

コンピュータ構造

1. 定義: コンピュータ組織は、コンピュータシステムの操作構造と実装に関するもので、ハードウェアコンポーネントが指示を実行する方法に焦点を当てています。
2. 歴史的な進化: 初期の機械式コンピュータから現代のマルチコアプロセッサまでの発展をたどります。
3. ボン・ノイマンアーキテクチャ: CPU、メモリ、I/O がバスを通じて接続される基本的なモデルです。
4. ハーバードアーキテクチャ: 命令とデータのストレージと信号バスを分離し、パフォーマンスを向上させます。
5. CPU コンポーネント: 算術論理ユニット (ALU)、制御ユニット (CU)、レジスタを含みます。
6. ALU の機能: 加算、減算、AND、OR などの算術および論理演算を実行します。
7. 制御ユニットの役割: プロセッサの操作を指示するために指示をデコードし、制御信号を生成します。
8. レジスタ: CPU 内の小さく高速なストレージ場所で、データと指示を一時的に保持します。
9. キャッシュメモリ: データアクセス時間を短縮するために CPU に近い高速メモリです。
10. メモリ階層: 速度とコストに基づいてレジスタ、キャッシュ、RAM、セカンダリストレージを含むメモリのレベルに分類します。
11. RAM (ランダムアクセスメモリ) : 現在使用中のデータとマシンコードを保存するための揮発性メモリです。
12. ROM (読み取り専用メモリ) : ファームウェアとシステムブート指示を保存するための非揮発性メモリです。
13. バス構造: コンピュータ内外のコンポーネント間でデータを転送する通信システムです。
14. データバス: 処理中の実際のデータを運びます。
15. アドレスバス: データが送信または取得される場所に関する情報を運びます。
16. 制御バス: CPU から他のコンポーネントへの制御信号を運びます。
17. 指令セットアーキテクチャ (ISA) : CPU が実行できる指令のセットを定義します。
18. RISC (リダクストインストラクションセットコンピューティング) : 小さく高度に最適化された指令セットを使用する ISA 設計哲学です。
19. CISC (コンプレックスインストラクションセットコンピューティング) : 大量の指令を含む ISA で、一部の指令は複雑なタスクを実行できます。
20. パイプライン: CPU のスループットを向上させるために複数の指令フェーズを重ねる技術です。
21. パイプラインステージ: 通常、フェッチ、デコード、実行、メモリアクセス、ライトバックを含みます。

22. パイプラインのハザード: データハザード、制御ハザード、構造ハザードなど、パイプラインフローを乱す問題です。
23. 分岐予測: パイプラインを満たすために分岐指令の方向を予測する方法です。
24. スーパースカラアーキテクチャ: 単一のパイプラインステージで複数の指令を同時に処理できます。
25. 並列処理: 複数のプロセッサまたはコアを使用して指令を同時実行します。
26. マルチコアプロセッサ: 単一のチップに複数の処理コアが統合されたCPUです。
27. SIMD (シングルインストラクション、マルチブルデータ) : 単一の指令が同時に複数のデータポイントに作用する並列処理アーキテクチャです。
28. MIMD (マルチブルインストラクション、マルチブルデータ) : 複数のプロセッサが異なるデータに対して異なる指令を実行する並列アーキテクチャです。
29. メモリ管理: ページングとセグメンテーションを含むメモリを効率的に管理および割り当てる技術です。
30. バーチャルメモリ: ディスクストレージに物理メモリを拡張し、システムがより大きなワークロードを処理できるようにします。
31. ページング: メモリ管理を簡素化し断片化を減少させるためにメモリを固定サイズのページに分割します。
32. セグメンテーション: 関数やデータ構造などの論理的な分割に基づいてメモリを可変サイズのセグメントに分割します。
33. キャッシュマッピング技術: ディレクトマップ、フルアソシエイティブ、セットアソシエイティブキャッシュを含みます。
34. キャッシュ置換ポリシー: 置換するキャッシュエントリを決定するもので、最新の使用 (LRU) または先入れ先出し (FIFO) などがあります。
35. キャッシュ貫性: マルチプロセッサシステム内の複数のキャッシュに保存されたデータの一貫性を確保します。
36. メモリー貫性モデル: システムの一貫性を維持するために操作が実行される順序を定義します。
37. 入出力システム: コンピュータと外部デバイス間の通信を管理します。
38. I/O デバイスの分類: 入力デバイス、出力デバイス、ストレージデバイスを含みます。
39. I/O インターフェース: デバイスがマザーボードと通信する方法を定義する標準で、USB、SATA、PCIe などがあります。
40. 直接メモリアクセス (DMA) : CPU の介入なしでデバイスがメモリにデータを転送できます。
41. インタラプト: CPU に即時の注意が必要なイベントを通知する信号で、非同期処理を可能にします。
42. インタラプト処理: CPU がインタラプトに応答するプロセスで、状態の保存とインタラプトサービスルーチンの実行を含みます。

43. DMA コントローラ: DMA 操作を管理し、CPU からデータ転送タスクを解放するハードウェアコンポーネントです。
44. デバイスドライバ: オペレーティングシステムがハードウェアデバイスと通信できるようにするソフトウェアです。
45. ペリフェラルコンポーネントインターフェース (PCI) : マザーボードにペリフェラルを接続するための標準です。
46. シリアル対パラレル通信: シリアルは 1 ビットずつデータを送信し、パラレルは複数のビットを同時に送信します。
47. シリアルポート: デバイスとのシリアル通信に使用されるインターフェースで、RS-232 などがあります。
48. パラレルポート: プリンターや他のペリフェラルとのパラレル通信に使用されるインターフェースです。
49. バス仲裁: 複数のデバイス間でバスへのアクセスを管理し、競合を防ぎます。
50. システムバス対ペリフェラルバス: システムバスは CPU、メモリ、主要コンポーネントを接続し、ペリフェラルバスは外部デバイスを接続します。
51. インタラプトベクタテーブル: インタラプトサービスルーチンのアドレスを保存するデータ構造です。
52. プログラマブルインタラプトコントローラ: 複数のインタラプト要求を管理し優先順位付けするハードウェアです。
53. バス幅: バスを通じて同時に転送できるビット数です。
54. クロック速度: CPU が指令を実行する速度で、GHz で測定されます。
55. クロックサイクル: CPU が基本操作を実行できる基本的な時間単位です。
56. クロックスキュー: 回路の異なる部分に到着するクロックスイグナルの到着時間の違いです。
57. クロック配布: CPU 内のすべてのコンポーネントにクロックスイグナルを配達する方法です。
58. 热放散: CPU の過熱を防ぐために過剰な熱を取り除くプロセスです。
59. クーリングソリューション: CPU 温度を管理するために使用されるヒートシンク、ファン、液冷システムを含みます。
60. 電源ユニット (PSU) : すべてのコンピュータコンポーネントに必要な電力を供給します。
61. 電圧調整器: CPU や他のコンポーネントに安定した電圧レベルを供給します。
62. マザーボードアーキテクチャ: CPU、メモリ、他の重要なコンポーネントを収容するメイン回路基板です。
63. チップセット: CPU、メモリ、ペリフェラル間のデータフローを管理する統合回路のグループです。
64. ファームウェア: ハードウェア機能を制御するために読み取り専用メモリにプログラムされた永続的なソフトウェアです。

65. BIOS/UEFI: ブートプロセス中にハードウェアを初期化し、ランタイムサービスを提供するファームウェアインターフェースです。
66. ブートプロセス: システムが電源を入れたときに初期化される操作のシーケンスです。
67. 指令パイプラインステージ: 通常、フェッチ、デコード、実行、メモリアクセス、ライトバックを含みます。
68. パイプラインの深さ: パイプラインのステージ数で、指令スループットとレイテンシーに影響を与えます。
69. パイプラインのバランス: 各ステージがほぼ等しい実行時間を持つようにして効率を最大化します。
70. データハザード: パイプライン内の前の指令の結果に依存する指令の状況です。
71. 制御ハザード: パイプラインフローを乱す分岐指令によるものです。
72. 構造ハザード: 同時にすべての可能な指令実行をサポートするためのハードウェアリソースが不足している場合に発生します。
73. フォワード (データバイパス): データハザードを軽減するためにデータをパイプラインステージ間で直接ルーティングする技術です。
74. スタル (パイプラインバブル): ハザードを解決するためにパイプラインに空のサイクルを挿入します。
75. アウトオブオーダー実行: リソースが利用可能になるにつれて指令を実行し、元のプログラム順序ではなく実行します。
76. 仮想実行: 必要かどうかが確認される前に指令を実行し、パフォーマンスを向上させます。
77. 分岐予測アルゴリズム: 分岐方向を予測するために使用される静的予測、動的予測、2段階適応予測などの技術です。
78. 指令レベル並列性 (ILP): 単一のCPUサイクル内で複数の指令を同時に実行できる能力です。
79. ループアンローリング: ループ制御のオーバーヘッドを減少させるためにループ本体を増やす最適化技術です。
80. スーパーパイプライン: より高いクロックスピードを許可するためにパイプラインステージの数を増やします。
81. VLIW (ベリーロングインストラクションウォード): 単一の指令ワードに複数の操作をエンコードできるアーキテクチャです。
82. EPIC (エクスプリシットパラレルインストラクションコンピューティング): コンパイラのアシストを通じて並列指令実行を可能にするアーキテクチャです。
83. レジスタリネーミング: 偽のデータ依存関係を排除するためにレジスタを動的に割り当てる技術です。
84. ハイパースレッディング: 単一のCPUコアが同時に複数のスレッドを実行できるIntelの技術です。

85. キャッシュメモリレベル: L1 (CPU に最も近く、最も速い)、L2、L3 キャッシュで、サイズとレイテンシーが増加します。
86. ライトスルー対ライトバックキャッシュ: ライトスルーはキャッシュとメモリを同時に更新し、ライトバックはキャッシュのみを更新し、メモリ更新を後で行います。
87. キャッシュのアソシエイティビティ: キャッシュラインがキャッシュセットにマッピングされる方法で、ヒット率とアクセス時間に影響を与えます。
88. プレフェッチ: アクセスレイテンシーを短縮するためにデータを実際に要求される前にキャッシュに読み込みます。
89. メモリアクセスパターン: シーケンシャル対ランダムアクセスとキャッシュパフォーマンスに与える影響です。
90. NUMA (ノンユニフォームメモリアクセス) : メモリアクセス時間がプロセッサに関連するメモリの場所によって異なるメモリ設計です。
91. SMP (シメトリックマルチプロセッシング) : 複数のプロセッサが単一の中央集権型メモリを共有するシステムです。
92. 分散メモリシステム: 各プロセッサが独自のプライベートメモリを持ち、ネットワークを通じて通信するシステムです。
93. インターコネクションサービス: 複数のプロセッサとメモリユニットを接続するためのトポロジーとプロトコルです。
94. スケーラビリティ: リソースを追加することでシステムのパフォーマンスを向上させる能力です。
95. フォールトレランス: 一部のコンポーネントが故障してもシステムが正常に動作し続ける能力です。
96. レダンダンシー: 信頼性と利用可能性を高めるために追加のコンポーネントを組み込みます。
97. エラーデテクションと修正: パリティビット、チェックサム、ECC (エラーコレクティングコード) などの技術を使用してデータエラーを検出および修正します。
98. 電力効率: パフォーマンスを維持しながら電力消費を最小限に抑えるシステムの設計です。
99. 熱設計電力 (TDP) : 典型的なワークロード下で CPU または GPU が生成する予想される最大の熱量です。
100. 将来のトレンド: クオンタムコンピューティング、ニューロモルフィックアーキテクチャ、フォトニックプロセッサなど、コンピュータ組織の将来を形作る進歩を探します。