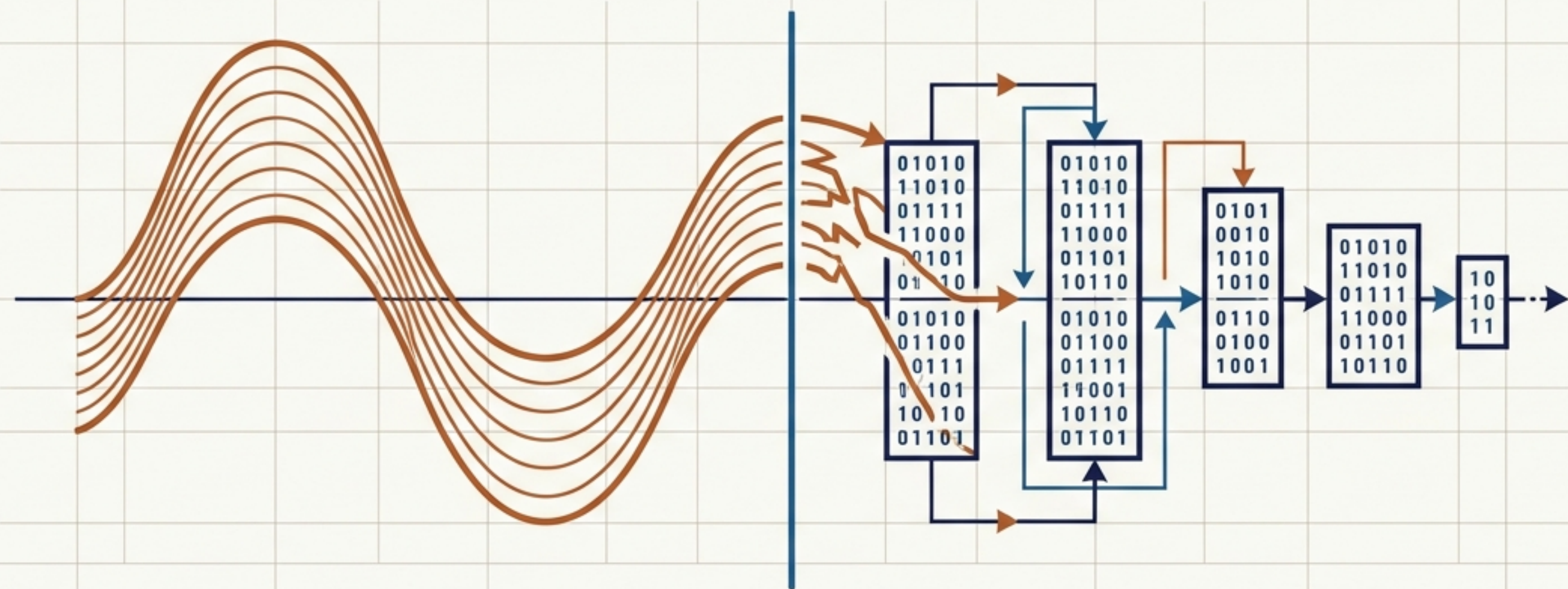


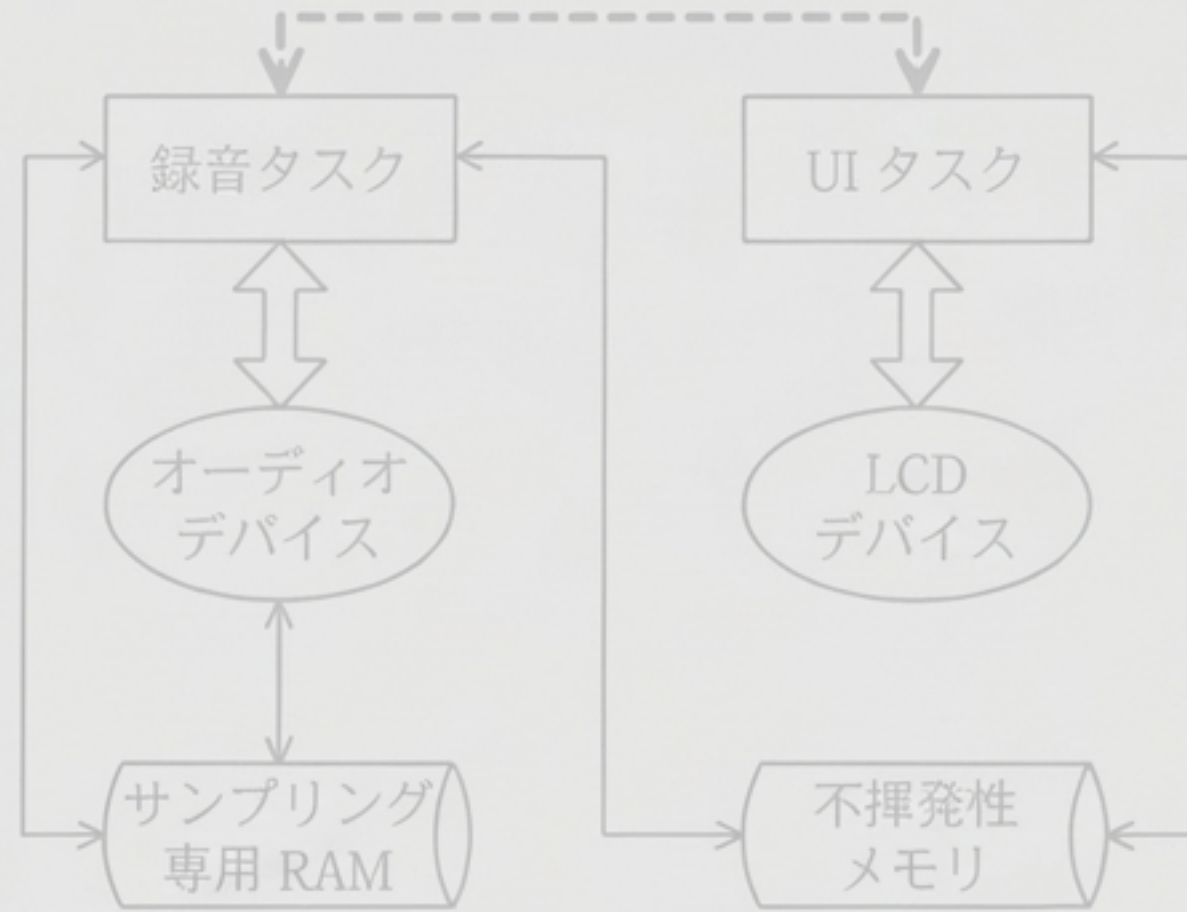


# リアルタイム音声処理の解体と再構築

組込みシステムにおけるデータ量計算とバッファアーキテクチャの思考プロセス

応用情報技術者試験 設問1 (1) (2) 徹底解剖

# ミッション・ブリーフィング：対象サブシステムの特定

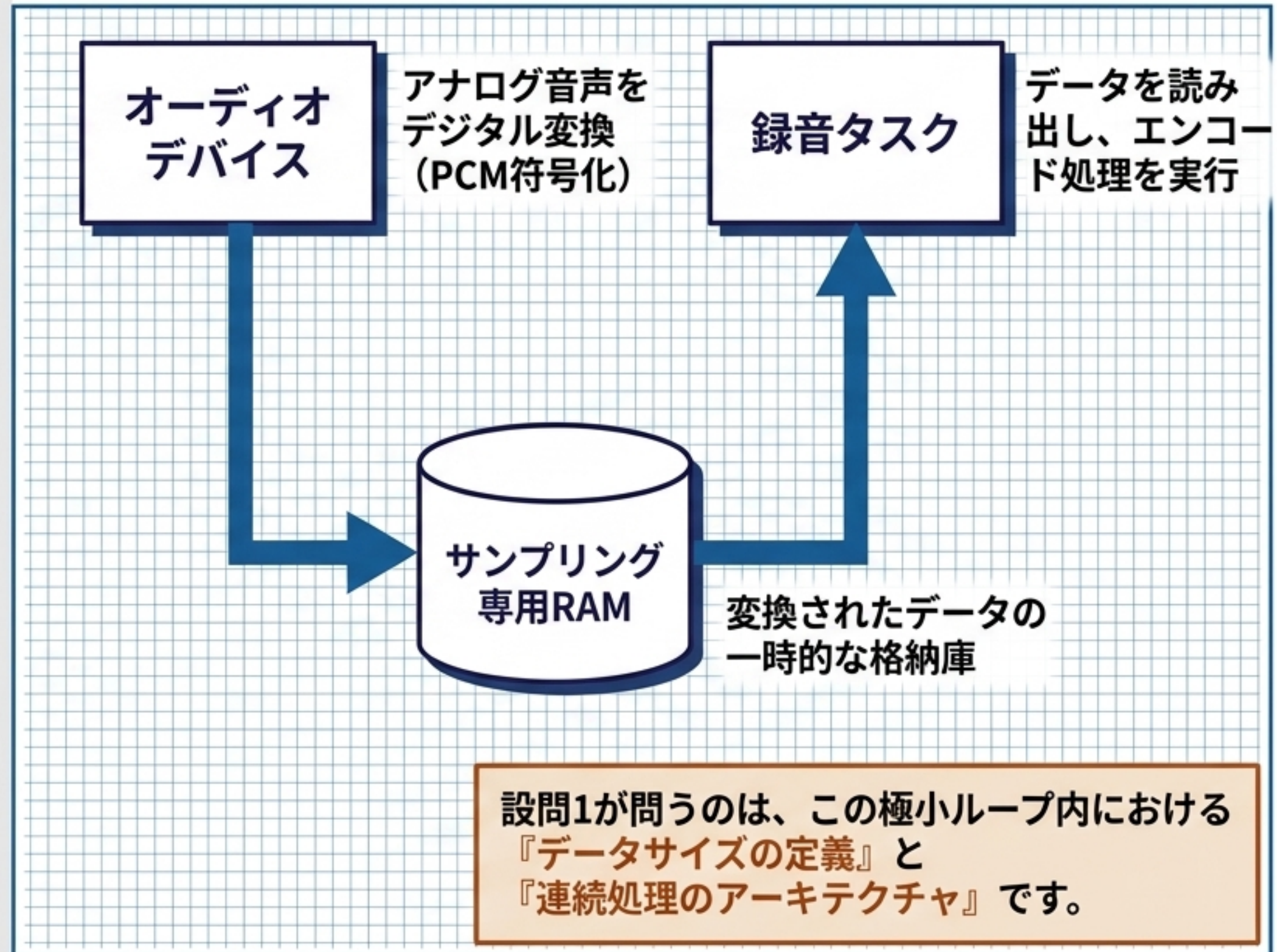


## 凡例

- : タスク
- : デバイス
- ▭ : メモリ
- ↔ : メモリアアクセス
- ↔---> : メッセージ

UIタスク：ユーザインタフェースタスク

図1 録音機能に関連するタスク、デバイス及びメモリの関係



# 要件の解剖：”音楽CD並み”とは何か？

録音データの音質は、音楽CD並み（サンプリング周波数 44.1kHz, 量子化ビット数 16ビット, チャンネル数 2ch）とする。

周波数 (Hz)

44,100  
(44.1kHz)

情報量 (Bit)

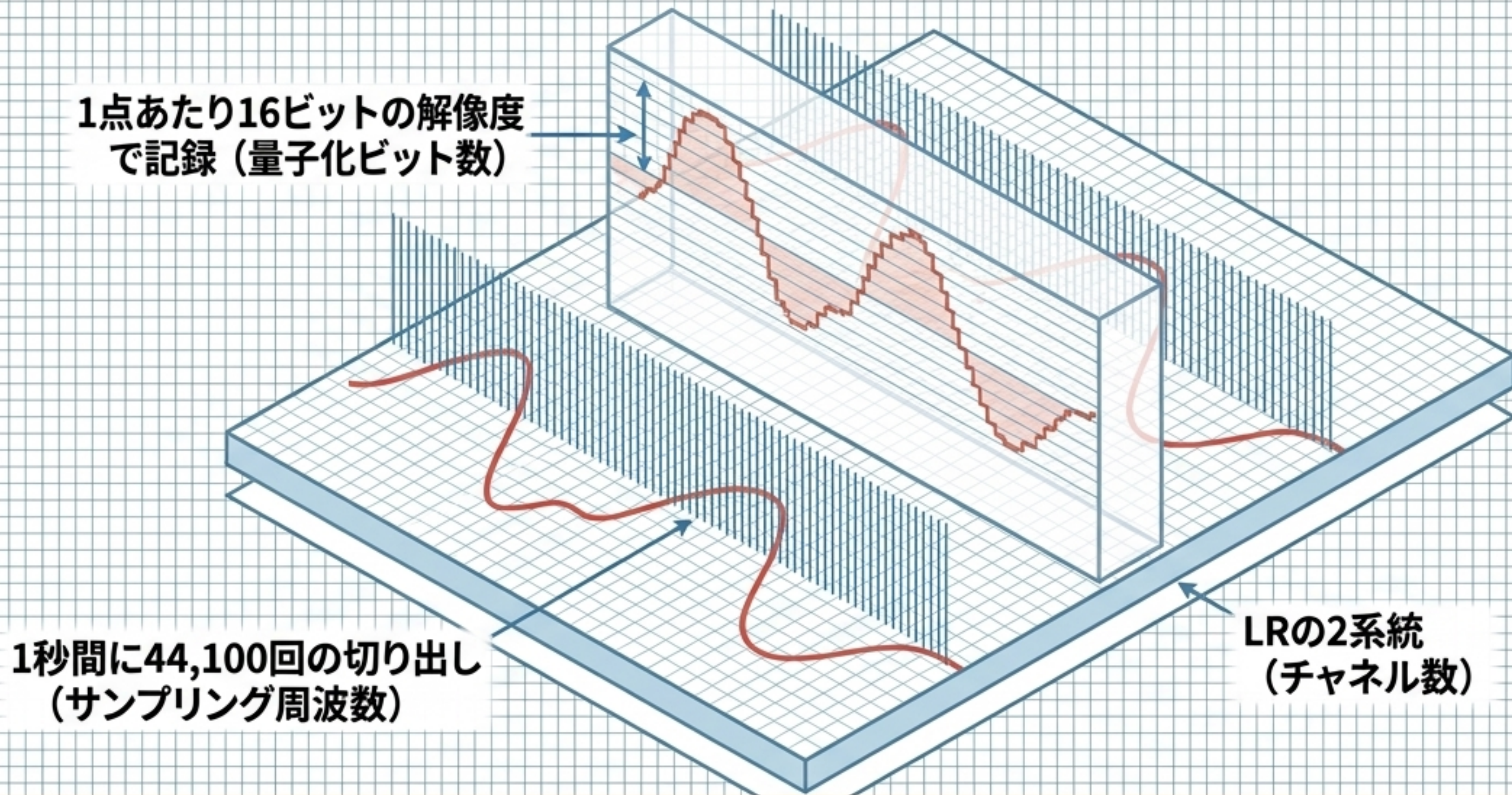
16

空間 (Ch)

2

抽象的な「音質」という言葉を、計算可能な3つのデジタル・パラメータに分解します。

# アナログからデジタルへの変換モデル



1秒間のデータ量 = 『切り出す回数』 × 『1回あたりのデータサイズ』 × 『系統数』

# データサイズ計算ロジックの構築

44,100

×

16

×

2

= 合計ビット数

ビット → バイト



÷ 8 を実行

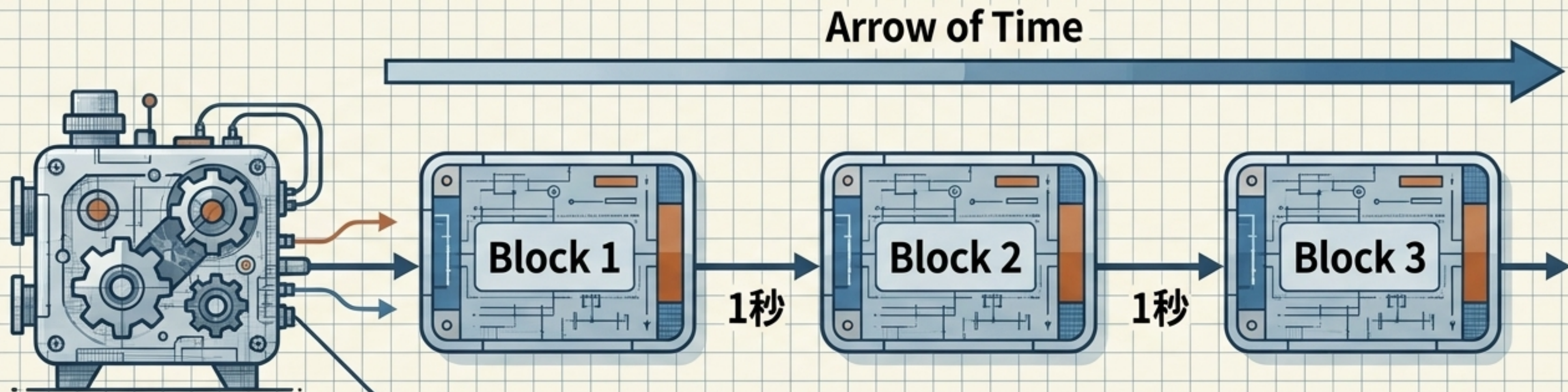
バイト → キロバイト



÷ 1,000 を実行 (設問指示)

1秒間 (1ブロック) あたりのサイズ =  kバイト

# リアルタイム性のジレンマ：絶え間ないデータストリーム

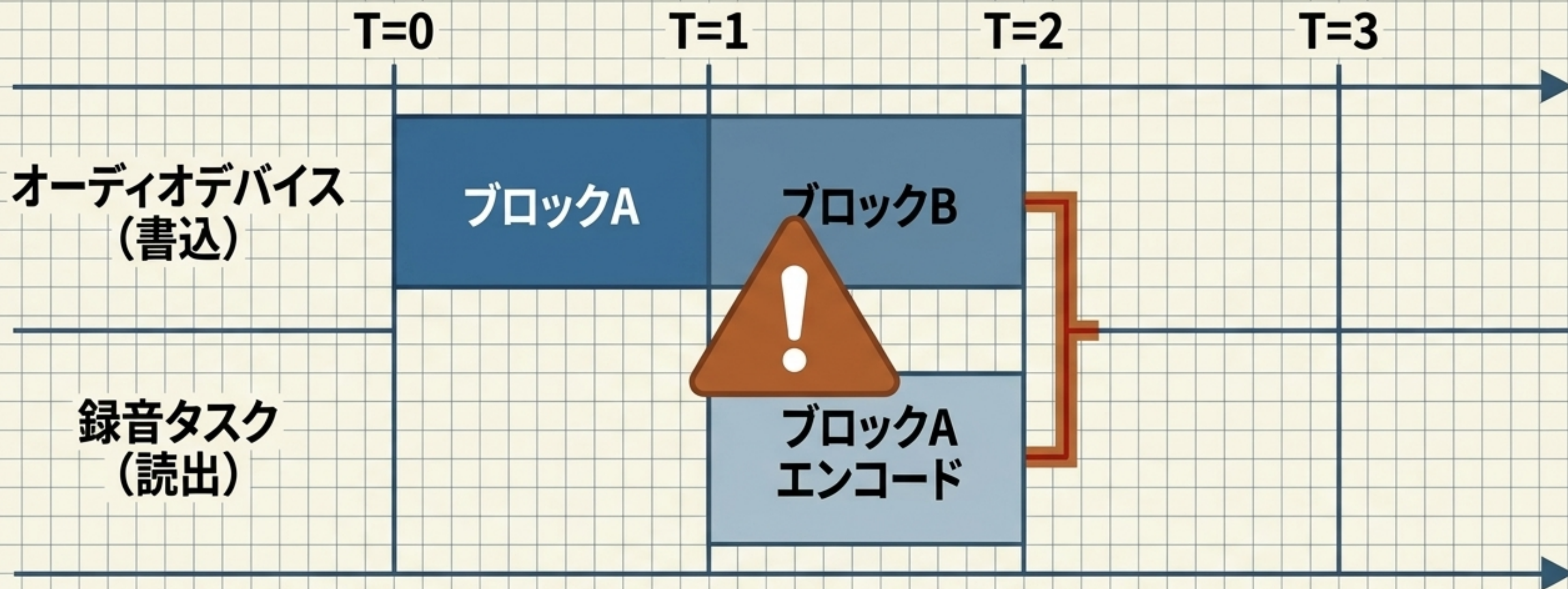


オーディオデバイスは待ってくれない。1秒ごとに新しい録音データをサンプリング専用RAMに書き込み、割込みを発生させる。



録音タスクがエンコードを『連続して』実行するためには、RAMはどのような構造であるべきか？

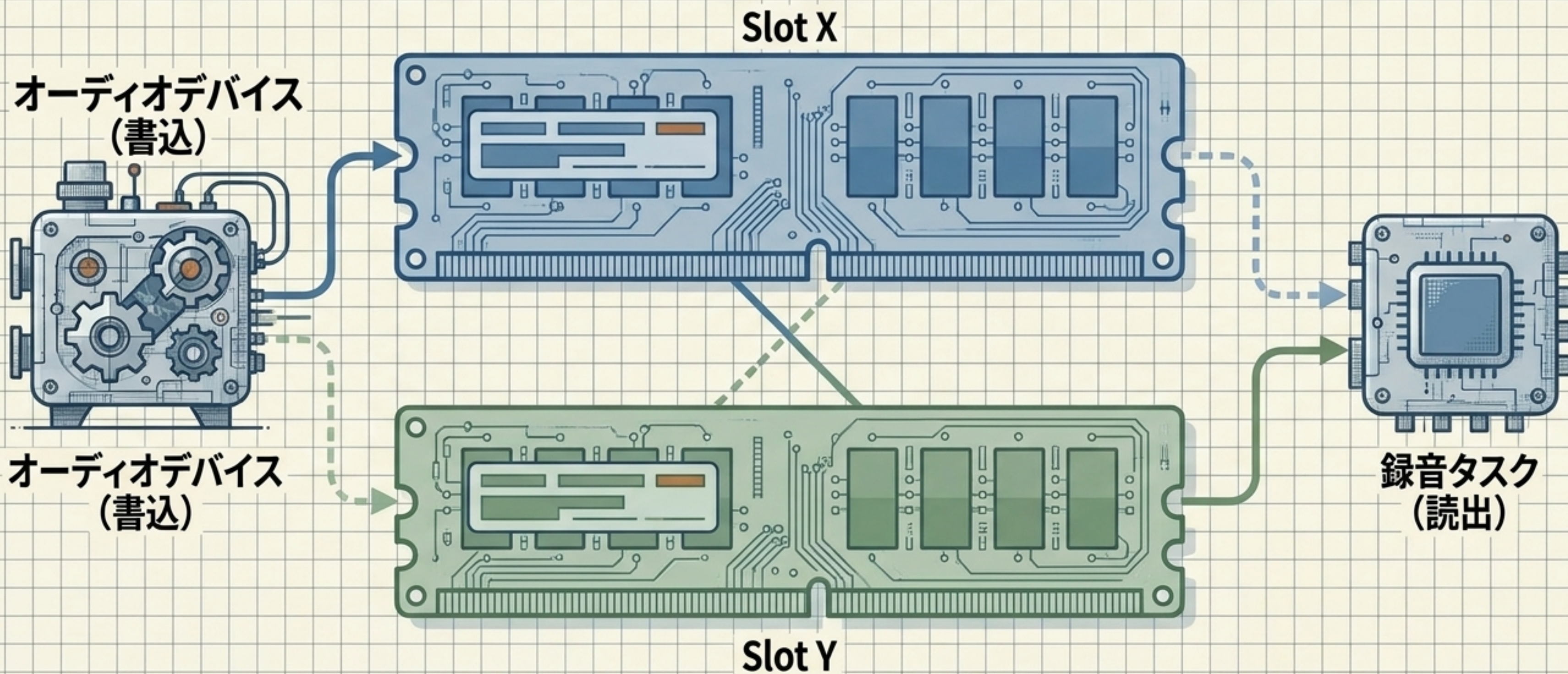
## 破綻のシナリオ (単一ブロック制約)



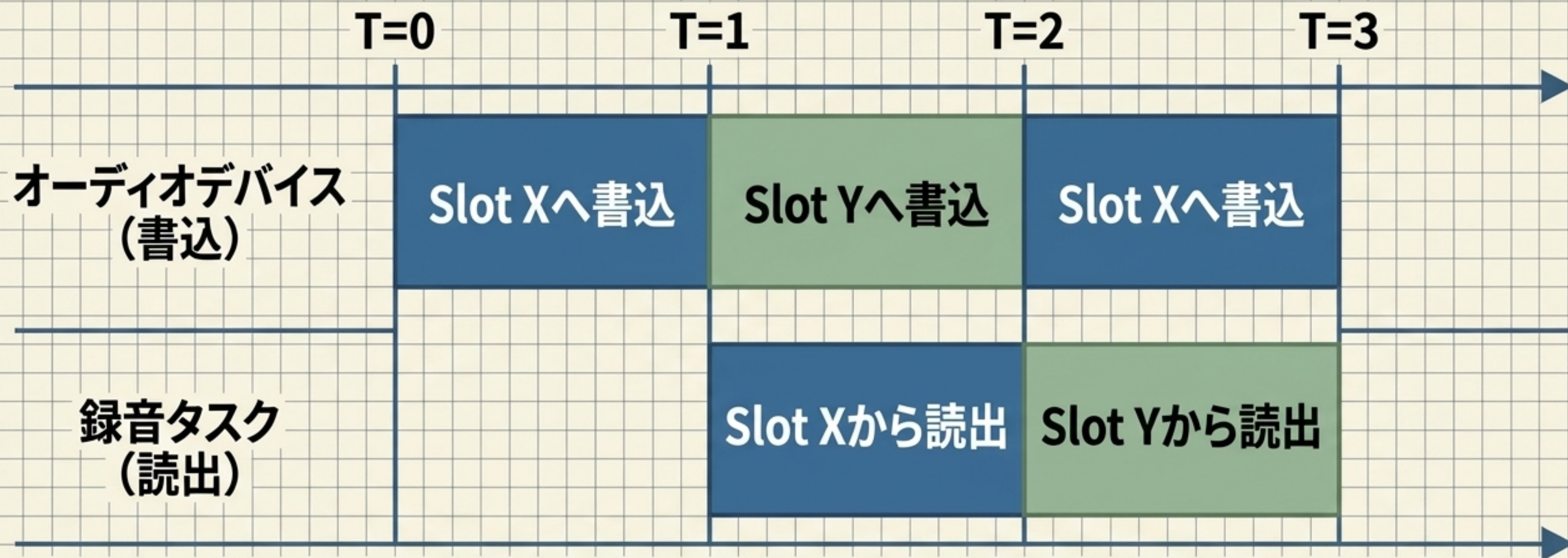
読み出し(エンコード)と書き込み(次のサンプリング)が同じメモリ空間で衝突し、データが破壊(上書き)される。

# アーキテクチャによる解決策：Ping-Pongアプローチ

空間を分割し、生産者（書込）と消費者（読出）が交互に異なる領域を使用するバッファリング戦略。



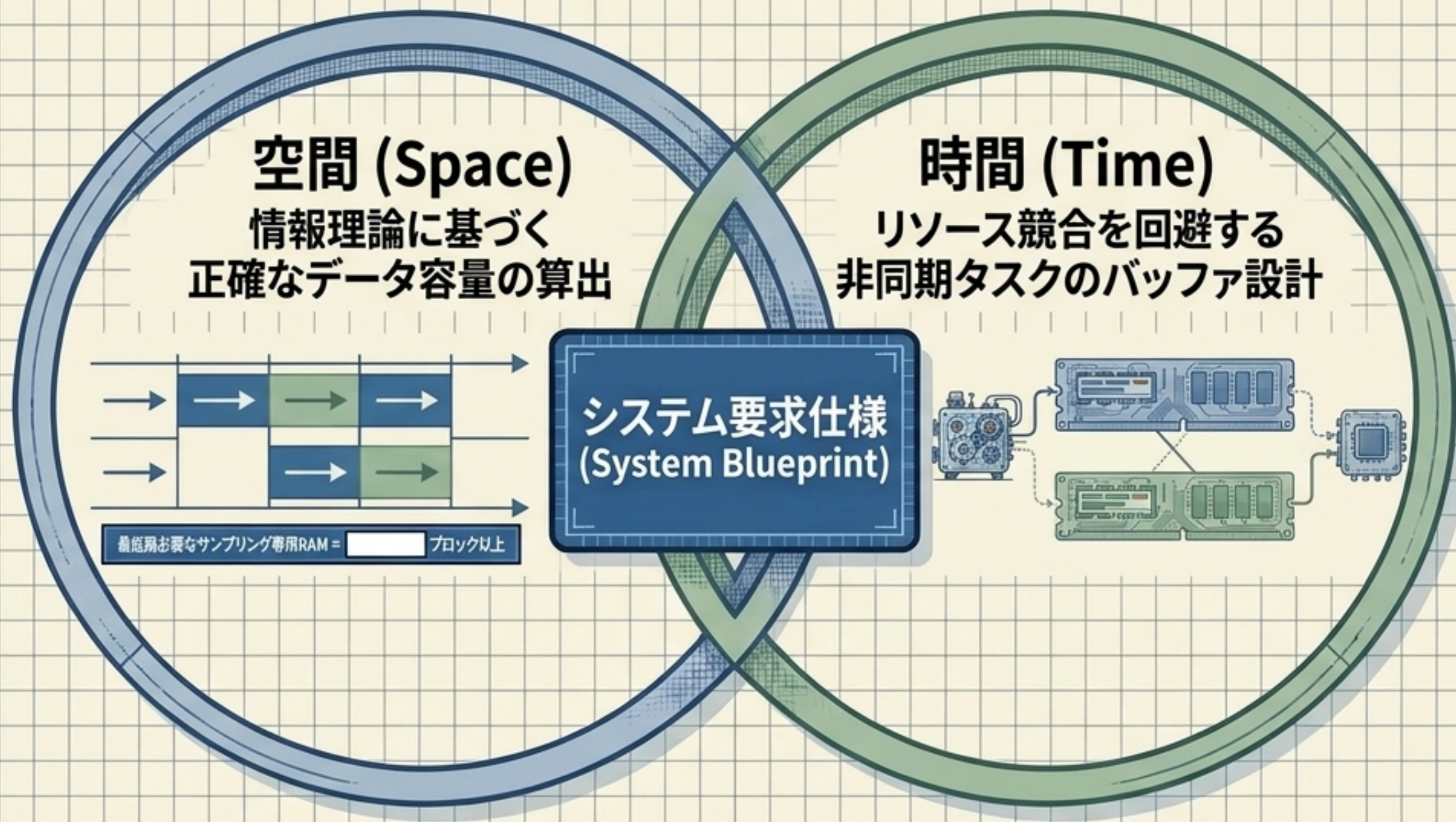
# 連続エンコードの実現



タイムラインの衝突を完全に回避し、途切れることなくデータを処理し続けるために、RAMには独立した領域がいくつ存在していたでしょうか？

最低限必要なサンプリング専用RAM =  ブロック以上

# Synthesis: 組み込みシステム設計の2つの次元



優れたエンジニアは、単に数値を計算するだけでなく、その数値がハードウェア上の時間軸の中でどのように振る舞うかを設計図として描くことができる。