

DCC192

2025/2



Desenvolvimento de Jogos Digitais

A16: Áudio I – Síntese

Prof. Lucas N. Ferreira

Plano de aula

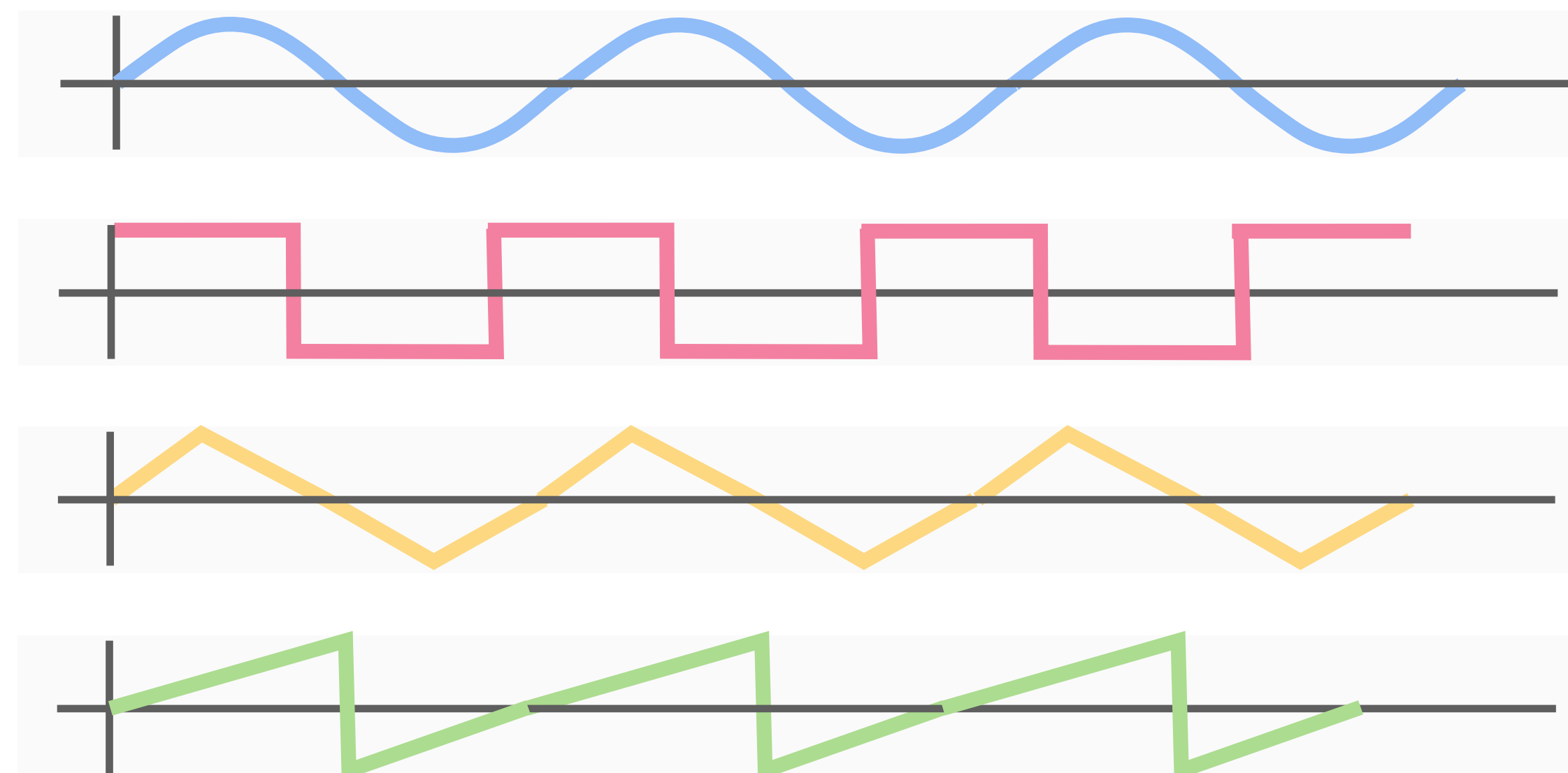


- ▶ Sinais de Áudio
 - ▶ Propriedades: amplitude, frequência e fase
 - ▶ Analógico vs. Digital
 - ▶ Sistemas de Reprodução
- ▶ Ondas Senoidais e Não-Senoidais
- ▶ Espectrogramas
- ▶ Soma de Sinais
- ▶ Sintetizando Músicas com Sequenciadores

Sintetizadores em Video Games



Uma forma bastante comum e histórica de se adicionar música em jogos é utilizando **sintetizadores** de áudio, os quais geram ondas sonoras a partir de modelos matemáticos:

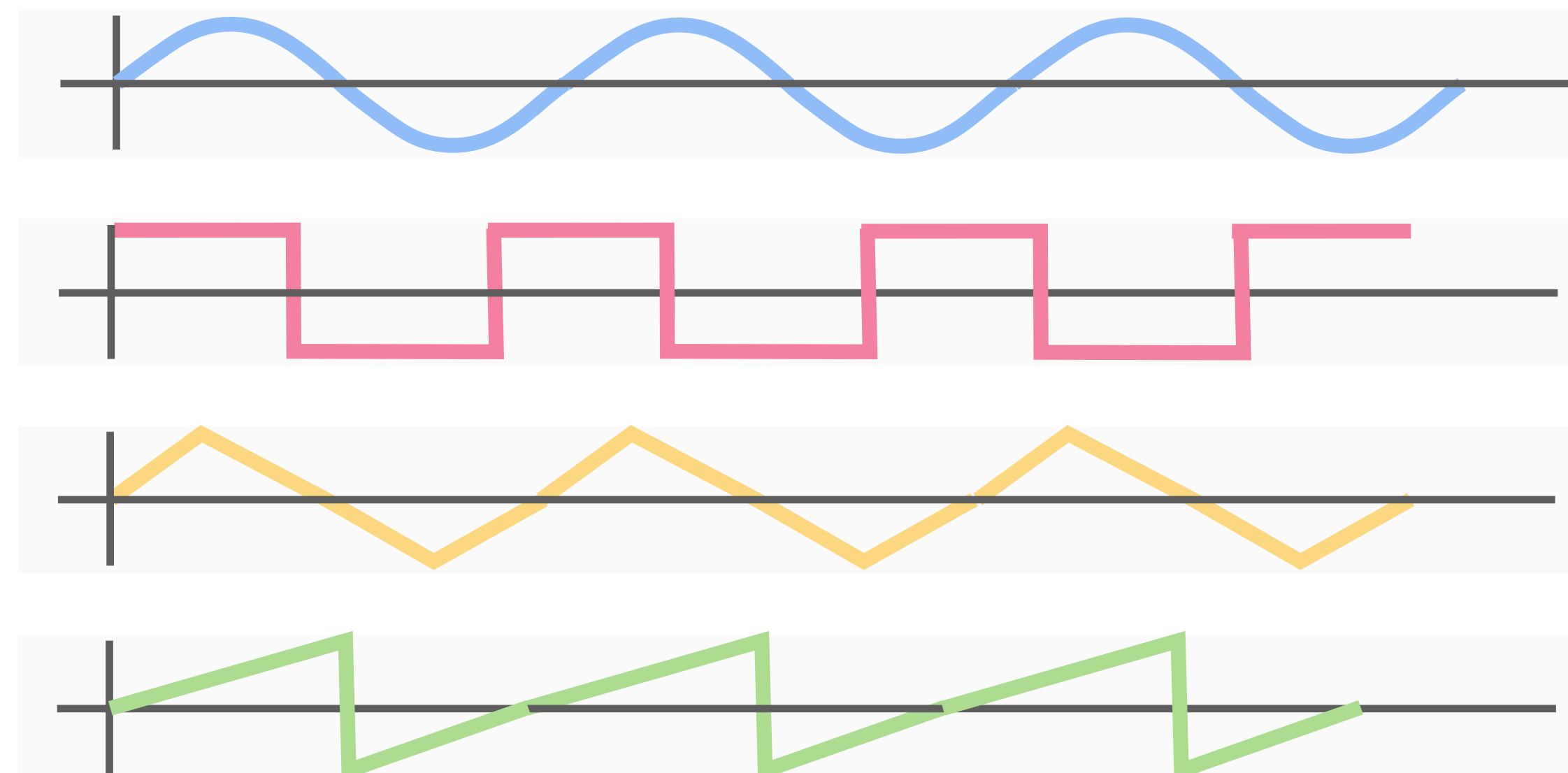
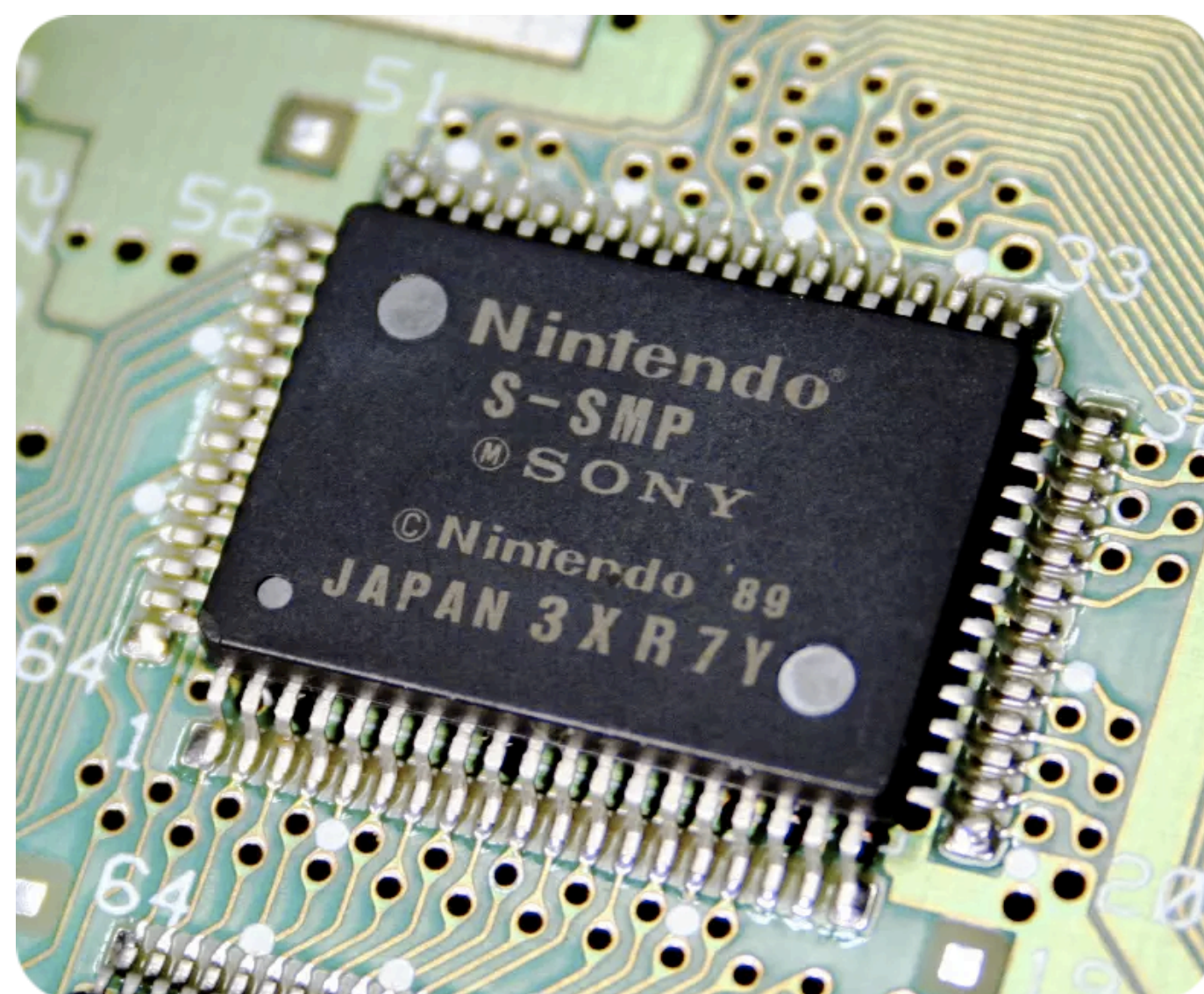


Sintetizadores são capazes de gerar diversos tipos, incluindo senoidal, quadrada, triangular e dente de serra.

Sintetizadores em Video Games



Uma forma bastante comum e histórica de se adicionar música em jogos é utilizando **sintetizadores** de áudio, os quais geram ondas sonoras a partir de modelos matemáticos:



Nos consoles de 8 e 16 bits (anos 80 e 90), os sintetizadores eram embarcados no hardware e eram programados (em assembly) para sintetizar ondas em tempo real.

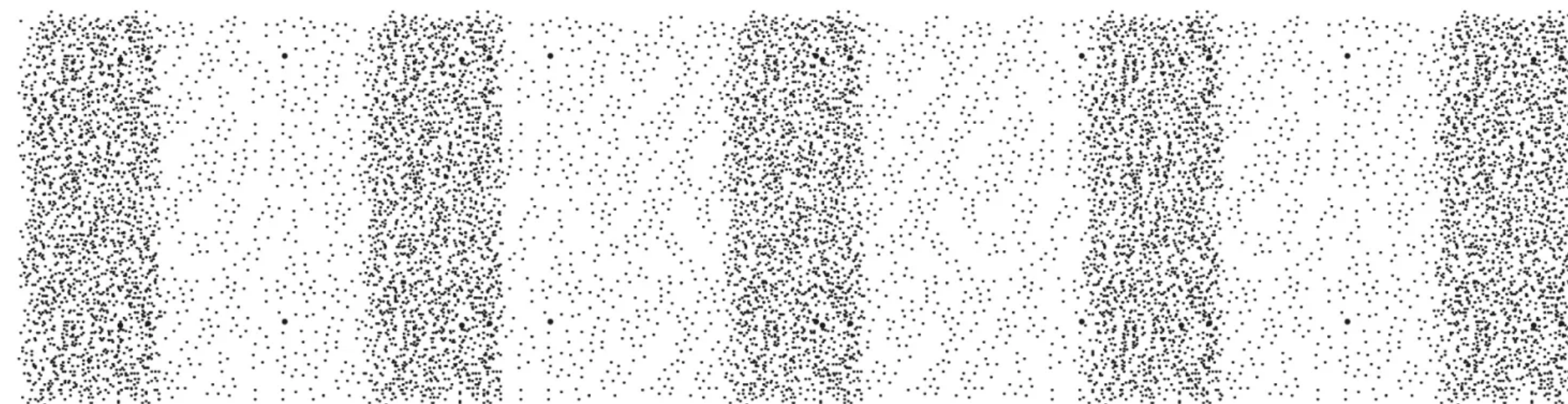
Sinal de Áudio



Um sinal de áudio é uma representação elétrica ou digital de ondas sonoras ao longo do tempo:

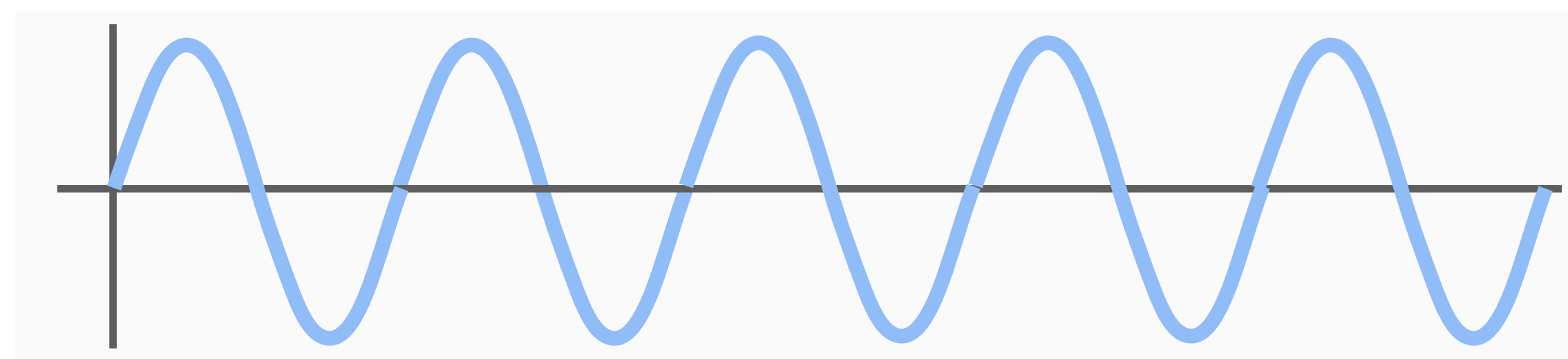
1. O ar ao nosso redor está cheio de moléculas.

2. Quando você reproduz um som em um alto-falante, ele vibra muito rápido.



3. Essa vibração se move através das moléculas do ar, como uma reação em cadeia, até chegar ao seu ouvido e ser processada pelo cérebro como som.

O sinal de áudio é definido por 3 parâmetros principais:



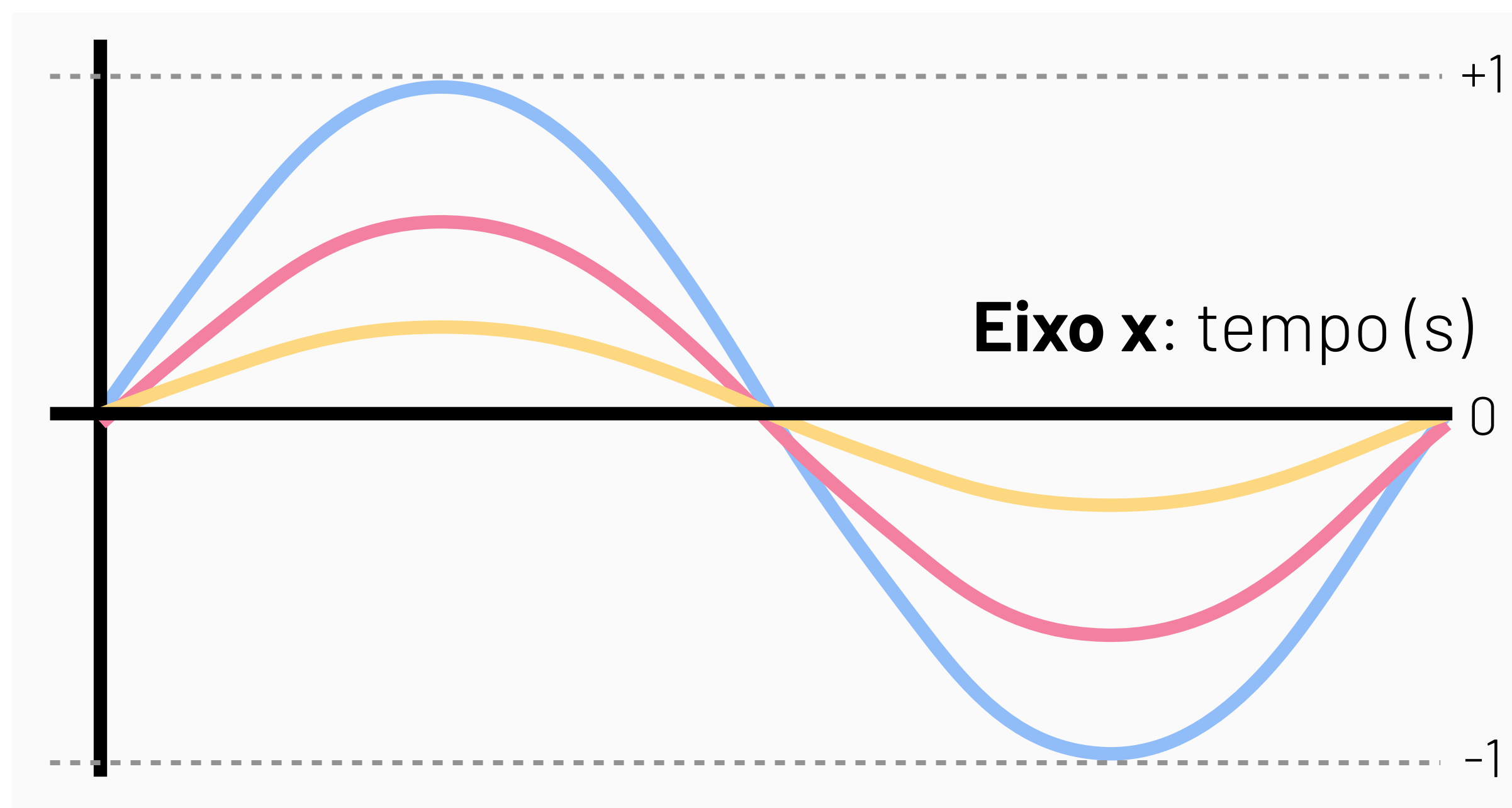
- ▶ Amplitude
- ▶ Frequência
- ▶ Fase

Amplitude



A **amplitude** de um sinal de áudio representa o deslocamento de moléculas de ar, geralmente numa escala entre -1 e +1:

Eixo y: amplitude



- ▶ **Amplitude = 1**
- ▶ **Amplitude = 0.5**
- ▶ **Amplitude = 0.25°**

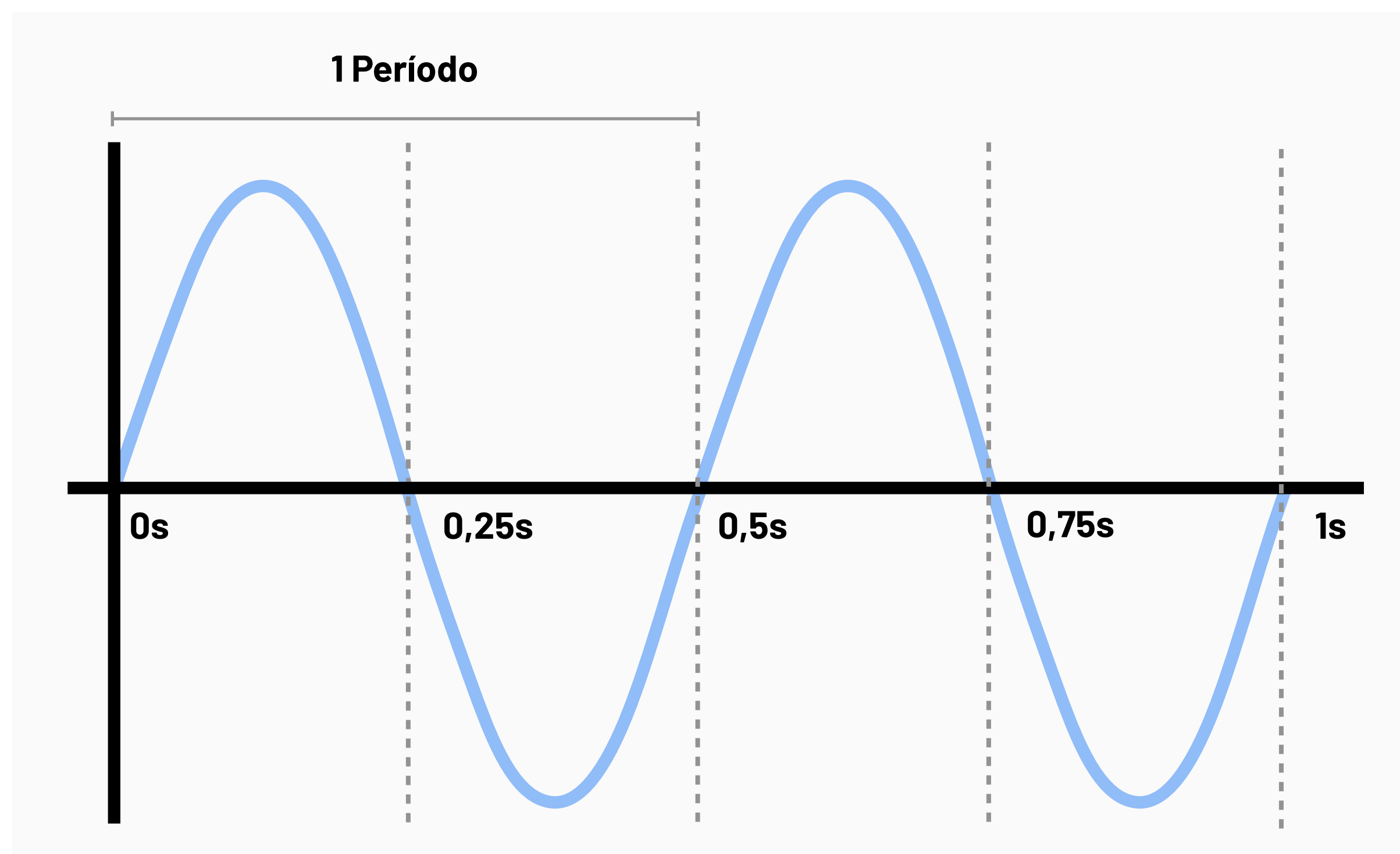
A amplitude está relacionada com o volume do som:

- ▶ Quanto maior a amplitude, mais alto você irá escutar o som
- ▶ Quanto menor a amplitude, mais baixo você irá escutar o som

Frequência



A **frequência** de um sinal de áudio mede (em Hz) quantas vezes ele se repete em um determinado período de tempo.



Frequência = 2Hz

2 períodos por segundo

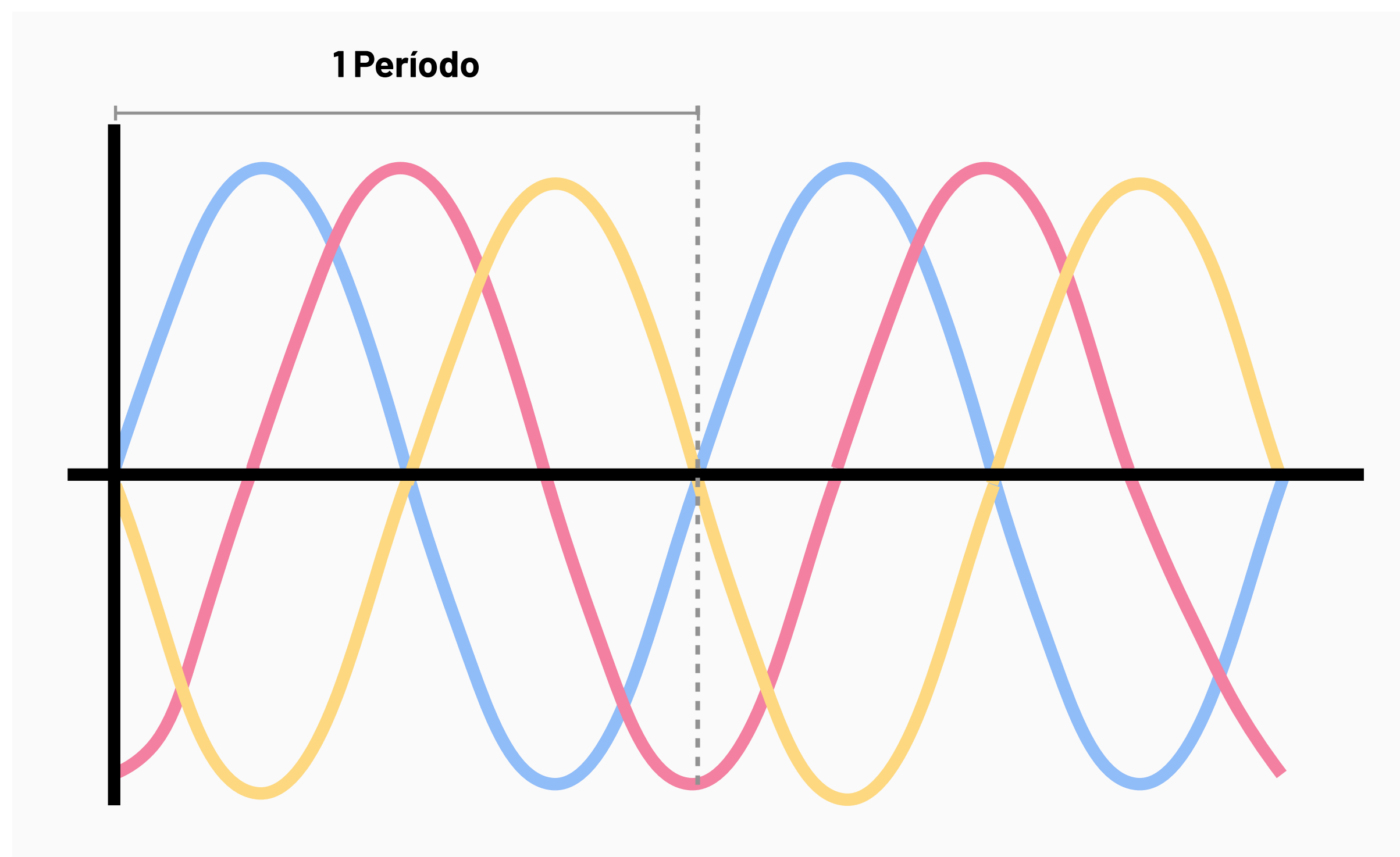
A frequência está relacionada com a *altura* do som:

- ▶ Quanto maior a frequência, mais agudo você irá escutar o som
- ▶ Quanto menor a frequência, mais grave você irá escutar o som

Fase



A **fase** é o deslocamento de um sinal, medido em graus ou radianos. Por exemplo, se um sinal está 180° fora de fase, isso significa que ele está atrasada em 50% do seu período:



- ▶ **Fase** = 0°
- ▶ **Fase** = 90°
- ▶ **Fase** = 180°

A fase se torna importante quando dois ou mais sinais são tocados ao mesmo tempo.

- ▶ Quando dois sinais estão em fase, eles atingem seus pontos de pico e vale no mesmo tempo, resultando em uma soma construtiva.
- ▶ Se estiverem fora de fase, os sinais se anulam parcialmente ou completamente, dependendo da diferença de fase.

Analógico vs Digital



Um sinal de áudio pode ser analógico ou digital:

Analógico

Um sinal de áudio analógico é um sinal contínuo representado por uma tensão ou corrente elétrica.

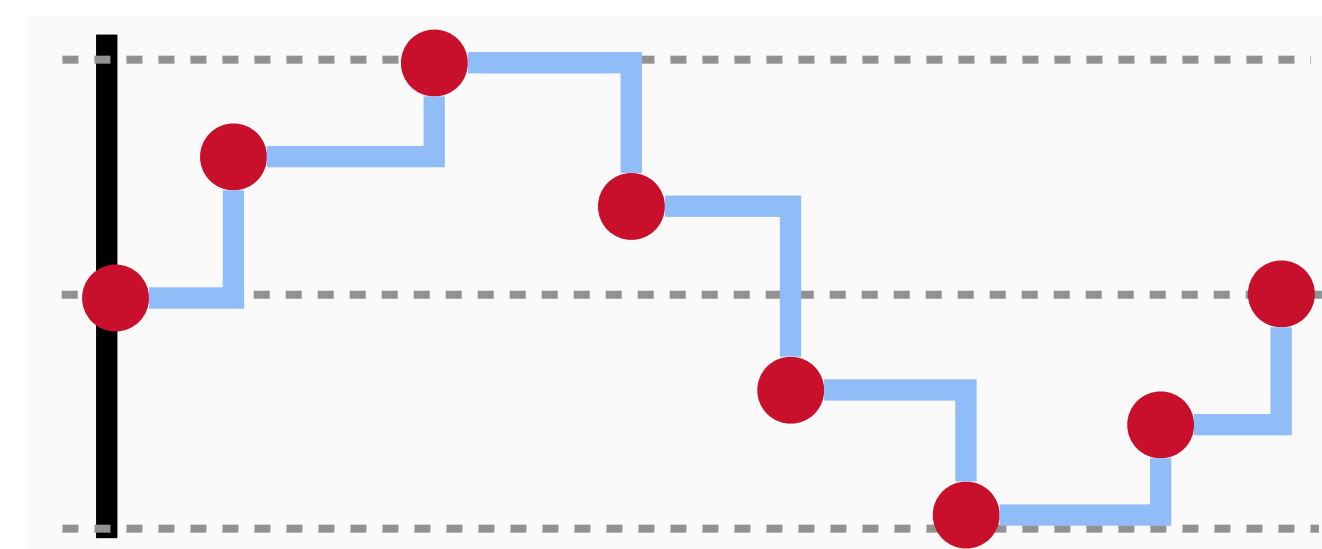
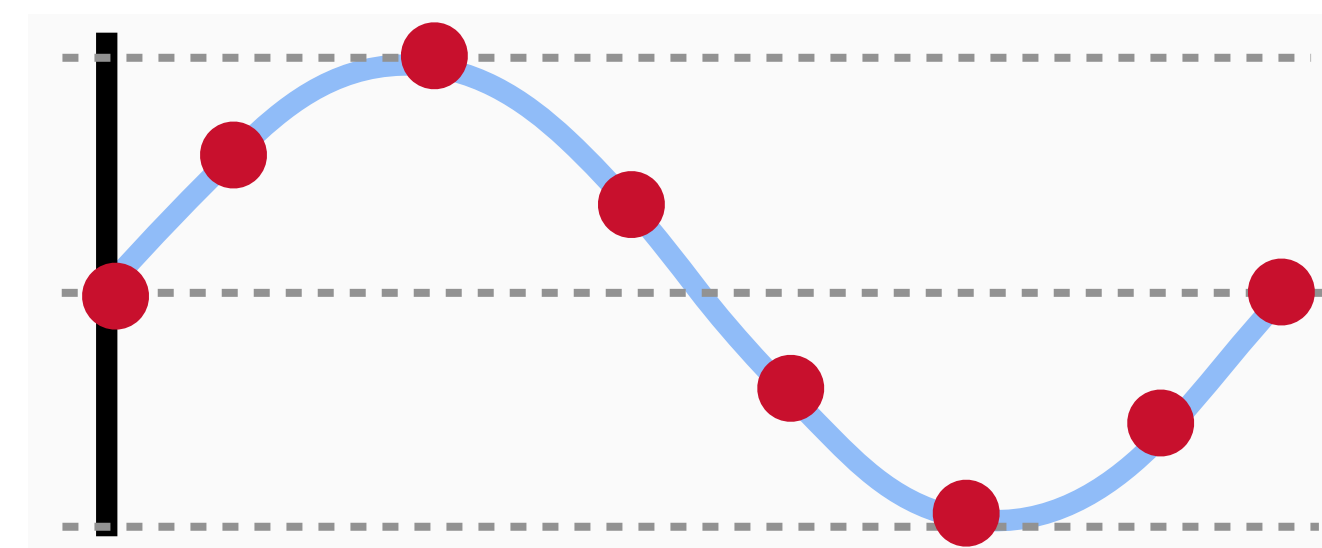
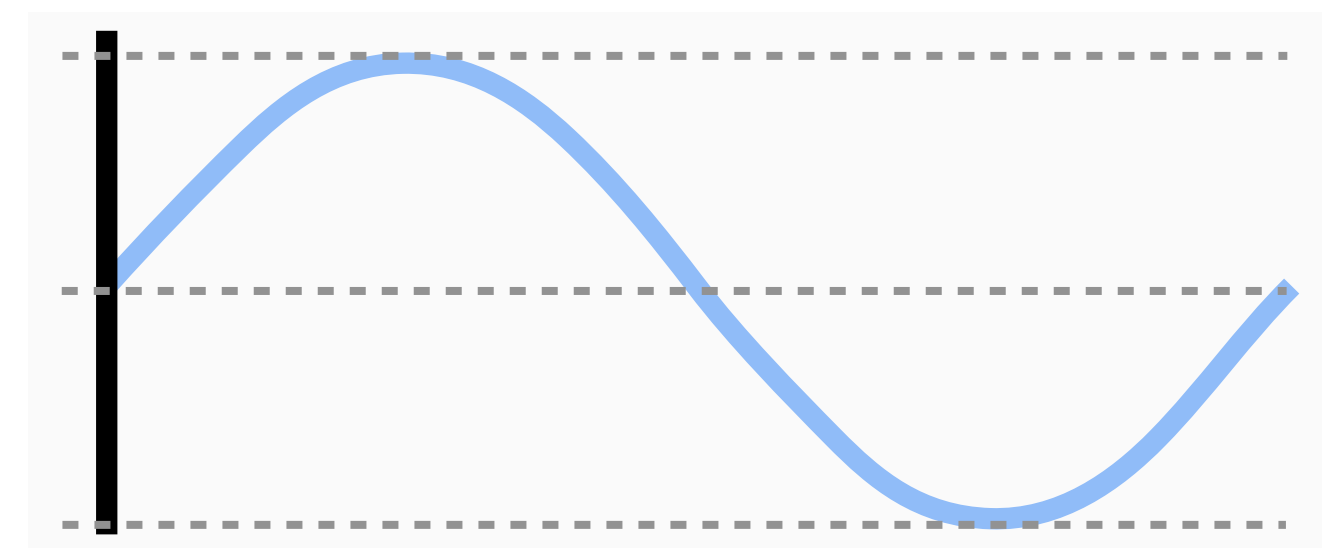


Conversor Analógico-Digital (ADC)

Amostragem ao longo do tempo

Digital

A representação digital expressa a forma de onda de áudio como uma sequência finita de amostras do sinal analógico

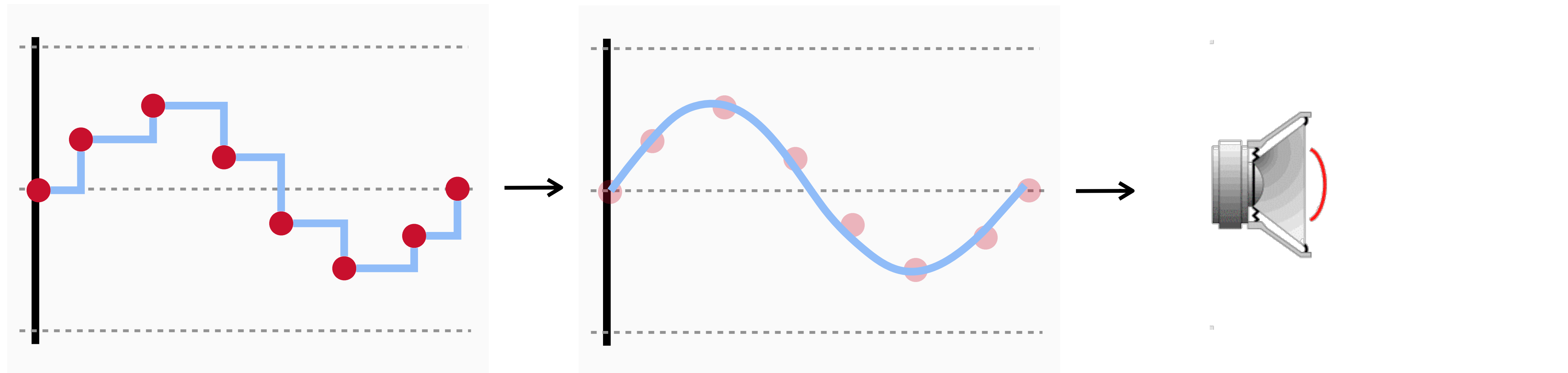


Sistemas de Reprodução de Áudio Digital



As amplitudes de um sinal de áudio digital são convertidos em tensões em um processo de conversão analógica-digital (DAC), as quais fazem o alto-falante vibrar:

Conversor Digital-Analógico (DAC)



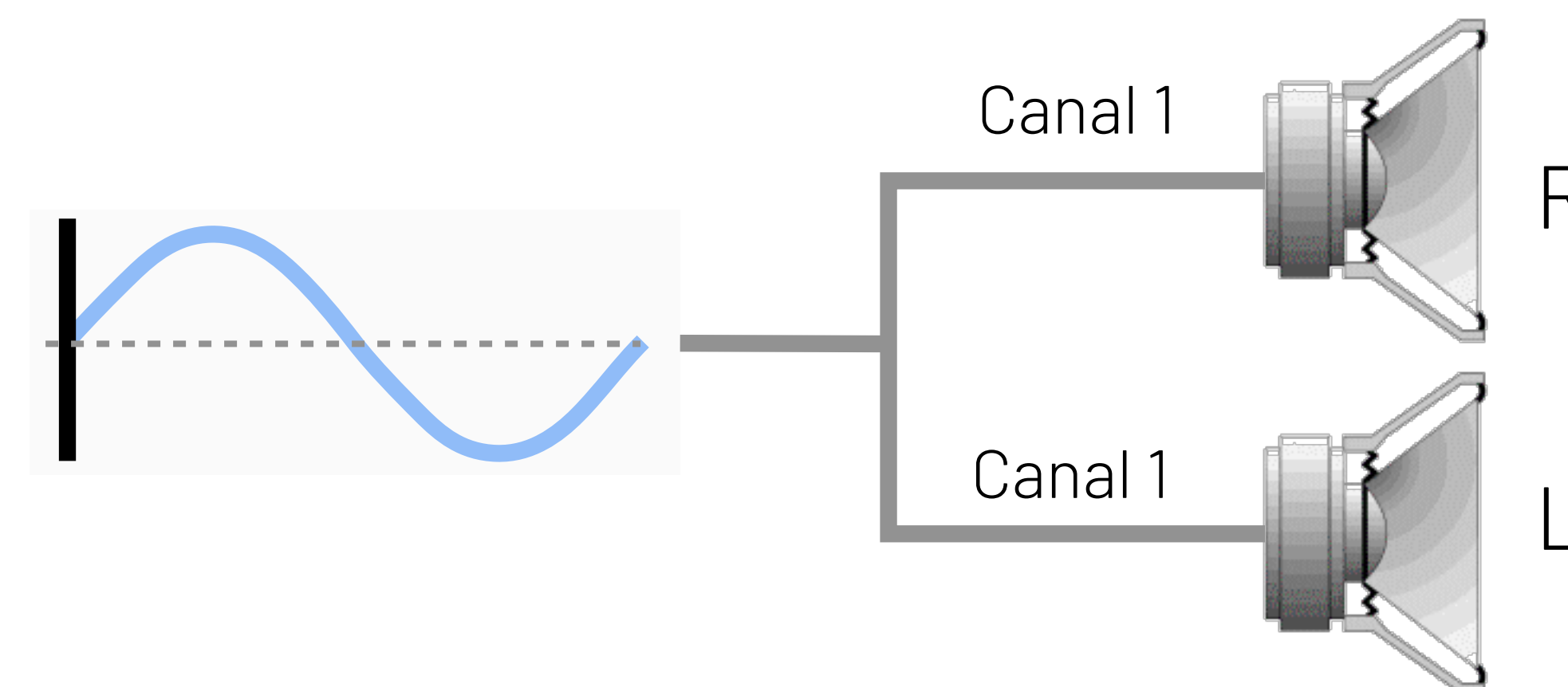
Sinais Mono Vs. Stereo



Um sinal de áudio pode ser mono ou estéreo:

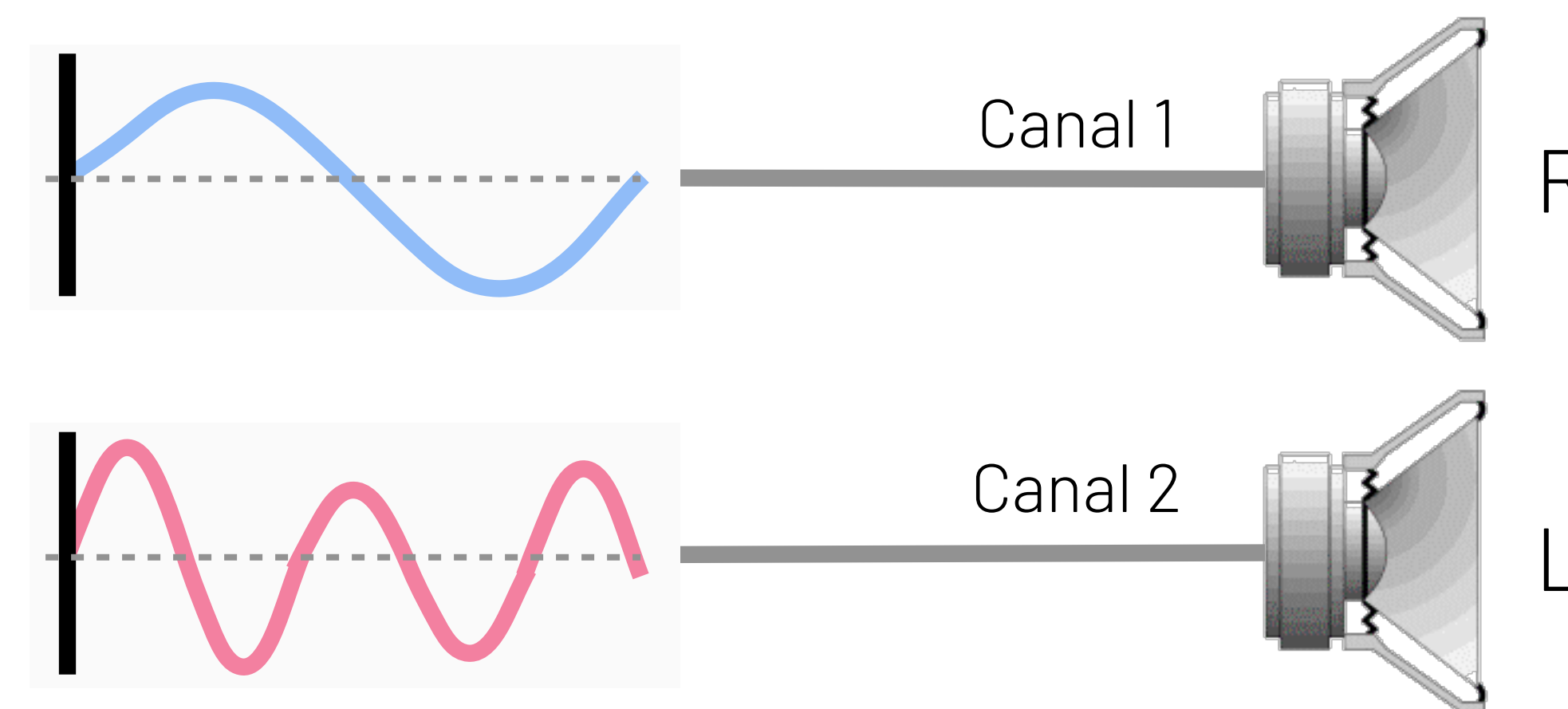
Mono

- ▶ O sinal mono representa uma única onda sonora.
- ▶ Todos os elementos sonoros (instrumentos, vocais, etc) são misturados em um único sinal e reproduzidos igualmente em ambos os alto-falantes.



Estéreo

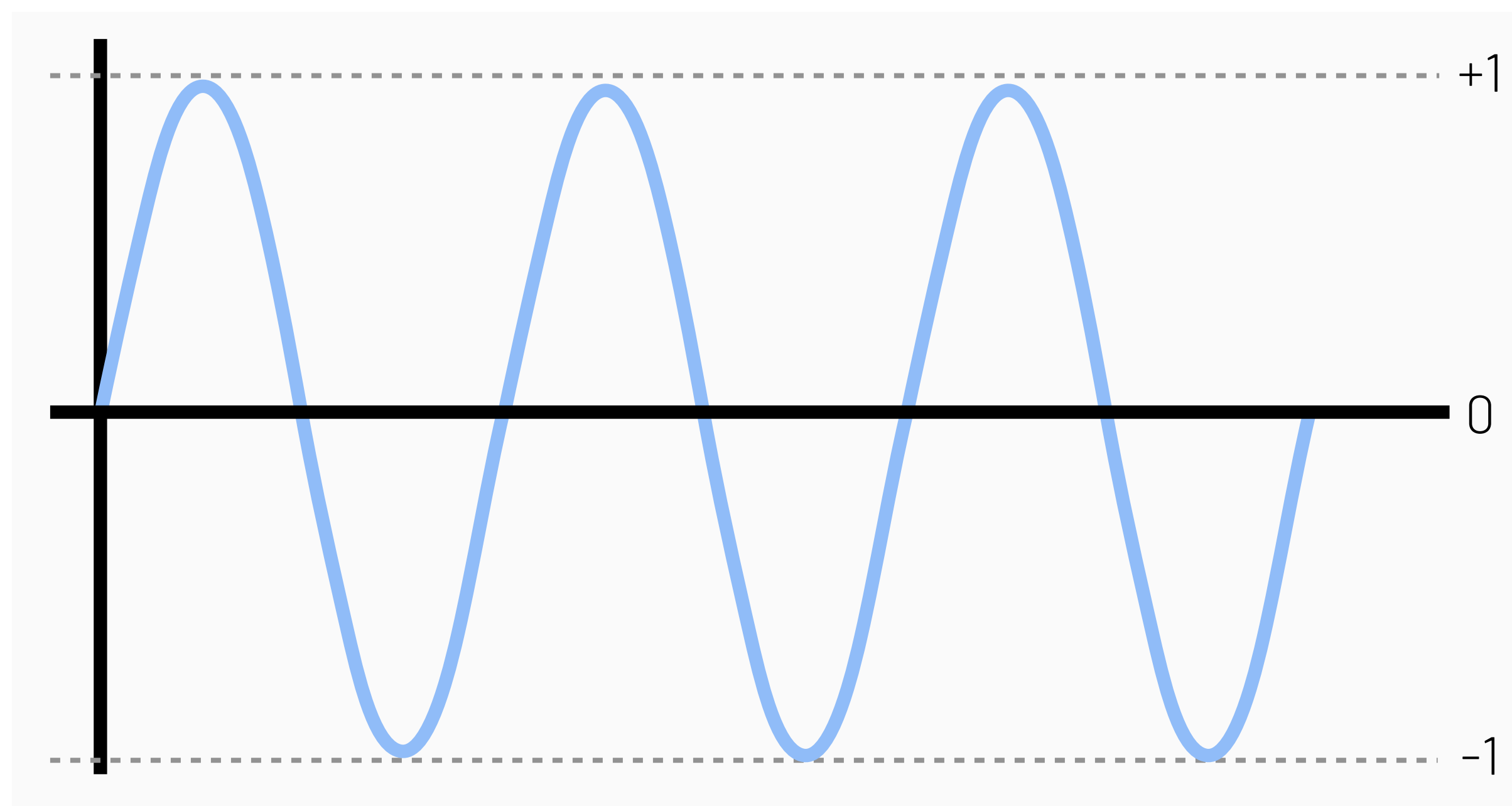
- ▶ O sinal estéreo representa duas ondas sonoras: um para o canal esquerdo e outra para o canal direito.
- ▶ É possível ouvir instrumentos ou vocais posicionados à esquerda, à direita ou no centro



Ondas Senoidais



As **ondas senoidais** são as ondas sonoras periódicas mais simples e por isso também são chamadas de ondas fundamentais:



Podemos gerar ondas senoidais com a seguinte fórmula:

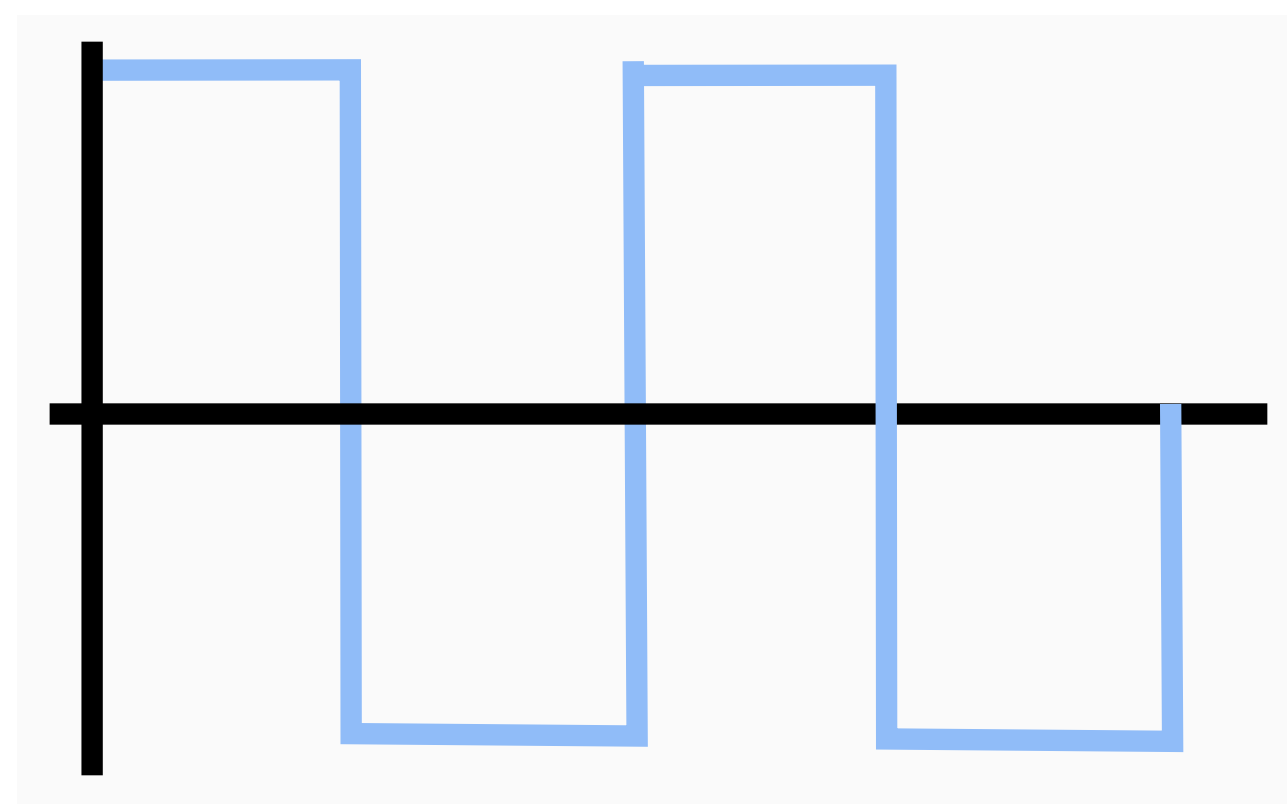
$$y(t) = A \sin(2\pi ft + \phi)$$

- ▶ A é a amplitude
- ▶ t : tempo em segundos
- ▶ f : frequência (número de ciclos por segundo)
- ▶ ϕ : fase (em graus ou radianos)

Ondas Não-senoidais

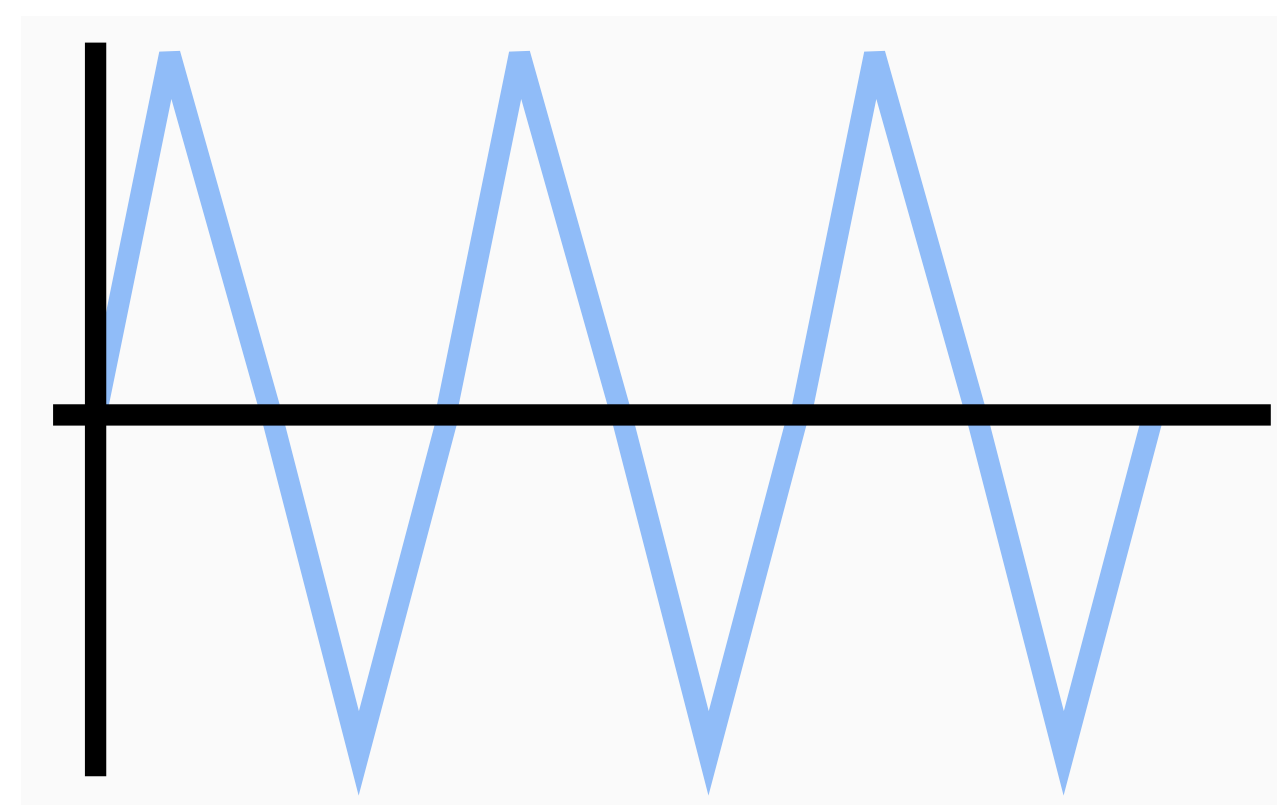


Outros formatos de onda comuns não-senoidais que podemos sintetizar matematicamente com facilidade incluem:



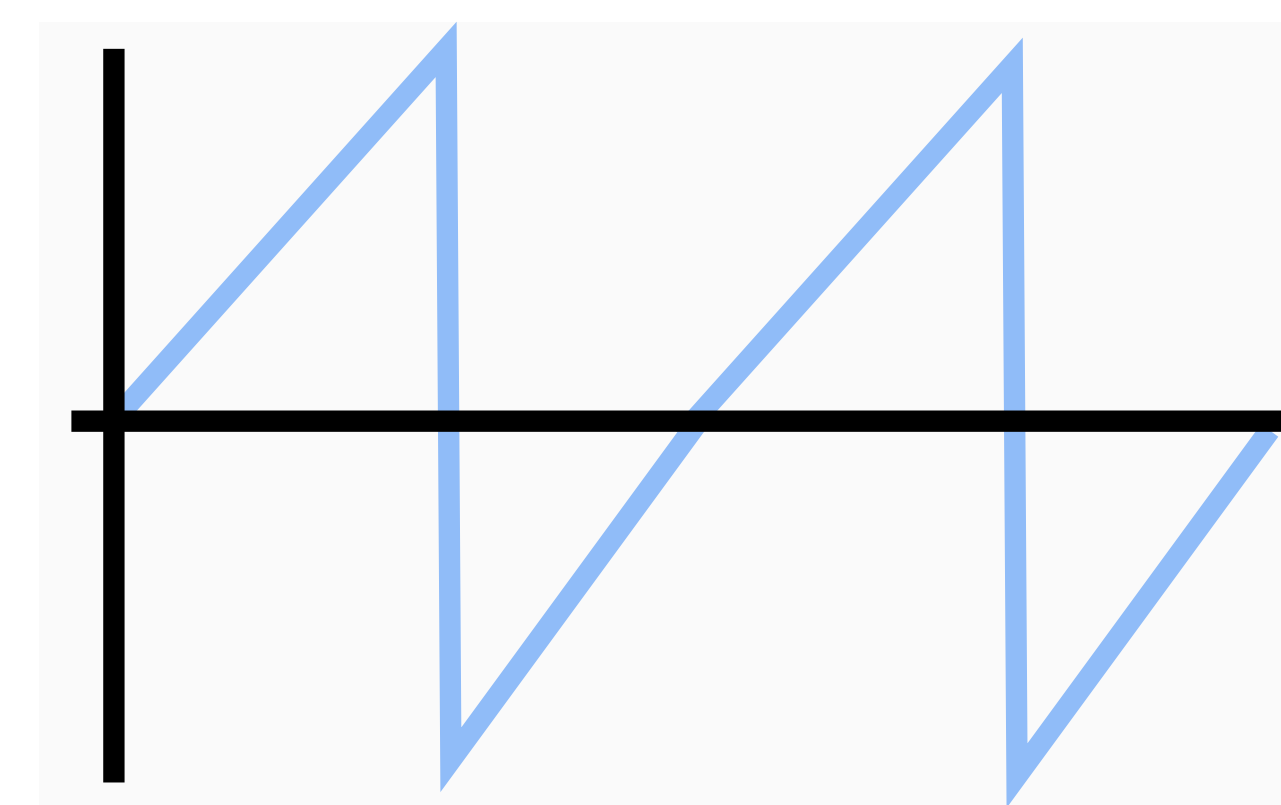
Onda Quadrada

$$y(t) = A \operatorname{sgn}(\sin(2\pi ft + \phi))$$



Onda Triangular

$$y(x) = \frac{4A}{p} \left| \left(\left(x + \frac{\phi}{2\pi} - \frac{p}{4} \right) \bmod p \right) - \frac{p}{2} \right| - A$$



Onda dente de serra

$$y(x) = \frac{2A}{p} \left(\left(x + \frac{\phi}{2\pi} \right) \bmod p - \frac{p}{2} \right)$$

Sintetizando sinais de áudio em SDL



Para gerar e reproduzir sinais de áudio em SDL, podemos usar a função `SDL_OpenAudioDevice`:

```
// Áudio callback que gera uma senoide contínua
void sinusoidal(void* userdata, Uint8* stream, int len) {
    static double phase = 0.0;
    float* buffer = reinterpret_cast<float*>(stream);
    int length = len / sizeof(float);

    for (int i = 0; i < length; ++i) {
        buffer[i] = amplitude * std::sin(phase);
        phase += 2.0 * PI * frequency / SAMPLE_RATE;
        if (phase >= 2.0 * PI)
            phase -= 2.0 * PI;
    }
}

// Configurar o áudio
SDL_AudioSpec desiredSpec;
SDL_zero(desiredSpec);
desiredSpec.freq = SAMPLE_RATE;
desiredSpec.format = AUDIO_F32SYS;
desiredSpec.channels = 1;
desiredSpec.samples = 1024;
desiredSpec.callback = sinusoidal;

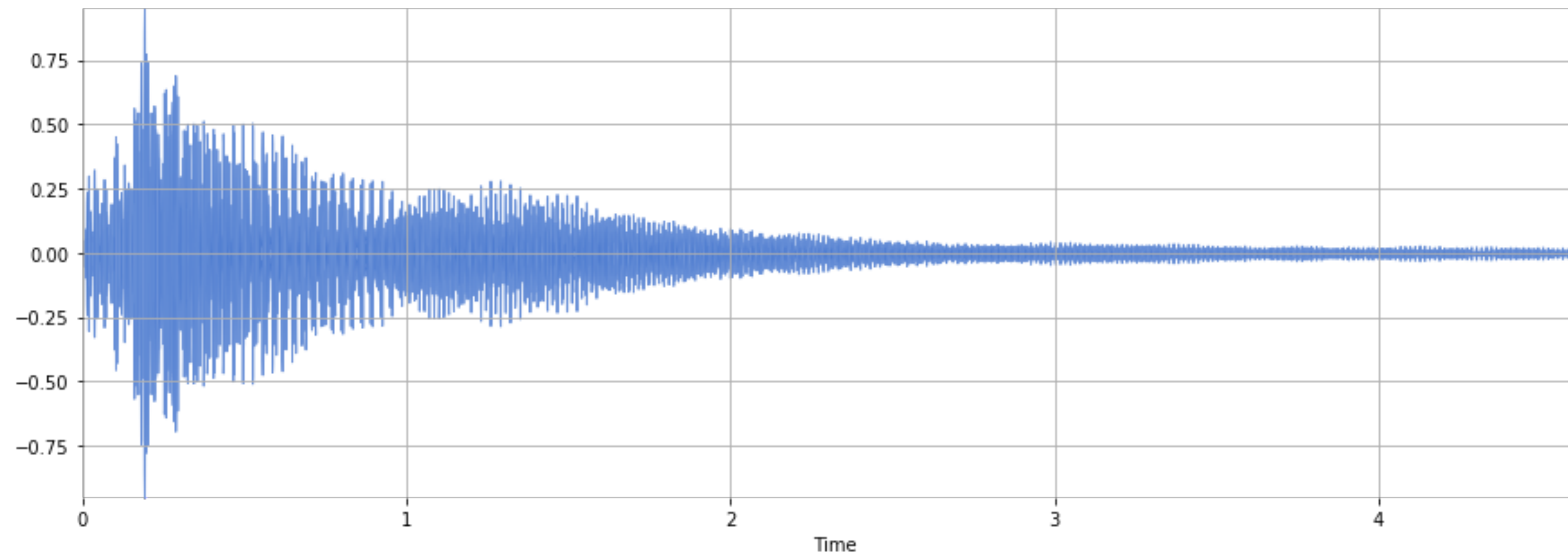
SDL_AudioSpec obtainedSpec;
SDL_AudioDeviceID deviceId = SDL_OpenAudioDevice(nullptr, 0, &desiredSpec, &obtainedSpec, 0);
SDL_PauseAudioDevice(deviceId, 0); // começar o áudio
```

Sinais de Áudio Reais



No mundo real, os sons geralmente não são periódicos, ou seja, eles não se mantêm constante ao longo tempo. O gráfico abaixo ilustra um sinal de áudio de um acorde de Dó Maior no violão:

Eixo y: amplitude — deslocamento das moléculas de ar

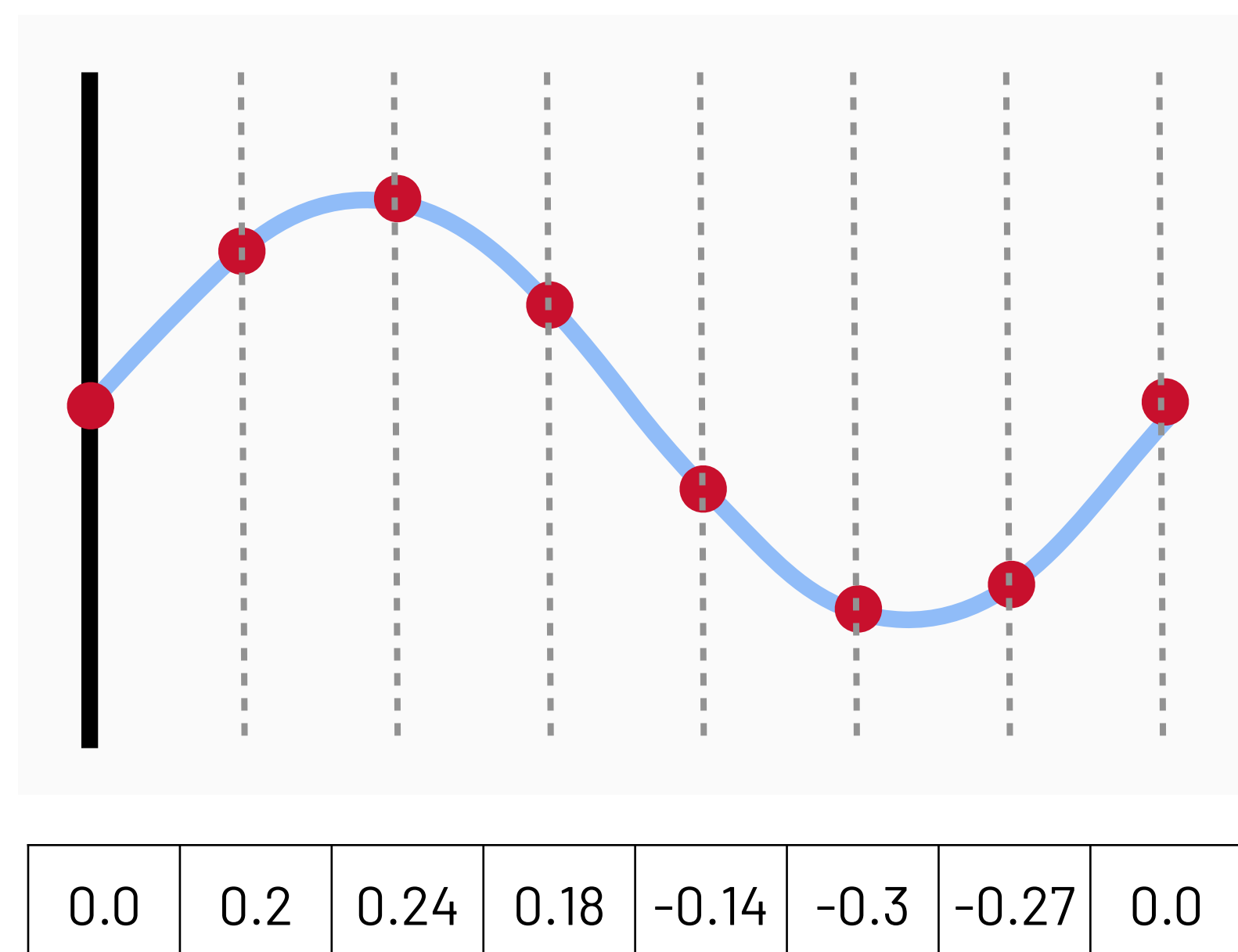


Eixo x: tempo (s)

Espectrogramas



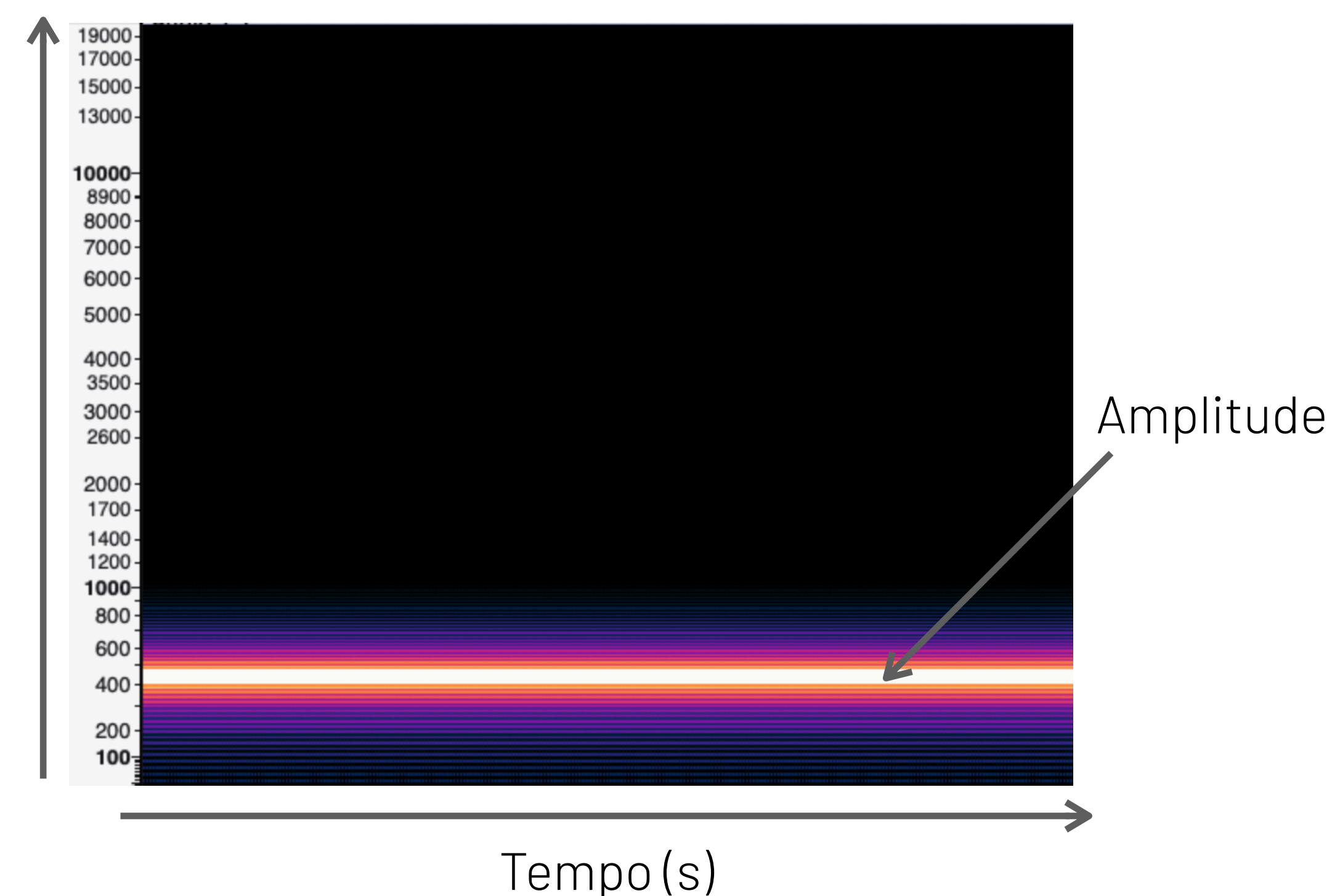
O sinal de áudio carrega informações de **amplitude** ao longo do tempo, mas não de **frequência**. Para visualizar a frequência de um sinal, podemos aplicar uma Transformada de Fourier:



→
**Transformada
de Fourier**

Frequência (Hz)

Spectrograma



Somando Sinais de Áudio



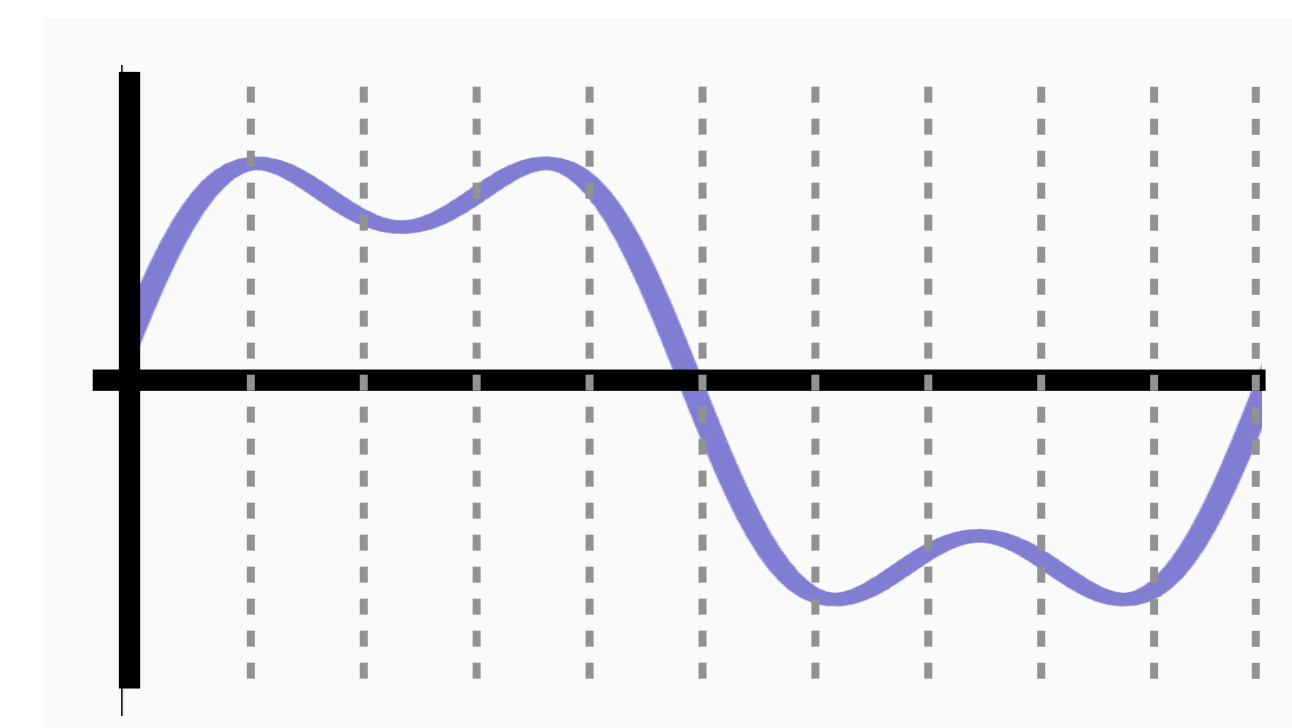
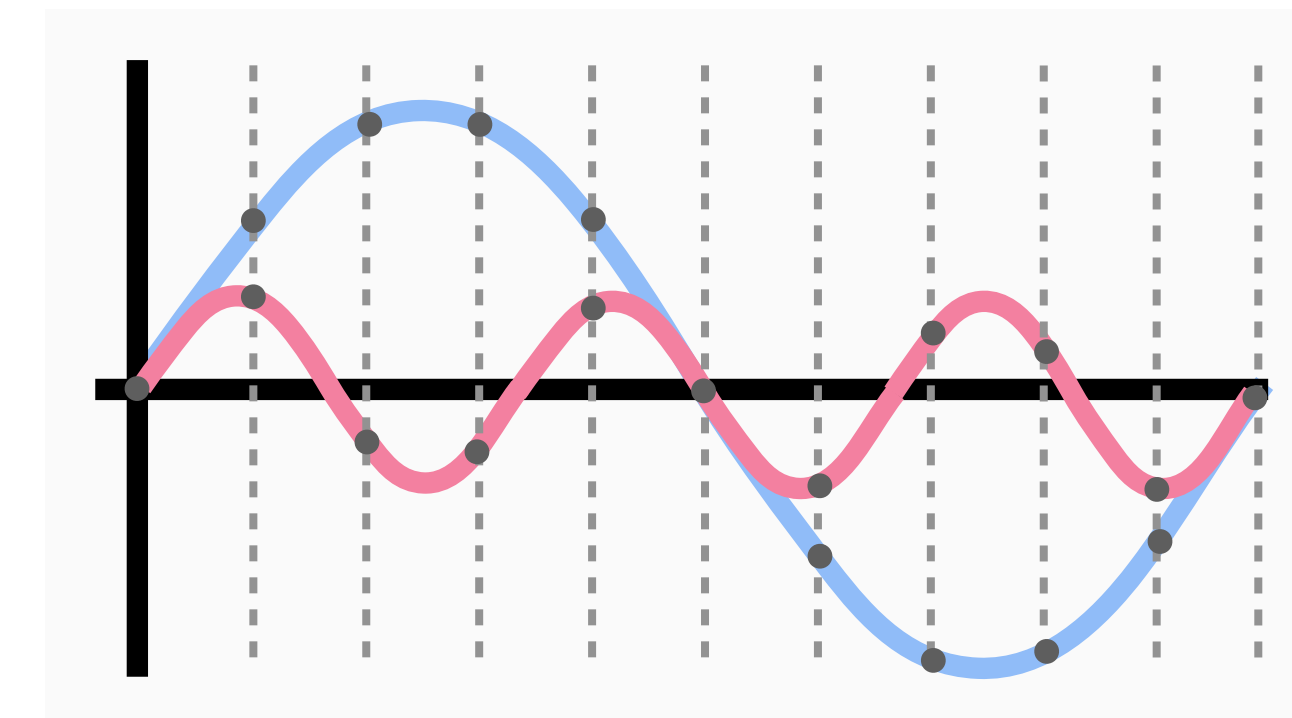
Quando dois ou mais sinais de áudio são tocados ao mesmo tempo, eles são **somados formando um único sinal**:



+



=



Sintetizando notas musicais

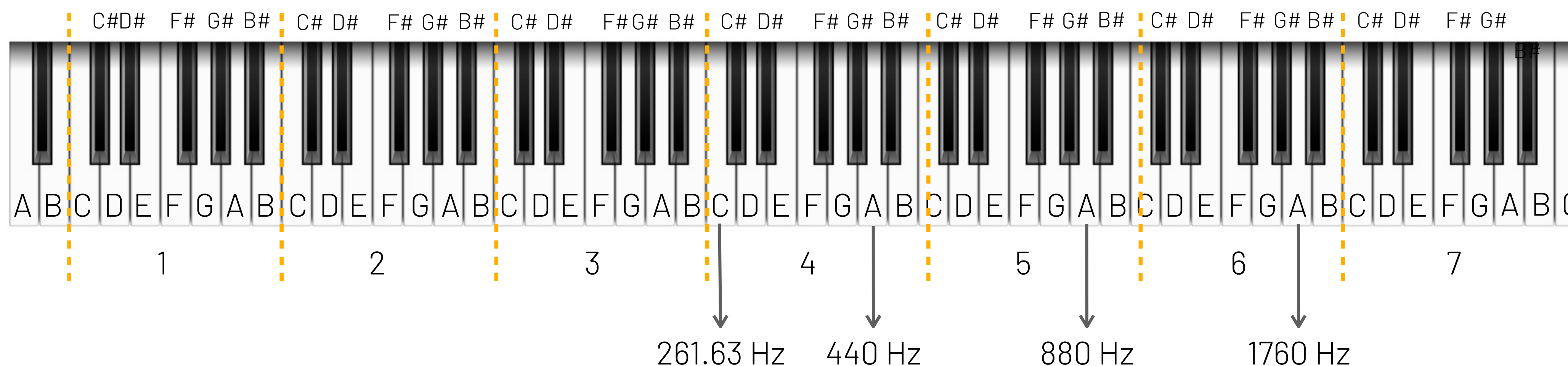


As notas musicais são símbolos que representam sons individuais com alturas (frequências) específicas, sendo o alfabeto fundamental da música. No sistema ocidental, temos 12 notas:

- ▶ Naturels: C, D, E, F, G, A, B
- ▶ Accidentals: C#, D#, F#, G#, A#

Oitava

As 12 notas se repetem em todas as oitavas



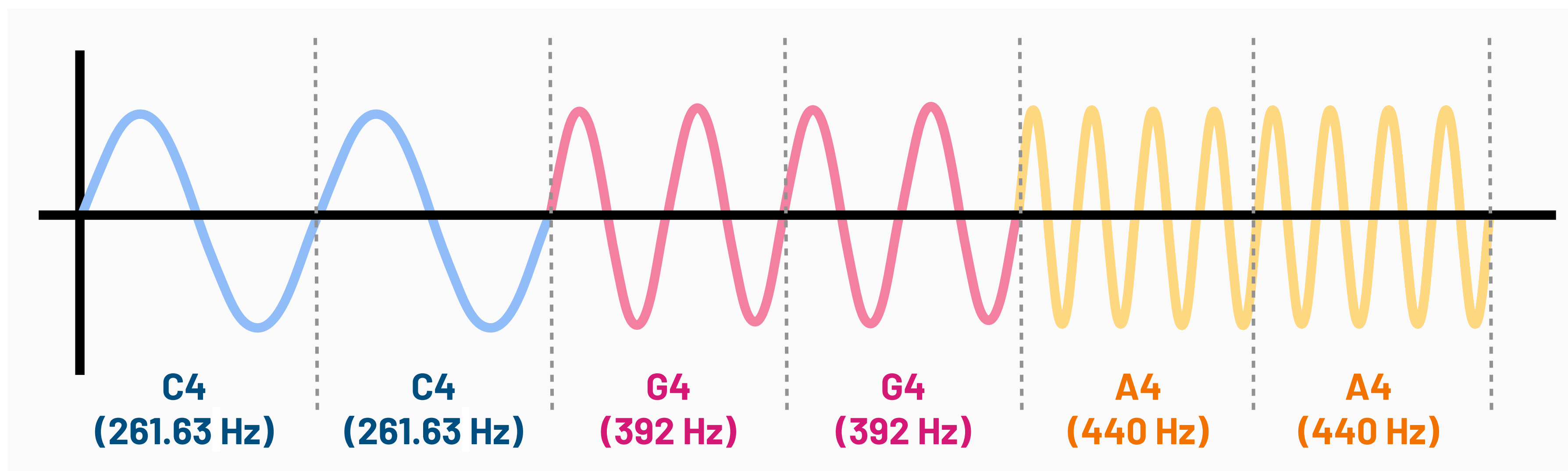
As frequências das notas dobram de um oitava para a outra! Por isso nós as percebemos como o mesmo som, apesar das frequências diferentes!

Sequenciadores



Uma forma de sintetizar músicas é usando um **sequenciador**, que sintetiza uma sequência de notas alterando a frequência f de uma onda senoide em intervalos de tempo constantes:

Melodia: "C4 C4 G4 G4 A4 A4 G4 F4 F4 E4 E4 D4 D4"



$$y(t) = A \sin(2\pi ft + \phi)$$

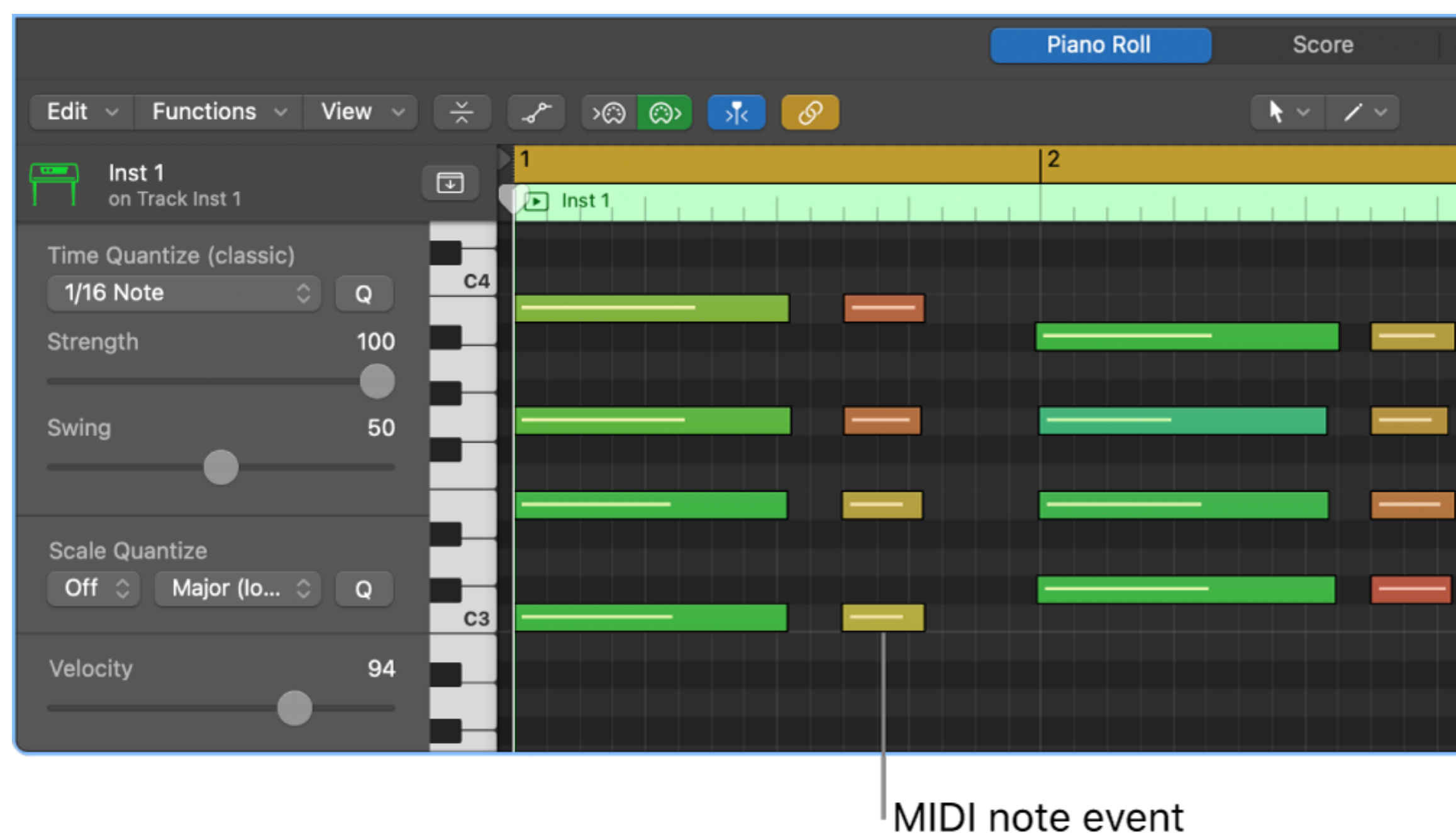
Sequenciadores MIDI



A maioria dos programas de produção musical, chamados de Digital Audio Workstations (DAWs) possuem um sequenciador MIDI, ou seja, a sequência de notas é definida pelo protocolo MIDI:

O protocolo MIDI representa músicas por meio de eventos musicais organizados no tempo:

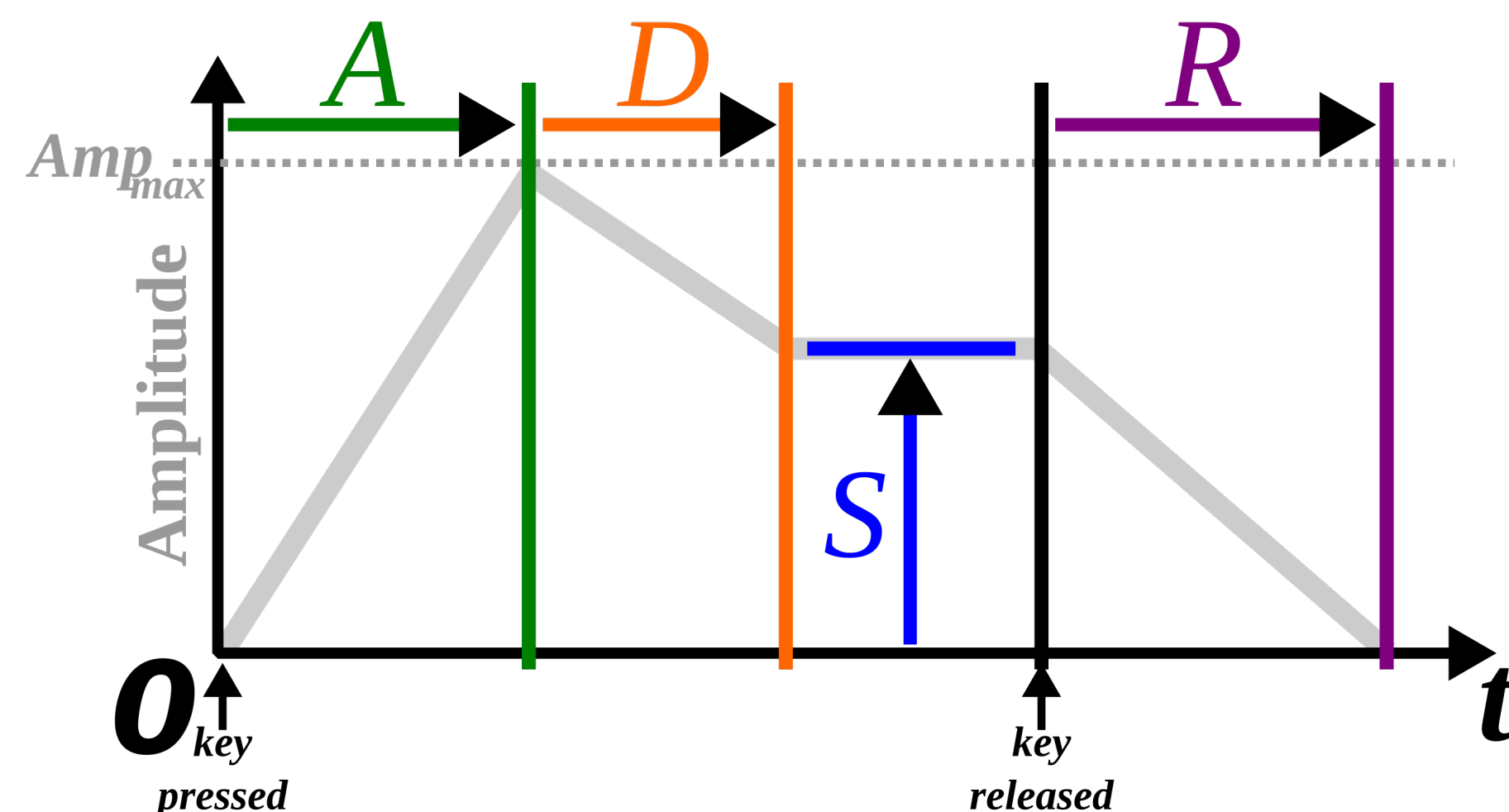
- ▶ Note on/off
 - ▶ Channel: 0-15
 - ▶ Velocity: 0-127
 - ▶ Pitch: 0-127
- ▶ Program Change
 - ▶ Channel: 0-15
 - ▶ Instrument number: 0-127
- ▶ ...



Envelopes



Um **envelope** é uma curva que controla o valor de amplitude (ou outros parâmetros) de um sinal de áudio ao longo do tempo. Geralmente, um envelope divide o sinal de áudio em 4 etapas:



- ▶ **Attack (A):** A fase inicial em que o volume do som sobe de zero até seu nível máximo.
- ▶ **Decay (D):** Período em que o volume do som diminui do nível de pico para o nível de sustentação.
- ▶ **Sustain (S):** O nível em que o volume do som permanece enquanto uma nota é mantida.
- ▶ **Release (R):** A fase em que o volume do som diminui de volta a zero após a nota ser liberada.

Envelopes são usados para sintetizar timbres diferentes!

Próxima aula



A17: Reprodução de Áudio

- ▶ Carregando e Reproduzindo Áudio em SDL
- ▶ Sistema de Gerenciamento de Áudio
- ▶ Lidando com Repetições
- ▶ Efeitos em Processamento de Sinais