam mendongkrak performa sistem, alur kerja operasionalnya dapat digambarkan melalui sebuah kerangka kerja terpadu. Kerangka ini memetakan seluruh proses, mulai dari penyerapan beban kerja awal, manajemen pemrosesan perangkat keras, hingga output kinerja yang dihasilkan.



Gambar 3.2. Framework Penelitian Pengaruh Arsitektur Komputer terhadap Kinerja Sistem Komputasi Modern

Gambar 2. Frame Work penelitian

Berdasarkan *Framework* Penelitian pada Gambar 3.2, mekanisme peningkatan performa komputasi modern dapat diuraikan menjadi empat tahapan terintegrasi sebagai berikut:

1. Tahap Input: Sistem menerima permintaan komputasi (*workload*) berskala besar yang mencakup data masukan mentah, instruksi program, dan beban kerja intensif (misalnya dataset pelatihan model pembelajaran mendalam atau kueri data raya) [7].
2. Tahap Arsitektur Komputer: Beban kerja dari tahap input dialokasikan ke lapisan perangkat keras yang heterogen. Di sini, CPU multi-core bertugas menangani logika kontrol yang kompleks [8], sementara GPU atau akselerator AI mengambil alih komputasi paralel intensif [12]. Hierarki memori (register, cache, RAM) bekerja secara simultan untuk memastikan ketersediaan data dengan latensi rendah, didukung oleh sistem interkoneksi berkecepatan tinggi yang memfasilitasi komunikasi antarkomponen tanpa hambatan.
3. Tahap Proses Komputasi: Pada tahap ini, arsitektur mengeksplorasi berbagai teknik optimasi. *Pipelining* digunakan untuk mempercepat *throughput* instruksi serial [13], *Parallel Processing* mengeksekusi ribuan thread secara bersamaan di GPU [4], *Memory Optimization* meminimalkan latensi akses memori utama [5], dan *Load Balancing* mendistribusikan beban kerja secara merata ke seluruh inti prosesor yang tersedia untuk mencegah adanya satu inti yang mengalami *overhead* sementara inti lainnya menganggur [8].
4. Tahap Output dan Evaluasi: Hasil akhir dari sinergi proses ini adalah lonjakan signifikan pada Kinerja Sistem Komputasi, yang ditandai dengan peningkatan nilai *throughput* (jumlah operasi yang diselesaikan per satuan waktu), penurunan latensi (waktu respon sistem), penghematan konsumsi daya (efisiensi energi), serta skalabilitas sistem yang jauh lebih adaptif [15]. Output ini kemudian dimonitor dan dievaluasi secara kontinu untuk memberikan umpan balik bagi penyempurnaan desain arsitektur komputer di masa yang akan datang.

Jika ditinjau dari Taksonomi Flynn, komputasi paralel pada arsitektur modern didominasi oleh pergeseran dari model SISD (*Single Instruction, Single Data*) menuju model SIMD (*Single Instruction, Multiple Data*) dan MIMD (*Multiple Instruction, Multiple Data*) [3]. GPU modern memanfaatkan arsitektur SIMT (*Single Instruction, Multiple Threads*) yang memungkinkannya menjalankan satu perintah matematika secara serentak terhadap ribuan elemen data dalam satu siklus waktu tunggal. Karakteristik arsitektur seperti ini sangat ideal untuk algoritma enkripsi data, rendering grafis tiga dimensi, serta kalkulasi bobot (*weights*) pada jaringan saraf tiruan *deep learning*. Melalui kombinasi kontrol logika presisi dari CPU (MIMD) dan kemampuan pemrosesan paralel masif dari GPU (SIMD), arsitektur komputer modern berhasil menciptakan ekosistem perangkat keras yang sangat tangguh untuk memikul beban kerja berskala industri.

Dengan adanya kemampuan akselerasi paralel ini, pemrosesan data berskala besar yang dahulu membutuhkan waktu lama, kini dapat diselesaikan dengan jauh lebih cepat dan efisien. Sinergi antara CPU multi-core dan kekuatan GPU dalam memproses data secara serentak menjadi kunci utama di balik jalannya aplikasi-aplikasi modern berkinerja tinggi. Rangkaian proses dari input hingga output dalam sistem ini secara nyata memberikan kontribusi terbesar terhadap peningkatan produktivitas teknologi saat ini.

Fleksibilitas komputasi paralel juga membuka peluang bagi integrasi sistem yang lebih luas di berbagai sektor industri digital modern. Melalui pemanfaatan kekuatan GPU dan sistem terdistribusi, pengembang perangkat lunak kini memiliki kebebasan untuk menciptakan aplikasi yang lebih kaya fitur tanpa perlu khawatir akan keterbatasan performa perangkat keras. Sinergi yang kuat antara arsitektur paralel dan kebutuhan perangkat lunak kontemporer ini secara nyata menjadi motor penggerak utama di balik percepatan inovasi teknologi informasi saat ini.

4. Kesimpulan

Arsitektur komputer memegang peran yang sangat krusial dan deterministik dalam menentukan batas atas kinerja serta efisiensi sistem komputasi modern. Seiring dengan ketidakmampuan industri manufaktur silikon untuk terus menaikkan frekuensi clock speed akibat kendala termal dan daya, inovasi pada tingkat arsitektur terbukti menjadi jalan keluar utama. Transformasi struktural dari era prosesor *single-core* menuju arsitektur heterogen yang menggabungkan CPU, GPU, dan akselerator AI terdedikasi telah berhasil memberikan lonjakan performa komputasi secara eksponensial melalui pergeseran model eksekusi berbasis SIMD, MIMD, dan SIMT. Optimasi pada komponen internal seperti penerapan teknik *pipelining* yang matang, *out-of-order execution*, serta manajemen hierarki memori cache yang cerdas terbukti efektif dalam meminimalkan dampak negatif dari fenomena *Memory Wall* yang selama ini menjadi momok bagi penurunan kecepatan akses data. Namun, inovasi

arsitektur mikro tersebut kini juga dituntut untuk menyeimbangkan antara peningkatan kecepatan ekstrem dan ketahanan keamanan pada tingkat perangkat keras (*hardware security*) guna memitigasi celah kerentanan spekulatif seperti Spectre dan Meltdown. Secara keseluruhan, integrasi berbagai inovasi teknologi ini tidak sekadar menyelesaikan kendala teknis pada tingkat perangkat keras, melainkan secara langsung meningkatkan responsivitas sistem dan memberikan pengalaman pengguna yang jauh lebih lancar serta stabil. Melalui koordinasi yang matang antara efisiensi instruksi internal CPU, manajemen sinkronisasi interkoneksi dan koherensi cache, serta kekuatan akselerasi paralel, hambatan penyaluran data (*bottleneck*) dapat dieliminasi secara efektif. Dampak praktisnya, sistem komputasi modern saat ini mampu memproses beban kerja data berskala besar dengan konsumsi daya yang jauh lebih hemat, yang pada akhirnya menjadi pilar utama dalam mendongkrak produktivitas teknologi digital. Meskipun pencapaian efisiensi sistem saat ini sudah sangat tinggi, tantangan-tantangan baru seperti tuntutan efisiensi energi yang lebih ketat pada pusat data (*green data center*) serta kebutuhan pemrosesan data real-time pada skala exabyte masih menjadi fokus utama dalam riset pengembangan arsitektur komputer masa depan. Oleh karena itu, arah riset selanjutnya diprediksi akan semakin bergeser ke arah arsitektur non-von Neumann, komputasi neuromorfik, hingga integrasi sistem komputasi kuantum tingkat awal. Pemahaman yang komprehensif mengenai interaksi antara perangkat keras dan karakteristik beban kerja perangkat lunak mutlak diperlukan oleh para insinyur sistem demi merancang infrastruktur komputasi masa depan yang tidak hanya bertenaga besar, tetapi juga adaptif dan berkelanjutan.

Referensi

- [1] W. Stallings, *Computer Organization and Architecture*, 11th ed., Pearson, 2022.
- [2] A. S. Tanenbaum dan T. Austin, *Structured Computer Organization*, 7th ed., 2021.
- [3] J. L. Hennessy dan D. A. Patterson, *Computer Architecture: A Quantitative Approach*, 7th ed., 2023.
- [4] S. Mittal, "Modern Processor Architectures and Trends in Parallel Computing," *IEEE Access*, 2023.
- [5] R. K. Gupta, "Memory Hierarchy Optimization Techniques," *Journal of Computer Systems*, 2024.
- [6] A. R. Putra et al., "Analisis Arsitektur CPU Modern," *Jurnal Teknologi Informasi*, 2025.
- [7] IEEE Computer Society, "Advances in Parallel Computing Architecture," *IEEE Computer*, 2024.
- [8] M. S. Khan et al., "Performance Analysis of Multi-Core Systems," *Future Computing Journal*, vol. 8, no. 2, pp. 115–128, 2026..
- [9] J. Dongarra, K. Moore, and T. Schuchart, "The Future of High-Performance Computing and Heterogeneous Architectures," *International Journal of High Performance Computing Applications*, vol. 36, no. 3, pp. 301–315, 2022.
- [10] X. Feng and Y. Chen, "Overcoming the Memory Wall: A Review of Emerging Memory Architectures," *ACM Computing Surveys*, vol. 55, no. 9, pp. 1–35, 2023.
- [11] T. Sterling, M. Anderson, and D. Brodowicz, *High Performance Computing: Modern Systems and Practices*, 2nd ed., Cambridge, MA: Academic Press, 2024.
- [12] H. Mutlu, L. E. Oden, and A. Bhatele, "A Review of Hardware Accelerators for Artificial Intelligence and Machine Learning," *IEEE Transactions on Parallel and Distributed Systems*, vol. 35, no. 1, pp. 45–62, 2025.
- [13] S. Raj and K. Kumar, "Pipelining and Out-of-Order Execution Optimization in Modern Microprocessors," *Journal of Systems Architecture*, vol. 142, p. 102931, 2023.
- [14] L. Nuñez and M. Silva, "Cloud Computing Performance Evaluation Based on Distributed Computer Architecture," *Journal of Cloud Computing*, vol. 13, no. 1, pp. 12–25, 2024.
- [15] R. Patterson, "Energy-Efficient Parallel Architectures for Big Data Processing," *Sustainable Computing: Informatics and Systems*, vol. 41, p. 100912, 2026