Manutenção de Computadores

Livro 2 — versão 0.1

João Paulo Ferreira Guimarães

4 de agosto de 2026

Sumário

1 Fundamentos e Abstração

Neste capítulo você vai estudar a manutenção de computadores como disciplina técnica — sua definição, sua divisão em manutenção corretiva e preventiva, e o raciocínio de diagnóstico que sustenta o trabalho do técnico de informática — e vai reconstruir, camada por camada, a ponte entre o software que se escreve e o hardware que efetivamente o executa, do transistor até a arquitetura de von Neumann. Este livro dá continuidade à disciplina de Organização e Montagem de Computadores, mas não pressupõe que ela tenha sido cursada: os conceitos de hardware, software e sistema computacional necessários são retomados aqui.

1.1 Manutenção de computadores: definição e escopo

Este livro adota a seguinte definição, comum na literatura técnica de manutenção industrial e plenamente aplicável à informática:

> "Manutenção é a combinação de todas as ações técnicas e administrativas, incluindo supervisão, destinadas a manter ou recolocar um item em estado no qual possa desempenhar uma função requerida." [1]

Essa definição contém uma implicação frequentemente ignorada por quem está começando na área: manutenção não é sinônimo de conserto. Todo computador precisa de manutenção — não apenas os que já apresentam defeito. Essa distinção é o que separa a **manutenção corretiva** (agir depois que a falha ocorreu) da **manutenção preventiva** (agir antes que ela ocorra), tratadas em detalhe na Seção 1.3.

1.2 O sistema computacional: hardware, software e pessoas

Como já visto no estudo introdutório de computadores, um sistema computacional não é formado apenas por hardware e software: ele inclui também as **pessoas** responsáveis por sua operação. Para a manutenção, essa tríade

— hardware, software, pessoas — é o primeiro filtro de qualquer diagnóstico: diante de um chamado técnico, a primeira pergunta não é “qual é o defeito?”, mas “em qual dessas três frentes está o problema?”.

Exemplo. Um chamado técnico relata que “o mouse não está funcionando”. Existem, ao menos, três hipóteses plausíveis, uma em cada frente do sistema computacional:

Frente	Hipótese	Diagnóstico
Software	O driver do *trackpad* do notebook não está instalado	O sistema operacional não reconhece o dispositivo apontador
Hardware	O mouse físico está com defeito ou sem pilha	O componente em si não opera
Pessoa (usuário)	O mouse é sem fio e não está pareado com o computador	O hardware e o software estão corretos; falta uma ação do usuário

Repare que, no terceiro caso, não há defeito de hardware nem de software: o problema está inteiramente na operação. É comum que problemas relatados como falha técnica sejam, na prática, erro de uso — por isso o técnico precisa investigar as três frentes antes de concluir qualquer diagnóstico.

Figura pendente

diagrama “sistema computacional = hardware + software + pessoas” com o exemplo do mouse ramificando nas três hipóteses

1.3 Manutenção corretiva e manutenção preventiva

- **Manutenção corretiva**: conjunto de ações tomadas para sanar um problema já identificado. O computador chega com sintomas de um comportamento indevido, e a tarefa do técnico é, a partir desses sintomas, identificar o defeito e realizar a substituição ou correção — seja em hardware (troca de um módulo), seja em software (reinstalação ou reconfiguração de um programa).
- **Manutenção preventiva**: conjunto de ações tomadas para reduzir a probabilidade de que um problema venha a ocorrer, executadas antes de qualquer falha se manifestar.

Não existe uma periodicidade única para a manutenção preventiva — ela depende do tipo de ação e das condições de uso do equipamento.

Exemplo. No ambiente de laboratório de um campus, os computadores costumam ser reinstalados uma vez por ano letivo, prática que reduz o índice de problemas ao longo do semestre; sob uso muito mais intenso, esse intervalo poderia cair para seis meses. Já a cópia de segurança (*backup*) de dados de uso profissional é uma ação de manutenção preventiva que deve ser executada diariamente: um computador sem rotina de backup que sofre uma falha grave perde integralmente os dados que não foram copiados.

Analogia. A ausência de uma regra fixa de periodicidade é comparável à manutenção automotiva: a troca de óleo e de determinadas peças de um carro de uso severo precisa ser mais frequente do que a de um carro de uso ocasional. Em ambos os casos — carro e computador —, a periodicidade correta é definida pelas condições reais de trabalho, não por uma tabela genérica.

Manutenção corretiva e preventiva se aplicam às três frentes do sistema computacional (Seção 1.2): existem ações preventivas relacionadas a hardware (limpeza, inspeção), a software (backup, atualização) e também a pessoas — treinamento e orientação do usuário para reduzir erros de operação recorrentes.

1.3.1 Condições reais e ideais de trabalho

A qualidade de uma manutenção não depende só do diagnóstico correto (Seção 1.4): depende também das condições físicas em que o reparo é executado. Três fatores concentram a maior parte do risco evitável numa bancada de manutenção.

Controle eletrostático (ESD). O corpo humano acumula, por atrito simples (caminhar sobre um carpete, por exemplo), uma carga eletrostática que pode chegar a milhares de volts — imperceptível ao toque, mas suficiente para danificar permanentemente um circuito integrado sensível, cuja tolerância a descargas é medida em dezenas ou centenas de volts [2]. A condição ideal de bancada inclui uma superfície de trabalho **antiestática** (um tapete condutor aterrado) e, quando disponível uma instalação de aterramento confiável (Capítulo 2, §2.4), uma pulseira antiestática conectando o técnico a essa mesma referência de terra — a mesma ressalva já feita no Capítulo 2 (§2.11.2) se aplica aqui: só vale a pena se conectar a um aterramento em que se confia.

Umidade e risco de condensação. Um componente eletrônico transportado de um ambiente frio (por exemplo, um veículo com ar-condicionado) para um ambiente quente e úmido pode sofrer condensação de água em sua superfície — o mesmo fenômeno que embaça um copo gelado num dia quente. Água condensada sobre um circuito energizado é um risco direto de curto-circuito. A prática recomendada é deixar o equipamento estabilizar à temperatura ambiente, ainda desligado, antes de energizá-lo.

Organização como parte do método, não só do ambiente. Ao desmontar um equipamento com múltiplos parafusos e cabos semelhantes entre si (Capítulo 4, do livro de Organização e Montagem de Computadores, trata o procedimento de desmontagem em detalhe), separar e identificar

cada peça na ordem de remoção evita o erro mais comum de reparo malsucedido: a remontagem incorreta. Essa organização não é uma questão de estética de bancada — é uma extensão direta do método de diagnóstico por hipóteses (Seção 1.4): um técnico que não sabe de onde veio cada parafuso não consegue isolar, com confiança, se um problema após a remontagem foi causado pelo reparo em si ou por uma montagem incorreta.

1.4 O raciocínio diagnóstico: hipóteses e método científico

A habilidade central desenvolvida ao longo desta disciplina é a de **isolar um problema** dentro do sistema computacional. Essa habilidade, como programar ou nadar, não se aprende apenas lendo — desenvolve-se pela prática repetida.

O procedimento subjacente ao diagnóstico técnico reproduz o método científico: a partir de uma observação (o sintoma relatado), formula-se uma **hipótese** que explique aquele comportamento, testa-se essa hipótese e verifica-se se ela é válida ou deve ser descartada em favor de outra.

Exemplo. Um computador entra em ciclo de reinicialização (POST bem-sucedido, tela de instalação do sistema operacional trava e reinicia). As hipóteses possíveis incluem: defeito na memória secundária, imagem de instalação corrompida, problema de configuração da máquina, ou defeito de hardware específico daquele computador. Um teste direcionado — por exemplo, conectar o disco em outra máquina e rodar um software de diagnóstico de saúde do disco (percentual de blocos defeituosos) — permite confirmar ou descartar a hipótese do disco sem precisar investigar as demais.

Um princípio prático orienta a ordem em que as hipóteses devem ser testadas: **teste primeiro a hipótese mais barata de verificar**, não necessariamente a mais provável. Tempo é um recurso escasso no trabalho técnico; investigar uma hipótese complexa (por exemplo, desmontar o computador para testar um componente em outra máquina) antes de descartar uma hipótese simples (por exemplo, trocar a imagem de instalação e tentar de novo) desperdiça tempo caso a causa fosse, de fato, a mais simples.

Esse tipo de raciocínio — geração de hipóteses, priorização por custo de verificação, teste e confirmação ou descarte — não é adquirido de uma vez; desenvolve-se ao longo da prática cotidiana como técnico, à medida que se acumula repertório de problemas já vistos e resolvidos.

1.5 Malware, vírus e cavalos de troia

Um problema de software identificado durante o diagnóstico pode ter origem em **software malicioso** (*malware*) instalado no computador sem o conhecimento ou consentimento do usuário.

Analogia. A relação entre vírus e malware é análoga à relação entre gripe e doença: toda gripe é uma doença, mas nem toda doença é uma gripe. Da mesma forma, todo vírus é um software malicioso, mas nem todo software malicioso é um vírus.

- **Vírus**: software malicioso cuja característica definidora é a **replicação** — a capacidade de se propagar de um computador para outro, contaminando novas máquinas.
- **Trojan** (cavalo de troia): software que se apresenta como um programa legítimo, mas que traz embutido um código malicioso oculto, executado em segundo plano após a instalação.

Exemplo. Em 2023, o maior canal de tecnologia do YouTube, o Linus Tech Tips, teve seus canais sequestrados após um funcionário da área de publicidade abrir um PDF contaminado recebido como proposta comercial. O software malicioso obteve acesso ao navegador da máquina infectada e, através das permissões já concedidas àquele computador, passou a publicar em todos os subcanais do grupo vídeos promovendo um golpe de criptomoeda associado à imagem de uma figura pública. Levou cerca de 24 horas para a equipe recuperar o controle dos canais [3] — todo o alcance da rede havia sido redirecionado para o conteúdo malicioso antes que o acesso fosse restabelecido.

Esse caso ilustra por que a origem de um software instalado importa: um malware pode chegar embutido em qualquer instalador, mesmo em arquivos aparentemente inofensivos como um documento PDF.

1.6 Reinstalação do sistema operacional como manutenção corretiva

Diante de um problema de software cuja causa exata não foi identificada, uma prática comum — ainda que nem sempre a mais eficiente — é reinstalar o sistema operacional por completo, apagando o disco e recomeçando do zero.

Analogia. Esse procedimento é comparado, na prática do curso, a "jogar uma bomba atômica em cima de uma mosca": resolve o problema, mas de forma desproporcional ao esforço realmente necessário. Um técnico experiente, que já reconhece o problema específico, tende a resolver a questão pontual (por exemplo, reconfigurar um ícone, restaurar uma barra de tarefas que desapareceu, desinstalar um programa específico) sem recorrer à reinstalação completa. Um técnico iniciante, sem esse repertório, tende a usar a reinstalação como solução padrão — o que resolve o problema, mas consome muito mais tempo do que seria necessário.

Ainda assim, a reinstalação continua sendo uma ferramenta legítima de manutenção corretiva, sobretudo quando o tempo de diagnóstico pontual excederia o tempo do próprio procedimento de reinstalação. Duas implicações são obrigatórias sempre que esse caminho é adotado:

1. **Perda de dados**: a reinstalação apaga o disco. É dever do técnico alertar o usuário e obter confirmação de que uma cópia de segurança (backup) dos dados relevantes foi realizada antes de iniciar o procedimento.
2. **A reinstalação só resolve problemas de software**: se o sintoma reaparecer após uma reinstalação bem-sucedida, a hipótese de defeito de hardware sobe de prioridade.

Figura pendente

fluxograma de decisão — sintoma relatado → hipótese hardware/software/usuário → teste da hipótese mais barata → correção pontual ou reinstalação

1.7 Criação de mídia de instalação: Rufus, MBR/GPT e UEFI

A instalação de um sistema operacional requer uma mídia de inicialização (pendrive ou disco físico) contendo a imagem do sistema, a partir da qual o computador é inicializado antes de o sistema instalado normalmente ser carregado.

Procedimento de segurança. A imagem de instalação (arquivo ISO) deve ser obtida diretamente do fabricante — no caso do Windows, do site oficial da Microsoft. Imagens obtidas de fontes intermediárias não confiáveis podem vir acompanhadas de software malicioso embutido no próprio instalador.

O procedimento de criação de mídia envolve duas etapas conceitualmente distintas: baixar a imagem do sistema operacional (arquivo `.iso`) e, em seguida, gravar essa imagem em um pendrive de forma que ele se torne inicializável.

1.7.1 BIOS/MBR e UEFI/GPT

Computadores mais antigos utilizam firmware **BIOS** com tabela de partição **MBR**; computadores mais recentes utilizam firmware **UEFI** com tabela de partição **GPT**. Uma mídia de instalação criada para um padrão não inicializa corretamente uma máquina que utiliza o outro.

Procedimento de identificação. Ao inicializar o computador e entrar no *setup* (menu de configuração do firmware, acessado durante o POST): se o menu responde ao mouse, o firmware é UEFI/GPT; se o menu só é navegável pelo teclado, o firmware é BIOS/MBR legado.

1.7.2 Ferramentas de criação de mídia

Ferramenta	Uso recomendado
Assistente de instalação da Microsoft (*Media Creation Tool*)	Gera diretamente um pendrive UEFI/GPT (acerta para a maioria dos computadores novos) ou permite apenas baixar o arquivo ISO para uso posterior com outra ferramenta
Rufus	Ferramenta leve (poucos megabytes) que permite escolher explicitamente o tipo de sistema-alvo (BIOS/MBR ou UEFI/GPT); recomendada para gravar uma única imagem por vez com controle preciso do padrão de destino
YUMI	Permite reunir múltiplos instaladores (por exemplo, várias distribuições Linux e versões do Windows) em um único pendrive, funcionando como um "GRUB de instaladores"; menos direta de configurar que o Rufus
Ventoy	Alternativa multi-imagem semelhante ao YUMI

Especificações e comportamento de cada ferramenta conforme suas páginas oficiais [4].

O Assistente de instalação da Microsoft, quando usado no modo "unidade flash USB", automaticamente grava a mídia como UEFI/GPT — o que funciona para a grande maioria dos computadores novos, mas falha em máquinas antigas com BIOS/MBR. Para instalar em uma máquina legada, é necessário baixar apenas o arquivo ISO e utilizar o Rufus, selecionando manualmente o esquema de partição BIOS/MBR compatível com aquele hardware.

1.8 Plataforma e abstração em camadas

Todo software é desenvolvido para ser executado em um determinado local — o que este livro chama de **plataforma**. A plataforma de um software pode ser um programa (por exemplo, um navegador específico), um sistema operacional, ou, em última instância, uma arquitetura de hardware.

Exemplo. Um bloqueador de anúncios desenvolvido para o navegador Firefox tem como plataforma o próprio Firefox: o software não funciona no Chrome nem no Safari. Por sua vez, o Firefox tem como plataforma diversos sistemas operacionais (Windows, macOS, Linux, Android, iOS). E cada um desses sistemas operacionais tem como plataforma hardwares fisicamente

distintos — um processador Snapdragon num smartphone Android, um chip Apple M1 num MacBook, um processador Intel ou AMD num desktop Windows.

Essa organização em camadas — software sobre navegador, navegador sobre sistema operacional, sistema operacional sobre hardware — é chamada **abstração em camadas**. Ela é o motivo pelo qual um desenvolvedor de software para navegador não precisa se preocupar com qual câmera, qual placa de rede ou qual processador está instalado no dispositivo do usuário final: cada camada delega à camada abaixo dela a responsabilidade por lidar com a complexidade que está fora do seu escopo.

A estratégia de camadas não é exclusiva da relação hardware-software: a mesma lógica organiza, por exemplo, os protocolos de rede em camadas (física, enlace, transporte, aplicação). Este capítulo adota uma abordagem **top-down** (do software visível ao usuário até o hardware que o executa) para, nas seções seguintes, reconstruir a mesma pilha de forma **bottom-up** (do transistor até a linguagem de programação).

Figura pendente

pilha de camadas — software aplicativo → navegador → sistema operacional → hardware, com setas de dependência apontando para baixo

1.9 Do transistor às portas lógicas

Para entender como um software é efetivamente executado por um hardware, é necessário responder a uma pergunta simples em aparência: como um circuito eletrônico realiza uma operação lógica ou aritmética?

1.9.1 O combinado entre bit e tensão

Um programa como `x = 2` instrui o computador a reservar um espaço de memória, nomeá-lo `x` e armazenar ali o valor 2. Esse número, em base decimal, é convertido para binário antes de ser armazenado, porque a eletrônica digital opera sobre um **combinado** (convenção) entre os símbolos binários (0 e 1) e grandezas elétricas mensuráveis. Na lógica TTL, por exemplo, o combinado usual é: 5 volts representa o bit 1, e 0 volts representa o bit 0 [5]. Esse combinado é arbitrário — poderia ser outro par de tensões — mas precisa ser fixado entre fabricante e desenvolvedor para que a informação seja interpretada de forma consistente. **Nota técnica:** na prática, os níveis TTL reais são faixas, não valores exatos — um nível alto válido é qualquer tensão a partir de aproximadamente 2,4 V, e um nível baixo válido vai até aproximadamente 0,4–0,5 V; “5 V e 0 V” é a simplificação didática usual para os valores nominais de saída.

1.9.2 O semicondutor e o transistor

Um **semicondutor** é um material que, dependendo das condições (como a temperatura), pode se comportar como isolante ou como condutor — o silício é o exemplo mais usado na indústria. Ao dopar o silício com impurezas específicas, obtêm-se dois tipos de material: o tipo **N** (com excesso de elétrons) e o tipo **P** (com déficit de elétrons, ou "buracos"). A junção entre um material tipo P e um tipo N forma um **diodo**, que conduz corrente em apenas um sentido quando polarizado corretamente.

Empilhando três camadas alternadas (P-N-P ou N-P-N), obtém-se o **transistor**: um dispositivo de três terminais — coletor, base e emissor — que se comporta como uma chave eletromecânica. Quando há nível lógico alto (1) na base, o transistor fecha o contato entre coletor e emissor, permitindo a passagem de corrente; quando há nível lógico baixo (0) na base, o contato permanece aberto.

1.9.3 Construindo portas lógicas com transistores

A partir dessa chave elementar, é possível construir as portas lógicas estudadas em eletrônica digital:

- **Porta AND**: dois transistores ligados em série. A saída só apresenta nível lógico 1 se ambas as entradas estiverem em 1 — cada transistor precisa estar "fechado" para que a tensão chegue até a saída.
- **Porta OR**: dois transistores ligados em paralelo. A saída apresenta nível lógico 1 se qualquer uma das entradas estiver em 1 — basta que um dos dois caminhos esteja fechado.
- **Porta NOT (inversora)**: um único transistor usado como chave que inverte o sinal de entrada — 1 na entrada produz 0 na saída, e vice-versa.

Nota técnica. Esse modelo (série = AND, paralelo = OR) é uma simplificação pedagógica — um "modelo de chave" coerente com a lógica CMOS moderna — e não corresponde exatamente à topologia de circuitos comerciais TTL/DTL, que tradicionalmente implementam AND/OR com lógica a diodo e usam um transistor inversor separado [6]. O modelo aqui é útil para a intuição, mas não deve ser confundido com a topologia exata de um circuito integrado comercial.

Analogia. O ponto notável desse processo — e a razão de ele ser descrito em aula como "extraordinário" — é que um punhado de areia (silício) e um pouco de fio de cobre, arranjados segundo esses princípios, realizam uma operação lógica. Não há nada além de física de materiais e geometria de circuito por trás da porta lógica que se manipula em alto nível de abstração.

Figura pendente

três circuitos lado a lado — porta AND (transistores em série), porta OR (transistores em paralelo) e porta NOT (transistor inversor), com

tabela-verdade de cada uma

1.10 Circuitos aritméticos: o meio-somador

Se é possível construir portas lógicas com transistores, também é possível construir um circuito capaz de somar dois números binários.

A adição em binário segue a mesma lógica da adição em decimal, mas com apenas dois símbolos disponíveis:

A	B	Soma	Vai-um (*carry*)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Observando essa tabela, nota-se que a coluna “Soma” corresponde exatamente à tabela-verdade da porta **OU exclusivo** (XOR), e a coluna “Vai-um” corresponde exatamente à tabela-verdade da porta **AND**.

Analogia. A porta XOR é ilustrada pela missão “vá à padaria e traga coca-cola ou guaraná”: trazer apenas um dos dois cumpre a missão; não trazer nenhum, ou trazer os dois, não cumpre. É a diferença entre “ou” inclusivo (porta OR comum) e “ou” exclusivo.

O circuito que combina uma porta XOR (para o bit de soma) e uma porta AND (para o bit de vai-um), ambas recebendo as mesmas duas entradas A e B, é chamado **meio-somador** (*half adder*). Ele soma dois bits e produz dois resultados: o bit de soma e o bit de transporte para a próxima posição. Encadeando vários meios-somadores (com o acréscimo da entrada de vai-um recebido da posição anterior, o que dá origem ao **somador completo**, ou *full adder*), constrói-se um circuito capaz de somar números de qualquer quantidade de bits — o tamanho da palavra binária que um processador consegue somar de uma vez (32 bits, 64 bits) é justamente definido por quantos desses circuitos estão encadeados no hardware.

Figura pendente

circuito do meio-somador — porta XOR e porta AND recebendo as entradas A e B, produzindo Soma e Carry

1.11 Memória: o flip-flop e os sinais de controle

Se as portas lógicas resolvem o problema das operações lógicas e aritméticas, resta um terceiro problema: como armazenar um valor de forma per-

sistente, para além do instante em que ele é calculado.

A célula de memória mais básica é o **flip-flop**: um circuito construído a partir da combinação de portas lógicas (e, portanto, em última instância, de transistores) capaz de armazenar um bit e responder a duas operações — leitura ("qual valor você tem?") e escrita ("guarde este valor").

A tabela-verdade do flip-flop tipo JK, um dos modelos mais estudados, resume seu comportamento:

J	K	Comportamento
0	0	Mantém o valor atual (memória não é alterada)
0	1	Escreve 0
1	0	Escreve 1
1	1	Inverte o valor atual

O flip-flop só responde a essas entradas no instante em que recebe um sinal de **clock** — um pulso periódico que sincroniza a operação. É esse sinal de clock, gerado em alta frequência, que corresponde à frequência de operação anunciada para um processador (em megahertz ou gigahertz).

1.11.1 Sinais de controle

A operação de leitura e escrita não esgota o funcionamento de um chip de memória real. Um registrador construído com flip-flops — como o circuito integrado 74HC173, um registrador de 4 bits disponível comercialmente [7] — expõe também:

- **Reset**: zera o valor armazenado, independentemente do estado do clock.
- **Habilitação de saída** (*output enable*): liga ou desliga a conexão entre o valor armazenado e a saída do chip — relevante, por exemplo, quando a saída está ligada a um motor que precisa estar energizado antes de receber o sinal.
- **Habilitação de escrita** (*write enable*): determina se o chip está em modo de leitura ou de escrita naquele instante, mesmo que o sinal de clock continue chegando.

Exemplo. Um simulador de eletrônica digital utilizado em aula demonstra por que os sinais de controle são indispensáveis: ao conectar um circuito somador simples diretamente a um decodificador binário e um display de sete segmentos, o resultado exibido fica incorreto se o segundo operando for alterado antes de o primeiro ter sido efetivamente processado. Os sinais de controle são o que permite dizer ao circuito "carregue este valor agora" e "leia o resultado agora" em momentos distintos e bem definidos — sem eles, não há garantia de que a operação foi concluída antes da leitura.

Esses sinais — de ligar/desligar, de habilitar saída, de selecionar entre operações possíveis de um chip — reaparecem em praticamente todo circuito digital: um somador, um circuito de memória, um controlador de

barramento. Entender que eles existem é o que permite compreender a camada seguinte da abstração, tratada na próxima seção.

Figura pendente

registrador de 4 bits com entradas de dados, clock, reset e habilitação de saída identificadas

1.12 A hierarquia de linguagens: de L0 a L5

A hierarquia de níveis apresentada nesta seção segue o modelo de máquinas multiníveis proposto por Tanenbaum [8]. A lógica digital descrita nas seções anteriores é poderosa, mas impraticável de programar diretamente: escrever um programa inteiro em sequências de 0 e 1 é uma tarefa inviável para qualquer problema não trivial. A solução histórica para esse problema foi a criação de sucessivas camadas de linguagem, cada uma mais próxima do raciocínio humano do que a anterior, e cada uma traduzida (por um interpretador, compilador ou circuito de controle) para a camada imediatamente abaixo.

Nível	Nome	Função
L0	Lógica digital	Portas lógicas, somadores, memórias (Seções 1.9–1.11)
L1	Microarquitetura (microprograma)	Controla o fluxo de dados: o que é carregado, de onde, para onde, acionando a camada L0
L2	Arquitetura do conjunto de instruções (ISA)	Define se o processador é RISC ou CISC, e qual conjunto de instruções ele reconhece (x86, ARM, etc.)
L3	Sistema operacional	Permite a execução de múltiplos programas e o gerenciamento do hardware por eles
L4	Tradução (compiladores/montadores)	Converte código de alto nível em código de máquina executável pelo sistema operacional

Nível	Nome	Função
L5	Linguagem orientada a problema	Python, C++, JavaScript — nível em que a maioria dos desenvolvedores trabalha

Nota terminológica. É comum que a imprensa especializada em hardware anuncie uma "nova arquitetura" ao descrever um novo processador quando, tecnicamente, o que mudou foi a **microarquitetura** (nível L1) — a forma como aquele conjunto de instruções é implementado internamente —, enquanto a arquitetura do conjunto de instruções (nível L2, por exemplo x86-64) permanece a mesma.

1.12.1 Exemplo: de Python ao circuito

Retomando o programa `x = 2; y = x + 1`: em L5 (Python), a atribuição e a soma são expressas em sintaxe legível por humanos. O interpretador ou compilador (L4) traduz essas instruções para código de máquina. O sistema operacional (L3) aloca memória e agenda a execução do processo. O processador consulta seu conjunto de instruções (L2) para decodificar cada instrução recebida. O microprograma (L1) aciona os circuitos específicos — leitura de memória, soma, escrita de memória — que fisicamente existem no nível de lógica digital (L0), executando a operação por meio de tensões elétricas.

Antes da existência de linguagens de alto nível, a única alternativa a programar diretamente em código de máquina (sequências binárias) era a **linguagem Assembly**: um conjunto de mnemônicos (como `ADD`, `MOV`) que correspondem quase diretamente às instruções do processador, mas ainda exigem gerenciamento manual explícito de registradores e endereços de memória — por isso, extremamente trabalhosa para programas complexos.

1.12.2 O nível além de L5

Um nível ainda não formalizado na literatura clássica de arquitetura de computadores, mas cada vez mais concreto na prática, é o de **linguagem natural**: comandos dados a um modelo de linguagem (LLM) que gera código executável em Python, C++ ou outra linguagem de nível L5 a partir de um pedido descrito em português ou inglês comum. Ferramentas como assistentes de voz (Alexa, Google Assistant) representaram uma primeira aproximação limitada desse nível; modelos de linguagem contemporâneos ampliam significativamente essa capacidade, embora ainda não exista uma camada intermediária formal, treinada especificamente para maximizar a taxa de acerto entre o pedido em linguagem natural e o código gerado.

Nota sobre o uso de ferramentas de IA. Um modelo de linguagem gera texto com alta probabilidade estatística de ser relevante — não é um inter-

locutor consciente, ainda que a fluência de suas respostas possa sugerir o contrário. Isso reforça um ponto prático para o técnico de informática: modelos de linguagem e agentes de IA são ferramentas, e a responsabilidade por uma tarefa delegada a eles continua sendo de quem concedeu essa liberdade — da mesma forma que dizer "o macaco trocou o pneu do carro" ignora que foi a pessoa quem usou o macaco.

1.13 A arquitetura de von Neumann

O conceito de **programa armazenado** — a ideia de que as instruções de um programa, e não apenas os dados que ele manipula, residem na mesma memória do computador — já apresentado como fundamento histórico da computação de uso geral, foi formalizado por escrito no relatório "First Draft of a Report on the EDVAC" (jun. 1945), de John von Neumann, e detalhado em 1946 em coautoria com Arthur Burks e Herman Goldstine [9]. A disputa de crédito mais discutida na literatura histórica não é com Alan Turing, mas com **Eckert e Mauchly**, engenheiros do grupo ENIAC/EDVAC que consideravam a ideia um resultado coletivo e ficaram descontentes por o relatório circular só com o nome de von Neumann. Turing, por sua vez, produziu um relatório equivalente sobre programa armazenado (o projeto ACE, apresentado ao National Physical Laboratory em fevereiro de 1946, já em tempos de paz) — o sigilo que de fato cercou o trabalho de Turing durante a guerra foi o da criptoanálise em Bletchley Park, não o do próprio projeto ACE [10]. Ainda assim, "arquitetura de von Neumann" é o nome consagrado que este livro adota para a formalização.

Essa arquitetura organiza o computador em quatro unidades funcionais:

- **Unidade de entrada** — capta dados do mundo externo (teclado, mouse, câmera).
- **Unidade de saída** — expõe dados processados ao mundo externo (monitor, alto-falante).
- **Unidade de processamento (CPU)** — subdividida em **unidade de controle** (coordena a sequência de operações) e **unidade lógica e aritmética — ULA** (executa as operações lógicas e aritméticas propriamente ditas).
- **Unidade de memória** — armazena tanto os dados quanto o próprio programa em execução.

Analogia. O processador ocupa, nessa arquitetura, o papel do cozinheiro que segue uma receita: recebe as entradas (leite condensado, ovos, leite), utiliza a memória como bancada de trabalho, processa segundo uma sequência de passos e produz a saída (o pudim pronto). A unidade de controle é o que garante que os passos da receita sejam seguidos na ordem correta.

No computador desktop moderno, a via de dados que interliga processador, memória e dispositivos de entrada e saída — o **barramento** — é fisicamente provida pela placa-mãe.

Figura pendente

diagrama da arquitetura de von Neumann — CPU (unidade de controle + ULA), memória, entrada e saída interligados por um barramento central

1.14 O gargalo de von Neumann

A separação física entre processador e memória — necessária, já que nenhum dispositivo conhecido realiza simultaneamente as funções de processamento e de armazenamento — traz uma limitação estrutural conhecida como **gargalo de von Neumann**.

Analogia. Um engarrafamento rodoviário ocorre quando várias faixas de tráfego convergem para uma via mais estreita — todo o fluxo de veículos, independentemente de sua origem, precisa passar pelo mesmo ponto de estrangulamento. De forma equivalente, toda instrução e todo dado que o processador precisa manipular precisam trafegar pelo barramento que o conecta à memória. Por mais rápido que o processador seja, e por mais rápida que a memória seja, a taxa de transferência entre os dois chips é limitada pela largura e pela velocidade dessa via de comunicação — e essa via é, estruturalmente, o ponto mais lento do sistema.

Abordagens modernas de arquitetura, como o **System-on-Chip (SoC)**, aproximam fisicamente processador e memória dentro de um único encapsulamento, reduzindo a latência de comunicação entre eles. Essa aproximação mitiga o problema, mas não o elimina: processamento e memória continuam sendo entidades fisicamente distintas, e o gargalo de von Neumann permanece um problema em aberto na arquitetura de computadores.

1.15 Computação além do silício

A abstração em camadas construída ao longo deste capítulo — da tensão elétrica à porta lógica, da porta lógica ao circuito aritmético, do circuito à linguagem de programação — revela algo importante sobre a natureza da computação: o que importa é a estrutura lógica das operações, não o material físico que as implementa.

Exemplo. Utilizando o sistema de circuitos do jogo *Minecraft* (o mecanismo de *redstone*), é possível reproduzir portas lógicas, meios-somadores e células de memória, e a partir deles construir um computador funcional dentro do próprio jogo — capaz, por exemplo, de executar um programa que calcula a sequência de Fibonacci. Um projeto ainda mais extremo levou essa ideia ao limite: como o computador construído em *redstone* é, em si, capaz de executar qualquer programa, um desenvolvedor conseguiu rodar uma cópia do próprio *Minecraft* dentro do computador que havia construído dentro do *Minecraft* — a um custo de desempenho da ordem de milhões de vezes mais lento, mas funcionalmente completo [11].

Outros projetos substituem o transistor por outros meios físicos para implementar as mesmas portas lógicas — por exemplo, circuitos hidráulicos que implementam portas AND, OR e NOT utilizando água e tubulações [12]. O romance de ficção científica *O Problema dos Três Corpos*, do autor chinês Liu Cixin, descreve um exército de seres humanos treinados para atuar, cada um, como uma porta lógica — formando coletivamente um computador funcional operado inteiramente por pessoas [13].

Esses exemplos, por mais lúdicos que pareçam, sustentam um ponto central deste capítulo: a computação é uma estrutura lógica de camadas de abstração, e o silício é apenas o meio físico mais eficiente conhecido atualmente para implementá-la — não o único possível.

1.16 Síntese do capítulo

Este capítulo apresentou a manutenção de computadores como disciplina fundamentada na distinção entre manutenção corretiva e preventiva, no raciocínio de diagnóstico por hipóteses e no reconhecimento do sistema computacional como uma tríade de hardware, software e pessoas — incluindo casos concretos de manutenção corretiva de software, como a identificação de malware e a reinstalação de sistemas operacionais. Em seguida, reconstruiu a ponte entre software e hardware por meio da abstração em camadas: da plataforma de execução ao transistor, do transistor às portas lógicas, das portas lógicas aos circuitos aritméticos e de memória, e destes à hierarquia de linguagens que culmina na arquitetura de von Neumann e em sua limitação estrutural, o gargalo de von Neumann. Esses fundamentos — sobretudo a noção de que todo problema pode ser localizado numa camada específica dessa pilha — sustentam o capítulo seguinte, que trata da camada mais concreta de todas: a eletricidade e a fonte de alimentação que energizam fisicamente cada uma dessas camadas.

1.17 Referências

1. ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. *NBR 5462: Confiabilidade e mantenabilidade — Terminologia*. Rio de Janeiro: ABNT, 1994.
2. ESD ASSOCIATION. *Fundamentals of Electrostatic Discharge*. Disponível em: `https://www.esda.org`; ou ABNT NBR IEC 61340-5-1 (controle eletrostático em ambientes que manuseiam dispositivos sensíveis).
3. TECHSPOT. “YouTube channel Linus Tech Tips terminated after it was hacked to show crypto-scam videos.” 2023. Disponível em: `https://www.techspot.com/news/98047-`; DIGITAL TRENDS. “Linus Tech

Tips restored after crypto scam hack." Disponível em: https://www.digitaltrends.com/computing/linus-tech-tips-offline-after-cryptosca
4. Páginas oficiais: RUFUS, https://rufus.ie; VENTOY, https://www.ventoy.net; YUMI, https://www.pendrivelinux.com.
5. MALVINO, Albert Paul; BROWN, Jerald A. *Digital Computer Electronics*. 3. ed. Nova York: Glencoe/McGraw-Hill, 1993, Cap. 4 "TTL Circuits".
6. MALVINO, Albert Paul; BROWN, Jerald A. *Digital Computer Electronics*. 3. ed. Nova York: Glencoe/McGraw-Hill, 1993, Cap. 2 ("2-1 Inverters", "Diode OR Gate", "2-3 AND Gates", "Diode AND Gate").
7. NEXPERIA. "74HC173; 74HCT173 — Quad D-type flip-flop; positive-edge trigger; 3-state." Datasheet. Disponível em: https://assets.nexperia.com/documents/data-sheet/74HC_HCT173.pdf.
8. TANENBAUM, Andrew S.; AUSTIN, Todd. *Organização Estruturada de Computadores*. 6. ed. São Paulo: Pearson Education do Brasil, 2013, Seção 1.1.
9. BURKS, A. W.; GOLDSTINE, H. H.; VON NEUMANN, J. *Preliminary Discussion of the Logical Design of an Electronic Computing Instrument*. Princeton: Institute for Advanced Study, 1946.
10. COPELAND, B. Jack (org.). *Alan Turing's Automatic Computing Engine*. Oxford: Oxford University Press, 2005.
11. MOJANG. "Minecraftception." Minecraft.net. Disponível em: https://www.minecraft.net/en-us/article/-minecraftception.
12. "Hydraulic logic gates: building a digital water computer." *European Journal of Physics*, v. 39, 2018. IOP Publishing. Disponível em: https://iopscience.iop.org/article/10.1088/1361-6404/aa97fc.
13. LIU, Cixin. *O Problema dos Três Corpos*. São Paulo: Aleph. (Tradutor e ano da edição a confirmar pelo autor.)

2 Eletricidade e Fonte de Alimentação

Neste capítulo você vai estudar as grandezas elétricas fundamentais (tensão, corrente, resistência e potência), os princípios de proteção e aterramento de uma instalação elétrica residencial, o funcionamento de transformadores e circuitos retificadores, a diferença entre fontes lineares e fontes chaveadas, a arquitetura da fonte ATX e seus sinais de controle, os critérios de eficiência energética e fator de potência usados para classificar fontes comercialmente, e a metodologia de dimensionamento e diagnóstico de uma fonte de computador.

2.1 Grandezas elétricas fundamentais

2.1.1 Diferença de potencial (tensão)

Toda análise elétrica de um computador começa na tomada, e toda tomada é, fisicamente, uma **diferença de potencial elétrico** (DDP). O conceito de potencial elétrico pode ser compreendido por analogia com o potencial gravitacional.

Analogia. Considere um lápis apoiado sobre uma mesa (ponto A) e outro lápis apoiado no chão (ponto B). O lápis em A possui maior energia potencial gravitacional do que o lápis em B, porque existe uma diferença de altura entre os dois pontos. Se uma segunda mesa for empilhada sobre a primeira e um terceiro lápis for apoiado sobre ela (ponto C), esse lápis possui energia potencial gravitacional ainda maior do que os pontos A e B. Existe, portanto, uma diferença de potencial gravitacional entre cada par de pontos — sendo a maior delas entre C e B. Um ponto não possui diferença de potencial em relação a si mesmo: a diferença de potencial só existe *entre* dois pontos de observação.

O mesmo raciocínio se aplica à eletricidade: a diferença de potencial elétrico só existe entre dois pontos. É por essa razão que toda tomada elétrica possui, no mínimo, dois furos — um representando cada um dos dois pontos entre os quais existe a diferença de potencial. A grandeza que quantifica essa diferença de potencial é chamada de **tensão** ou **voltagem**, medida em **volts** (V).

Figura pendente

diagrama dos pontos A, B e C em mesas empilhadas, com setas indicando as diferenças de potencial gravitacional entre cada par

Quando um caminho condutor é fornecido entre dois pontos de potenciais elétricos diferentes, os elétrons se deslocam da região de maior potencial para a de menor potencial — exatamente como o lápis cai da mesa para o chão, transformando energia potencial gravitacional em energia cinética. Esse movimento de elétrons pode ser aproveitado para realizar trabalho: aquecer uma resistência, acender um LED, ou acionar os transistores que formam os circuitos lógicos de um computador.

Exemplo. Um dispositivo eletrônico é projetado para operar dentro de uma faixa de tensão específica — assim como uma roda-d'água é projetada para funcionar sob uma queda d'água de determinada altura. Uma roda-d'água dimensionada para uma queda de 2 metros não resiste a uma queda de 20 metros: a estrutura se rompe porque não foi projetada para aquela energia. Da mesma forma, um equipamento projetado para operar em 110 V, ao ser ligado em 220 V, recebe o dobro da tensão para a qual foi dimensionado — o que, pela Lei de Ohm (Seção 2.1.3), provoca uma corrente elétrica também maior do que a suportada pelos seus componentes, danificando-os.

2.1.2 Corrente elétrica

Corrente elétrica é o fluxo ordenado de elétrons através de um condutor, medido em **ampere** (A). Historicamente, antes de se compreender que a carga móvel era o elétron (de carga negativa), adotou-se a convenção de que a corrente flui do polo positivo para o polo negativo — o chamado **sentido convencional** da corrente. O **sentido real** do movimento dos elétrons é o oposto: do polo negativo (região com excesso de elétrons) para o polo positivo (região com falta de elétrons). Essa convenção histórica permanece em uso na análise de circuitos até hoje.

2.1.3 Lei de Ohm e resistência elétrica

A relação entre tensão, corrente e resistência é dada pela **primeira Lei de Ohm**:

$$V = R \times I$$

onde V é a tensão (volts), R é a **resistência elétrica** (ohms, Ω) e I é a corrente (amperes). A resistência mede a oposição de um material à passagem de corrente: materiais condutores (como cobre) têm resistência muito baixa; materiais isolantes (como borracha ou ar) têm resistência muito alta.

Exemplo. Em um circuito simples formado por uma fonte de tensão, um resistor e um LED em série, aumentar a tensão da fonte, mantendo a mesma resistência, provoca um aumento proporcional na corrente — e um LED

submetido a maior corrente brilha com maior intensidade. Tensão e corrente são, nesse circuito, grandezas diretamente proporcionais: quando uma sobe, a outra sobe na mesma proporção.

Essa relação também explica por que um **curto-circuito** é perigoso. Um curto-circuito ocorre quando um caminho de resistência próxima de zero é criado entre dois pontos de potenciais diferentes — por exemplo, um pequeno objeto condutor (um clipe de papel, um grampo metálico) tocando acidentalmente os dois pinos de uma tomada. Como a corrente é inversamente proporcional à resistência, uma resistência próxima de zero produz uma corrente extremamente alta — teoricamente, o máximo que a fonte for capaz de fornecer. Esse pico de corrente é o que provoca aquecimento excessivo dos condutores, podendo causar incêndio se não houver um dispositivo de proteção interrompendo o circuito (ver Seção 2.3.1).

Figura pendente

circuito simples fonte–resistor–LED com seta indicando corrente e legenda “V = R × I”

2.1.4 Corrente contínua e corrente alternada

A corrente elétrica pode ser **contínua** (CC ou, do inglês, DC) ou **alternada** (CA ou AC).

Na **corrente contínua**, os elétrons se deslocam sempre no mesmo sentido, produzida por uma fonte de tensão fixa ao longo do tempo — como uma pilha, cuja diferença de potencial entre os polos permanece constante em, por exemplo, 1,5 V. Representando essa tensão em um gráfico ao longo do tempo, obtém-se uma linha reta e constante.

Na **corrente alternada**, a tensão da fonte varia periodicamente entre valores positivos e negativos, seguindo um comportamento **senoidal**: a tensão sobe até um valor máximo, desce até um valor mínimo (negativo) e repete esse ciclo indefinidamente. Como consequência, os elétrons não se deslocam sempre no mesmo sentido — eles oscilam, ora em uma direção, ora na direção oposta, tantas vezes por segundo quanto for a **frequência** da onda. A tomada residencial no Rio Grande do Norte fornece uma tensão alternada de 220 V (valor eficaz, ou RMS) a 60 Hz [1] — ou seja, o ciclo de oscilação se repete 60 vezes por segundo.

Exemplo. O funcionamento de um chuveiro elétrico não depende do sentido em que os elétrons se movem: o simples ir e vir do elétron, induzido pela tensão alternada, já é suficiente para aquecer a resistência do chuveiro e, consequentemente, a água. O trabalho realizado (aquecimento) não exige que a corrente seja contínua.

A razão pela qual a energia elétrica é gerada, transportada e distribuída na forma alternada — e não contínua — está diretamente ligada à minimização de perdas no transporte, tratada em detalhe na Seção 2.2.

2.2 Potência elétrica e efeito Joule

Potência elétrica é a capacidade de um dispositivo realizar trabalho por unidade de tempo, medida em **watts** (W). É calculada como o produto entre tensão e corrente:

$$P = V \times I$$

Combinando essa equação com a Lei de Ohm ($V = R \times I$), obtém-se uma segunda forma equivalente:

$$P = R \times I^2$$

Essa segunda forma é especialmente importante porque descreve a **potência dissipada por efeito Joule** — o calor gerado pela passagem de corrente através de um condutor com resistência. Quanto maior a corrente que passa por um condutor, maior é a potência perdida em forma de calor, e essa perda cresce com o **quadrado** da corrente.

Analogia. A diferença entre um dispositivo pouco potente e um dispositivo muito potente pode ser ilustrada pela comparação entre um Fusca antigo e uma Ferrari: ambos podem estar andando à mesma velocidade em um dado instante, mas a Ferrari possui um motor capaz de atingir velocidades muito maiores. Potência não é velocidade — é a *capacidade* de realizar mais trabalho, ainda que essa capacidade não esteja sendo totalmente utilizada no momento.

Exemplo. Um chuveiro elétrico ligado em 220 V, puxando uma corrente da ordem de 5 A, tem potência de $220 \times 5 = 1100$ W. Um carregador de celular, ligado na mesma tomada de 220 V mas puxando apenas cerca de 0,01 A, tem potência de $220 \times 0{,}01 = 2{,}2$ W. Ambos os dispositivos estão sujeitos à mesma tensão da tomada; o que os diferencia é a corrente que cada um demanda — e essa diferença de corrente é o que torna o chuveiro elétrico centenas de vezes mais potente que o carregador.

2.2.1 Por que a energia é transportada em alta tensão

A rede elétrica gera energia em uma determinada tensão e, antes de transportá-la por longas distâncias até os centros consumidores, eleva essa tensão para valores muito mais altos (por meio de transformadores, tratados na Seção 2.5), reduzindo-a de volta a 220 V apenas próximo ao ponto de consumo. Essa escolha de projeto decorre diretamente da equação $P = R \times I^2$: para transportar uma mesma potência, é possível usar uma tensão alta com corrente baixa, ou uma tensão baixa com corrente alta. Como a perda por efeito Joule cresce com o quadrado da corrente, transportar a energia com uma corrente menor (elevando a tensão) reduz drasticamente as perdas ao longo dos cabos de transmissão — o que explica também o zumbido característico dos transformadores de poste, resultado da conversão eletromagnética de tensão em andamento.

Figura pendente

diagrama simplificado da rede elétrica — geração, elevação de tensão para transporte, redução de tensão em subestações e no poste, chegada em 220 V à residência

2.3 Proteção da instalação elétrica

Normas aplicáveis. O conteúdo desta seção e da Seção 2.4 (disjuntores, dimensionamento de condutores, aterramento) segue as prescrições das normas técnicas brasileiras de instalações elétricas de baixa tensão e de segurança em instalações elétricas [2].

2.3.1 Disjuntores

Um **disjuntor** é um dispositivo eletromecânico que interrompe automaticamente a passagem de corrente elétrica quando ela ultrapassa um determinado limite. Seu funcionamento se baseia no fato de que toda corrente elétrica alternada produz um campo magnético proporcional à sua intensidade: dentro do disjuntor existe um sensor sensível a esse campo magnético; quando a corrente excede o limite para o qual o disjuntor foi dimensionado (por exemplo, 10 A), o campo magnético resultante aciona um mecanismo de mola que desarma o disjuntor, desconectando mecanicamente o circuito — produzindo o característico estalo audível.

Analogia. O disjuntor funciona como uma cancela de pedágio: os carros (a corrente) passam normalmente até que um volume de tráfego anormal é detectado, e a cancela se fecha, impedindo a passagem de mais carros.

Uma instalação elétrica é organizada hierarquicamente: da rede pública, a energia chega a um **quadro geral**, de onde é distribuída para um ou mais **quadros de distribuição**, cada um atendendo a um setor específico da edificação (um andar, uma sala). Cada circuito de um quadro de distribuição — iluminação, tomadas, ar-condicionado — é protegido por seu próprio disjuntor, de forma que uma falha em um circuito não derrube os demais. Esse isolamento por circuito é análogo ao encapsulamento em programação orientada a objetos: cada objeto (aqui, cada sala ou circuito) mantém seus próprios atributos e falhas isoladas do restante do sistema.

A instalação pode ser **monofásica** (uma única fase de 220 V mais um neutro de referência) ou **trifásica** (três fases de 220 V, cada uma referenciada ao mesmo neutro, usadas para distribuir cargas maiores entre três circuitos independentes).

Figura pendente

hierarquia poste → quadro geral → quadros de distribuição → disjuntores individuais por circuito

2.3.2 Dimensionamento de condutores

Cada condutor (fio) elétrico suporta uma corrente máxima, determinada pela sua **bitola** (área de seção transversal). Quanto maior a bitola do fio, maior a corrente que ele pode conduzir sem aquecer excessivamente.

Analogia. Um fio é como uma pista de rodovia: uma pista larga comporta mais tráfego (corrente) simultâneo do que uma pista estreita.

O disjuntor de um circuito deve ser dimensionado de acordo com a capacidade do fio que ele protege: se um fio suporta no máximo 10 A, o disjuntor daquele circuito deve desarmar antes que essa corrente seja ultrapassada. Uma causa comum de disjuntores desarmando repetidamente é a **sobrecarga**: ligar, em um mesmo circuito, equipamentos cuja soma de potências excede a capacidade projetada para aquele fio e aquele disjuntor — por exemplo, uma impressora de alto consumo e uma pistola de cola quente compartilhando a mesma tomada. Quando essa sobrecarga não é interrompida por um disjuntor subdimensionado (ou por um disjuntor trocado por um de corrente nominal mais alta sem trocar o fio correspondente), o próprio condutor pode superaquecer e provocar um curto-circuito ou incêndio.

2.4 Aterramento e segurança elétrica

2.4.1 O planeta Terra como referência de potencial

O terceiro pino presente na maioria das tomadas modernas corresponde ao **aterramento** (terra). O planeta Terra, por sua imensa massa e extensão, funciona como uma fonte praticamente infinita de cargas elétricas: ele pode absorver um excesso de cargas de qualquer ponto conectado a ele, ou fornecer cargas a um ponto que esteja com déficit, sempre tendendo ao equilíbrio elétrico.

Analogia. O comportamento do aterramento é semelhante ao equilíbrio térmico entre um sorvete retirado do freezer e uma panela quente retirada do fogo, colocados lado a lado: o sorvete tende a esquentar e a panela tende a esfriar, ambos convergindo para a temperatura ambiente. Da mesma forma, um excesso (ou déficit) de cargas elétricas tende a se equilibrar através de um caminho condutor até a Terra.

O **ar** é naturalmente isolante — seus átomos mantêm os elétrons presos com energia suficiente para impedir a condução em condições normais — o que explica por que é seguro aproximar a mão de um fio energizado sem tocá-lo diretamente. Quando, porém, a diferença de potencial entre dois

pontos se torna extrema (como entre uma nuvem carregada e o solo), a rigidez elétrica do ar é rompida, e ele passa a conduzir: esse fenômeno é o **raio**. O trajeto irregular de um raio decorre do fato de que a descarga segue o caminho de menor resistência entre as moléculas de ar naquele instante. Estruturas pontiagudas, como para-raios, favorecem a descarga por um efeito conhecido como **poder das pontas**: a geometria fina concentra o campo elétrico, oferecendo um caminho de menor resistência para a corrente até o fio terra.

Figura pendente

ilustração do trajeto irregular de um raio e de um para-raios conduzindo a descarga até o fio terra

Exemplo. Um carregador ou notebook com gabinete metálico, ligado a um cabo de alimentação de apenas dois pinos (sem aterramento), pode transmitir ao usuário uma leve sensação de formigamento ao ser tocado enquanto carrega. Essa sensação ocorre porque um pequeno vazamento de corrente se acumula no chassi metálico e, na ausência de um caminho de aterramento de baixa resistência, o próprio corpo do usuário se torna o caminho disponível para o escoamento dessas cargas até a terra.

2.4.2 Instalação de aterramento e o papel da concessionária

Em uma instalação elétrica residencial, os condutores de **fase** e **neutro** são fornecidos pela concessionária de energia (no Rio Grande do Norte, a Cosern): problemas nesses dois condutores são de responsabilidade dela. O condutor de **terra**, por outro lado, é de responsabilidade da instalação elétrica local — tipicamente executado com hastes metálicas cravadas no solo, muitas vezes aproveitando o próprio ferro da fundação da edificação, conectadas por fio a uma resistência de aterramento baixa o suficiente para escoar cargas com eficiência.

Como a tomada é uma diferença de potencial alternada, não existe, tecnicamente, um lado “certo” e um lado “errado” entre fase e neutro do ponto de vista da física básica — mas, na prática, muitos equipamentos e normas de instalação definem um lado correto para cada função, e essa correspondência deve ser verificada com instrumentos apropriados, nunca presumida.

Atenção. Um erro de instalação — por exemplo, o condutor de fase sendo conectado, por engano, ao ponto que deveria ser o terra — pode fazer com que toda a fiação de aterramento de uma edificação fique energizada em 220 V. Nessa condição, qualquer equipamento aterrado (como uma geladeira) torna-se, ele próprio, uma fonte de choque ao ser tocado. Por essa razão, um técnico nunca deve presumir que uma instalação elétrica está correta: a tomada deve ser testada com um multímetro antes de qualquer intervenção, identificando corretamente fase, neutro e terra.

2.4.3 Choque elétrico e o chuveiro elétrico

O choque elétrico é perigoso porque a corrente elétrica que atravessa o corpo humano pode interferir nos impulsos nervosos que controlam a musculatura — inclusive o músculo cardíaco. Uma corrente elétrica no momento errado pode desorganizar o ritmo cardíaco, causando uma disfunção potencialmente fatal.

Atenção. Antes de qualquer manuseio de fiação elétrica, energia deve ser desligada na fonte (disjuntor) e, sempre que possível, confirmada como desligada com um instrumento de medição. Nunca se deve presumir que um circuito está desenergizado apenas porque um interruptor foi acionado.

O **chuveiro elétrico** — uma invenção brasileira [3] — costuma causar estranhamento em visitantes de países onde o aquecimento de água é feito por gás ou por reservatórios térmicos (*boilers*), a ponto de gerar receio em relação ao seu uso. Esse receio é, tecnicamente, infundado: o chuveiro elétrico não estabelece contato elétrico com a água. A corrente elétrica atravessa apenas a resistência interna do aparelho, que se aquece por efeito Joule; a água, ao passar por essa resistência já aquecida, absorve o calor por condução térmica, sem que corrente elétrica normalmente flua através dela. Um choque em um chuveiro elétrico só ocorre diante de uma falha grave de isolamento ou de instalação — não é uma característica intrínseca do funcionamento do equipamento.

2.5 Transformadores e transporte de energia

Um **transformador** é o dispositivo responsável por elevar ou reduzir uma tensão alternada, sendo o elemento central tanto no transporte de energia em alta tensão (Seção 2.2) quanto no primeiro estágio de uma fonte de alimentação linear (Seção 2.6).

Seu princípio de funcionamento se baseia na **Lei de Faraday**: uma corrente elétrica variável produz um campo magnético; um campo magnético variável, por sua vez, induz uma corrente elétrica em um condutor próximo. Um transformador é fisicamente constituído por dois enrolamentos de fio (bobinas) — o primário e o secundário — em torno de um núcleo ferromagnético comum, sem contato elétrico direto entre eles. A corrente alternada que circula pela bobina primária gera um campo magnético variável no núcleo, que induz uma corrente na bobina secundária.

A relação entre a tensão de entrada e a tensão de saída é determinada pela razão entre o **número de espiras** (voltas do fio) em cada bobina: se o secundário possui mais espiras do que o primário, a tensão é elevada e a corrente correspondentemente reduzida (para conservar a potência); se possui menos espiras, a tensão é reduzida e a corrente aumentada.

Analogia. O funcionamento por número de espiras é o mesmo princípio empregado no captador de uma guitarra elétrica: a bobina do captador,

enrolada um determinado número de vezes, capta a vibração das cordas metálicas e converte essa vibração em um sinal elétrico cujo timbre depende justamente da quantidade de voltas do enrolamento.

Como a energia não se perde nessa conversão (idealmente), o produto tensão × corrente se mantém aproximadamente constante entre primário e secundário: elevar a tensão implica reduzir a corrente na mesma proporção, e vice-versa. É exatamente essa propriedade que permite elevar a tensão da rede elétrica para transporte de longa distância (reduzindo a corrente e, com ela, as perdas por efeito Joule) e depois reduzi-la de volta a 220 V para uso residencial — passando antes por transformadores de subestação e, por fim, pelos transformadores de poste que abaixam a tensão para o nível final entregue às residências.

Como o funcionamento do transformador depende de variação de campo magnético, ele **exige corrente alternada** para operar: não é possível usar um transformador diretamente sobre uma tensão contínua constante, pois esta não gera a variação de campo magnético necessária à indução.

Figura pendente

diagrama de um transformador com bobina primária e secundária em torno de um núcleo ferromagnético, indicando número de espiras e tensões de entrada/saída

---

2.6 Da corrente alternada à contínua: fontes lineares

Todos os componentes internos de um computador — processador, memória, circuitos lógicos — operam com **corrente contínua**. A função central de qualquer fonte de alimentação é, portanto, converter a corrente alternada da tomada em corrente contínua nos níveis de tensão exigidos por cada componente.

O modelo mais simples de conversão é a **fonte linear**, construída por meio de quatro estágios em sequência:

1. **Transformador** — reduz a tensão alternada de entrada (por exemplo, 220 V) para uma tensão alternada de menor amplitude (por exemplo, 10 V), conforme descrito na Seção 2.5.
2. **Retificador** (ponte de diodos) — inverte a parcela negativa da onda alternada para o lado positivo, já que um diodo permite a passagem de corrente em apenas um sentido. Um único diodo eliminaria completamente a metade negativa da onda; uma ponte de diodos “rebate” essa metade negativa para cima, produzindo uma onda inteiramente positiva, ainda que pulsante.

3. **Filtro capacitivo** — um capacitor de grande capacitância, ligado em paralelo com a carga, suaviza as oscilações (o chamado *ripple*) da onda retificada, aproximando-a de uma tensão contínua estável.
4. **Regulador de tensão** — ajusta e estabiliza a tensão final na saída.

Analogia. O papel do capacitor de filtro é análogo ao de uma caixa d'água doméstica: enquanto a torneira permanece fechada, a caixa acumula água vinda da rede de distribuição; quando a torneira é aberta, a água armazenada é consumida antes de nova reposição. Da mesma forma, o capacitor acumula carga elétrica durante os picos da onda retificada e a libera durante os vales, suavizando a tensão entregue à carga.

Fontes lineares são robustas e relativamente simples de projetar, mas apresentam uma desvantagem física significativa: para uma mesma potência, seus componentes (especialmente o transformador) são consideravelmente maiores e mais pesados do que os de uma fonte chaveada equivalente. Por essa razão, fontes lineares praticamente não são mais utilizadas em computadores modernos, restringindo-se a aplicações específicas, como equipamentos de áudio de alta fidelidade, em que a característica do circuito linear é tecnicamente desejável.

Figura pendente

diagrama de blocos de uma fonte linear — transformador → ponte de diodos → filtro capacitivo → regulador — com a forma de onda em cada estágio

---

2.7 Fontes chaveadas e modulação por largura de pulso (PWM)

A imensa maioria das fontes de alimentação usadas em computadores atuais — de smartphones a servidores — são **fontes chaveadas**. Em vez de reduzir a tensão de forma linear e contínua, esse tipo de fonte liga e desliga um circuito em altíssima frequência, controlando a fração do tempo em que o circuito permanece ligado. Essa técnica é chamada de **PWM** (*Pulse Width Modulation*, ou modulação por largura de pulso).

O princípio do PWM é simples: uma onda de tensão fixa (por exemplo, 10 V de amplitude) é ligada e desligada periodicamente. Se essa onda permanece "ligada" durante 50% de cada período, uma carga conectada a esse circuito "enxerga" uma tensão efetiva de 5 V — a metade da amplitude máxima. Se a onda permanece ligada apenas 23% do tempo, a carga enxerga aproximadamente 2,3 V.

Exemplo. Para gerar uma saída de 2,3 V a partir de uma onda de amplitude máxima de 10 V, basta manter o sinal em nível alto durante 23% de

cada período (*duty cycle* de 23%) e em nível baixo nos 77% restantes. Circuitos com comportamento predominantemente passivo — como motores e LEDs — respondem a essa alternância rápida como se a tensão fosse, de fato, constante e igual à fração correspondente da amplitude máxima.

A vantagem central do chaveamento sobre outras formas de redução de tensão (como um simples divisor resistivo) é a **eficiência**: um divisor de tensão dissipa parte da energia em forma de calor através dos resistores, enquanto o chaveamento — ligando e desligando o circuito, em vez de dissipar energia continuamente — perde muito menos energia nessa conversão. É por isso que as fontes de computador atuais são chamadas de **fontes chaveadas**: internamente, elas empregam circuitos de chaveamento em alta frequência (controlados por PWM) para produzir as diversas tensões contínuas exigidas pelos componentes, de forma muito mais compacta e eficiente do que uma fonte linear equivalente.

Figura pendente

forma de onda PWM com diferentes duty cycles (10%, 50%, 90%) e a tensão média efetiva resultante em cada caso

---

2.8 A fonte ATX

2.8.1 Padrão ATX e o conector principal

A **fonte ATX** é o padrão universal de fonte de alimentação para computadores desktop. Diferentemente de fontes com uma chave seletora de tensão (110 V/220 V) — cuja posição incorreta pode queimar o equipamento se ligado na tensão errada —, boa parte das fontes modernas de melhor qualidade é do tipo **full range** (ou *auto switch*): elas aceitam qualquer tensão alternada de entrada dentro de uma faixa ampla, tipicamente de 100 V a 240 V, ajustando-se automaticamente sem necessidade de seleção manual.

Atenção. Em fontes com chave seletora de tensão, a posição da chave deve sempre ser conferida antes de conectar o equipamento à tomada. Ligar uma fonte selecionada para 115 V em uma tomada de 220 V resulta, tipicamente, em dano imediato e irreversível ao equipamento — o mesmo princípio descrito na Seção 2.1.1.

A fonte ATX se conecta à placa-mãe por meio de um conector principal de **20 ou 24 pinos** [4] (dependendo do modelo da placa-mãe), fornecendo simultaneamente diversas tensões contínuas distintas, identificadas por um padrão de cores nos fios [5]:

Cor do fio	Tensão / sinal	Função
Laranja	+3,3 V	Alimentação de baixa tensão (memórias, lógica moderna)
Vermelho	+5 V	Alimentação de componentes diversos
Amarelo	+12 V	Alimentação de motores (ventoinhas, discos) e do processador
Roxo	+5 V (auxiliar/*standby*)	Alimentação permanente, mesmo com o computador desligado
Preto	COM (referência/terra)	Referência de potencial (0 V) para todas as demais tensões
Verde	PS_ON (*Power Supply On*)	Sinal de comando da placa-mãe para a fonte
Cinza	Power OK	Sinal de confirmação da fonte para a placa-mãe

Além dos fios listados, a fonte também disponibiliza tensões negativas (-12 V e -5 V) para funções específicas de compatibilidade histórica. Um segundo conector, tipicamente de 4 ou 8 pinos, fornece alimentação dedicada ao processador — algumas placas-mãe mais recentes exigem inclusive **dois** conectores de 8 pinos para essa finalidade, tornando necessário verificar a compatibilidade entre placa-mãe e fonte antes da montagem.

O padrão ATX substituiu o padrão anterior, chamado **AT**, cuja principal limitação era a ausência de comunicação eletrônica entre a placa-mãe e a fonte para o desligamento: em computadores com fonte AT, o comando de desligar (pelo sistema operacional) apenas exibia uma mensagem informando que o computador já podia ser desligado com segurança, mas o desligamento físico ainda dependia de o usuário acionar uma chave mecânica. O padrão ATX introduziu o desligamento controlado por software, que se tornou padrão em todos os computadores modernos [6].

Figura pendente

conector ATX de 24 pinos com o código de cores dos fios sobreposto

2.8.2 Sinais de controle: PS_ON e Power OK

O procedimento de inicialização de um desktop ilustra como a fonte, o botão de energia e a placa-mãe se comunicam eletricamente.

O botão de *power* do painel frontal do gabinete é, fisicamente, apenas um par de fios que, quando o botão é pressionado, se conectam momentaneamente — fechando um pequeno curto-circuito entre dois pinos específicos no conector do painel frontal da placa-mãe. Esse contato momentâneo é interpretado por um circuito da placa-mãe como um comando de ligar.

Ao reconhecer esse comando, a placa-mãe sinaliza à fonte que deve energizar seus circuitos de saída, unindo eletricamente o pino verde (**PS_ON**) ao pino preto (**COM**) — ou seja, colocando o pino de comando no mesmo nível de potencial que a referência de terra. Quando a fonte detecta essa condição, ela liga seus circuitos internos e passa a fornecer as tensões de saída (3,3 V, 5 V, 12 V etc.).

Esse acionamento não é instantâneo: existe um pequeno intervalo de tempo entre o comando de ligar e o momento em que todas as tensões de saída atingem seus valores nominais e estáveis. Ao final desse intervalo, a fonte sinaliza à placa-mãe, através do pino **Power OK**, que as tensões estão estabilizadas e prontas para uso. Somente após receber esse sinal a placa-mãe libera a energização dos demais componentes e inicia o procedimento de *boot*, chamando o primeiro programa executado pela placa-mãe — o POST.

Exemplo (metodologia de diagnóstico). Esse mecanismo fornece um procedimento sistemático para isolar problemas de energização em um computador que não liga:

1. Testar o botão físico do painel frontal, substituindo-o por um curto manual (por exemplo, com uma chave de fenda) diretamente nos pinos correspondentes da placa-mãe. Se o computador ligar dessa forma, o problema estava no botão ou em sua fiação — não na fonte nem na placa-mãe.
2. Se o curto manual no botão não resolver, desconectar a fonte da placa-mãe e testar a fonte isoladamente, unindo o pino verde (PS_ON) a qualquer pino preto (COM) no próprio conector da fonte. Se a fonte ligar (ventoinha gira, LEDs acendem), o problema está na placa-mãe. Se a fonte não ligar, o problema está na fonte.

Esse procedimento exemplifica, no contexto elétrico, a metodologia geral de diagnóstico por isolamento de módulos já apresentada no Capítulo 1: identificar o submódulo defeituoso testando cada elo da cadeia (botão → placa-mãe → fonte) separadamente.

Atenção. A fonte ligar durante esse teste — girando a ventoinha e produzindo as tensões nominais — indica apenas que ela consegue entregar tensão. Não garante que ela consegue entregar a **potência** total sob carga real: uma fonte deteriorada pode fornecer tensões corretas em vazio e ainda assim falhar ao alimentar componentes de alto consumo, como uma placa de vídeo. O teste do jumper é uma condição necessária, mas não suficiente, para validar uma fonte.

Figura pendente

fluxo botão → pinos do painel frontal → sinal PS_ON → fonte → sinal Power OK → placa-mãe, com indicação dos pontos de teste para diagnóstico

2.8.3 VRM: uma segunda fonte na placa-mãe

As tensões fornecidas pela fonte ATX (3,3 V, 5 V, 12 V) não correspondem, na maioria dos casos, às tensões efetivamente exigidas pelo processador e por outros componentes modernos, que frequentemente operam em tensões muito mais baixas e específicas. Por esse motivo, a própria placa-mãe contém uma "segunda fonte" interna: o **VRM** (*Voltage Regulator Module*), responsável por converter as tensões recebidas da fonte ATX nas tensões finais exigidas por cada componente. De forma equivalente, alguns processadores contam com um módulo de conversão próprio, às vezes chamado de PPM (*Processor Power Module*) [7].

A razão histórica para manter essa conversão adicional na placa-mãe — e não na fonte — é a compatibilidade: o padrão de saída da fonte ATX permaneceu estável ao longo de gerações de processadores e memórias, permitindo que um usuário reaproveite a mesma fonte em upgrades de placa-mãe, processador ou memória, desde que as novas exigências específicas de tensão sejam resolvidas pela conversão adicional na própria placa-mãe.

2.9 Eficiência energética e fator de potência

2.9.1 Selos de eficiência (80 Plus e ETA-Lambda)

Toda conversão de energia envolve perdas: parte da energia retirada da tomada é dissipada em forma de calor durante as sucessivas transformações (AC–AC, AC–DC, DC–DC) realizadas dentro da fonte. A **eficiência** de uma fonte é definida como a razão entre a potência efetivamente entregue aos componentes do computador e a potência total consumida na tomada.

Órgãos de certificação independentes, como o **80 Plus**, realizam ensaios de bancada e atribuem selos de eficiência às fontes comerciais. Mais recentemente, a Cybenetics Labs passou a manter dois programas de certificação distintos: **ETA** (eficiência energética) e **LAMBDA** (ruído acústico produzido pela fonte sob diferentes cargas) — são dois selos separados, avaliados independentemente; uma fonte pode ter um sem o outro [8]. Esses ensaios avaliam a eficiência em três níveis de carga, expressos como percentual da potência nominal da fonte:

- **Carga leve** — cerca de 20% da potência nominal.
- **Carga típica** — cerca de 50% da potência nominal.

- **Carga máxima** — 100% da potência nominal.

O selo 80 Plus é concedido em diferentes categorias (Standard, Bronze, Silver, Gold, Platinum, Titanium, em ordem crescente de exigência), cada uma exigindo um patamar mínimo de eficiência nas três cargas. Por exemplo, o selo **Standard** exige 80% de eficiência nas três cargas (80/80/80); o selo **Platinum** exige patamares mais altos, como 90% em carga leve, 92% em carga típica e 89% em carga máxima [9].

Exemplo. Uma fonte é tipicamente projetada para atingir sua eficiência máxima justamente na região de carga típica (em torno de 50% da potência nominal) — motivo pelo qual o dimensionamento recomendado de uma fonte para um computador (Seção 2.10) busca deixar o consumo real do sistema próximo dessa faixa.

Um fator adicional observado experimentalmente é que uma fonte tende a ser mais eficiente quando alimentada em 220 V do que em 110 V, para a mesma carga [10] — consequência direta do mesmo princípio de efeito Joule discutido na Seção 2.2: em uma tensão de entrada mais alta, a corrente de entrada correspondente é menor, reduzindo as perdas internas.

Selos de eficiência não devem ser confundidos com selos de potência: uma fonte de "650 W" descreve a potência que ela é capaz de fornecer, e sua eficiência (Standard, Bronze, Gold etc.) descreve a proporção dessa energia que é efetivamente aproveitada, e não perdida em calor, na conversão a partir da tomada.

2.9.2 Fator de potência e PFC

O **fator de potência** é uma segunda métrica de qualidade de uma fonte, distinta da eficiência. É definido como a razão entre a **potência ativa** (a potência que efetivamente realiza trabalho) e a **potência aparente** (a potência total que o circuito precisa "reservar" para operar, incluindo a energia armazenada temporariamente em componentes reativos, como capacitores e indutores, sem realizar trabalho útil):

$$\text{Fator de potência} = \frac{P_{\text{ativa}}}{S_{\text{aparente}}}$$

Analogia. A relação entre potência ativa e potência aparente pode ser comparada ao enchimento de uma caixa d'água residencial. A água usada para encher a caixa d'água tem um custo (é fornecida pela concessionária), mas, enquanto está apenas armazenada na caixa, não está realizando nenhum trabalho útil (lavar, limpar, beber). Só quando a torneira é aberta e a água efetivamente flui é que o trabalho é realizado. Da mesma forma, capacitores e indutores dentro de um circuito de fonte precisam ser "carregados" (o que consome energia da rede), mas essa energia armazenada, por si só, não realiza trabalho computacional — apenas parte dela é convertida, de fato, em tensões contínuas úteis aos componentes do computador.

Fontes com correção de fator de potência (**PFC**, *Power Factor Correction*) empregam circuitos adicionais — passivos (bancos de capacitores e induto-

res) ou ativos (circuitos eletrônicos dedicados) — para reduzir essa parcela reativa e aproximar a potência aparente da potência ativa. No Brasil, consumidores residenciais não são cobrados por energia reativa (essa cobrança se aplica a consumidores do Grupo A — média/alta tensão, o que inclui grandes consumidores industriais e comerciais, não apenas industriais) [11], de forma que a presença de PFC em uma fonte doméstica não reduz a conta de energia do usuário — mas é, ainda assim, um indicador de qualidade construtiva do circuito.

Analogia. A presença de PFC ativo em uma fonte é comparável à presença de airbag em um automóvel: não torna o carro mais rápido nem mais barato de operar, mas é um diferencial de qualidade e segurança de projeto, frequentemente destacado em anúncios comerciais.

Atenção. O fator de potência e a eficiência de uma fonte são métricas independentes: uma fonte com PFC ativo não é, por esse motivo isolado, necessariamente mais eficiente — a eficiência é determinada pelo ensaio de conversão de energia (Seção 2.9.1), e não pelo fator de potência.

2.10 Dimensionamento e diagnóstico de fontes

2.10.1 Trilhos de potência (rails)

Ao especificar uma fonte, o fabricante não apenas informa a potência total nominal (por exemplo, 650 W), mas também detalha, em uma tabela de especificações, a corrente máxima que cada **trilho de tensão** (*rail*) — 3,3 V, 5 V, 12 V, além dos trilhos negativos e auxiliares — é capaz de fornecer individualmente.

Exemplo. Considere uma fonte nominal de 650 W com a seguinte especificação de trilhos:

Trilho	Tensão	Corrente máxima	Potência do trilho
A	3,3 V	25 A	82,5 W
B	5 V	25 A	125 W
A + B (combinados)	—	—	150 W (máximo compartilha
C	12 V	52 A	624 W
D	12 V (auxiliar)	0,5 A	6 W
E	5 V (auxiliar)	2,5 A	12,5 W

Somando a potência máxima teórica de todos os trilhos individualmente ($150 + 624 + 6 + 12{,}5 = 792{,}5$ W), o resultado ultrapassa os 650 W nominais da fonte. Isso não significa que o fabricante esteja anunciando uma especificação incorreta: significa que os 650 W nominais são **compartilhados** entre os trilhos, e que não é possível extrair a corrente máxima de todos

os trilhos simultaneamente. Na prática, o trilho de 12 V (tipicamente o que alimenta processador e placa de vídeo, os dois maiores consumidores de um desktop) concentra a maior reserva de potência da fonte, e os demais trilhos (3,3 V e 5 V, tipicamente usados por memória, armazenamento e portas USB) compartilham uma parcela menor do total.

Figura pendente

tabela de trilhos de uma fonte real com tensões e correntes máximas por rail

2.10.2 Metodologia de dimensionamento

Ao dimensionar uma fonte para um computador desktop, os dois maiores consumidores de energia — e, portanto, os fatores decisivos no dimensionamento — são o **processador** e a **placa de vídeo**: uma GPU de alto desempenho pode consumir, sozinha, uma potência muito superior à soma de todos os demais componentes do sistema. Componentes adicionais (memórias, unidades de armazenamento, ventoinhas) contribuem com potências individuais menores, mas devem ser somados quando existem em grande quantidade — como em servidores com múltiplos discos.

Fabricantes de fontes disponibilizam calculadoras de dimensionamento *on-line* que, a partir da lista de componentes do sistema (processador, GPU, memória, armazenamento, ventoinhas), estimam o consumo total e recomendam uma potência de fonte adequada. Essas calculadoras tipicamente recomendam uma margem confortável acima do consumo estimado do sistema — não porque o sistema vá efetivamente consumir aquele valor, mas para que o consumo real do sistema recaia na faixa de carga típica (em torno de 50% da potência nominal), onde a fonte opera com eficiência máxima (Seção 2.9.1).

Analogia. A relação entre carga leve, típica e máxima de uma fonte pode ser comparada ao esforço de um atleta correndo em terrenos diferentes: correr em terreno plano corresponde a uma carga típica; correr ladeira abaixo, a uma carga leve; correr ladeira acima, a uma carga máxima — e o consumo de “oxigênio” (corrente) do atleta varia de acordo.

Atenção. Capacitores de grande capacitância no interior de uma fonte podem reter carga elétrica significativa por dias mesmo após o equipamento ser desconectado da tomada. Uma fonte nunca deve ser aberta sem os devidos cuidados de segurança, e o escopo deste curso não inclui a manutenção interna do circuito da fonte (troca de capacitores, indutores ou outros componentes) — apenas o diagnóstico que permite decidir pela substituição do módulo completo.

2.10.3 Roteiro prático de diagnóstico

Somando os conceitos apresentados neste capítulo, o diagnóstico de um computador que apresenta suspeita de falha na fonte segue, tipicamente,

esta sequência:

1. **Inspeção visual** — verificar conexões soltas, a posição correta da chave seletora de tensão (em fontes não *full range*) e sinais evidentes de dano (capacitores estufados, vazamentos, queimaduras).
2. **Isolamento do ambiente de teste** — testar o computador com um monitor e uma tomada sabidamente funcionais, para que qualquer falha observada seja atribuída ao computador, e não ao ambiente de teste.
3. **Observação dos indicadores de energização** — ao ligar o computador, observar se a ventoinha da fonte gira, se a ventoinha do processador gira e se LEDs de atividade acendem.
4. **Verificação do POST** — a ausência do bipe característico do POST pode indicar um problema anterior ao teste do BIOS (fonte, botão, conexão) ou, simplesmente, um *speaker* (alto-falante interno de aviso) desconectado ou invertido em polaridade — o *speaker* é um componente polarizado (positivo e negativo) e deve ser conectado na orientação correta indicada na placa-mãe.
5. **Teste isolado da fonte**, conforme descrito na Seção 2.8.2, unindo o pino PS_ON (verde) a um pino COM (preto) diretamente no conector da fonte.
6. **Manutenção preventiva de contatos** — mau contato nos módulos de memória RAM é uma causa frequente de falha intermitente no POST (por exemplo, o computador não bipar em uma tentativa e bipar normalmente na tentativa seguinte). O procedimento de limpeza consiste em remover os módulos de memória, aplicar um produto de limpeza de contatos eletrônicos (álcool isopropílico ou produto similar, altamente volátil) sobre os pontos de contato dourados, aguardar a evaporação completa e reinserir o módulo, certificando-se de que o encaixe nas travas do slot esteja completo (indicado por um clique audível em ambos os lados do slot).

Esse roteiro exemplifica, de forma concreta, a metodologia de manutenção corretiva por formulação e teste de hipóteses apresentada no Capítulo 1: cada etapa isola uma parte do sistema, permitindo concluir, por eliminação, em qual submódulo está o problema.

2.11 Integração da placa-mãe: chipset, painel frontal e aterramento do gabinete

Como visto nas seções anteriores, a fonte se conecta à placa-mãe em dois pontos (o conector principal e o conector de alimentação do processador). A placa-mãe, por sua vez, é responsável por distribuir essa energia — e por integrar todos os demais componentes do computador — através de um conjunto de circuitos controladores conhecido como **chipset**.

O chipset é, como o nome sugere, um *conjunto de chips* que provê as funcionalidades de interconexão da placa-mãe: controladores de portas USB, de rede Ethernet, de Wi-Fi (quando presente), de linhas de memória, entre outros. Um mesmo chipset (identificado por um código de modelo, como "B860") pode ser implementado por diferentes fabricantes de placa-mãe, cada um oferecendo uma organização física e um conjunto de recursos adicionais diferentes — explicando por que placas-mãe de fabricantes distintos, baseadas no mesmo chipset, podem ter preços significativamente diferentes: a diferença normalmente está em funcionalidades extras (como Wi-Fi integrado) e não necessariamente em qualidade superior de um fabricante sobre o outro.

Manual da placa-mãe. O manual técnico fornecido pelo fabricante de cada placa-mãe é a fonte de referência definitiva para identificar a pinagem de conectores, os requisitos de alimentação do processador (por exemplo, se são necessários um ou dois conectores de 8 pinos) e os procedimentos específicos daquele modelo, como o Clear CMOS (Capítulo 1). Um técnico deve sempre consultar o manual de cada placa-mãe individualmente, em vez de presumir que todos os modelos seguem exatamente o mesmo layout de pinos.

2.11.1 Painel frontal

O conector de **painel frontal** (*front panel* ou *system panel header*) da placa-mãe recebe os fios provenientes do gabinete: o botão de energia, o botão de reinicialização (*reset*), o LED de energia e o LED de atividade do disco. Como o gabinete e a placa-mãe são fabricados por empresas diferentes, a correspondência exata de pinos varia de modelo para modelo e deve ser conferida no manual da placa-mãe.

Uma distinção importante deve ser observada nesses conectores: **botões** (power e reset) não possuem polaridade — o fio pode ser conectado em qualquer uma das duas orientações possíveis, sem alterar o funcionamento. **LEDs**, por serem diodos emissores de luz, possuem polaridade definida (um terminal positivo e um negativo): conectados na orientação invertida, simplesmente não acendem, sem dano ao componente.

2.11.2 Aterramento do gabinete

O chassi metálico do gabinete deve manter continuidade elétrica com o condutor de aterramento da instalação — tanto através do cabo de alimentação da fonte quanto da fixação mecânica da própria fonte e da placa-mãe ao gabinete. Essa continuidade permite que eventuais excessos de carga acumulados no chassi (Seção 2.4.1) sejam escoados de forma segura para a terra, em vez de se acumularem e serem sentidos pelo usuário ao tocar o gabinete.

Atenção. A verificação de continuidade entre o terra da tomada e o chassi do gabinete deve sempre ser feita com um multímetro, nunca por contato

direto. Uma instalação elétrica com fase e terra invertidos por erro de instalação pode energizar todo o chassi de um computador aterrado incorretamente — situação já registrada em ambientes reais, inclusive institucionais, e potencialmente fatal.

Pulseiras antiestáticas de aterramento, usadas para escoar cargas eletrostáticas do próprio corpo do técnico durante o manuseio de componentes sensíveis, dependem da confiabilidade da instalação elétrica ao qual são conectadas: um técnico deve avaliar, caso a caso, se confia o suficiente na instalação disponível antes de se conectar diretamente a ela por meio de uma pulseira — preferindo, em caso de dúvida, outras formas de controle eletrostático que não impliquem conexão direta a uma instalação de aterramento não verificada.

Figura pendente

diagrama do painel frontal de uma placa-mãe com os pinos de power switch, reset, HDD LED e power LED identificados, indicando polaridade dos LEDs

---

2.12 Síntese do capítulo

Este capítulo apresentou as grandezas elétricas fundamentais — tensão, corrente, resistência e potência — e sua aplicação direta ao diagnóstico de falhas na cadeia que vai da tomada até a placa-mãe: proteção por disjuntores, aterramento e segurança contra choque elétrico, conversão de energia por transformadores e circuitos retificadores, a distinção entre fontes lineares e chaveadas, a arquitetura e os sinais de controle da fonte ATX, e os critérios de eficiência, fator de potência e dimensionamento que orientam a escolha e o diagnóstico de uma fonte real. Do ponto de vista metodológico, o capítulo reforça e aplica em contexto elétrico o princípio de diagnóstico por isolamento de módulos apresentado no Capítulo 1 — testar cada elo da cadeia tomada–fonte–placa-mãe isoladamente para localizar a origem de uma falha.

A energia entregue e regulada pela fonte, tratada aqui, é justamente o que torna possível o funcionamento do componente estudado no Capítulo 3: o processador. Os conceitos de tensão, corrente e dissipação de potência por efeito Joule retornarão, em outra escala, na discussão sobre consumo energético, dissipação térmica e desempenho de processadores — onde o mesmo compromisso entre potência entregue e eficiência de conversão, já estabelecido neste capítulo para a fonte, reaparece na análise de benchmark e desempenho da CPU.

2.13 Referências

1. NEOENERGIA COSERN. “Normas Técnicas — Padrão de Entrada de Energia.” Disponível em: https://www.neoenergia.com/web/rn/normas-tecni
2. ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. *NBR 5410: Instalações elétricas de baixa tensão*. ABNT. (Edição a confirmar pelo autor.) BRASIL. Ministério do Trabalho e Emprego. *NR-10: Segurança em Instalações e Serviços em Eletricidade*. (Edição a confirmar pelo autor.)
3. Engenharia 360. “O inventor brasileiro do chuveiro elétrico.” Disponível em: https://engenharia360.com/o-inventor-brasileiro-do-chuveiro-e WIKIPÉDIA. “Chuveiro elétrico.” Disponível em: https://pt.wikipedia.org/wiki/Chuveiro_el%C3%A9trico.
4. WIKIPEDIA. “ATX.” Disponível em: https://en.wikipedia.org/wiki/ATX.
5. INTEL CORPORATION. *ATX12V Power Supply Design Guide*. (Versão a confirmar pelo autor.) Como fonte secundária: eTechnophiles. “ATX Power Supply Connector Pinout (20 & 24 pins).” Disponível em: https://www.etechnophiles.com/atx-power-supply-connector-pinout/.
6. WIKIPEDIA. “ATX.” Disponível em: https://en.wikipedia.org/wiki/ATX; COMPUTER HOPE. “What Is ATX?” Disponível em: https://www.computerhope.com/jargon/a/atx.htm.
7. WIKIPEDIA. “Voltage regulator module.” Disponível em: https://en.wikipedia.org/wiki/Voltage_regulator_module.
8. CYBENETICS LABS. “ETA — PSU Efficiency Certification.” Disponível em: https://www.cybenetics.com/index.php?option=eta; “LAMBDA — PSU Noise Level Certification.” Disponível em: https://www.cybenetics.com/index.php?option=lambda-%28power-supplies%29.
9. WIKIPEDIA. “80 Plus.” Disponível em: https://en.wikipedia.org/wiki/80_Plus.
10. ANANDTECH. “The Seasonic Focus Plus Gold 750FX 750W PSU Review.” Disponível em: https://www.anandtech.com/show/14338/the-seasonic 3.
11. AGÊNCIA NACIONAL DE ENERGIA ELÉTRICA (ANEEL). Resolução Normativa nº 414, de 9 de setembro de 2010. (Edição/atualizações a confirmar pelo autor.)

3 Processador, Benchmark e Dimensionamento

Neste capítulo você vai estudar a evolução histórica e os limites físicos do processador moderno — da lei de Moore ao problema do calor —, a distinção entre arquiteturas RISC e CISC, a organização interna de um processador multicore (núcleos, threads, cache e pipeline), a relação entre resolução de vídeo e demanda de processamento, e a metodologia de benchmark que permite comparar hardware de fabricantes e gerações diferentes para realizar o dimensionamento de um computador dentro de um orçamento.

3.1 O processador como peça central da especificação

Dentro da tarefa de especificar uma máquina para uma determinada demanda — um dos objetivos centrais desta disciplina —, o processador (CPU, *Central Processing Unit*) ocupa posição crítica. A escolha da CPU não é uma decisão isolada: ela impõe restrições sobre a placa-mãe (via soquete e chipset) e sobre a fonte de alimentação (via potência exigida, tema do Capítulo 2), e é frequentemente o fator que define o desempenho final da máquina — o chamado **gargalo** (*bottleneck*) da especificação.

Se o processador fosse tratado como uma classe de programação, seus atributos característicos seriam:

- **Arquitetura** (RISC ou CISC, tratada na Seção 3.2)
- **Litografia** (o processo de fabricação, tratado na Seção 3.3)
- **Fabricante**
- **Soquete** (o encaixe físico na placa-mãe, tratado na Seção 3.7)
- **Geração**
- **Número de núcleos** (*cores*)
- **Número de threads**
- **Potência** (em watts)
- **Clock** (frequência de trabalho)

Esses atributos formam a "ficha técnica" completa de um processador, tal como aparece nos anúncios comerciais. O restante deste capítulo desenvolve cada um desses atributos e apresenta a metodologia — baseada em *benchmark* — que permite comparar processadores diferentes de forma objetiva.

Figura pendente

anúncio real de processador com os atributos arquitetura, litografia, soquete, núcleos, threads, potência e clock destacados

3.2 Arquiteturas RISC e CISC

No nível do *Instruction Set Architecture* (ISA — o conjunto de instruções que uma CPU é capaz de executar), o mercado de processadores se organiza historicamente em dois grandes paradigmas:

- **CISC** (*Complex Instruction Set Computer*) — conjunto de instruções complexo. Seu grande diferencial é a **retrocompatibilidade**: um processador CISC lançado hoje ainda é capaz de executar instruções presentes em processadores lançados décadas atrás. Essa característica gera discussões reais na engenharia de software — por exemplo, entre os desenvolvedores do núcleo (*kernel*) do sistema operacional Linux há debates recorrentes sobre manter, ou não, suporte a instruções herdadas de arquiteturas de computadores como o 386 e o 486, da década de 1990 [1].
- **RISC** (*Reduced Instruction Set Computer*) — conjunto de instruções reduzido e simplificado. Instruções mais simples exigem hardware mais simples, o que resulta em maior eficiência energética (mais desempenho por watt consumido) em comparação a um processador CISC equivalente.

Exemplo. Processadores da família Pentium 2, Pentium 3, Atom, e as diversas gerações da linha Ryzen são exemplos de processadores CISC (arquitetura x86), demonstrando décadas de retrocompatibilidade dentro dessa família.

Por muito tempo, essa escolha de arquitetura era irrelevante para o técnico de manutenção de desktops e notebooks: processadores RISC ficavam restritos a smartphones, onde a eficiência energética é decisiva para a autonomia de bateria. Esse cenário mudou. Hoje, processadores RISC (como os chips Apple Silicon e os processadores Snapdragon) também competem no mercado de desktops e laptops. Consequentemente, ao especificar computadores portáteis — por exemplo, para uma equipe de professores que leva o equipamento para dar aula fora do laboratório —, a arquitetura passa a ser uma escolha técnica relevante: um computador de arquitetura CISC de alto desempenho pode exigir tantos acessórios de suporte (fonte robusta, ventilação adicional) que se torna impraticável para uso móvel, enquanto um equivalente RISC entrega autonomia de bateria muito superior.

> **Atualidade de mercado — o chip RISC da NVIDIA para computação de IA local**
>
> Em 1º de junho de 2026, durante a feira Computex em Taipei, a NVIDIA — historicamente fabricante de placas de vídeo GeForce

— anunciou o **RTX Spark**, seu primeiro chip de arquitetura RISC (especificamente Arm) voltado a computadores pessoais, com memória unificada (nos moldes do que a Apple já pratica em seus chips Apple Silicon) e capacidade de até 128 GB de memória compartilhada entre CPU, GPU e NPU [2]. A NVIDIA e a Microsoft, em conjunto, descreveram o lançamento como parte de uma "reinvenção do computador".

A novidade central não é apenas a arquitetura RISC em si, mas o fato de que o chip foi projetado com instruções voltadas ao chamado *loop agêntico*: da mesma forma que o ciclo clássico de busca–decodificação–execução (Seção 3.6) fundamenta o funcionamento de qualquer CPU, o RTX Spark inclui, em nível de processador, suporte a um ciclo de raciocínio–execução–validação voltado à execução local de agentes de inteligência artificial (assistentes de programação e automação que operam diretamente na máquina do usuário, sem depender inteiramente de processamento em nuvem).

O chip já nasce com parcerias fechadas com fabricantes de notebooks como Lenovo, HP, Dell (Microsoft Surface), Asus e MSI, colocando pressão competitiva simultânea sobre Intel, AMD e Apple. Trata-se de uma notícia de mercado recente — o leitor deste livro deve tratá-la como indicativo de tendência, não como fato estático: a configuração exata de produtos comerciais baseados nesse chip deve ser verificada em fontes atualizadas no momento da leitura.

3.3 A Lei de Moore e os limites físicos da litografia

Historicamente, observou-se que os processadores lançados pela Intel dobravam a quantidade de transistores em relação ao modelo anterior a cada aproximadamente dois anos. Essa observação empírica foi formalizada como meta de desenvolvimento da indústria e ficou conhecida como **lei de Moore**.

Como o transistor é a unidade fundamental de construção de portas lógicas, circuitos aritméticos, circuitos de controle e circuitos de memória, um processador com o dobro de transistores tende a oferecer maior poder computacional. O gráfico histórico de contagem de transistores por chip, de 1970 a meados da década de 2010, mostra crescimento de milhares para dezenas de bilhões de transistores por chip.

Analogia. Se um saco comporta dez bolas de um determinado tamanho e a meta passa a ser encaixar vinte bolas no mesmo saco, a única solução é diminuir o tamanho das bolas. Da mesma forma, para dobrar a contagem de transistores mantendo o tamanho físico do chip aproximadamente constante, a indústria precisou miniaturizar continuamente os transistores

— processo que trouxe a fabricação de semicondutores à escala dos **nanômetros** (1 nm = 10^{-9} m, um bilionésimo de metro; a título de comparação, uma folha de papel comum tem cerca de 100.000 nm de espessura, ou 0,1 mm) [3].

Fabricante/arquitetura	Litografia aproximada
Intel, 12ª geração	10 nm
AMD Zen 3 (Ryzen série 5000)	7 nm
AMD Zen 4 (Ryzen série 7000)	5 nm
Apple Silicon (M1)	5 nm
Qualcomm Snapdragon (gerações recentes)	~3 nm

Dados de litografia levantados em 2026; a litografia real muda a cada nova geração de produto [4].

Reduzir a litografia traz dois benefícios simultâneos e vinculados por física básica: como a velocidade de deslocamento do elétron é aproximadamente constante, diminuir a distância física que o elétron percorre entre terminais do transistor reduz o tempo de processamento (maior velocidade) e reduz a perda de energia por efeito Joule (maior eficiência energética). Um processador fabricado numa litografia menor é, por concepção de projeto, mais rápido e mais eficiente do que um equivalente fabricado numa litografia maior.

O processo de fabricação. O silício, elemento com quatro elétrons na camada de valência, é abundante — extraído da areia, não de jazidas raras — e forma cristais estáveis por ligação covalente entre átomos vizinhos. Após purificado e cristalizado, o silício é fatiado em discos finos chamados *wafers*. Sobre o *wafer*, um processo óptico de precisão chamado **litografia** projeta, com o auxílio de lasers e lentes especiais, a imagem do circuito do processador, "queimando" no silício o desenho dos transistores já interligados em portas lógicas. O processo de **dopagem** (introdução controlada de impurezas, como boro, na estrutura cristalina) cria as regiões de tipo N e tipo P que formam diodos e transistores. Concluída a gravação, o *wafer* é fatiado em unidades individuais (os *dies*), que recebem encapsulamento protetor e se tornam o chip vendido no mercado, no formato que se conecta ao soquete da placa-mãe.

Historicamente, a Intel controlava tanto o projeto quanto a fabricação (fundição) de seus próprios chips. Outros fabricantes — AMD, Apple, Qualcomm — projetam seus chips mas terceirizam a fabricação para fundições especializadas, principalmente a TSMC (*Taiwan Semiconductor Manufacturing Company*), sediada em Taiwan. A dificuldade da Intel em reduzir sua própria litografia abaixo de 10 nm é um fator relevante da perda de competitividade da empresa nos últimos anos, a ponto de ela também ter passado a recorrer a fundições terceirizadas para parte de sua produção. A concentração geográfica dessa capacidade de fabricação de semicondutores de ponta em Taiwan é, adicionalmente, um fator de tensão geopolítica relevante, dado o contexto de disputa territorial entre Taiwan e a China —

um ponto que tem desdobramentos diretos sobre preço e disponibilidade de chips no mercado global.

Figura pendente

fotografia de um wafer de silício antes e depois do processo de litografia] [IMAGEM: gráfico da contagem de transistores por chip ao longo do tempo, 1970–2016 — se adaptado de fonte de terceiros (ex.: Our World in Data), creditar a fonte e confirmar a licença antes de publicar; alternativa: construir com dados brutos próprios/públicos para evitar qualquer dúvida de direito autoral

3.4 Calor, thermal throttling e o fim do núcleo único

Alocar mais transistores no mesmo espaço físico tem uma consequência inevitável: mais calor gerado na mesma área. Processadores das gerações mais antigas (até o início dos anos 2000) nem sempre possuíam proteção térmica — e a diferença entre ter ou não essa proteção podia ser dramática: um vídeo de demonstração histórico e amplamente citado (Tom's Hardware, ~2000–2001) comparou um Intel Pentium III e um AMD Athlon "Thunderbird", ambos sem *cooler*, sob carga. O Pentium III, equipado com um monitor térmico que desligava o processador ao atingir a temperatura crítica, travou o sistema mas sobreviveu; o Athlon, sem qualquer proteção térmica, literalmente queimou e soltou fumaça em menos de um segundo, chegando a 370 °C no núcleo [5].

Os processadores modernos resolvem esse risco com o mecanismo de **thermal throttling** (acelerador térmico): ao detectar que a temperatura se aproxima de um limite pré-definido, o processador reduz automaticamente sua frequência de processamento; ao esfriar, ele volta a acelerar. O resultado é um ciclo contínuo de aceleração e desaceleração, governado pela temperatura.

O *thermal throttling* deixou de ser apenas uma medida de emergência e passa a ser usado como característica de projeto: um MacBook sem ventoinha reduz o clock progressivamente conforme aquece, enquanto um MacBook Pro com ventoinha ativa consegue sustentar clocks mais altos por mais tempo, adiando o momento do *throttling*.

O limite que originou o multicore. No início dos anos 2000, a Intel chegou a planejar um Pentium 4 de 4 GHz, mas cancelou o lançamento em outubro de 2004: a dissipação de calor necessária tornava o produto inviável para os computadores da época [6]. A solução encontrada pela indústria não foi continuar aumentando a densidade de transistores num único núcleo, mas **duplicar o número de núcleos de processamento** dentro do mesmo encapsulamento — cada um mais simples e mais frio do que seria um único núcleo hipertrofiado. Nasceu assim a era **multicore** no mercado de desktop, por volta de 2005, com o lançamento do Pentium D

pela Intel e do Athlon X2 pela AMD [7]. Essa mudança de direção arquitetural se mantém até hoje: não houve retorno ao paradigma de núcleo único, apenas a adição de novas unidades de processamento especializadas (GPU, NPU), tratadas ao final deste capítulo.

Figura pendente

gráfico comparando temperatura e FPS de um processador com e sem cooler durante um jogo

3.5 A era multicore: núcleos, cache e hyper-threading

Um **core** (núcleo) é uma unidade de processamento física — hardware. Um processador multicore contém, dentro de um único encapsulamento, múltiplas cópias completas da estrutura de processamento (unidade lógico-aritmética, registradores, unidades de controle de fluxo), cada uma com sua própria memória cache de nível 1 (L1) e nível 2 (L2). Externamente aos núcleos, mas ainda dentro do chip, fica a cache de nível 3 (L3), compartilhada entre todos os núcleos — funcionando como uma mesa de trabalho comum para a troca de dados entre eles quando uma tarefa é dividida entre múltiplos núcleos.

Nota sobre controle de qualidade. Nem todo processador de uma mesma linha de produção nasce igual: após a fabricação, cada chip passa por uma bancada de testes. Esse processo, chamado *binning*, é real e bem documentado na indústria [8]: às vezes, um chip com todos os núcleos aprovados no controle de qualidade é vendido como a linha superior (por exemplo, um "i7"), enquanto um chip do mesmo *die*, com núcleos defeituosos desabilitados, é vendido como um produto de linha inferior ("i5" ou "i3"). Essa não é, porém, a explicação universal da diferenciação entre linhas: em boa parte dos casos — especialmente em gerações mais recentes — i5 e i3 são fabricados a partir de *dies* fisicamente menores e diferentes do i7, não do mesmo chip com núcleos desabilitados.

Uma **thread**, por sua vez, é uma unidade lógica — uma simulação em software da existência de mais processadores do que os fisicamente presentes. A tecnologia responsável por essa simulação é chamada de **hyper-threading** (nomenclatura da Intel) ou **SMT**, *simultaneous multi-threading* (nomenclatura da AMD). Um núcleo físico com hyper-threading de grau 2 é enxergado pelo sistema operacional como dois processadores lógicos.

Analogia. Um único atendente de balcão que tanto cobra quanto embala a compra do cliente está, na prática, desempenhando duas tarefas — mas continua sendo uma única pessoa fazendo alternadamente uma coisa e outra, aproveitando os intervalos ociosos de uma tarefa para avançar a outra. É exatamente esse aproveitamento de tempo ocioso do hardware, viabilizado pela técnica de pipeline (Seção 3.6), que permite a um núcleo físico simular o comportamento de dois núcleos lógicos.

Exemplo. Um anúncio real de processador Intel Core i5-12400F (12ª geração) especifica 6 núcleos físicos (*cores*) com hyper-threading, totalizando 12 threads [9] — ou seja, o sistema operacional identifica 12 processadores lógicos disponíveis, embora fisicamente existam apenas 6.

Essa complexidade, no entanto, tem um custo que sobe para as camadas de software: programação paralela (a técnica de dividir um algoritmo para ser executado simultaneamente em múltiplos núcleos) é uma disciplina distinta e mais difícil do que a programação sequencial convencional, e a maioria dos programas aplicativos comuns não é otimizada para tirar proveito de mais do que quatro a seis núcleos simultâneos. A consequência prática desse descompasso — poder computacional disponível, mas ocioso — é discutida na Seção 3.11.

Figura pendente

fotografia (die shot) de um processador com quatro núcleos, cache L1/L2 por núcleo e cache L3 compartilhada identificados

3.6 Pipeline: sobreposição de estágios de execução

O ciclo básico de funcionamento de uma CPU consiste em três etapas repetidas continuamente: **busca** da instrução na memória, **decodificação** dessa instrução e **execução** da instrução decodificada. A técnica de **pipeline** consiste em sobrepor essas etapas para diferentes instruções simultaneamente, em vez de executar o ciclo completo de uma instrução antes de iniciar a próxima.

Analogia (adaptada da clássica analogia da lavanderia de Patterson & Hennessy [10]). Considere uma lavanderia com quatro estações — lavar, secar, dobrar e guardar —, em que cada etapa leva uma hora. Numa execução sem pipeline, o ciclo completo (lavar → secar → dobrar → guardar) leva 4 horas, e o próximo ciclo só começa depois que o anterior termina inteiramente: quatro ciclos completos consumiriam 16 horas (4 etapas × 1 hora × 4 ciclos), com cada máquina ficando ociosa na maior parte do tempo. Com pipeline, assim que a máquina de lavar termina uma carga e a transfere para a secadora, ela já recebe uma nova carga de roupa suja — nenhum equipamento fica parado. Nesse esquema, em sete horas obtêm-se quatro ciclos completos de saída, contra as dezesseis horas necessárias sem sobreposição — um ganho de mais do que o dobro de produtividade utilizando exatamente o mesmo hardware, apenas evitando ociosidade.

É essa mesma lógica de aproveitamento de estágios ociosos do circuito de busca–decodificação–execução que permite ao hyper-threading (Seção 3.5) fazer um único núcleo físico atender a duas instruções em estágios diferentes do pipeline ao mesmo tempo, simulando dois processadores lógicos. O termo *pipeline* não é exclusivo da arquitetura de computadores — é usado

de forma equivalente em engenharia de produção industrial para descrever qualquer processo organizado em estágios reaproveitáveis e encadeados.

Figura pendente

diagrama de linha do tempo mostrando quatro estágios de pipeline sobrepostos ao longo de sete unidades de tempo

3.7 Soquete, geração e compatibilidade de mercado

O **soquete** é o conector físico da placa-mãe onde o processador é encaixado. Cada geração de processador é compatível apenas com um conjunto específico de soquetes: uma placa-mãe fabricada para processadores Intel não aceita processadores AMD, e vice-versa — e mesmo dentro de um mesmo fabricante, gerações diferentes de processador frequentemente exigem soquetes diferentes.

Esse detalhe tem consequência direta para o trabalho de manutenção. Historicamente, a AMD tem mantido o mesmo soquete por múltiplas gerações de processador — o soquete AM4 atendeu várias gerações consecutivas de Ryzen, e o soquete AM5, lançado mais recentemente, teve seu suporte estendido pela AMD até o ano de 2029 [11]. Isso significa que uma placa-mãe AM5 fabricada hoje continuará aceitando processadores lançados anos depois. A Intel, por sua vez, tende a alterar o soquete a cada uma ou duas gerações.

Consequência prática para o técnico. Diante de uma placa-mãe com defeito comprovado, cujo processador continua funcional, a estratégia de diagnóstico modular (transferir o processador para outra placa-mãe compatível) é significativamente mais viável em um ecossistema AMD — em que placas-mãe compatíveis com processadores mais antigos continuam disponíveis no mercado — do que em um ecossistema Intel, no qual encontrar uma placa-mãe compatível com um processador de poucos anos atrás pode ser difícil e caro o suficiente para tornar mais econômico descartar o processador funcional e comprar um conjunto novo (processador e placa-mãe).

Esse tipo de escolha de projeto — que reduz a vida útil economicamente viável de um componente ainda funcional — se aproxima do fenômeno mais amplo de **obsolescência programada**, observável em outras indústrias. Do ponto de vista puramente técnico, isso reforça por que a análise de soquete e longevidade de plataforma deve fazer parte da recomendação de compra ao cliente, e não apenas a comparação de desempenho bruto — que é tratada, com apoio de benchmark, na Seção 3.9. Cabe notar, entretanto, que a Intel frequentemente compensa essa desvantagem de longevidade praticando preços mais agressivos, de modo que a escolha entre os dois fabricantes é sempre uma decisão de custo-benefício, não uma resposta única.

Figura pendente

foto comparativa de soquete AMD (AM4/AM5) e soquete Intel (LGA) com os pinos/contatos visíveis

3.8 Resolução, taxa de quadros e demanda gráfica

Um monitor é, fisicamente, uma matriz de milhões de **pixels**, cada um capaz de assumir uma cor por meio da combinação aditiva das cores primárias de luz — vermelho, verde e azul (RGB, *Red-Green-Blue*). A síntese aditiva parte do preto (monitor desligado) e soma luz até formar as demais cores; o processo é diferente da impressão em papel, que parte do branco e usa síntese subtrativa de tinta (ciano, magenta, amarelo).

A **resolução** de um monitor descreve a quantidade estática de pixels — largura por altura:

Nome comercial	Resolução (pixels)
HD	1280 × 720 (também chamado 720p)
Full HD	1920 × 1080 (também chamado 1080p)
2K / QHD	2560 × 1440
4K	3840 × 2160

Quanto maior a resolução, maior o número de pixels que o hardware precisa calcular, transportar e copiar continuamente — trabalho que recai, em última instância, sobre o processador (e sobre a GPU, tratada na Seção 3.9).

A segunda variável relevante é a **taxa de quadros por segundo** (FPS, *frames per second*): quantas vezes por segundo a imagem inteira é atualizada. O olho humano percebe movimento contínuo a partir de aproximadamente 24 atualizações por segundo — o mesmo princípio de um desenho animado feito quadro a quadro num caderno.

Exemplo. Para estimar o volume de dados de vídeo que o hardware precisa processar por segundo, multiplica-se a quantidade de pixels de um quadro pela taxa de quadros por segundo:

- 720p a 30 FPS: 1.280 × 720 × 30 ≈ 27,6 milhões de pixels processados por segundo.
- Full HD (1080p) a 60 FPS: 1.920 × 1.080 × 60 ≈ 124,4 milhões de pixels processados por segundo.

A diferença entre os dois cenários é de aproximadamente 4,5 vezes — um salto de resolução de 720p para Full HD, combinado com o dobro da taxa de quadros, quase quintuplica a demanda computacional. Esse cálculo explica por que jogos e aplicações gráficas listam requisitos mínimos e recomendados atrelados tanto a uma resolução quanto a uma taxa de FPS específicas.

Vale registrar que nem toda queda de desempenho percebida em jogos multijogador tem origem no processamento local: quando o computador atua como cliente de um servidor remoto (típico de jogos *online*), atrasos de rede (Wi-Fi, latência do provedor, disponibilidade do servidor) também produzem sensação de travamento, independentemente da capacidade da CPU e da GPU locais.

Figura pendente

comparação lado a lado da mesma imagem renderizada em diferentes resoluções, evidenciando o tamanho dos pixels

3.9 Benchmark: metodologia de comparação e dimensionamento

Ao especificar um computador para uma finalidade concreta — por exemplo, atender aos requisitos mínimos de um jogo —, o técnico se depara com um problema: os requisitos publicados pelo fabricante do software costumam ser descritivos (sistema operacional, um modelo específico de CPU, quantidade de memória RAM, um modelo específico de GPU, espaço de armazenamento), e não numéricos. Memória RAM e armazenamento são diretamente comparáveis em gigabytes — qualquer módulo de 8 GB atende a um requisito de 8 GB. CPU e GPU não: não existe uma unidade simples que permita comparar diretamente um processador de um fabricante, arquitetura e geração com outro processador de fabricante, arquitetura e geração diferentes.

A solução adotada pela indústria é o **benchmark** — literalmente, "bancada de testes". Todo processador ou placa de vídeo submetido ao mesmo teste padronizado recebe uma nota (*score*) comparável.

Analogia. Um dinamômetro automotivo submete motores diferentes ao mesmo teste padronizado, produzindo uma medida comparável (cavalos de potência, torque). Da mesma forma, o Enem submete candidatos de escolas e contextos diferentes à mesma prova, nas mesmas condições de tempo e sem consulta, produzindo uma nota comparável entre eles. O benchmark de hardware cumpre exatamente esse papel para CPUs e GPUs.

As duas notas de CPU. Ferramentas de benchmark para processador (a mais usada em sala é o PassMark, cujo software CPU-Z é abordado na Seção 3.10) produzem duas notas distintas:

- **Single-thread rating** — desempenho de um único núcleo trabalhando sozinho.
- **Multi-thread rating** — desempenho de todos os núcleos e threads trabalhando simultaneamente.

Essa distinção é decisiva na prática: a maioria dos softwares de uso comum (planilhas, navegadores) e a maioria dos jogos não são otimizados

para dezenas de núcleos simultâneos — eles dependem, sobretudo, do desempenho *single-thread*. Cargas de trabalho de servidor, renderização e simulação, por outro lado, se beneficiam diretamente do desempenho *multi-thread*.

Exemplo. Uma comparação real conduzida em sala, entre um processador Intel (lançado em 2015) e um processador AMD Ryzen (lançado em 2017), ilustra o ponto: o processador Intel obteve nota *single-thread* de aproximadamente 2.315 pontos contra aproximadamente 2.000 pontos do AMD (uma diferença de cerca de 10% a favor da Intel); já na nota *multi-thread*, o AMD obteve cerca de 12.000 pontos contra 6.305 pontos da Intel — quase o dobro. *(Nota: dado autoral de demonstração em sala; os modelos exatos de CPU não são nomeados aqui — vale documentá-los, ou anexar a captura de tela do PassMark usada na aula, para que o exemplo seja reproduzível.)* A conclusão prática: para um computador cuja finalidade é jogar (dependente de desempenho *single-thread*), o processador Intel seria a escolha mais adequada, apesar de mais antigo; para um servidor cuja carga se distribui por múltiplas threads simultâneas, o AMD seria a escolha correta. O benchmark permite essa comparação mesmo entre processadores de fabricantes, arquiteturas, gerações, caches e potências diferentes, porque ambos foram submetidos exatamente à mesma prova.

Ganho geracional. Comparações de longo prazo dentro de uma mesma linha de produto evidenciam o efeito acumulado da miniaturização (Seção 3.3) e de melhorias de arquitetura. Um exemplo apresentado em aula, de uma mesma família de processador ao longo de gerações sucessivas: em 2017, um modelo de 65 W entregava cerca de 2.000 pontos *single-thread*; em 2018, a geração seguinte, operando a 105 W, chegou a cerca de 2.400 pontos; já em 2024, uma geração mais recente voltou a operar a 65 W (a mesma potência de 2017) e alcançou aproximadamente 4.500 pontos *single-thread* e cerca de 30.000 pontos *multi-thread* — mais do que o dobro do desempenho *single-thread* de sete anos antes, com potência igual à do ponto de partida. *(Nota: família de processador específica a documentar pelo autor, para que a série seja rastreável.)* Esse é o retrato numérico do que a Seção 3.3 descreve de forma qualitativa: litografias menores entregam, ao mesmo tempo, mais desempenho e mais eficiência energética.

Benchmark de GPU e custo-benefício. O mesmo tipo de ferramenta existe para placas de vídeo. Ao comparar duas GPUs reais de gerações próximas, por exemplo, uma custando cerca de R$ 2.890 com nota de aproximadamente 22.000 pontos, contra outra custando cerca de R$ 4.600 com nota de aproximadamente 28.000 pontos, *(GPUs específicas e data de consulta de preço a documentar pelo autor, já que preços em reais mudam rapidamente)* o técnico consegue avaliar objetivamente se o ganho de desempenho (cerca de 27%) justifica o acréscimo de custo (cerca de 60%) para aquele cliente específico — e ainda precisa verificar se a fonte de alimentação do computador suporta a potência adicional exigida pela placa mais cara, sob risco de o upgrade da GPU obrigar também um upgrade da fonte.

Uma forma particularmente útil de visualizar essas comparações é o gráfico de dispersão (*scatter plot*), com a nota de benchmark no eixo horizontal

e o preço no eixo vertical: quanto mais à direita e mais abaixo estiver um produto nesse gráfico, melhor o seu custo-benefício. O cálculo de custo-benefício em si é simples — nota de benchmark dividida pelo preço do componente.

Exemplo — RISC contra CISC, lado a lado. A comparação mais didática de eficiência energética entre arquiteturas usa um processador RISC de origem móvel (o Apple A18 Pro, originalmente projetado para iPhone e reaproveitado num MacBook) contra um processador CISC de laptop equivalente: o A18 Pro obteve cerca de 4.000 pontos *single-thread* e quase 12.000 pontos *multi-thread*, consumindo entre 4 W e 10 W [12]; o processador CISC equivalente consumiu cerca de dez vezes mais potência para entregar uma nota inferior *(modelo CISC específico a documentar pelo autor)*. É esse tipo de comparação, quantificada por benchmark, que explica por que a autonomia de bateria de dispositivos com chips RISC é tão superior — e por que a indústria de desktops e laptops caminha na direção descrita no quadro da Seção 3.2.

Figura pendente

captura de tela do PassMark comparando dois processadores lado a lado, com notas single-thread e multi-thread destacadas] [IMAGEM: gráfico de dispersão (scatter plot) de nota de benchmark por preço, com pontos coloridos por fabricante

3.10 Diagnóstico em campo: CPU-Z e HWMonitor

Duas ferramentas de software, de uso corrente entre técnicos, permitem levantar as características de um processador e monitorar seu comportamento sob carga sem a necessidade de desmontar o computador.

CPU-Z lê e apresenta, a partir do sistema operacional em execução, informações detalhadas de hardware: nome comercial e codinome do processador, litografia, potência máxima, família de instruções, faixa de clock em operação, tamanho das caches L1/L2/L3, contagem de núcleos e threads, fabricante e versão de BIOS da placa-mãe, configuração de canais de memória (single ou dual channel) e as GPUs disponíveis no sistema. Uma aba específica, chamada *bench*, aplica o teste de benchmark descrito na Seção 3.9 diretamente no computador em uso.

Analogia. Processadores híbridos modernos combinam núcleos de desempenho (*performance cores*, ou P-cores) e núcleos econômicos (*efficient cores*, ou E-cores) — uma arquitetura adotada pela Intel, entre outros, em resposta à eficiência energética típica de processadores RISC. A diferença entre os dois tipos de núcleo é comparável à diferença entre um carro 1.0 (mais econômico, menos potente) e um carro 2.0 (mais potente, mais consumo): o sistema operacional aloca tarefas leves (como notificações e aplicativos em segundo plano) aos núcleos econômicos, reservando os núcleos

de desempenho para tarefas que realmente demandam potência, como jogos.

HWMonitor é um painel de sensores em tempo real — o equivalente a um monitor cardíaco preso a um atleta correndo em uma esteira. Ele reporta, para cada componente monitorado, os valores atual, mínimo e máximo registrados de temperatura, potência (watts) e outras grandezas.

Exemplo (demonstração em sala). Em um notebook equipado com processador Intel i7 de 13ª geração e 32 GB de RAM, em repouso a temperatura do pacote de processamento ficava próxima de 50–59 °C. Ao aplicar um teste de estresse (o mesmo tipo de ferramenta de benchmark da Seção 3.9, usada aqui para forçar a carga máxima), a potência consumida saltou para a faixa de 45–51 W e a temperatura chegou a 96–97 °C — no limite de segurança do fabricante. Nesse ponto, o *thermal throttling* (Seção 3.4) reduziu a potência entregue para cerca de 19 W, e o ciclo de aceleração e desaceleração térmica ficou visível em tempo real no HWMonitor, com a potência oscilando repetidamente entre esses dois extremos. Ao conectar o notebook a um carregador de 100 W, o sistema operacional liberou automaticamente o "modo de alto desempenho" — deixando de gerenciar a bateria — e o processador passou a sustentar potências mais altas por mais tempo antes de sofrer *throttling* novamente. Esse comportamento — desempenho diferente conforme o notebook está ou não conectado à tomada — é característico de notebooks Windows; processadores Apple Silicon, por comparação, mantêm o mesmo desempenho ligados ou desligados da tomada, justamente por serem RISC e demandarem uma fração da potência.

Esse teste de estresse cumpre dupla função de diagnóstico: verifica, simultaneamente, se a **fonte de alimentação** (ou a bateria, no caso de notebooks) é capaz de entregar a potência de pico exigida pelo hardware sob carga máxima — tema do Capítulo 2 —, e se a **refrigeração** é adequada para sustentar essa carga sem superaquecimento. É o procedimento indicado, por exemplo, para validar se a troca de pasta térmica resolveu um quadro de reinicializações intermitentes atribuídas a superaquecimento.

Figura pendente

captura de tela do CPU-Z mostrando núcleos, threads, caches e clock de um processador real] [IMAGEM: captura de tela do HWMonitor durante um teste de estresse, evidenciando o ciclo de thermal throttling

3.11 Gargalo e dimensionamento orientado à aplicação

O **gargalo** de um sistema computacional é o componente que, em um dado momento, limita o desempenho do conjunto — e ele não é fixo: ao melhorar um componente, o gargalo simplesmente se desloca para outro ponto do sistema (processador, GPU, armazenamento, memória RAM ou rede).

Exemplo. Um usuário que investiu uma quantia elevada em um processador de altíssimo desempenho, mas obteve resultado pior do que o esperado em um jogo específico, ilustra bem o problema: o jogo em questão não estava otimizado para tirar proveito de dezenas de núcleos de processamento, de modo que apenas uma pequena fração dos núcleos disponíveis era efetivamente utilizada, e o restante do investimento em poder computacional ficou ocioso.

Exemplo. Em um cenário de download de arquivos em uma rede universitária extremamente rápida, o gargalo real não estava na velocidade da internet, nem na memória RAM, nem no processador, mas na velocidade de escrita do disco rígido (HD) mecânico, incapaz de gravar os dados recebidos na mesma velocidade em que chegavam pela rede. Substituir esse HD por um SSD, de escrita muito mais rápida, não elimina o conceito de gargalo — apenas desloca o ponto de estrangulamento para outro componente do sistema, tipicamente a rede ou o servidor remoto.

Método de dimensionamento. A metodologia apresentada ao longo deste capítulo pode ser resumida em cinco passos:

1. Definir a **aplicação** do computador (jogo, estação de trabalho, servidor, uso de escritório etc.), pois é ela que determina quais notas de benchmark (single-thread, multi-thread, GPU) são relevantes.
2. Levantar os requisitos mínimos e recomendados publicados pelo fabricante do software-alvo.
3. Converter esses requisitos descritivos em notas de benchmark comparáveis, usando ferramentas como o PassMark.
4. Pesquisar, dentro do orçamento disponível, componentes cuja nota de benchmark atenda ou supere a meta, comparando custo-benefício entre alternativas de fabricantes e gerações diferentes.
5. Verificar se a escolha de um componente mais potente não desloca o gargalo, ou a exigência de potência, para outro ponto do sistema — tipicamente a fonte de alimentação.

Exemplo — orçamento de potência de um servidor. Considere um servidor local com processador de 150 W de consumo máximo, duas GPUs idênticas de 250 W cada, oito discos de armazenamento de 15 W cada, e um consumo combinado de placa-mãe, memória RAM e ventoinhas de 80 W:

Componente	Consumo máximo
Processador	150 W
GPUs (2 × 250 W)	500 W
Discos (8 × 15 W)	120 W
Placa-mãe, RAM e ventoinhas	80 W
Total de pico	**850 W**

Esse valor de 850 W representa o **pico teórico**, isto é, a soma dos consumos máximos individuais — um cenário raro na prática, já que dificilmente

todos os componentes operam simultaneamente no limite. Em uso típico, um servidor como esse opera perto de 40% da carga de pico (cerca de 340 W neste exemplo). Retomando o Capítulo 2: a eficiência elétrica de uma fonte ATX é máxima justamente perto de 50% de sua carga nominal. Escolher uma fonte de exatos 850 W deixaria a operação típica numa faixa de baixa eficiência (cerca de 40% de 850 W) e sem margem alguma para upgrades futuros. A escolha tecnicamente correta é uma fonte de capacidade nominal maior — por exemplo, entre 1.000 W e 1.500 W —, de modo que a operação típica do servidor caia próxima da faixa de melhor eficiência da fonte, com folga de segurança para os momentos de pico.

Esse raciocínio fecha o ciclo entre o dimensionamento do processador (e dos demais componentes de processamento) e o dimensionamento da fonte de alimentação: nenhuma escolha de hardware de processamento pode ser tomada de forma isolada da capacidade de entrega de energia do sistema.

Figura pendente

tabela de orçamento de potência de um servidor, com a soma dos componentes e a faixa de operação típica marcada sobre a curva de eficiência de uma fonte ATX

3.12 Manutenção preventiva e corretiva em notebooks

As seções anteriores deste capítulo trataram o processador de forma geral, aplicável tanto a desktops quanto a notebooks. Esta seção fecha o capítulo tratando especificamente do que muda quando o computador em questão é portátil — um recorte que soma, mas não substitui, tudo o que já foi visto sobre refrigeração (livro de Organização e Montagem, Capítulo 4), fonte de alimentação (Capítulo 2) e thermal throttling (Seção 3.4).

3.12.1 Bateria: o componente que só o notebook tem

A bateria de um notebook é uma célula de íon de lítio (ou polímero de lítio), sujeita a um processo de degradação química irreversível — diferente de qualquer componente puramente eletrônico já estudado neste livro, que falha por defeito, não por desgaste químico natural. Duas práticas de manutenção preventiva reduzem essa degradação:

- **Evitar ciclos completos de descarga.** Descarregar a bateria até 0% com frequência acelera sua degradação química, de forma análoga (guardadas as proporções) ao desgaste por ciclo de escrita da memória flash (livro de Organização e Montagem, Capítulo 2, §2.12): a bateria de íon de lítio se degrada menos quando mantida, na maior parte do

tempo, numa faixa intermediária de carga (por exemplo, entre 20% e 80%) do que quando ciclada repetidamente entre 0% e 100%.
- **Evitar calor prolongado.** A degradação química da bateria acelera com a temperatura — manter o notebook conectado à energia e sob carga de processamento pesada por longos períodos, numa superfície que restrinja a ventilação (uma cama, um sofá), combina justamente os dois fatores que mais aceleram esse desgaste: calor e carga elétrica sustentada.

Diagnóstico corretivo. Uma bateria degradada não perde a capacidade de operação instantaneamente — ela perde **capacidade total** (a autonomia cai progressivamente) e pode passar a apresentar inchaço físico, visível como uma leve deformação do chassi ou do touchpad. Um notebook com a bateria fisicamente inchada não deve continuar em uso: o inchaço indica acúmulo de gases internos por decomposição química da célula, um risco real de incêndio caso a célula seja perfurada ou continue sendo carregada.

3.12.2 Refrigeração: o desafio do espaço reduzido

A Seção 4.7 do livro de Organização e Montagem apresentou dissipador e ventoinha como os dois componentes do resfriamento a ar. Num notebook, o mesmo princípio se aplica, mas dentro de um volume drasticamente menor — o que tem duas consequências práticas diretas:

- **Acúmulo de poeira em menos espaço, com efeito proporcionalmente maior.** As aletas do dissipador de um notebook são mais finas e mais próximas entre si do que as de um cooler de desktop, para caber no chassi fino. Uma mesma quantidade de poeira acumulada obstrui, proporcionalmente, uma fração muito maior da área de passagem de ar — o que explica por que notebooks tendem a apresentar thermal throttling (Seção 3.4) por acúmulo de poeira num intervalo de tempo menor do que um desktop equivalente. A limpeza preventiva do dissipador e da ventoinha (normalmente com ar comprimido, sem desmontar o notebook por completo) é, por essa razão, mais frequente em notebooks do que em desktops.
- **Repasse de pasta térmica mais delicado.** O procedimento de troca de pasta térmica (livro de Organização e Montagem, Capítulo 4, §4.8) segue o mesmo princípio físico num notebook, mas a desmontagem para acessar o processador é, em geral, mais invasiva: exige remover a placa inteira do chassi em muitos modelos, em vez de apenas liberar um dissipador preso por presilhas externas como num desktop. Por isso, esse é um procedimento tipicamente reservado a um técnico com experiência prévia em desmontagem daquele modelo específico — um manual de serviço (*service manual*) do fabricante, quando disponível, é a referência mais confiável para essa etapa.

3.12.3 Componentes soldados: o que muda no diagnóstico por substituição

O Capítulo 1 do livro de Organização e Montagem (§1.11.2) já apresentou o compromisso entre modularidade e portabilidade: memória RAM e, cada vez mais, o próprio processador vêm soldados à placa num notebook. Isso tem uma consequência direta sobre o método de diagnóstico por substituição de módulo (livro de Organização e Montagem, Capítulo 1, §1.6.2): quando o componente suspeito está soldado, não é possível simplesmente trocá-lo por um equivalente para testar a hipótese. As alternativas de diagnóstico corretivo, nesse cenário, são:

- Testar o mesmo sintoma num live USB (Capítulo 3 do livro de Organização e Montagem, §3.9) para isolar hardware de software, já que essa técnica não depende de trocar peça alguma.
- Testar o componente suspeito em outro notebook do mesmo modelo, quando disponível (por exemplo, numa bancada com máquinas de reposição), como forma indireta de aplicar o mesmo princípio de substituição sem precisar desoldar nada.
- Quando a suspeita se confirma e o componente é de fato o defeituoso, a correção deixa de ser uma troca de módulo e passa a ser retrabalho de solda em nível de placa — fora do escopo de manutenção de bancada convencional na maioria dos casos, e que costuma justificar, financeiramente, a comparação entre o custo do reparo e o custo de um equipamento novo (a mesma lógica de obsolescência econômica já discutida a propósito de soquetes de CPU, Seção 3.7).

3.12.4 Reparos mais comuns: tela e teclado

Diferente do processador e da memória, tela e teclado são, na prática de bancada, os componentes de notebook mais frequentemente substituídos — não por serem tecnologicamente mais frágeis, mas por estarem fisicamente mais expostos a dano por impacto, derramamento de líquido e desgaste de uso repetitivo. Em praticamente todo modelo de notebook, ambos são módulos destacáveis (presos por parafusos e conectores do tipo *ribbon cable*, uma fita plana de contatos), o que os torna reparáveis por substituição direta mesmo em modelos que soldam processador e RAM — uma peça de reposição compatível, comprada do fabricante ou de um fornecedor terceirizado, restaura o equipamento sem exigir retrabalho de solda.

Figura pendente

notebook com a tampa inferior removida, evidenciando bateria, dissipador miniaturizado e conectores tipo ribbon cable da tela e do teclado

3.13 Síntese do capítulo

Este capítulo apresentou o processador como peça central da tarefa de especificação de um computador: a distinção entre arquiteturas RISC e CISC, os limites físicos impostos pela litografia e pelo calor, a organização interna de um processador multicore (núcleos, threads, cache e pipeline), a relação entre resolução de vídeo e demanda computacional, e a metodologia de benchmark que transforma requisitos descritivos de software em notas numéricas comparáveis entre fabricantes, arquiteturas e gerações diferentes. O exemplo de orçamento de potência da Seção 3.11 retoma diretamente o dimensionamento de fonte de alimentação estudado no Capítulo 2, evidenciando que a escolha de processador, GPU e armazenamento nunca é independente da capacidade de entrega de energia do sistema. O capítulo fechou aplicando esses mesmos princípios — refrigeração, energia, diagnóstico por substituição — ao caso específico do notebook, cujo espaço reduzido e componentes soldados exigem adaptações do método geral. Os conceitos de diagnóstico por eliminação de módulos, teste de estresse e identificação de gargalo, exercitados aqui com CPU-Z e HWMonitor, serão retomados e sistematizados no Capítulo 4, dedicado ao atendimento e suporte técnico.

3.14 Referências

1. PHORONIX. “The Linux Kernel May Finally Phase Out Intel i486 CPU Support.” Disponível em: https://www.phoronix.com/news/Intel-i486-Linux THE REGISTER. Cobertura relacionada. Disponível em: https://forums.theregister.com/forum/all/2025/05/07/linux_kernel_drops_486/.
2. NVIDIA NEWSROOM/GEFORCE NEWS. “NVIDIA at COMPUTEX 2026: NVIDIA RTX Spark…” Disponível em: https://www.nvidia.com/en-us/geforce/news/computex-2026-nvidia-geforce-rtx-announcements/; TOM’S HARDWARE. “Nvidia unveils RTX Spark Superchip for laptops and desktop PCs at Computex 2026.” Disponível em: https://www.tomshardware.com/laptops/nvidia-unveils-rtx-spark-superchip-at-comp
3. Dados de litografia por fabricante/arquitetura conferidos via TechInsights/TrendForce/roteiro de nós de processo TSMC; cobertura em notebookcheck.net e techpowerup.com.
4. NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY (NIST). “A sheet of paper is about 100,000 nanometers thick.” Disponível em: https://x.com/NIST/status/1792590000179064919; NATIONAL NANOTECHNOLOGY INITIATIVE. “Just How Small Is Nano?” Disponível em: https://www.nano.gov/about-nanotechnology/just-how-small-is-nano
5. TOM’S HARDWARE. “Hot Spot: How Modern Processors Cope With Heat Emergencies.” Disponível em: https://www.tomshardware.com/reviews/hot-spot,365-4.html.

6. CHANNEL INSIDER. “Intel Cancels 4-GHz Pentium 4.” Disponível em: https://www.channelinsider.com/news-and-trends/intel-cancels-4-ghz- PC PERSPECTIVE. “Intel Cancels the 4GHz Prescott.” Disponível em: https://pcper.com/2004/10/intel-cancels-the-4ghz-prescott/.
7. INTEL NEWSROOM. “Dual Core Era Begins, PC Makers Start Selling Intel-Based PCs.” 18 abr. 2005. Disponível em: https://www.intel.com/pressroom/archive/releases/2005/20050418comp.htm.
8. Documentação técnica geral sobre segmentação de produto e *binning* de CPUs (Intel/AMD); artigos técnicos sobre *yield* de semicondutores — fonte específica a confirmar pelo autor.
9. INTEL. Especificações oficiais, “Intel® Core™ i5-12400F Processor (18M Cache, up to 4.40 GHz).” Disponível em: https://www.intel.com/content/www/us/en/products/sku/134587/intel-core-i512400f-proces specifications.html.
10. PATTERSON, David A.; HENNESSY, John L. *Computer Organization and Design: The Hardware/Software Interface* — RISC-V Edition. Cambridge, MA: Morgan Kaufmann, 2017. ISBN 978-0-12-812275-4.
11. TECHPOWERUP. “AMD Announces Socket AM5 Longevity till 2029.” Disponível em: https://www.techpowerup.com/349541/amd-announces-socket VIDEOCARDZ. Disponível em: https://videocardz.com/newz/amd-extends-am
12. PASSMARK/CPU BENCHMARK. “Apple A18 Pro (MacBook Neo) Benchmark.” Disponível em: https://www.cpubenchmark.net/cpu.php?cpu=Apple+A18+Pro+%28MacBook+Neo%29&id=7232.

4 Suporte e Gestão de Serviços

Neste capítulo você vai estudar a organização do atendimento técnico ao usuário: a central de serviços (*service desk*) como ponto único de entrada das demandas, a categorização de chamados em evento, incidente e solicitação, os critérios de priorização por impacto e urgência, os níveis de suporte e o custo crescente de cada escalonamento, e as ferramentas e princípios éticos que regem o acesso remoto a máquinas de terceiros.

4.1 Central de serviços (service desk)

Quando um problema técnico surge — por exemplo, a internet de um laboratório para de funcionar —, o primeiro passo do diagnóstico é distinguir se a causa é **local** (a máquina específica) ou de **infraestrutura** (algo fora do alcance daquele posto de trabalho, que depende do setor de Tecnologia da Informação). Uma vez identificado que o problema é de infraestrutura, a demanda deve ser encaminhada a um ponto único de contato.

Esse ponto único é a **central de serviços**, também chamada de *service desk*. Trata-se de um hub para onde convergem todas as demandas de um setor ou organização, e a partir do qual elas recebem tratamento e encaminhamento adequados. Em vez de o usuário procurar diretamente um técnico específico, toda solicitação passa por esse canal centralizado, que a registra, categoriza e distribui.

Exemplo. Um caso amplamente divulgado na imprensa e em redes sociais em Natal/RN ilustra o que ocorre quando uma central de serviços é sobrecarregada ou mal gerida: em 2026, clientes da operadora **Alares** relataram instabilidade prolongada de internet, TV e telefone em bairros como Ponta Negra, Pitimbu e Candelária, sem conseguir contato pelos canais oficiais nem para suporte técnico nem para cancelamento de serviço — a ponto de precisarem procurar a sede da empresa pessoalmente —, o que motivou fiscalização do Procon-RN na sede da empresa [2]. O episódio mostra que a central de serviços não é um detalhe administrativo: é a peça que determina se a organização consegue, de fato, responder às demandas de seus usuários. *(Nota: caso identificado por correspondência forte com a descrição do capítulo — confirmar com o autor se é este o episódio pretendido.)*

No âmbito do IFRN, a central de serviços é operada pelo sistema **SUAP** (Sistema Unificado de Administração Pública) [1], tratado em detalhe na Seção 4.6.

Figura pendente

fluxograma — usuário identifica problema → local ou infraestrutura? → abertura de chamado na central de serviços → triagem

4.2 Categorização das demandas: evento, incidente e solicitação

Nem toda demanda que chega a uma central de serviços tem a mesma natureza. A prática de gestão de serviços de TI — estudada em profundidade em cursos de Administração e Engenharia de Produção, sob frameworks como o ITIL (*Information Technology Infrastructure Library*, atualmente em transição da versão 4 para a versão 5, lançada em fevereiro de 2026) [3] — classifica as demandas em três categorias.

4.2.1 Evento

Seguindo o vocabulário de gerenciamento de eventos do ITIL [4], um **evento** é uma ocorrência detectável que **não interrompe** o trabalho do usuário, mas gera um sinal de atenção. Está fortemente associado à manutenção preventiva: o evento é o alerta que permite agir antes que um problema real se manifeste.

Exemplo. O uso de CPU de um servidor atinge 80% de ocupação; o nível de tinta ou toner de uma impressora cai abaixo de determinado percentual e o equipamento emite uma notificação; uma rotina de backup semanal projeta que o disco de armazenamento ficará sem espaço disponível dali a três semanas. Em nenhum desses casos o usuário deixou de trabalhar — mas em todos eles existe um sinal que, se ignorado, tende a se converter em um problema real.

4.2.2 Incidente

De acordo com o glossário oficial do ITIL [5], um **incidente** é uma interrupção não planejada ou uma redução da qualidade de um serviço de TI. Ao contrário do evento, o incidente **interrompe** o fluxo de trabalho do usuário e está associado à manutenção corretiva: exige uma ação de resposta, geralmente com maior urgência.

Exemplo. O computador não liga; a impressora trava; o usuário perde a conexão com a internet. Em todos esses casos, alguém que precisa de um recurso computacional está impedido de usá-lo.

4.2.3 Solicitação

No mesmo vocabulário do ITIL [6], uma **solicitação de serviço** é um pedido previsível, que não decorre de uma falha. Pode ser periódica ou não,

mas segue o fluxo padrão de atendimento, sem a urgência associada a um incidente.

Exemplo. Recuperar uma senha esquecida; solicitar a instalação de uma impressora nova; pedir a expansão do sinal de Wi-Fi em um setor; solicitar a troca da conexão de um computador de rede sem fio para rede cabeada; os procedimentos de matrícula e criação de e-mail institucional de um aluno recém-ingresso.

4.2.4 Quando uma solicitação vira incidente

A fronteira entre as três categorias depende do critério de **interrupção do fluxo do usuário**, e não apenas da natureza superficial do pedido. Um exemplo recorrente: solicitar a substituição de um mouse com defeito é, em princípio, uma solicitação de serviço. Mas se aquele mouse é a ferramenta de trabalho de um funcionário que, sem ele, não consegue produzir, a demanda deixa de ser uma solicitação comum e passa a ser tratada como **incidente** — porque há um custo real (o salário daquela pessoa, parada) sendo desperdiçado a cada minuto sem solução.

A tabela a seguir resume os três critérios de categorização:

Categoria	Interrompe o trabalho do usuário?	Tipo de manutenção associada	Exemplos
Evento	Não	Preventiva	CPU do servidor em 80%; toner baixo; projeção de disco cheio
Incidente	Sim	Corretiva	Computador não liga; impressora travada; sem internet
Solicitação	Não (em geral)	Preventiva ou corretiva, conforme o caso	Redefinição de senha; instalação de software; expansão de Wi-Fi

4.3 Priorização: impacto, urgência e prioridade

Como as demandas de uma central de serviços não chegam de forma organizada nem podem ser todas atendidas simultaneamente, é preciso um critério para decidir a ordem de atendimento. Esse critério não é a ordem

de chegada — nem todo incidente é atendido na sequência em que foi registrado. O que determina a ordem de atendimento é a **prioridade**, calculada a partir de dois fatores:

- **Impacto**: quantas pessoas são afetadas pelo problema e quão crítico é o serviço afetado.
- **Urgência**: quão rapidamente a situação precisa ser resolvida antes de gerar prejuízo maior.

A combinação desses dois fatores define a prioridade de atendimento, seguindo o modelo da matriz de prioridade do ITIL (impacto × urgência) [7]:

Impacto \ Urgência	Urgência baixa	Urgência alta
Impacto baixo	Prioridade baixa — ex.: substituir teclados antigos, sujos, mas ainda funcionais	Prioridade moderada
Impacto alto	Prioridade alta — ex.: trocar cadeiras que causam problema de postura para toda uma equipe	Prioridade crítica — ex.: internet caiu; monitor quebrou; servidor da secretaria parou no dia da matrícula

Exemplo. Um servidor da secretaria escolar cair justamente no dia de matrícula tem impacto alto (afeta um processo crítico e um grande volume de pessoas) e urgência alta (o processo tem prazo), resultando em prioridade crítica: todo o trabalho em andamento é interrompido para atender essa demanda primeiro. Já a ausência do papel de parede padrão em um computador é um problema de impacto e urgência baixos, que pode aguardar.

Quando não existe um sistema formal de chamados — como ocorre em equipes pequenas de suporte —, a definição de prioridade cabe ao critério do supervisor responsável, que aplica a política da organização mesmo sem registrá-la formalmente em um sistema. Um técnico que ingressa em uma empresa recebe, na fase de treinamento, a orientação sobre qual política de priorização deve seguir.

Figura pendente

matriz impacto x urgência com quadrantes de prioridade destacados

4.4 Níveis de suporte e o custo do atendimento

Uma vez aberto, o chamado (também chamado de **ticket** — o nome vem do francês antigo *estiquet* (“nota anexada”), a mesma raiz de “etiqueta” em português [8]) segue por níveis de suporte sucessivos, até ser resolvido. A cada nível não resolvido, o problema é **escalado** para o nível seguinte.

Nível	Descrição	Modo típico de atendimento
Nível 0	Autoatendimento: o próprio usuário resolve o problema com apoio de um guia ou tutorial fornecido pelo sistema, sem necessidade de abrir chamado	Tutoriais, guias de configuração, e cada vez mais chatbots baseados em IA generativa como camada intermediária
Nível 1	Triagem e suporte básico	Telefone, WhatsApp, acesso remoto
Nível 2	Suporte técnico local	Visita física ao local, diagnóstico dedicado
Nível 3	Especialistas ou fabricantes do equipamento	Escalonamento para equipes especializadas externas

Modelo de níveis de suporte (0 a 3) [9]; a nomenclatura e os limites exatos entre níveis variam conforme a fonte consultada.

A escalada de um nível para outro não é gratuita: cada nível tem um custo mais alto do que o anterior. O custo de um técnico que se desloca fisicamente é maior do que o de um atendente de central telefônica; a esse custo de mão de obra somam-se custos de equipamento, transporte e logística. Por isso, o técnico de informática deve buscar, sempre que possível, resolver o problema no nível mais baixo em que a solução for viável — o que reforça a importância do acesso remoto como ferramenta de contenção de custo (Seção 4.5).

Exemplo real de escalonamento. A operadora de internet que enfrentou uma queda prolongada e generalizada em sua qualidade de serviço (o mesmo caso citado na Seção 4.1) precisou recorrer ao Nível 3 — especialistas e fabricantes de equipamento de infraestrutura [2] — porque o problema não se resolveu em um dia, nem em uma semana, evidenciando uma falha estrutural que os níveis anteriores de suporte não tinham capacidade de resolver sozinhos.

Figura pendente

diagrama de escalonamento nível 0 → nível 1 → nível 2 → nível 3, com custo crescente indicado

4.5 Acesso remoto: ferramentas e uso ético

Grande parte dos problemas resolvidos nos níveis 1 e 2 de suporte envolve apenas configuração de software, e não troca física de peças. Nesses casos, o **acesso remoto** permite ao técnico operar o computador do usuário a distância, evitando o custo de deslocamento. Entre as ferramentas mais usadas estão o SSH (acesso remoto via terminal, tipicamente para servidores e equipamentos de rede), o TeamViewer, o AnyDesk e o RustDesk, além da própria funcionalidade de Área de Trabalho Remota do Windows.

O acesso físico à máquina continua sendo necessário apenas quando o problema exige a substituição de um componente de hardware — tudo o que envolve exclusivamente configuração de software pode, em princípio, ser resolvido remotamente.

Exemplo. Um problema doméstico comum — um arquivo PDF que passou a abrir automaticamente no Word em vez do leitor de PDF, por associação de tipo de arquivo alterada — foi resolvido a distância por meio do AnyDesk: o técnico recebeu um código de acesso gerado no computador remoto e, a partir dele, assumiu o controle de teclado e mouse da máquina até corrigir a associação de arquivo.

Por envolver o controle total do computador de outra pessoa — muitas vezes com acesso a e-mails, mensagens, fotos e documentos pessoais —, o acesso remoto exige a observância de princípios éticos claros:

- **Consentimento explícito.** O acesso remoto só deve começar depois que o usuário autoriza expressamente o início da sessão.
- **Transparência durante a operação.** O técnico deve narrar o que está fazendo a cada passo (por exemplo, “vou abrir a configuração de rede”), para que o usuário se sinta seguro de que nada além do necessário está sendo acessado.
- **Não acessar dados pessoais.** Arquivos e mensagens pessoais não devem ser abertos, exceto quando estritamente necessários à resolução do problema relatado.
- **Encerramento comunicado e registrado.** Ao final, o técnico deve informar explicitamente que a sessão está sendo encerrada, e registrar o que foi feito, por quem, quando e onde. Esse registro protege tanto o usuário quanto o técnico: sem ele, um uso indevido do computador feito pelo próprio usuário depois do atendimento poderia ser injustamente atribuído ao técnico que teve acesso remoto anteriormente.

Esses princípios se relacionam diretamente com a **Lei Geral de Proteção de Dados (LGPD, Lei nº 13.709/2018)** [10]: dados pessoais de terceiros

não podem ser tratados sem o devido consentimento, o que se aplica tanto ao conteúdo acessado durante uma sessão remota quanto aos registros gerados por ela.

Figura pendente

tela de autenticação de uma ferramenta de acesso remoto, com destaque para o código de sessão e o aviso de consentimento

4.6 Sistemas de chamados na prática

O registro e o acompanhamento de chamados são feitos por sistemas de *ticketing*. O **GLPI** [11], por exemplo, é um software livre amplamente usado para essa finalidade, que organiza os chamados pendentes e os já designados a um responsável, funcionando como um painel supervisório do trabalho da equipe de suporte.

4.6.1 O SUAP como central de serviços do IFRN

No IFRN, a abertura de chamados é feita pelo módulo de serviços do SUAP. O usuário escolhe primeiro a **área** do serviço (Administração, Comunicação, EAD, Ensino, Gestão de Pessoas, Manutenção Predial, Pesquisa ou Tecnologia da Informação) [12] e, dentro dela, uma **subcategoria** cada vez mais específica — por exemplo, dentro de Tecnologia da Informação: Redes e Internet, Data Center, Equipamentos, Segurança da Informação, Sistemas e Aplicativos, entre outras.

Exemplo de simulação de abertura de chamado. Para um problema de sinal de Wi-Fi instável em um laboratório, o caminho na árvore de categorias é: Tecnologia da Informação → Redes e Internet → Rede sem fio, onde o sistema oferece três opções específicas: informar problema de acesso à rede sem fio, solicitar acesso à rede corporativa, ou solicitar expansão de rede sem fio. Ao abrir o chamado, o usuário preenche uma descrição do problema (por exemplo, “verificar viabilidade de expansão da rede sem fio nos laboratórios de manutenção”), pode adicionar outros interessados (como o coordenador do setor afetado, que passa a acompanhar as atualizações do chamado), anexar evidências (como um print do sinal fraco) e, dependendo da categoria, escolher entre atendimento pela equipe de TI local ou pela equipe de TI remota — alguns serviços, como VPN, só são resolvidos por esta última. Uma vez aberto, o chamado entra na fila de atendimento com um prazo de resposta esperado, tipicamente entre 48 e 120 horas conforme a categoria [13].

Um mesmo caso é útil para revisar a categorização apresentada na Seção 4.2: um sinal de Wi-Fi instável (mas não totalmente indisponível) não interrompe o uso, então não é um incidente; também não é um evento, porque não se trata de um sinal detectado por infraestrutura de monitoramento; é

uma **solicitação**, e por não interromper o fluxo de ninguém, recebe prioridade baixa.

Antes de permitir a abertura de um chamado, o próprio SUAP oferece, quando disponível, um guia de solução (por exemplo, um tutorial de configuração de rede) — uma tentativa de resolver o problema em Nível 0, sem necessidade de acionar a equipe de TI.

Figura pendente

captura de tela do SUAP — árvore de categorias de serviço até “Rede sem fio”] [IMAGEM: captura de tela do SUAP — formulário de abertura de chamado preenchido, com campos de descrição, interessados e anexo

4.6.2 O mesmo modelo no desenvolvimento de software

A lógica de chamados categorizados, priorizados e distribuídos entre responsáveis não é exclusiva da manutenção de hardware. No desenvolvimento de software, o equivalente ao chamado é a **issue**, registrada em plataformas de controle de versão (como o Git). Em equipes que seguem a metodologia Scrum, as tarefas são organizadas dentro de um intervalo de tempo fixo (*sprint*) — mas quem as distribui não é o *Scrum Master*: o time de desenvolvimento é **auto-organizado**, e nem mesmo o Scrum Master tem autoridade para dizer ao time como realizar seu trabalho [14]. O papel do Scrum Master é dar suporte ao time e servir de ponte de comunicação com quem está fora dele; quem mais se aproxima de “distribuir” prioridades é o *Product Owner*, ao ordenar o Backlog do Produto — e, mesmo assim, sem atribuir tarefas individualmente. Outra abordagem comum é o método **Kanban** [15], no qual cada tarefa avança por colunas visuais que tornam o fluxo de trabalho visível a toda a equipe simultaneamente — por exemplo, do *backlog* (tarefas pendentes) para “em resolução”, depois “teste”, “avaliação” e, por fim, “finalizado”. O Kanban, como método, define princípios (visualizar o fluxo, limitar trabalho em andamento) e não uma lista fixa de nomes de coluna — o número e os nomes das colunas variam de equipe para equipe.

Vale registrar uma ressalva prática sobre sistemas de chamados em geral: a exigência de registrar formalmente cada atendimento pode gerar uma burocracia que, paradoxalmente, reduz a produtividade — por exemplo, quando um técnico já resolveu informalmente um problema simples e precisa, ainda assim, abrir um chamado só para que a solução conste no sistema. Esse é um custo real da formalização do atendimento, que deve ser equilibrado com os benefícios de rastreabilidade e priorização que o sistema oferece.

4.7 Troubleshooting: problemas comuns e possíveis soluções

Encerrado o percurso teórico do livro, esta seção consolida, em formato de referência rápida, os problemas mais comumente relatados por um usuário e o roteiro de hipóteses (Seção 1.4) que um técnico deveria seguir para diagnosticá-los — sem repetir a explicação de cada mecanismo, já detalhada no capítulo correspondente.

4.7.1 Computador não liga

Hipótese, em ordem de custo de verificação	Onde aprofundar
Tomada, régua ou disjuntor sem energia	Capítulo 2, §2.3
Cabo de força ou fonte com defeito (teste do jumper PS_ON/COM)	Capítulo 2, §2.8.2
Botão do painel frontal ou sua fiação	Livro de Organização e Montagem, Capítulo 4, §4.5.2
Mau contato de memória RAM	Capítulo 2, §2.10.3

4.7.2 Liga, mas não dá POST (sem imagem, com ou sem bipes)

Hipótese	Onde aprofundar
Memória RAM ausente, com defeito ou mal encaixada	Livro de Organização e Montagem, Capítulo 4, §4.3.1 e §4.3.3
Fonte energiza, mas não sustenta carga real	Capítulo 2, §2.8.2 (nota de atenção)
Speaker de aviso desconectado ou invertido (falso negativo de “sem bipe”)	Capítulo 2, §2.10.3
CPU incompatível com a placa-mãe (soquete/geração)	Livro de Organização e Montagem, Capítulo 4, §4.6.1

4.7.3 Trava, reinicia sozinho ou perde desempenho sob carga

Hipótese	Onde aprofundar
Thermal throttling por refrigeração insuficiente ou pasta térmica ressecada	Seção 3.4; livro de Organização e Montagem, Capítulo 4, §4.8
Fonte incapaz de sustentar o pico de potência da configuração	Seção 3.11
Memória RAM configurada acima da frequência suportada pelo conjunto placa-mãe/processador	Livro de Organização e Montagem, Capítulo 2, §2.5
Acúmulo de poeira no dissipador (mais comum em notebooks)	Seção 3.12.2

4.7.4 Sem som, Wi-Fi, Bluetooth ou touchpad

Hipótese	Onde aprofundar
Driver ausente ou desatualizado (causa mais provável)	Livro de Organização e Montagem, Capítulo 3, §3.12
Isolar hardware vs. software testando em um Live USB	Livro de Organização e Montagem, Capítulo 3, §3.9
Dispositivo fisicamente desabilitado no firmware	Livro de Organização e Montagem, Capítulo 3, §3.6.2

4.7.5 Computador lento no uso geral

Hipótese	Onde aprofundar
Pouca memória RAM disponível, sistema recorrendo a memória virtual	Livro de Organização e Montagem, Capítulo 2, §2.9
Malware em execução em segundo plano	Seção 1.5
Armazenamento mecânico (HD) como gargalo do sistema	Livro de Organização e Montagem, Capítulo 2, §2.10; Seção 3.11
Fragmentação de disco (sistemas com HD)	Livro de Organização e Montagem, Capítulo 3, §3.2.2

4.7.6 Perda de dados ou suspeita de arquivo corrompido/apagado

Hipótese	Onde aprofundar
Arquivo apagado, mas cluster ainda não sobrescrito (recuperável)	Livro de Organização e Montagem, Capítulo 3, §3.3.1
Backup ausente ou desatualizado no momento da falha	Seção 1.3; livro de Organização e Montagem, Capítulo 3, §3.8.1
Mídia de armazenamento com falha física iminente	Livro de Organização e Montagem, Capítulo 2, §2.10 e §2.12

Como usar esta tabela. Cada linha é ordenada, sempre que possível, da hipótese mais barata de testar para a mais cara — o mesmo princípio de priorização apresentado na Seção 1.4. Esta seção não substitui o raciocínio de diagnóstico: é um ponto de partida para gerar hipóteses, não uma lista de respostas prontas a aplicar sem investigação.

4.8 Síntese do capítulo

Este capítulo apresentou a organização do suporte técnico: a central de serviços como ponto único de entrada das demandas, a categorização em evento, incidente e solicitação, a priorização por impacto e urgência, o escalonamento por níveis de suporte com custo crescente, o uso ético do acesso remoto e o funcionamento prático de um sistema de chamados institucional, o SUAP. Fechou com uma referência rápida de troubleshooting, remetendo os problemas mais comuns relatados por usuários de volta aos mecanismos explicados ao longo dos quatro capítulos.

Com isso se encerra a formação teórica do curso de Manutenção de Computadores. O percurso começou pela definição do computador e pelo princípio de modularidade que estrutura todo diagnóstico técnico (Capítulo 1); avançou para a base elétrica do funcionamento de um computador, com a fonte de alimentação como ponto de partida de qualquer diagnóstico de hardware (Capítulo 2); tratou do processador e dos critérios objetivos de avaliação de desempenho por meio de benchmarks (Capítulo 3); e fecha, neste capítulo, com a dimensão de atendimento da manutenção — como organizar, priorizar e conduzir eticamente o suporte ao usuário. Resta ao curso a etapa final, prática: a apresentação dos trabalhos de dimensionamento de computadores para perfis de uso específicos, na qual os conceitos estudados ao longo dos quatro capítulos são aplicados a um caso concreto de especificação de hardware.

4.9 Referências

1. IFRN. “Suap.” Portal IFRN — Tecnologia da Informação. Disponível em: `https://portal.ifrn.edu.br/institucional/tecnologia-da-informaca`

servicos/suap/.
2. VIACERTANATAL. “Instabilidade na Alares gera reclamações e leva clientes a buscar cancelamento em Natal.” 2026. Disponível em: https://www.viacertanatal.com.br/2026/07/instabilidade-na-alares-gera.html; BLOG DO BG. “Clientes da Alares relatam problemas nos serviços de internet há semanas e comemoram chegada da fiscalização do Procon à empresa.” 2026. Disponível em: https://www.blogdobg.com.br/video-clientes-da-alares-relatam-problemas-nos-servicos-de-i (Caso a confirmar pelo autor.)
3. PEOPLECERT. “ITIL (Version 5) explained.” 2026. Disponível em: https://www.peoplecert.org/news-and-announcements/itil-version-5-ex ADVISEDSKILLS. “ITIL (Version 5) in 2026: What is confirmed, what is available, and what comes next.” Disponível em: https://www.advisedskills.com/blog/it-service-management/itil-version-5-in-2026-what-is-confi
4. PORTAL GSTI. “Gerenciamento de Eventos x Gerenciamento de Incidentes da ITIL.” Disponível em: https://www.portalgsti.com.br/2013/01/gerenciamento-de-eventos-x-gerenciamento-de-incidentes-da-i.html.
5. AXELOS/PEOPLECERT. *ITIL 4 Glossary of Terms*. Fonte secundária em português: HDI BRASIL. “Incidentes ou Requisições: como classificar problemas de desempenho de serviços.” Disponível em: https://hdibrasil.com.br/conteudo/incidentes-ou-requisicoes-como-classifica
6. AXELOS/PEOPLECERT. *ITIL 4 Glossary of Terms*.
7. INVGATE. “ITIL® Priority Matrix: How to Build And Use It in Your Service Desk.” Disponível em: https://blog.invgate.com/itil-priority-matrix
8. ONLINE ETYMOLOGY DICTIONARY. “Ticket.” Disponível em: https://www.etymonline.com/word/ticket.
9. TOPDESK. “Understanding service desk tiers: A guide to Tier 0 through Tier 3.” Disponível em: https://www.topdesk.com/en/blog/service-desk-tiers-explained/; INVGATE. “IT Support Levels Explained From Tier 0 to Tier 4.” Disponível em: https://blog.invgate.com/the-5-levels-of-it-support.
10. BRASIL. Lei nº 13.709, de 14 de agosto de 2018. Lei Geral de Proteção de Dados Pessoais (LGPD). Disponível em: https://www2.camara.leg.br/legin/fed/lei/2018/lei-13709-14-agosto-2018-787077-publicacaoori.html.
11. GLPI PROJECT. Página oficial. Disponível em: https://glpi-project.org/.
12. Captura de tela do próprio SUAP (evidência primária); IFRN. “Como abrir chamado no Suap?” Disponível em: https://portal.ifrn.edu.br/institucional/tecnologia-da-informacao/tutoriais/suap/como-abrir (Estabilidade da lista de áreas/subcategorias a confirmar pelo autor na data de publicação.)
13. Documentação interna da DIGTI/IFRN sobre SLA por categoria de serviço — não localizada publicamente nesta revisão; faixa a confirmar pelo autor com documentação interna ou captura de tela antes de publicar como fato geral do módulo de TI do SUAP.

14. Base local: MARÇULA, Marcelo; BENINI FILHO, Pio Armando. *Informática: Conceitos e Aplicações*; SCRUM.ORG. *The Scrum Guide*. Disponível em: https://scrumguides.org/scrum-guide.html.
15. KANBAN UNIVERSITY / David J. Anderson, Método Kanban; TARGET TEAL. “Método Kanban: um guia (quase) completo.” Disponível em: https://targetteal.com/blog/metodo-kanban/.