



KOMTELKA: Jurnal Komputer Telekomunikasi dan Elektronika, Volume 01 Nomor 1 Tahun 2026
Tersedia online di
<https://jurnal.rajawalicemerlangabadi.com/index.php/jpek>



Tinjauan Arsitektur Komputer Kuantum dan Potensinya dalam Transformasi Komputasi Modern

Ira Sri Kusuma¹, Iswandi², Arrisky Ardian³

^{1,2,3} Sekolah Tinggi Manajemen Informatika dan Komputer

email: ¹ irasrikusuma.stmikpi@gmail.com

Kata kunci: arsitektur komputer kuantum; Qubit; NISQ; koreksi kesalahan kuantum; komputasi hibrida

Keywords: quantum computer architecture; Qubit; NISQ; quantum error correction; hybrid computing

ABSTRAK

Komputer kuantum merupakan paradigma komputasi yang memanfaatkan prinsip superposisi, interferensi, dan keterikatan kuantum untuk memproses informasi dalam bentuk qubit. Kajian terdahulu umumnya membahas satu lapisan arsitektur secara terpisah—misalnya hanya platform qubit atau hanya koreksi kesalahan—sehingga keterkaitan antarlapisan dan posisinya terhadap kesiapan integrasi hibrida belum banyak dianalisis secara terpadu (research gap). Kajian ini bertujuan meninjau arsitektur komputer kuantum dari lapisan fisik hingga lapisan perangkat lunak secara berjenjang dan saling terkait, sehingga menghasilkan kerangka integratif lapisan arsitektur–NISQ–hibrida yang membedakan kajian ini dari review lain di bidang quantum computing (novelty), serta menganalisis potensinya dalam transformasi komputasi modern. Metode yang digunakan adalah studi literatur naratif (bukan sistematis), dengan tahapan identifikasi topik, seleksi referensi, klasifikasi arsitektur, analisis potensi, dan sintesis kesimpulan, terhadap publikasi ilmiah, dokumentasi industri, dan standar teknologi terbaru dalam kurun lima tahun terakhir. Hasil kajian menunjukkan bahwa arsitektur komputer kuantum tidak hanya bergantung pada jumlah qubit, tetapi juga pada kualitas qubit, fidelitas gerbang, konektivitas, sistem kontrol, mekanisme mitigasi dan koreksi kesalahan, serta integrasi hibrida dengan komputer klasik. Platform seperti superkonduktor, ion terperangkap, atom netral, fotonik, dan spin silikon memiliki keunggulan serta keterbatasan masing-masing. Potensi transformasi terbesar terlihat pada simulasi molekuler, optimasi, kecerdasan buatan, keamanan siber, dan komputasi ilmiah berskala besar. Namun, pemanfaatan praktis masih menghadapi kendala berupa derau, dekoherensi, kebutuhan koreksi kesalahan, biaya infrastruktur, dan keterbatasan algoritma. Dengan demikian, arah pengembangan paling realistis pada masa kini adalah arsitektur hibrida CPU/GPU/QPU yang memadukan kekuatan komputasi klasik dan kuantum. Kontribusi utama kajian ini adalah kerangka integratif yang menghubungkan kelima lapisan arsitektur secara kritis, sehingga melampaui pemaparan deskriptif dan memberikan dasar evaluasi bagi pemilihan arah riset maupun platform kuantum yang sesuai kebutuhan.

ABSTRACT

Quantum computers represent a computing paradigm that uses superposition, interference, and entanglement to process information in the form of qubits. Prior reviews typically address a single architectural layer in isolation – for example, only the qubit platform or only error correction – leaving the cross-layer relationships and their bearing on hybrid-integration readiness largely unaddressed (the research gap this paper targets). This paper reviews quantum computer architecture from the physical layer to the software layer as an interconnected stack rather than as separate topics, offering an integrative architecture–NISQ–hybrid framework that distinguishes this review from others in the field (the novelty) and analyzes its potential role in transforming modern computing. The method used is a narrative (not systematic) literature review of recent scientific publications, industrial documentation, and technology standards published within the last five years, following stages of identification, selection, classification, analysis, and synthesis. The review shows that quantum computer architecture is not determined solely by qubit count, but also by qubit quality, gate fidelity, connectivity, control systems, error mitigation and correction mechanisms, and hybrid integration with classical computers. Superconducting, trapped-ion, neutral-atom, photonic, and silicon-spin platforms each offer distinct strengths and limitations. The most significant transformation potential appears in molecular simulation, optimization, artificial intelligence, cybersecurity, and large-scale scientific computing. Nevertheless, practical adoption still faces noise, decoherence, error-correction overhead, infrastructure cost, and algorithmic maturity challenges. Therefore, the most realistic development direction today is a hybrid CPU/GPU/QPU architecture that combines the complementary strengths of classical and quantum computing. The main contribution of this review is an integrative framework that critically links all five architectural layers, moving beyond descriptive exposition to provide a basis for evaluating research directions and platform choices.

PENDAHULUAN

Perkembangan komputasi modern selama beberapa dekade banyak ditopang oleh peningkatan kerapatan transistor, optimasi arsitektur prosesor, komputasi paralel, dan akselerator seperti GPU. Akan tetapi, peningkatan kinerja komputer klasik menghadapi batas fisik dan ekonomi, terutama ketika ukuran transistor semakin kecil, konsumsi daya meningkat, serta masalah yang harus diselesaikan memiliki ruang keadaan sangat besar. Pada konteks tersebut, komputer kuantum dipandang sebagai salah satu arah komputasi baru karena memproses informasi menggunakan *qubit*, bukan bit biner konvensional. *Qubit* dapat berada pada kombinasi keadaan melalui superposisi dan dapat saling berkorelasi melalui entanglement, sehingga sejumlah algoritma kuantum berpotensi mengeksplorasi ruang solusi tertentu dengan cara yang tidak tersedia pada komputer klasik [1].

Walaupun potensinya besar, komputer kuantum tidak dapat dipahami sebagai pengganti langsung komputer klasik. Sistem kuantum modern masih berada pada fase *noisy intermediate-scale quantum* (NISQ), yaitu fase ketika perangkat memiliki jumlah *Qubit* yang terus bertambah, tetapi masih dibatasi oleh derau, kesalahan gerbang, keterbatasan waktu koherensi, dan belum sepenuhnya menerapkan

komputasi kuantum toleran-kesalahan. Kajian kompleksitas NISQ menunjukkan bahwa perangkat kuantum bising dapat memiliki kemampuan tertentu yang berbeda dari komputer klasik, tetapi kemampuannya tetap tidak sama dengan komputer kuantum *fault-tolerant* yang ideal [2]. Oleh karena itu, analisis arsitektur menjadi penting agar pembahasan komputer kuantum tidak hanya berfokus pada jumlah *qubit*, tetapi juga mencakup kualitas perangkat, konektivitas, sistem kontrol, *compiler*, hingga integrasi dengan komputasi klasik.

Dalam beberapa tahun terakhir, riset komputer kuantum berkembang sangat cepat. Eksperimen *surface code* menunjukkan bahwa peningkatan ukuran kode dapat menekan kesalahan logis jika kualitas perangkat cukup baik [3]. Hasil lain menunjukkan perkembangan menuju koreksi kesalahan di bawah ambang *surface code*, yang menjadi indikator penting untuk membangun komputer kuantum yang lebih andal [4]. Di sisi lain, platform atom netral menunjukkan prosesor logis yang dapat diprogram dengan ratusan *qubit* fisik dan arsitektur zona yang mendukung operasi logis [5]. Perkembangan ini memperlihatkan bahwa arsitektur komputer kuantum merupakan bidang multidisipliner yang menggabungkan fisika, elektronika, rekayasa sistem, algoritma, dan ilmu komputer. Tinjauan terdahulu seperti kajian kompleksitas NISQ [2] dan tinjauan kemajuan komputasi era NISQ [8] cenderung berhenti pada level fenomena—menjelaskan keterbatasan derau dan dekoherensi—tanpa menghubungkannya secara eksplisit dengan arsitektur sistem kontrol, transpilasi, maupun jalur integrasi hibrida CPU/GPU/QPU. Demikian pula, laporan teknis IBM [10][11] dan Microsoft [9][14] membahas integrasi hibrida dan estimasi sumber daya secara terpisah dari diskusi platform *qubit* dan koreksi kesalahan. Posisi artikel ini dalam kajian arsitektur komputer kuantum (state of the art) adalah menyatukan ketiga ranah tersebut—perangkat fisik, mekanisme NISQ/QEC, dan integrasi hibrida—ke dalam satu kerangka analisis yang berjenjang, sehingga keterkaitan kausal antarlapisan menjadi lebih eksplisit dibandingkan kajian-kajian sebelumnya yang membahasnya secara parsial.

Urgensi pembahasan ini juga muncul dari dampak potensial komputer kuantum terhadap transformasi komputasi modern. Pada bidang sains dan industri, komputer kuantum berpotensi mempercepat simulasi material, kimia kuantum, optimasi kompleks, pembelajaran mesin, dan pemodelan sistem yang sulit ditangani

oleh metode klasik. Pada bidang keamanan siber, perkembangan komputer kuantum mendorong migrasi menuju kriptografi pascakuantum karena komputer kuantum berskala besar dapat mengancam beberapa skema kriptografi kunci publik yang digunakan saat ini [13]. Dengan demikian, pembahasan arsitektur komputer kuantum perlu dihubungkan dengan kesiapan ekosistem komputasi modern, bukan hanya dengan teori komputasi. Tujuan artikel ini adalah meninjau arsitektur komputer kuantum secara sistematis, mulai dari lapisan perangkat keras *qubit*, sistem kontrol, lapisan sirkuit dan *compiler*, mekanisme koreksi kesalahan, hingga model integrasi hibrida CPU/GPU/QPU. Artikel ini juga menganalisis potensi komputer kuantum dalam transformasi komputasi modern serta menguraikan keterbatasan yang masih perlu diatasi agar teknologi ini dapat digunakan secara lebih luas. Kebaruan (novelty) artikel ini terletak pada penyatuan lapisan perangkat keras, kontrol, koreksi kesalahan, dan integrasi hibrida ke dalam satu kerangka analisis yang berjenjang dan saling terkait—berbeda dari kajian terdahulu yang umumnya membahas salah satu lapisan tersebut secara berdiri sendiri—sehingga artikel ini memberikan basis evaluasi lintas-lapisan yang belum banyak tersedia pada literatur sejenis.

METODE PENELITIAN

Penelitian ini menggunakan metode studi literatur naratif (narrative review), bukan tinjauan sistematis (systematic review). Metode ini dipilih karena topik komputer kuantum masih berkembang cepat dan memiliki cakupan multidisipliner, sehingga kajian perlu menggabungkan hasil penelitian ilmiah, dokumentasi teknologi, standar, serta arah pengembangan industri, sumber yang heterogen dan tidak selalu terindeks secara seragam pada basis data tunggal sehingga belum memenuhi prasyarat pelaporan tinjauan sistematis seperti PRISMA. Tinjauan sistematis lebih sesuai untuk pertanyaan penelitian yang sempit dan terukur (misalnya membandingkan efektivitas satu jenis decoder), sedangkan tujuan kajian ini adalah membangun pemahaman konseptual lintas-lapisan, sehingga pendekatan naratif lebih sesuai. Metode sintesis yang digunakan bersifat tematik (thematic analysis) dengan elemen perbandingan (comparative analysis): literatur dikelompokkan ke dalam tema-tema lapisan arsitektur (tematik), kemudian platform dan pendekatan dalam setiap tema diperbandingkan berdasarkan keunggulan,

keterbatasan, dan kesesuaian aplikasi (komparatif). Fokus kajian tidak diarahkan untuk menguji satu hipotesis kuantitatif, melainkan untuk menyusun pemahaman konseptual mengenai arsitektur komputer kuantum dan implikasinya terhadap komputasi modern.

Sumber literatur dipilih berdasarkan tiga kriteria utama. Pertama, sumber diterbitkan dalam kurun lima tahun terakhir agar sesuai dengan perkembangan mutakhir teknologi kuantum. Kedua, sumber berasal dari publikasi ilmiah, dokumentasi resmi, atau lembaga standar yang dapat diakses secara terbuka. Ketiga, sumber memiliki relevansi langsung terhadap arsitektur komputer kuantum, NISQ, koreksi kesalahan kuantum, integrasi hibrida, dan potensi aplikasi. Literatur yang digunakan meliputi artikel jurnal Nature, Nature Communications, dokumentasi IBM Quantum, dokumentasi Microsoft Azure Quantum, dan rilis standar NIST.

Tahapan penelitian meliputi identifikasi topik, seleksi referensi, klasifikasi arsitektur, analisis potensi, dan sintesis kesimpulan. Alur penelitian ditunjukkan pada Gambar 1. Pada tahap identifikasi, topik dibatasi pada struktur arsitektur komputer kuantum dan dampaknya terhadap komputasi modern. Pada tahap seleksi referensi, sumber yang tidak relevan, tidak jelas asal-usulnya, atau terlalu lama tidak digunakan sebagai rujukan utama. Pada tahap klasifikasi, literatur dikelompokkan ke dalam lapisan perangkat keras, sistem kontrol, perangkat lunak, koreksi kesalahan, serta aplikasi. Pada tahap analisis, setiap kelompok dievaluasi berdasarkan kelebihan, kendala, dan kontribusinya terhadap transformasi komputasi.



Gambar 1. Alur metode studi literatur yang digunakan dalam kajian

Tabel 1. Kriteria pemilihan literatur dalam kajian

Kriteria	Penjelasan	Tujuan Penggunaan
Rentang waktu	Literatur diprioritaskan dari tahun 2021-2026.	Menjaga relevansi dengan perkembangan terbaru.
Kredibilitas sumber	Jurnal ilmiah, dokumentasi resmi, dan lembaga standar.	Menghindari rujukan yang tidak dapat diverifikasi.
Relevansi topik	Membahas <i>qubit</i> , arsitektur, NISQ, <i>compiler</i> , QEC, dan aplikasi.	Menyesuaikan literatur dengan rumusan kajian.
Keterbukaan akses	Memiliki DOI, laman resmi, atau tautan yang dapat dibuka.	Memudahkan pembaca menelusuri daftar pustaka.

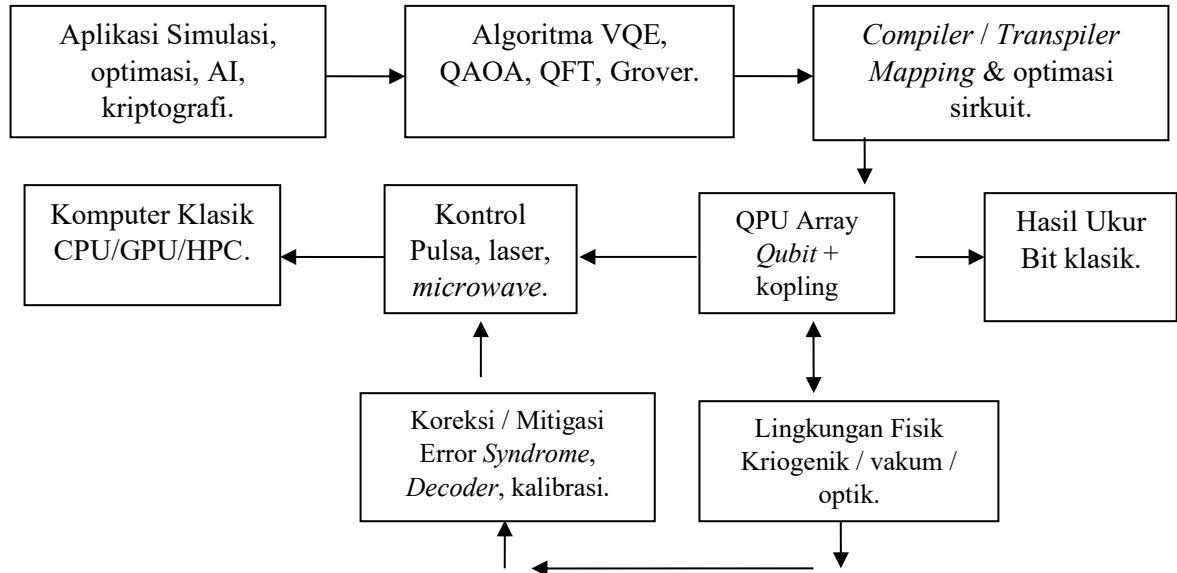
PEMBAHASAN

1. Konsep Dasar Arsitektur Komputer Kuantum

Arsitektur komputer kuantum dapat dipahami sebagai susunan lapisan yang memungkinkan informasi kuantum disiapkan, dimanipulasi, dikoreksi, diukur, dan diterjemahkan menjadi keluaran klasik. Pada komputer klasik, satuan informasi adalah bit dengan nilai 0 atau 1. Pada komputer kuantum, satuan informasi adalah *qubit* yang secara matematis dapat direpresentasikan sebagai kombinasi linier dari dua keadaan basis. Karakteristik ini membuat pemrosesan kuantum sangat berbeda dari pemrosesan klasik karena operasi dilakukan melalui gerbang kuantum yang bersifat reversibel sebelum hasil akhirnya diukur menjadi bit klasik [1].

Secara umum, arsitektur komputer kuantum terdiri dari lapisan aplikasi, algoritma, *compiler* atau *transpiler*, sistem kontrol, *quantum processing unit* (QPU), lingkungan fisik, serta sistem koreksi atau mitigasi kesalahan. Lapisan aplikasi menentukan masalah yang ingin diselesaikan, misalnya simulasi molekuler atau optimasi. Lapisan algoritma menerjemahkan masalah menjadi prosedur kuantum seperti *variational quantum eigensolver* (VQE), *quantum approximate optimization*

algorithm (QAOA), *quantum fourier transform* (QFT), atau metode pembelajaran mesin kuantum. Lapisan *compiler* atau *transpiler* kemudian menyesuaikan sirkuit dengan topologi perangkat nyata karena setiap QPU memiliki konektivitas, basis gerbang, dan tingkat kesalahan yang berbeda [12].



Gambar 2. Arsitektur konseptual komputer kuantum modern berbasis integrasi klasik-kuantum

Tabel 2. Lapisan arsitektur konseptual komputer kuantum modern

Lapisan	Komponen Utama	Fungsi	Contoh Teknologi
Aplikasi	Problem formulation	Menentukan masalah yang diselesaikan	Simulasi, optimasi, AI, kriptografi
Algoritma	Quantum circuits	Menerjemahkan masalah ke prosedur kuantum	VQE, QAOA, QFT, Grover
Compiler / Transpiler	Circuit mapping & optimization	Menyesuaikan sirkuit ke topologi QPU	Qiskit Transpiler, tket
Kontrol	Pulse generator, laser, microwave	Menerjemahkan instruksi ke sinyal fisik	AWG, FPGA, kontrol kriogenik

QPU	Qubit array + kopling	Eksekusi operasi kuantum	IBM Eagle, Google Sycamore
Mitigasi / Koreksi Error	Syndrome measurement, decoder	Menekan atau mengoreksi kesalahan	Surface code, AlphaQubit
Lingkungan Fisik	Kriogenik, vakum, optik	Menjaga kondisi fisik Qubit	Dilution refrigerator, optical trap
Integrasi Klasik	CPU / GPU / HPC	Orkestrasi, optimasi parameter, pengolahan hasil	IBM QC Supercomputing, Azure Quantum

Gambar 2 menunjukkan arsitektur konseptual komputer kuantum modern. Diagram tersebut memperlihatkan bahwa QPU tidak bekerja sendiri. QPU selalu terhubung dengan komputer klasik untuk menyusun sirkuit, mengoptimalkan parameter, mengirim instruksi kontrol, membaca hasil pengukuran, serta melakukan koreksi atau mitigasi error. Pola kerja ini menjelaskan mengapa perkembangan komputer kuantum modern semakin diarahkan pada integrasi hibrida antara CPU, GPU, HPC, dan QPU, bukan pada pemisahan total antara komputer klasik dan komputer kuantum [11].

2. Lapisan Fisik *Qubit* dan Platform Perangkat Keras

Lapisan fisik merupakan fondasi utama dalam komputer kuantum karena kualitas *qubit* menentukan seberapa panjang sirkuit dapat dijalankan sebelum informasi rusak oleh derau. Beberapa platform *qubit* yang banyak diteliti meliputi *qubit* superkonduktor, ion terperangkap, atom netral, fotonik, dan spin silikon. *qubit* superkonduktor banyak digunakan karena kompatibel dengan fabrikasi mikroelektronika dan memiliki kecepatan gerbang tinggi, tetapi biasanya memerlukan lingkungan kriogenik dengan suhu sangat rendah. Ion terperangkap memiliki koherensi dan fidelitas tinggi, tetapi operasi gerbangnya relatif lebih lambat serta memerlukan sistem vakum dan kontrol laser yang kompleks. Atom netral menawarkan skala array yang besar dan konektivitas yang dapat dikonfigurasi ulang, sedangkan fotonik menarik untuk komunikasi dan komputasi

berbasis pengukuran. Spin silikon berpotensi kompatibel dengan proses CMOS, tetapi tantangan fabrikasi dan kontrol presisi masih besar.

Perbandingan platform *qubit* tidak dapat dinilai dari satu parameter saja. Jumlah *qubit* yang besar tidak cukup jika *error rate* tinggi, konektivitas terbatas, atau sistem kontrol tidak stabil. Sebaliknya, *qubit* dengan fidelitas tinggi juga belum tentu mudah diskalakan jika kebutuhan infrastruktur terlalu kompleks. Eksperimen pada atom netral menunjukkan bahwa arsitektur berbasis array yang dapat dikonfigurasi ulang mampu mengoperasikan *qubit* logis dengan jumlah *qubit* fisik besar dan kontrol logis yang fleksibel [5]. Sementara itu, eksperimen pada perangkat superkonduktor menunjukkan bahwa peningkatan ukuran *surface code* dapat menurunkan error logis apabila kualitas fisik perangkat berada pada tingkat yang memadai [3]. Hal ini menunjukkan bahwa kedua hasil eksperimen tersebut sebenarnya mengonfirmasi prinsip yang sama dari sisi berbeda: skalabilitas kode koreksi kesalahan hanya bermanfaat jika error fisik dasar sudah berada di bawah ambang tertentu, sehingga perlombaan menambah jumlah *qubit* tanpa memperbaiki fidelitas gerbang berisiko tidak produktif. Implikasi praktisnya, pemilihan platform tidak dapat dilakukan hanya dengan membandingkan satu metrik unggulan masing-masing platform (misalnya kecepatan gerbang superkonduktor versus koherensi ion terperangkap), tetapi harus mempertimbangkan keseluruhan kebutuhan infrastruktur kontrol yang menyertainya karena kebutuhan kontrol inilah yang pada akhirnya menentukan kelayakan operasional setiap platform pada subbagian berikutnya.

Tabel 3. Perbandingan umum beberapa platform *qubit* dalam arsitektur komputer kuantum

Platform <i>Qubit</i>	Keunggulan	Keterbatasan	Kebutuhan Infrastruktur	Kesesuaian Aplikasi
Superkonduktor	Gerbang cepat,	Dekoherensi, kriogenik, noise kontrol	<i>Dilution refrigerator</i> ,	NISQ, QEC, benchmarking

	ekosistem		<i>microwave</i>	
	industri kuat		<i>control</i>	
Ion terperangkap	Fidelitas tinggi, koherensi panjang	Gerbang relatif lambat, sistem optik kompleks	Vakum, laser, <i>trap chip</i>	Simulasi, algoritma presisi
Atom netral	Array besar, konektivitas fleksibel	Kontrol laser dan pembacaan kompleks	<i>Optical tweezer</i> , laser stabil	QEC, simulasi <i>many-body</i>
Fotonik	Baik untuk jaringan kuantum	Kehilangan foton, sumber/detektor sulit	Sumber foton, interferometer, detektor	Komunikasi, MBQC
Spin silikon	Potensi kompatibel CMOS	Kontrol dan fabrikasi presisi tinggi	Kriogenik, chip silikon isotopik	Integrasi jangka panjang

3. Sistem Kontrol, Konektivitas, dan Transpilasi Sirkuit

Sistem kontrol berfungsi menghubungkan instruksi perangkat lunak dengan dinamika fisik *qubit*. Pada platform superkonduktor, kontrol umumnya dilakukan melalui pulsa *microwave* dan pengaturan frekuensi. Pada ion terperangkap dan atom netral, kontrol banyak bergantung pada laser, medan elektromagnetik, serta sistem pembacaan optik. Setiap perintah logika kuantum harus diterjemahkan menjadi sinyal fisik yang tepat karena kesalahan amplitudo, fase, waktu pulsa, atau *crosstalk* dapat menurunkan fidelitas operasi. Dengan demikian, arsitektur komputer kuantum mencakup tidak hanya chip QPU, tetapi juga rak kontrol, elektronik cepat, kabel kriogenik, sistem sinkronisasi, dan perangkat lunak kalibrasi.

Konektivitas *qubit* juga menentukan efisiensi eksekusi sirkuit. Jika dua *qubit* yang harus berinteraksi tidak terhubung langsung, *compiler* perlu menambahkan operasi penukar keadaan seperti SWAP. Operasi tambahan ini memperpanjang kedalaman sirkuit dan meningkatkan akumulasi error. Karena itu, proses transpilasi menjadi komponen penting dalam arsitektur kuantum. Dokumentasi IBM Quantum menjelaskan bahwa *transpilation* adalah proses menulis ulang sirkuit agar sesuai dengan topologi perangkat tertentu dan mengoptimalkan instruksi untuk komputer kuantum bising [10]. Artinya, kualitas hasil komputasi tidak hanya bergantung pada algoritma, tetapi juga pada kemampuan *compiler* mengurangi kedalaman sirkuit, memilih *mapping qubit*, dan memanfaatkan basis gerbang perangkat.

Pada praktiknya, pengguna jarang berinteraksi langsung dengan sinyal fisik *qubit*. Pengguna menulis sirkuit pada level abstrak, kemudian perangkat lunak mengubahnya menjadi rangkaian gerbang yang kompatibel dengan *hardware*. Lapisan ini membuat komputer kuantum semakin mirip ekosistem komputasi modern, yaitu memiliki tumpukan perangkat lunak, *runtime*, *middleware*, dan layanan *cloud*. Namun, perbedaannya adalah proses kompilasi kuantum harus mempertimbangkan noise *hardware* yang berubah dari waktu ke waktu. Oleh sebab itu, arsitektur masa depan membutuhkan integrasi erat antara *compiler*, sistem monitoring perangkat, dan kalibrasi otomatis. Kebutuhan ini terlihat nyata pada eksekusi algoritma seperti VQE untuk simulasi molekuler atau QAOA untuk optimasi kombinatorial: kedua algoritma menjalankan sirkuit parameterik berulang kali pada hardware yang karakter noise-nya berubah dari satu eksekusi ke eksekusi berikutnya, sehingga setiap iterasi kalibrasi yang terlambat dapat membiaskan estimasi parameter dan memperlambat konvergensi algoritma. Ketergantungan inilah yang menjelaskan mengapa pengelolaan derau dan error tidak dapat dipisahkan dari kualitas sistem kontrol, dan menjadi pintu masuk pembahasan NISQ serta koreksi kesalahan kuantum pada subbagian berikutnya.

4. NISQ, Mitigasi Error, dan Koreksi Kesalahan Kuantum

Kendala paling mendasar dalam komputer kuantum adalah error. *Qubit* sangat sensitif terhadap gangguan lingkungan, sehingga keadaan kuantum dapat rusak melalui dekoherensi, kesalahan gerbang, kesalahan pembacaan, dan *crosstalk*. Pada era NISQ, perangkat kuantum dapat menjalankan sirkuit tertentu, tetapi belum mampu menjalankan algoritma panjang secara andal tanpa koreksi kesalahan penuh. Kajian kompleksitas menunjukkan bahwa NISQ memiliki karakteristik tersendiri karena sebuah algoritma NISQ dapat dipandang sebagai interaksi antara komputer klasik dan perangkat kuantum bising [2].

Mitigasi error dan koreksi kesalahan kuantum memiliki peran yang berbeda. Mitigasi error berusaha memperbaiki estimasi hasil tanpa menghilangkan error sepenuhnya dari proses fisik. Teknik ini lebih sesuai untuk perangkat NISQ karena tidak membutuhkan overhead *qubit* sebesar koreksi kesalahan penuh. Sebaliknya, *quantum error correction* (QEC) menyandikan satu *qubit* logis ke banyak *qubit* fisik agar informasi kuantum tetap dapat dipertahankan walaupun sebagian *qubit* mengalami kesalahan. Tantangannya, QEC membutuhkan banyak *qubit* fisik tambahan, pengukuran *syndrome* berulang, *decoder* cepat, serta fidelitas operasi yang cukup tinggi.

Surface Code menjadi salah satu pendekatan QEC yang banyak diteliti karena cocok untuk arsitektur dua dimensi dengan interaksi lokal. Eksperimen Google Quantum AI menunjukkan bahwa *surface code distance-5* dapat memberikan kinerja logis sedikit lebih baik dibandingkan rata-rata beberapa *logical qubit distance-3*, yang menandakan bahwa peningkatan ukuran kode dapat mulai memberi manfaat ketika perangkat fisik cukup baik [3]. Perkembangan berikutnya menunjukkan hasil koreksi kesalahan di bawah ambang *surface code*, yang menjadi tonggak penting karena secara teori error logis dapat ditekan dengan memperbesar kode apabila error fisik berada di bawah ambang tertentu [4].

Selain perbaikan *hardware*, *decoder* juga menjadi bagian penting dari arsitektur *fault-tolerant*. *Decoder* bertugas menafsirkan hasil *syndrome* untuk menentukan koreksi yang paling mungkin. Penelitian Alpha *qubit* menunjukkan bahwa *decoder* berbasis jaringan saraf dapat meningkatkan akurasi decoding *surface code* pada data

eksperimental dan tetap kompetitif pada skala lebih besar [6]. Hal ini menunjukkan bahwa transformasi arsitektur komputer kuantum tidak hanya terjadi di level *hardware*, tetapi juga pada integrasi kecerdasan buatan, algoritma decoding, dan kontrol *real-time*. Namun, keberhasilan decoder berbasis jaringan saraf pada data eksperimental belum menjawab pertanyaan mengenai latensi real-time pada skala qubit logis yang jauh lebih besar, karena setiap siklus koreksi kesalahan harus diselesaikan lebih cepat daripada waktu dekoherensi qubit. Dengan demikian, mitigasi error, QEC, dan decoding bukan tiga solusi alternatif yang dapat dipilih salah satu, melainkan tiga lapisan pertahanan yang saling melengkapi pada tingkat overhead yang berbeda – dan ketiganya pada akhirnya membutuhkan kapasitas komputasi klasik berkecepatan tinggi untuk berjalan beriringan dengan QPU. Kebutuhan inilah yang membawa pembahasan pada arsitektur integrasi hibrida CPU/GPU/QPU sebagai satu sistem yang utuh, bukan sebagai lapisan terpisah.

5. Integrasi Hibrida CPU/GPU/QPU dalam Komputasi Modern

Arah pengembangan yang paling realistis saat ini adalah komputasi hibrida. Dalam model ini, komputer klasik menjalankan bagian komputasi yang deterministik, pengolahan data, optimasi parameter, serta orkestrasi tugas, sedangkan QPU digunakan untuk subproses tertentu yang berpotensi memperoleh keuntungan dari dinamika kuantum. Pola ini terlihat pada algoritma *variational quantum eigensolver* dan QAOA, ketika QPU mengeksekusi sirkuit parameterik dan komputer klasik memperbarui parameter berdasarkan hasil pengukuran. Karena hasil pengukuran kuantum bersifat probabilistik, proses ini biasanya dilakukan berulang hingga memperoleh estimasi yang stabil.

Konsep *quantum-centric supercomputing* menempatkan QPU sebagai bagian dari ekosistem superkomputasi, bukan sebagai perangkat terpisah. IBM mendefinisikan *quantum-centric supercomputing* sebagai pendekatan yang menggabungkan quantum computing dengan *high-performance computing* melalui infrastruktur, middleware, dan aplikasi yang dirancang bersama [11]. Model ini penting karena banyak masalah praktis membutuhkan kombinasi antara simulasi klasik, akselerasi GPU, penyimpanan data besar, dan komputasi kuantum. Dengan demikian,

transformasi komputasi modern kemungkinan tidak berbentuk migrasi total dari klasik ke kuantum, tetapi berbentuk penambahan akselerator kuantum ke dalam *workflow* komputasi ilmiah dan industri.

Resource estimation juga menjadi komponen penting dalam arsitektur hibrida dan *fault-tolerant*. Sebelum algoritma dijalankan pada komputer kuantum berskala besar, peneliti perlu memperkirakan jumlah *qubit* logis, jumlah *qubit* fisik, kedalaman sirkuit, waktu eksekusi, dan kebutuhan koreksi kesalahan. Microsoft *Azure Quantum Resource Estimator* dirancang untuk membantu menganalisis kebutuhan sumber daya algoritma kuantum pada berbagai konfigurasi *hardware* dan *error rate* [14]. Dengan alat seperti ini, pengembangan aplikasi kuantum dapat dilakukan lebih realistis karena algoritma tidak hanya dinilai dari kompleksitas teoritis, tetapi juga dari kebutuhan perangkat fisik. Relevansi kerangka ini terhadap komputasi modern terlihat pada beberapa contoh konkret. Pada sektor kimia dan farmasi, VQE mulai digunakan oleh penyedia layanan cloud kuantum sebagai modul yang dipanggil dari *workflow* simulasi material berbasis HPC, bukan sebagai aplikasi yang berjalan sepenuhnya pada QPU. Pada sektor logistik dan keuangan, QAOA diuji untuk masalah optimasi kombinatorial berskala kecil hingga menengah, dengan QPU menangani eksplorasi ruang solusi dan komputer klasik menangani agregasi serta validasi hasil. Pada keamanan siber, migrasi menuju kriptografi pascakuantum [13] menunjukkan bahwa dampak komputer kuantum terhadap industri tidak selalu berbentuk akselerasi, tetapi juga berbentuk mitigasi risiko terhadap infrastruktur klasik yang sudah berjalan. Ketiga contoh ini memperkuat argumen bahwa arsitektur hibrida bukan sekadar solusi sementara pada era NISQ, melainkan pola integrasi yang kemungkinan tetap relevan bahkan setelah komputer kuantum *fault-tolerant* tercapai, karena tugas pengolahan data berskala besar tetap lebih efisien diselesaikan secara klasik.

6. Potensi Transformasi dan Keterbatasan Komputer Kuantum

Potensi transformasi komputer kuantum terletak pada beberapa domain utama:

- 1. Simulasi molekuler dan material: Algoritma seperti VQE berpotensi mempercepat penemuan material baru, katalis, dan obat-obatan dengan mensimulasikan dinamika elektron secara akurat.
- 2. Optimasi kombinatorial: QAOA berpotensi memberikan keuntungan pada masalah optimasi seperti penjadwalan logistik dan alokasi sumber daya, meskipun keunggulan atas metode klasik terbaik masih terus diteliti.
- 3. Kecerdasan buatan: Quantum kernel methods dan quantum neural networks sedang dieksplorasi, meskipun keunggulan nyata atas model klasik masih terbatas pada kasus tertentu.
- 4. Keamanan siber: Algoritma Shor berpotensi memecahkan kriptografi RSA pada komputer kuantum berskala besar, mendorong NIST merilis standar kriptografi pascakuantum pada tahun 2024 [13].
- 5. Komputasi ilmiah berskala besar: Simulasi sistem fisika kompleks, meteorologi, dan astrofisika dapat memperoleh akselerasi signifikan dari komputasi kuantum hibrida.

Namun, sebagian besar aplikasi kuantum yang dipublikasikan masih bersifat demonstrasi konsep (proof of concept), bukan implementasi yang mengalahkan metode klasik terbaik secara konsisten [8]. Dekoherensi, noise, dan kedalaman sirkuit yang terbatas menjadi hambatan utama. Tabel 4 merangkum perbandingan antara fase NISQ saat ini dan visi FTQC masa depan.

Tabel 4. Karakteristik NISQ vs. Fault-Tolerant Quantum Computing (FTQC)

Aspek	NISQ (Saat Ini)	FTQC (Target Masa Depan)
Jumlah Qubit fisik	Puluhan – ribuan	Jutaan (estimasi)
Qubit logis	Belum / sangat terbatas	Ribuan – jutaan
Error rate gerbang	~0,1% – 1%	<0,001% (di bawah ambang)
Koreksi kesalahan	Mitigasi error (parsial)	QEC penuh (Surface code, dll.)
Kedalaman sirkuit	Puluhan – ratusan gerbang	Miliaran gerbang logis

Contoh algoritma	VQE, QAOA (terbatas)	Shor (kriptografi), simulasi kimia
Status kesiapan	Tersedia secara komersial (cloud quantum)	Estimasi 10–20 tahun ke depan
Keunggulan atas klasik	Terbatas / dalam penelitian	Diharapkan pada problem spesifik

KESIMPULAN

Berdasarkan kajian yang telah dilakukan, arsitektur komputer kuantum merupakan sistem berlapis yang mencakup *qubit* fisik, konektivitas, kontrol perangkat, *compiler*, sirkuit kuantum, sistem mitigasi atau koreksi kesalahan, serta integrasi dengan komputer klasik. Jumlah *qubit* bukan satu-satunya ukuran kemajuan karena kualitas *qubit*, fidelitas gerbang, kedalaman sirkuit, kemampuan koreksi kesalahan, dan stabilitas kontrol memiliki pengaruh besar terhadap kinerja keseluruhan.

Platform *qubit* seperti superkonduktor, ion terperangkap, atom netral, fotonik, dan spin silikon memiliki karakteristik yang berbeda. Tidak ada satu platform yang sepenuhnya unggul untuk semua kebutuhan. Superkonduktor kuat dari sisi ekosistem dan kecepatan gerbang, ion terperangkap unggul pada fidelitas dan koherensi, atom netral menjanjikan skala array dan konektivitas fleksibel, fotonik relevan untuk jaringan kuantum, sedangkan spin silikon menarik untuk integrasi mikroelektronika jangka panjang.

Potensi transformasi komputer kuantum terletak pada simulasi molekuler, material, optimasi, kecerdasan buatan, keamanan siber, dan komputasi ilmiah berskala besar. Namun, pemanfaatan praktis masih menghadapi tantangan noise, dekoherensi, overhead koreksi kesalahan, keterbatasan algoritma, dan kebutuhan infrastruktur. Karena itu, arah yang paling realistis saat ini adalah arsitektur hibrida CPU/GPU/QPU yang memadukan kekuatan komputasi klasik dan kuantum. Secara teoritis, kajian ini berkontribusi dengan menempatkan jumlah *qubit*, fidelitas gerbang, koneksi kontrol, dan kapasitas koreksi kesalahan sebagai satu rantai dependensi

lintas-lapisan, bukan sebagai daftar metrik yang berdiri sendiri, sehingga kemajuan arsitektur komputer kuantum dapat dievaluasi secara lebih sistemik. Secara praktis, kerangka ini berimplikasi pada pengambilan keputusan: pengembang aplikasi dapat menggunakannya untuk menilai kesiapan suatu platform berdasarkan kebutuhan kedalaman sirkuit dan toleransi error aplikasinya, sementara pembuat kebijakan riset dapat menggunakannya untuk mengarahkan investasi pada lapisan yang paling membatasi kinerja sistem secara keseluruhan, bukan semata-mata pada peningkatan jumlah qubit.

Keterbatasan artikel ini adalah sifatnya sebagai kajian literatur naratif sehingga belum melakukan pengujian eksperimen terhadap perangkat kuantum tertentu, dan klasifikasi keunggulan-keterbatasan platform pada Tabel 2 disusun dari laporan masing-masing pengembang teknologi sehingga berpotensi belum sepenuhnya netral antarplatform. Kerangka integratif yang diajukan juga belum diuji secara kuantitatif terhadap metrik kinerja nyata (misalnya quantum volume atau circuit layer operations per second), sehingga sifatnya masih konseptual. Penelitian selanjutnya disarankan membandingkan performa beberapa platform cloud quantum melalui studi kasus algoritma yang sama, misalnya VQE atau QAOA, serta mengevaluasi pengaruh transpiler, noise model, dan error mitigation terhadap kualitas hasil. Secara lebih spesifik, penelitian lanjutan dapat diarahkan pada tiga jalur: pertama, benchmarking kuantitatif satu algoritma (misalnya QAOA pada masalah max-cut) pada minimal dua penyedia cloud kuantum berbasis platform berbeda (superkonduktor vs ion terperangkap) dengan metrik approximation ratio yang konsisten; kedua, evaluasi kuantitatif pengaruh tingkat optimasi transpiler terhadap kedalaman sirkuit dan fidelitas hasil pada sirkuit berukuran sedang (10–20 qubit); dan ketiga, pengembangan model estimasi biaya komputasi hibrida yang menggabungkan waktu eksekusi klasik dan kuantum sebagai metrik tunggal, sehingga kelayakan arsitektur hibrida dapat dinilai secara kuantitatif, bukan hanya konseptual seperti pada kajian ini.

DAFTAR PUSTAKA

- [1]. J. Preskill, "Quantum computing 40 years later," *arXiv:2106.10522*, 2021.
- [2]. S. Chen, J. Cotler, H.-Y. Huang, and J. Li, "The complexity of NISQ," *Nature Communications*, vol. 14, Art. no. 6001, 2023.

- [3]. Google Quantum AI, "Suppressing quantum errors by scaling a *Surface code* logical Qubit," *Nature*, vol. 614, pp. 676–681, 2023.
- [4]. Google Quantum AI and Collaborators, "Quantum error correction below the *Surface code* threshold," *Nature*, vol. 638, pp. 920–926, 2025.
- [5]. D. Bluvstein et al., "Logical quantum processor based on reconfigurable atom arrays," *Nature*, vol. 626, pp. 58–65, 2024.
- [6]. J. Bausch et al., "Learning high-accuracy error decoding for quantum processors," *Nature*, vol. 635, pp. 834–840, 2024.
- [7]. A. Morvan et al., "Phase transitions in random circuit sampling," *Nature*, vol. 634, pp. 328–333, 2024.
- [8]. X. D. Xie et al., "Advances in quantum computation in NISQ era," *Entropy*, vol. 27, no. 10, Art. no. 1074, 2025.
- [9]. W. van Dam, M. Mykhailova, and M. Soeken, "Using Azure Quantum Resource Estimator for assessing performance of fault tolerant quantum computation," *arXiv:2311.05801*, 2023.
- [10]. IBM Quantum, "IBM Quantum Roadmap," *IBM Technology Atlas*, 2024.
- [11]. R. Mandelbaum, J. Schneider, and I. Smalley, "What is *quantum-centric supercomputing?*," *IBM Think*, Sep. 12, 2024, updated Apr. 14, 2026.
- [12]. IBM Quantum Documentation, "Introduction to *transpilation*," *IBM Quantum Platform Documentation*, 2026.
- [13]. National Institute of Standards and Technology, "NIST releases first 3 finalized post-quantum encryption standards," *NIST*, Aug. 13, 2024.
- [14]. Microsoft Learn, "Introduction to the resource estimator - Azure Quantum," *Microsoft Azure Quantum Documentation*, Jan. 29, 2026.