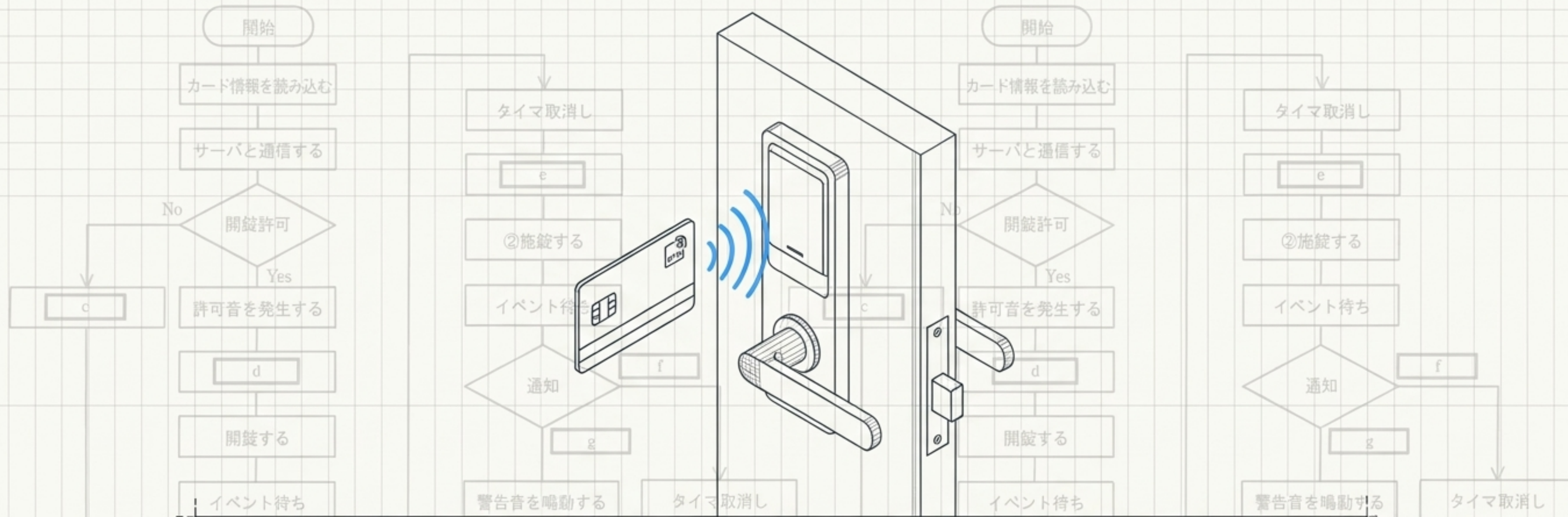




電子扉システムの論理を解説せよ

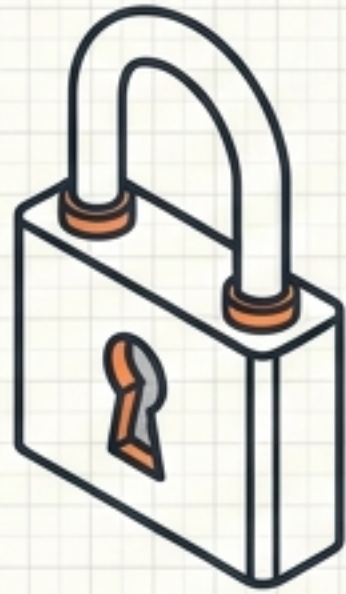
応用情報技術者試験 午後問7：設問2 完全攻略ガイド（解答非公開・ヒント特化型）

QA

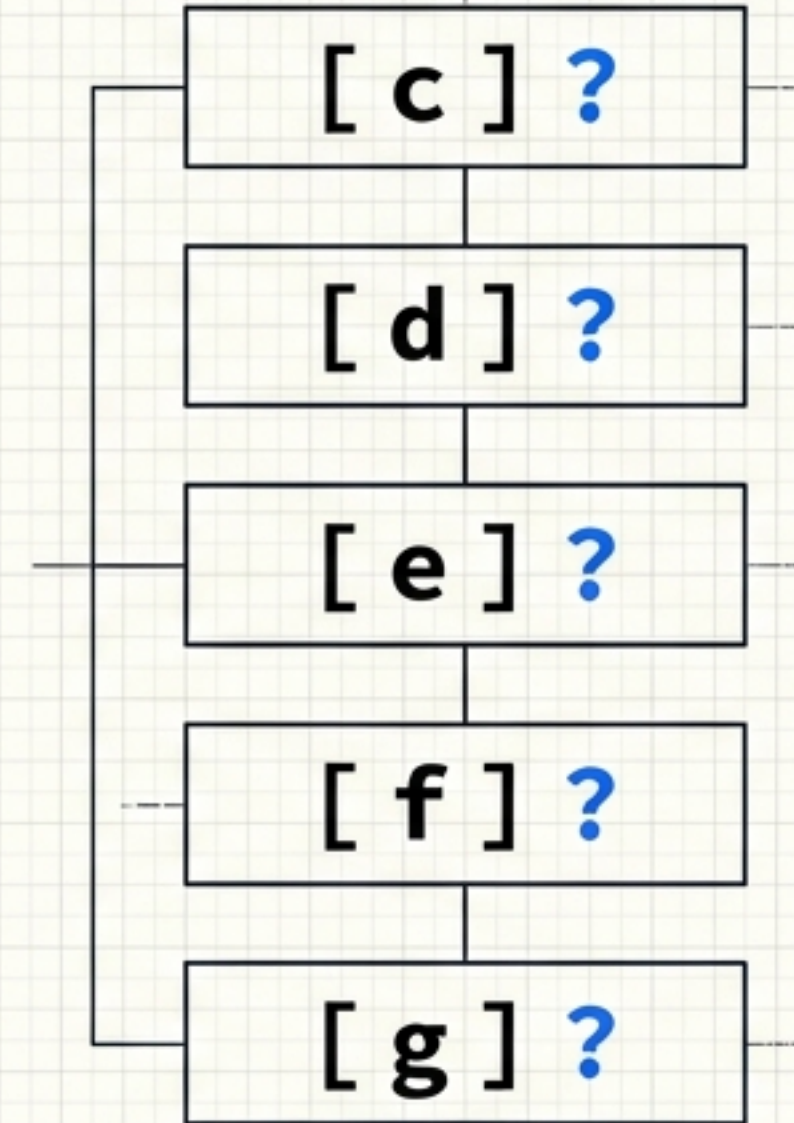
答えを暗記するのではなく、「導き出す思考プロセス」を身につけるための特別スライド。

本資料のミッション

設問2では、制御プログラムの処理フロー（図3）の空欄を埋めることが求められます。



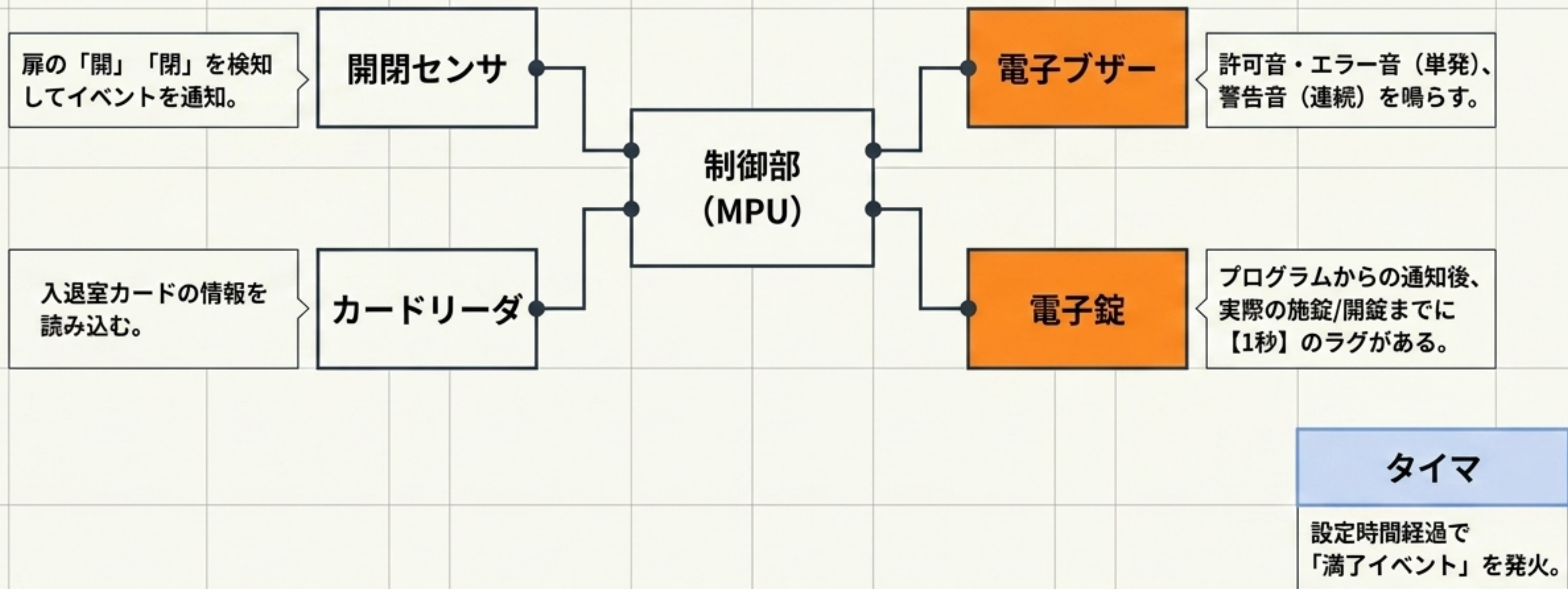
- ✗ 解答をそのまま提示することはしません。
- 仕様書（本文）のどこに着目し、どう論理を繋げばよいのか、極上のヒントと考察のフレームワークを提供します。



物理的な「扉の動き」と、プログラム上の「処理」をいかにリンクさせるかが攻略の鍵。

ヒントの源泉①：物理デバイスの役割を整理する

プログラムは物理的なハードウェアを制御するために存在します。
何が・どう動くのかを把握しましょう。



ヒントの源泉②：動作仕様をタイムラインで視覚化

複雑な箇条書きの仕様は、時系列のフローに変換することで全体像が見えます。

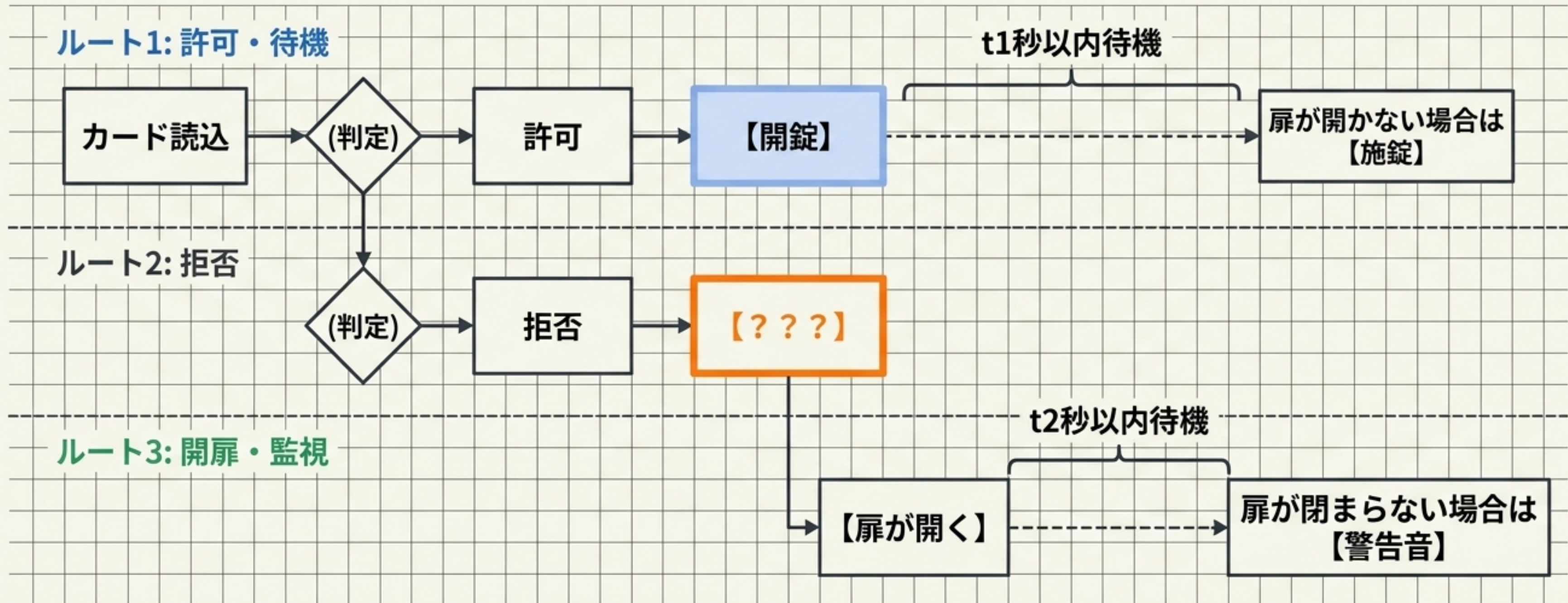
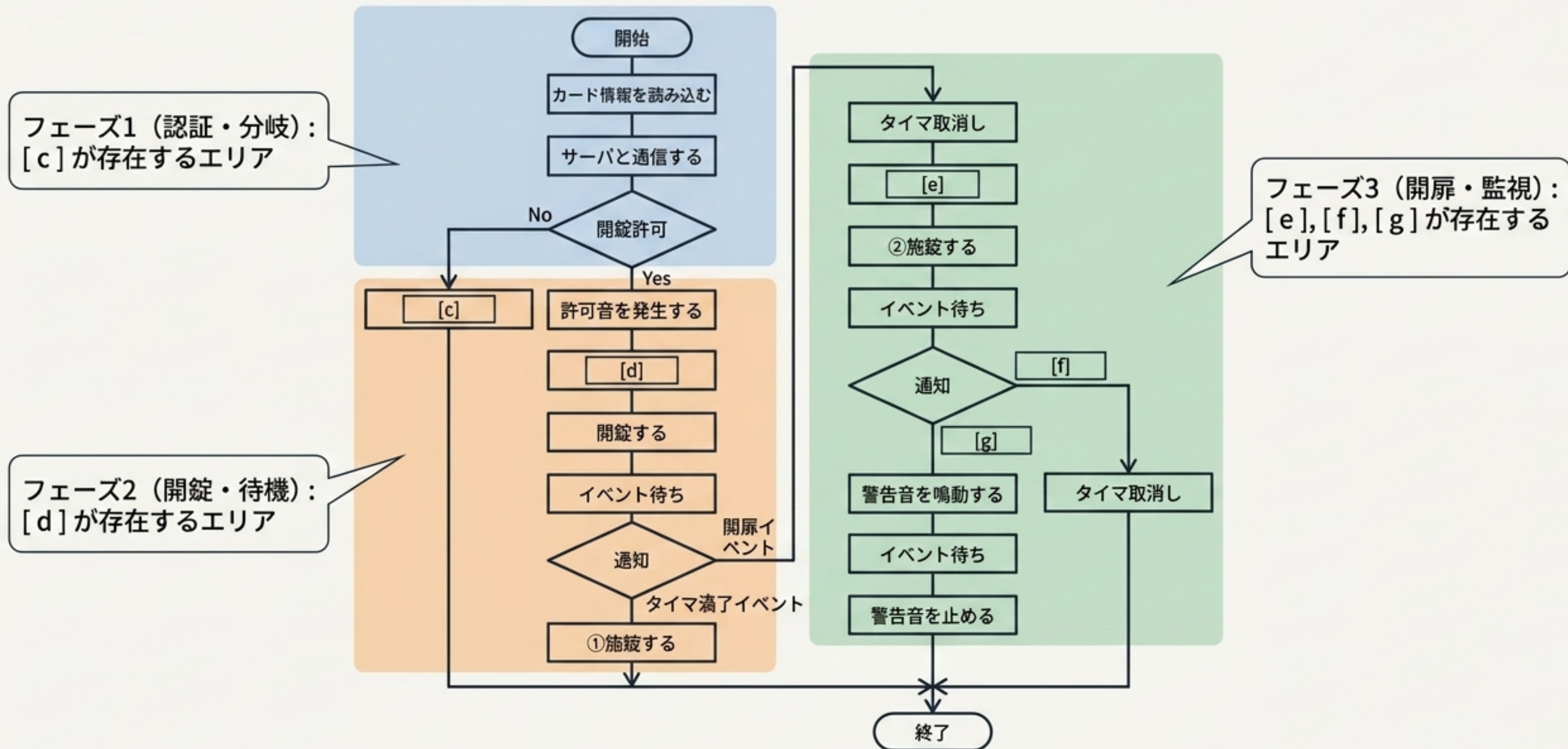
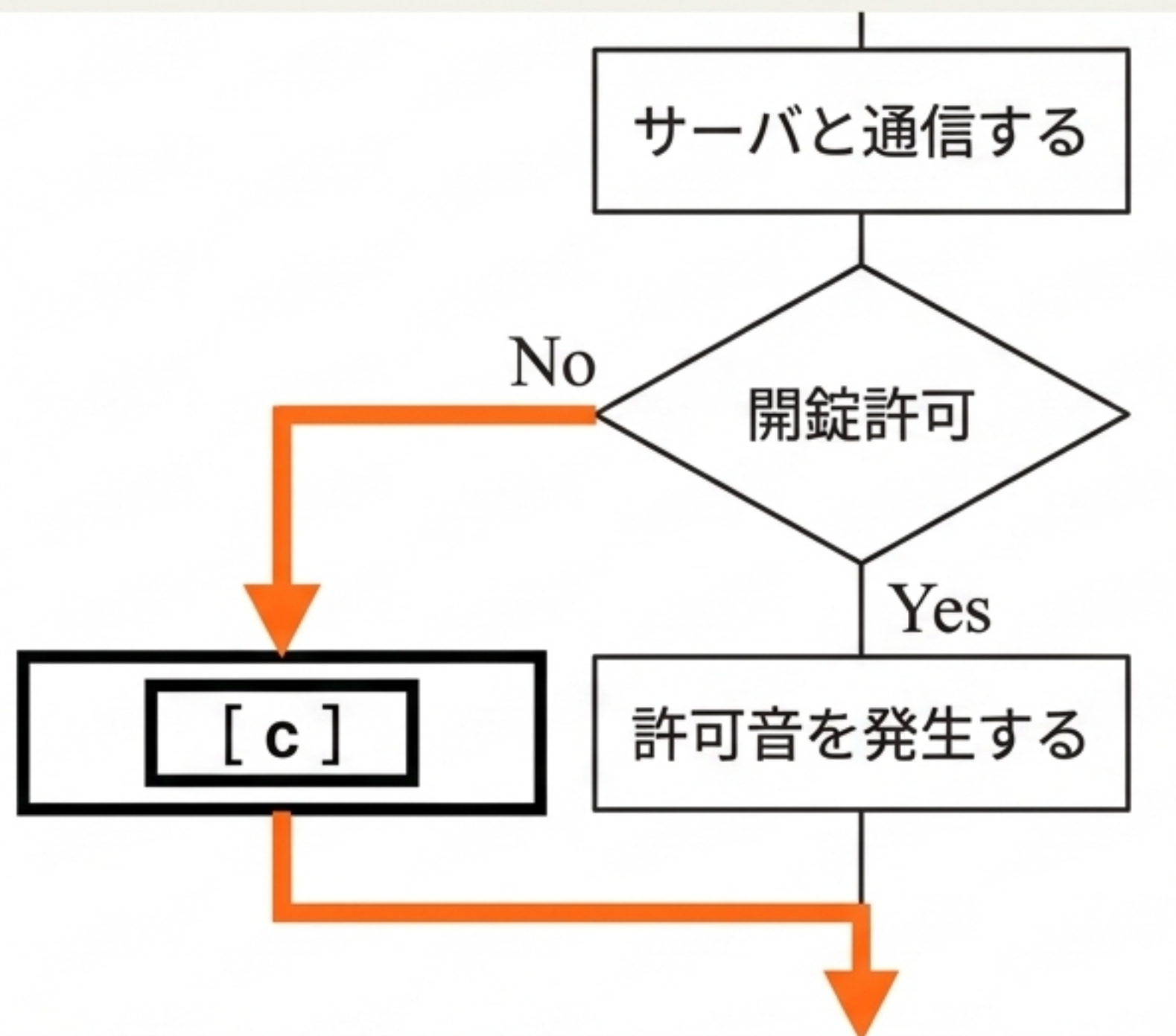


図3：制御プログラムを3つのフェーズで捉える

空欄を単発で見のではなく、プログラムが今どのフェーズにいるのかを意識してください。



空欄 [c] のヒント：認証に失敗した物理的な結果



注目すべき仕様：〔電子扉システムの動作〕 (3)

サーバと通信した結果、開錠許可が「No」だったルートです。

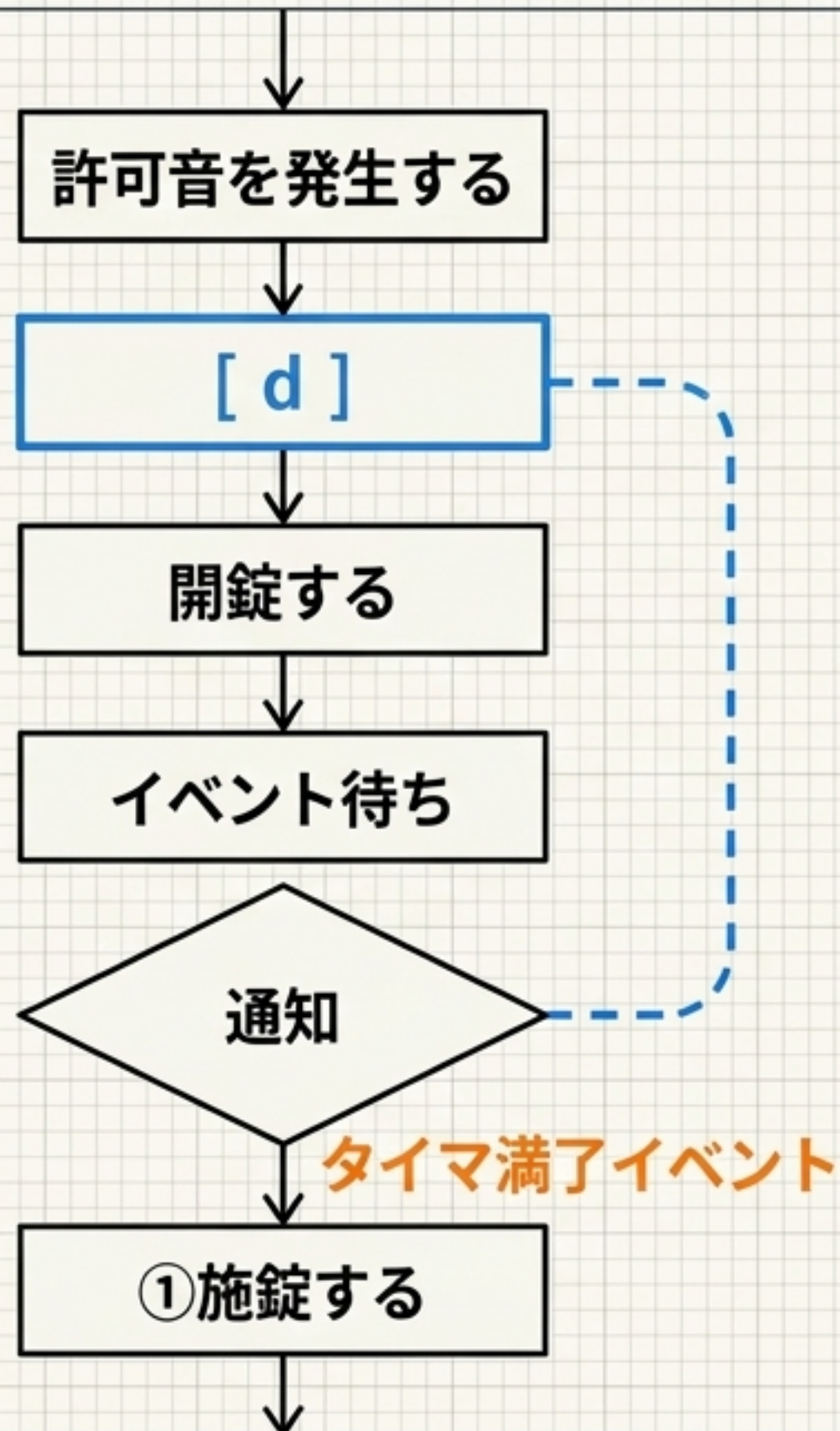
考察のヒント

- ・ 扉を開けられない利用者に、システムはどうやって「NG」であることを伝えるべきでしょうか？
- ・ 使えるハードウェアの中に、音を出す装置がありましたね。どの種類の音を鳴らすのが適切ですか？

攻略の鍵：2つのタイマ（t1とt2）の役割の違い

	t1 タイマ	t2 タイマ
目的	開錠後、放置されるのを防ぐ	開扉後、開けっ放しを防ぐ
計測開始の瞬間	開錠した直後	扉が開いた直後
期待する正常動作	扉が「開く」こと	扉が「閉まる」こと
時間切れ時のペナルティ	強制的に「施錠」する	「警告音」を鳴らし続ける

空欄 [d] のヒント：開錠直後の時限爆弾

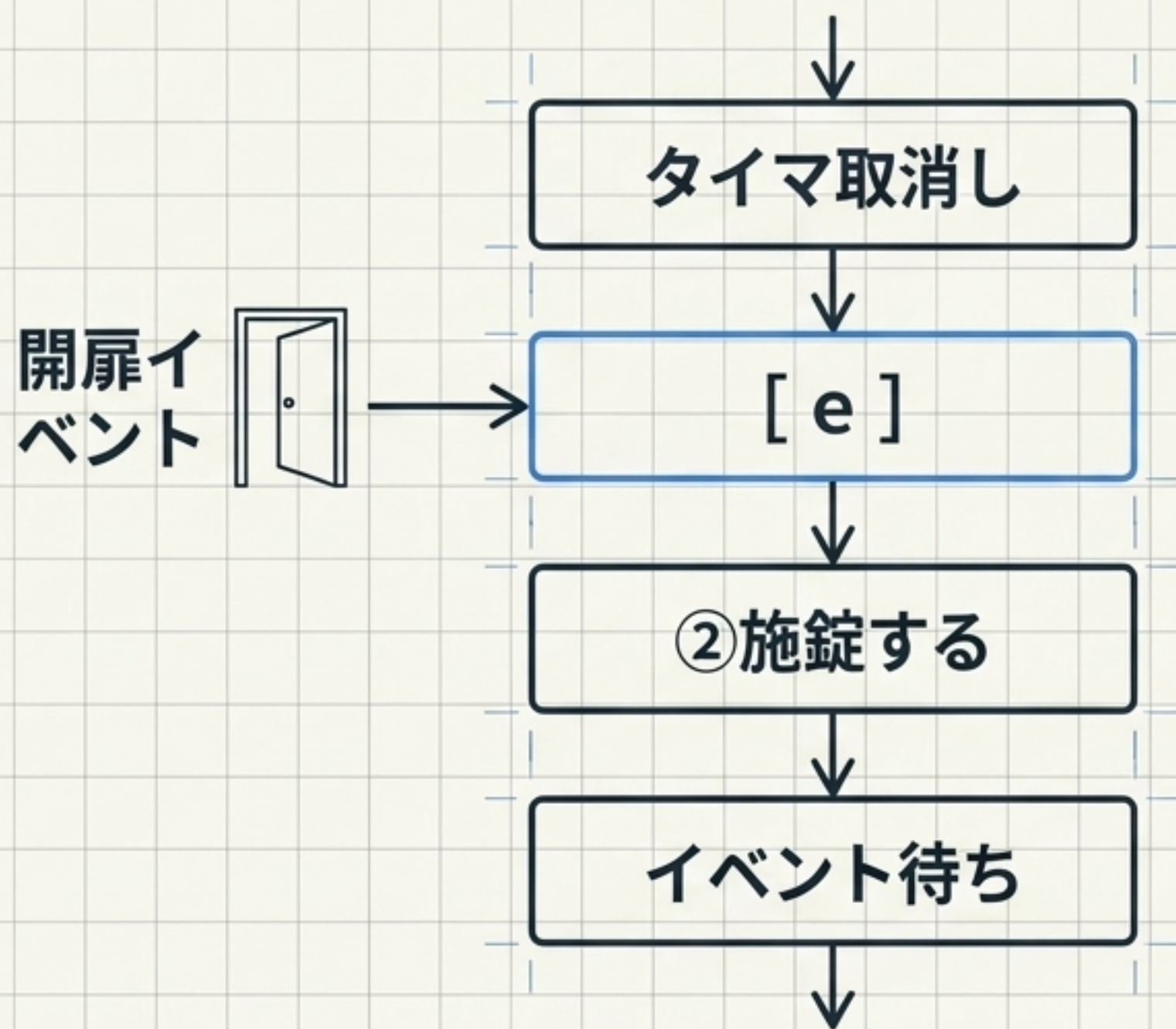


注目すべき仕様：〔電子扉システムの動作〕 (2)

「許可音を発生する」と「開錠する」の間に挟まる処理です。

- 開錠した直後から「ある時間」の計測が始まらなければなりません。
- その時間が満了すると「①施錠する」が実行されます。
- ここでセットすべき時間は、 t_1 でしょうか？それとも t_2 でしょうか？（スライド7の表を思い出してください）

空欄 [e] のヒント：扉が開いた後の新たな監視



Hint Box

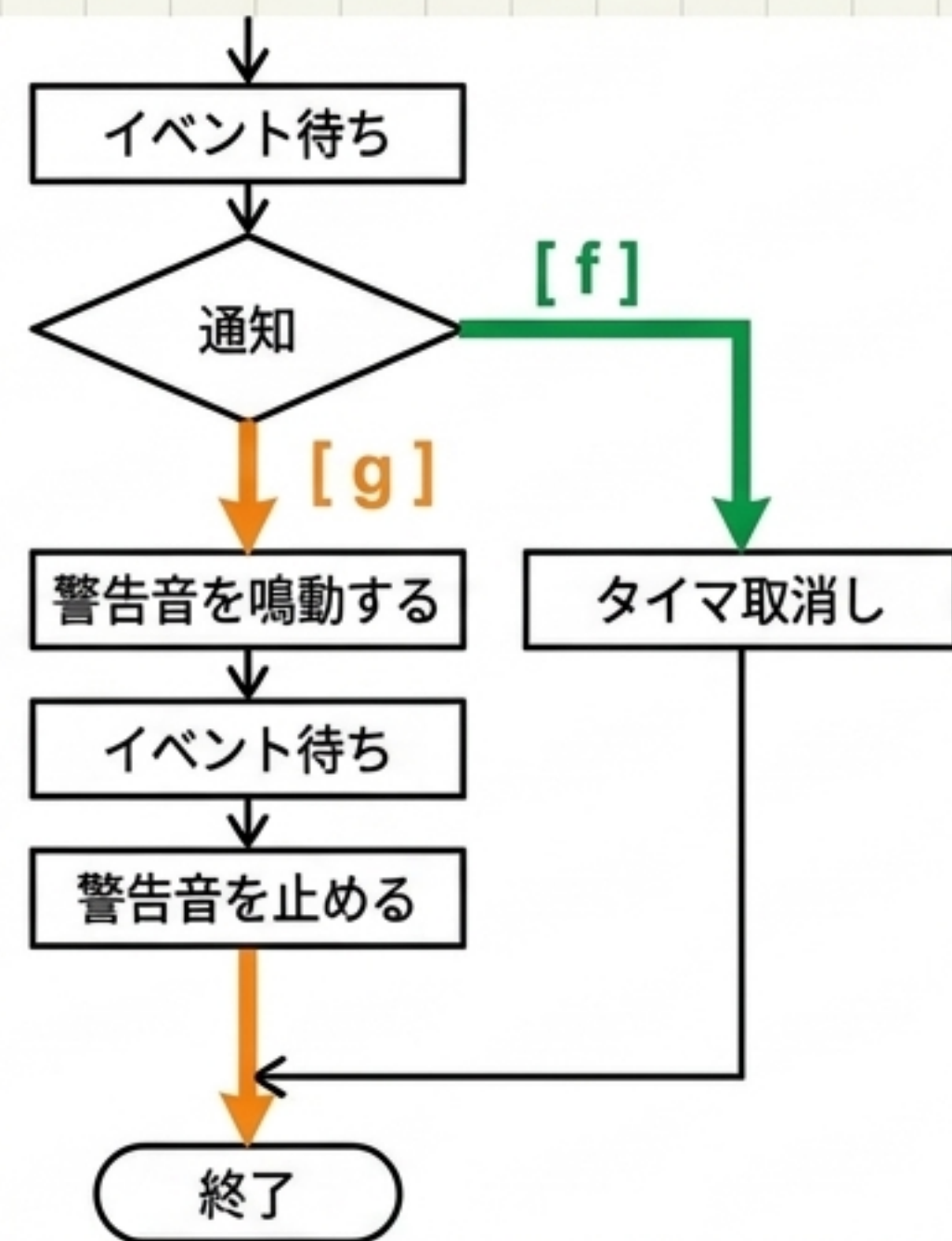
注目すべき仕様：「電子扉システムの動作」(4)

開扉イベントが発生し、不要になった前のタイマを取り消した直後の処理です。

- 扉が開いた状態になりました。次にシステムが監視しなければならないのは「開けっ放し（閉め忘れ）」です。
- 警告音を鳴らす条件となる時間はどれでしたか？
- 「イベント待ち」に入る前に、タイマにどの時間をセットすべきでしょうか？

空欄 [f] [g] のヒント：イベントの正体

[e] でタイマをセットし、扉が開いた状態（施錠済み）でシステムは「イベント」を待ち受けています。ここでのイベント候補は「閉扉イベント」か「タイマ満了イベント」の二択です。



✓ ルートF（右へ進む）
タイマを取消して終了する
＝無事にミッションクリアした状態。

✗ ルートG（下へ進む）
警告音を鳴動する
＝制限時間をオーバーした状態。

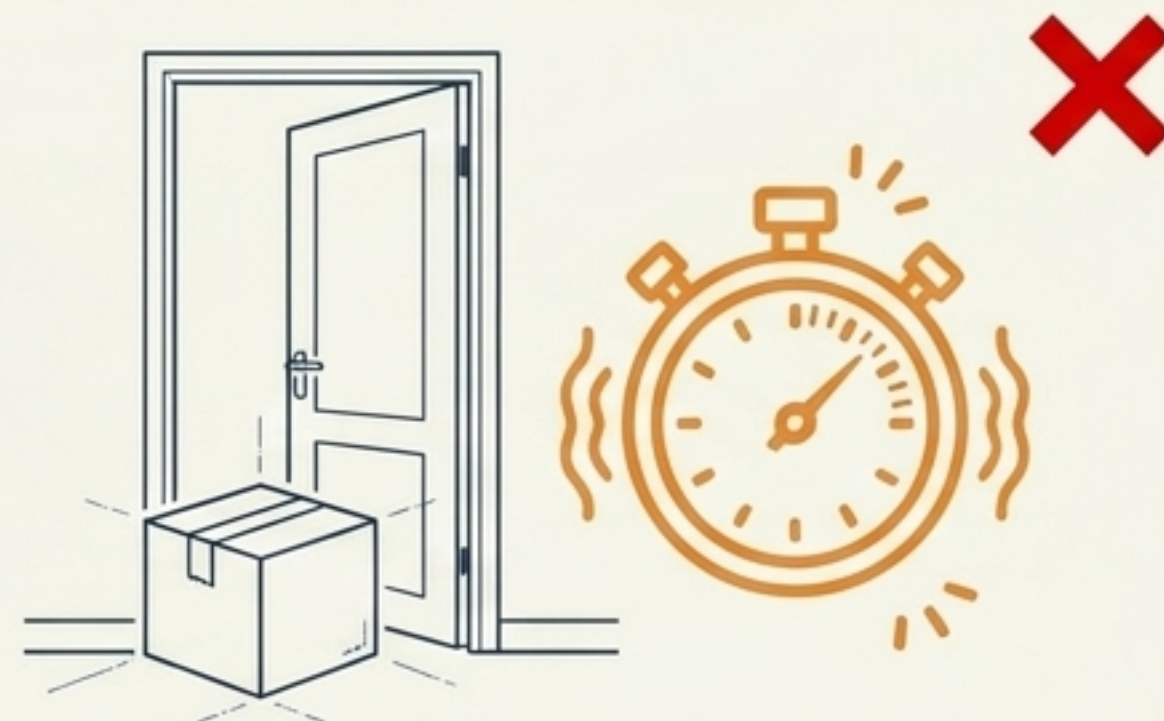
アクションと結果を繋ぎ合わせる

● ルートF の考察



警告音が鳴る前にタイマを取り消せる
(=安心できる) のは、センサーが何を
検知した時ですか？
扉がどうなれば良いのでしょうか？

● ルートG の考察



警告音が鳴り始める直接のトリガー
(通知) は何ですか？
設定した時間が「どうなった」ことを
プログラムが検知した瞬間でしょうか？

発展：このフローチャートに潜む「罠」

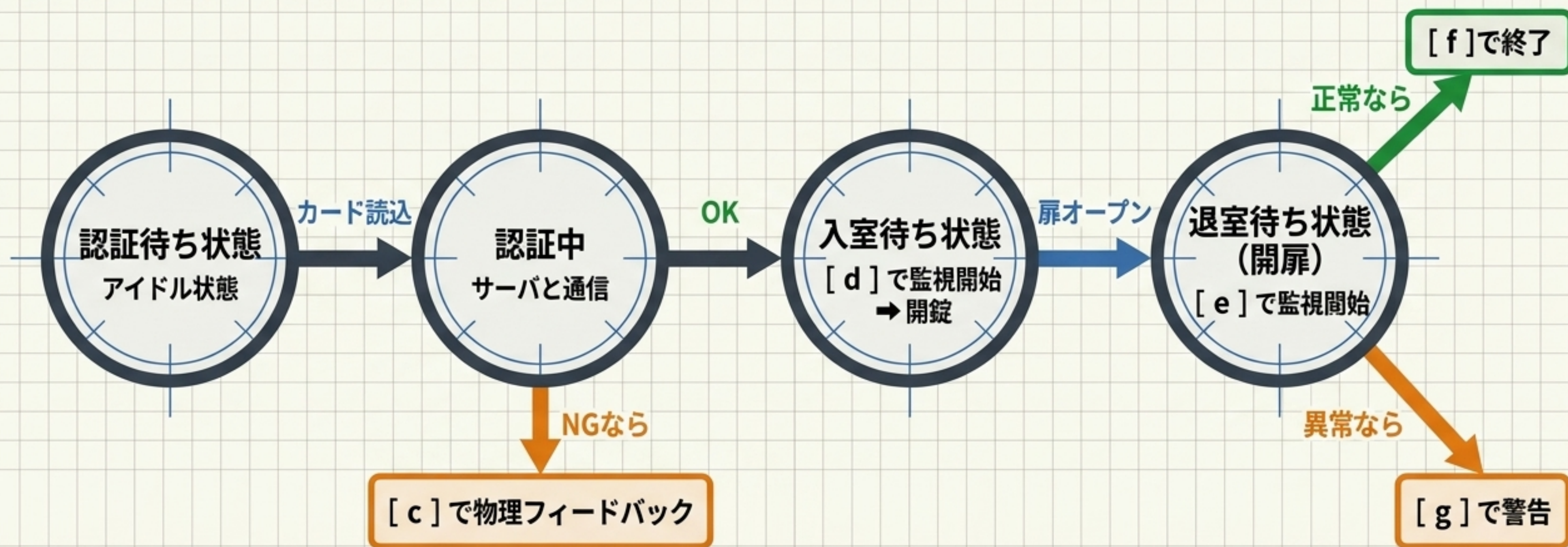
設問3に繋がる重要な仕様にも触れておきましょう。



Insight

- ・プログラムが「①施錠する」を命令しても、物理的な電子錠が動くまでに「1秒」の隙（ラグ）があります。
- ・この1秒間は、プログラム上は施錠処理中ですが、物理的にはまだ扉が開けられる状態です。
- ・※このズレが、システムの重大な不具合を引き起こす原因となります。

全体統合：システム状態とフローの完全対応



全体の構造（アーキテクチャ）が見えれば、空欄は論理的な必然として埋まります。

最終確認用チェックリスト

自分の導き出した答えが正しいか、以下のテストで検証してください。

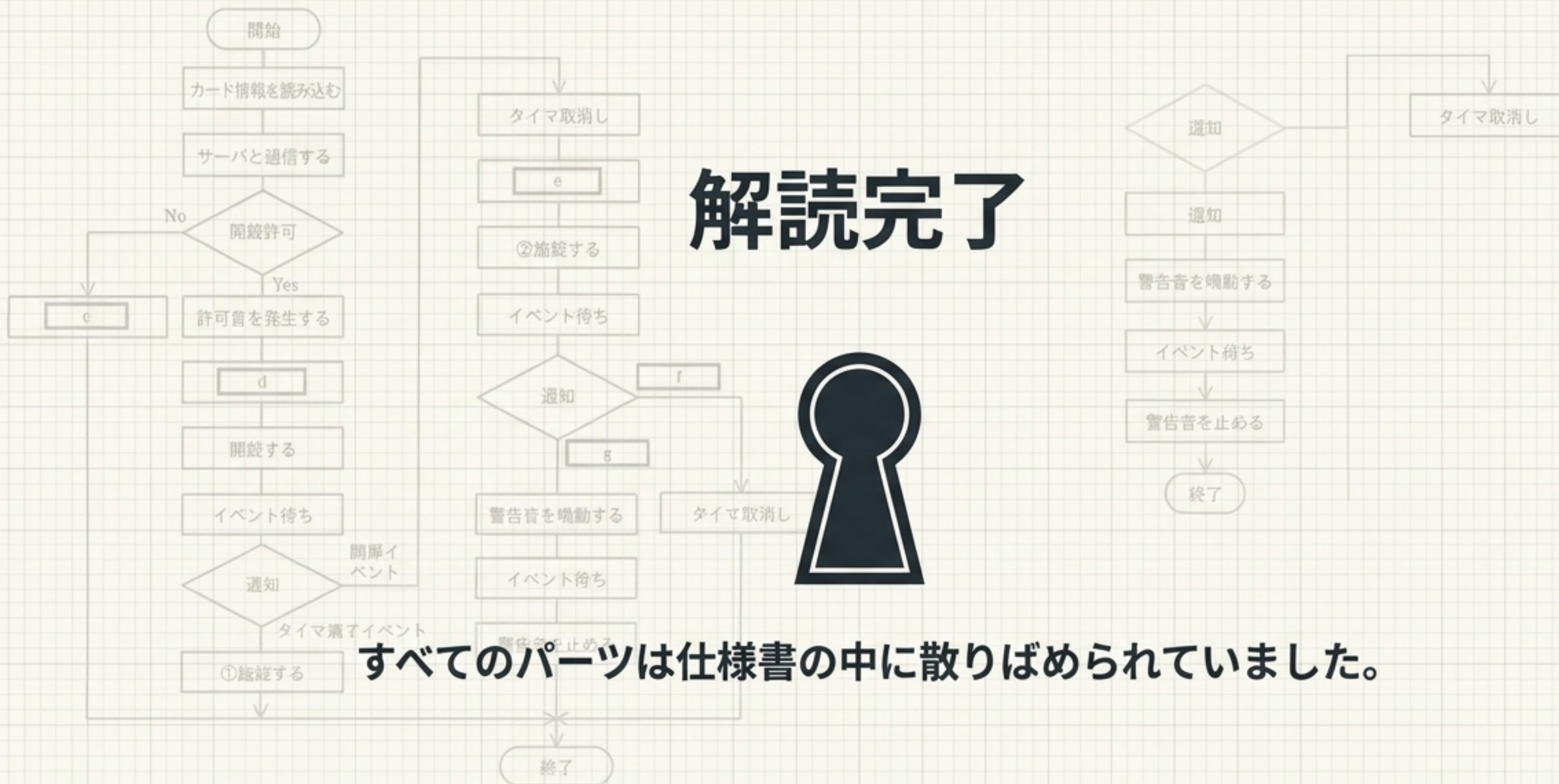
QA Testing Protocol

- ☐ [c] には、利用者にエラーを知らせる「物理的アクション」が入っているか？
- ☐ [d] には、開錠後から「自動施錠」までの時間を設定しているか？
- ☐ [e] には、開扉後から「警告音」までの時間を設定しているか？
- ☐ [f] には、警告音を鳴らさずに済む「正常な動作」のイベントが入っているか？
- ☐ [g] には、時間をオーバーしたことを示すイベントが入っているか？

解読完了



すべてのパーツは仕様書の中に散りばめられていました。



知識の暗記ではなく、与えられたルール（仕様）をモデリングし、論理的なパズルを解く力が応用情報技術者に：は求められます。この思考プロセスを武器に、本番に挑んでください。