

## 第6章 制御アーキテクチャ

---

---

# 命令実行サイクルの制御

- ▶ ノイマン型コンピュータでは,
  - ▶ メインメモリに格納されている命令を取り出す. (命令フェッチ)
  - ▶ 取り出した命令を解読する. (命令デコード)
  - ▶ 解読した情報に基づいて, 所定の処理を実行する. (実行)という動作を繰り返して行うことにより, 命令が順次実行される.



コンピュータの内部では, どのようにして,  
「命令フェッチ」, 「命令デコード」, 「実行」を実現しているのか?



プロセッサ内に分散実装してある制御回路が,  
時間とともに変化する制御信号を, 各論理回路に与えることにより,  
「命令フェッチ」, 「命令デコード」, 「実行」を実現している.

# 制御アーキテクチャ

---

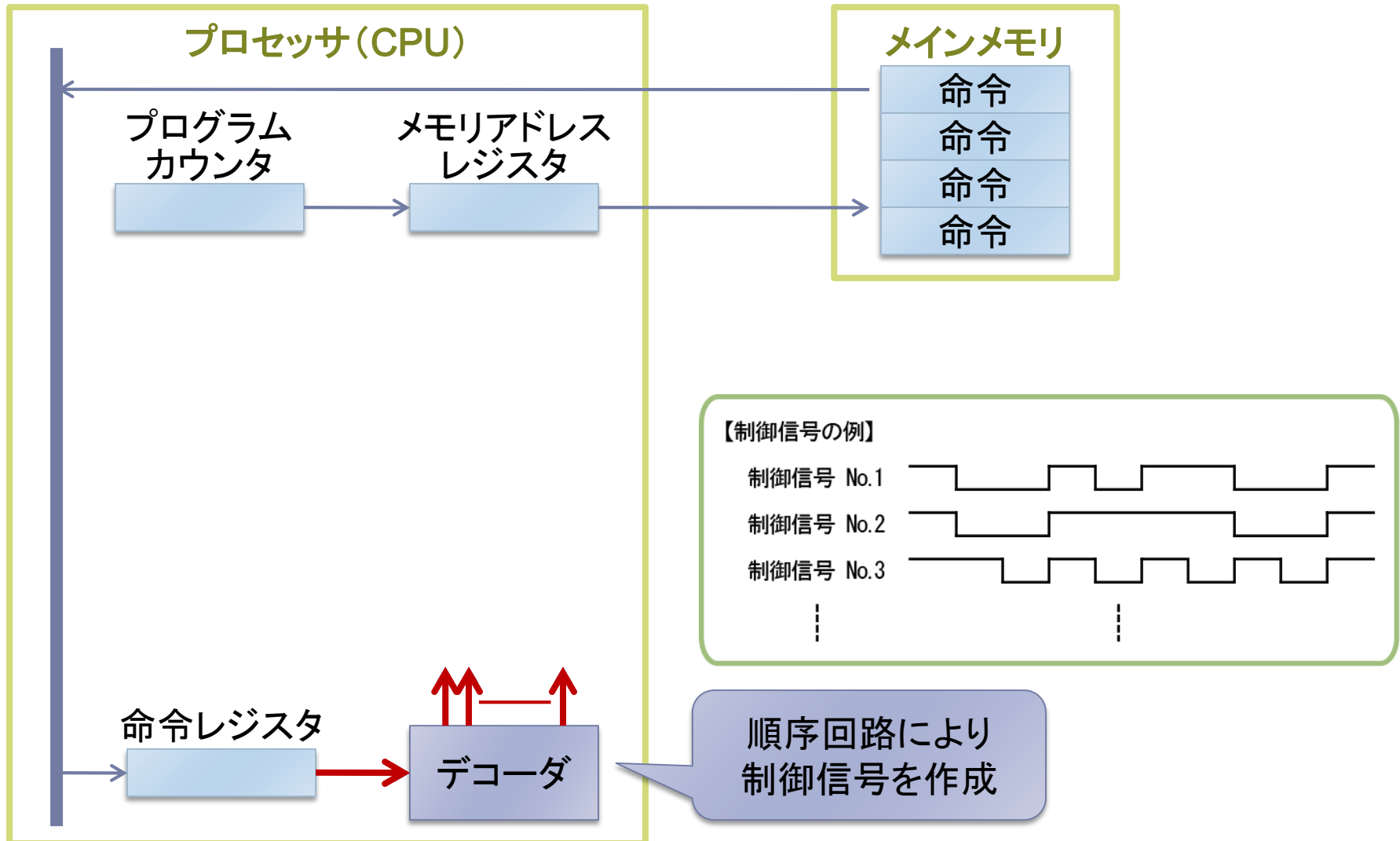
- ▶ 制御アーキテクチャ
  - ▶ 制御信号をどのように生成するのか, という指針や制御方式のこと.
- ▶ 制御方式
  - ▶ 現代の制御方式は, 以下の2つに大別される.
    - ▶ ワイヤードロジック制御方式 (wired-logic control)
    - ▶ マイクロプログラム制御方式 (micro-programmed control)

# ワイヤードロジック制御方式

---

- ▶ ワイヤードロジック制御方式
  - ▶ ハードウェアすなわち順序論理回路で、制御信号を生成する方式.
- ▶ 特徴
  - ▶ 長所
    - ▶ 高速制御が可能.
  - ▶ 短所
    - ▶ 制御機構の設計が複雑で修正が困難.

# ワイヤードロジック制御方式の概念図

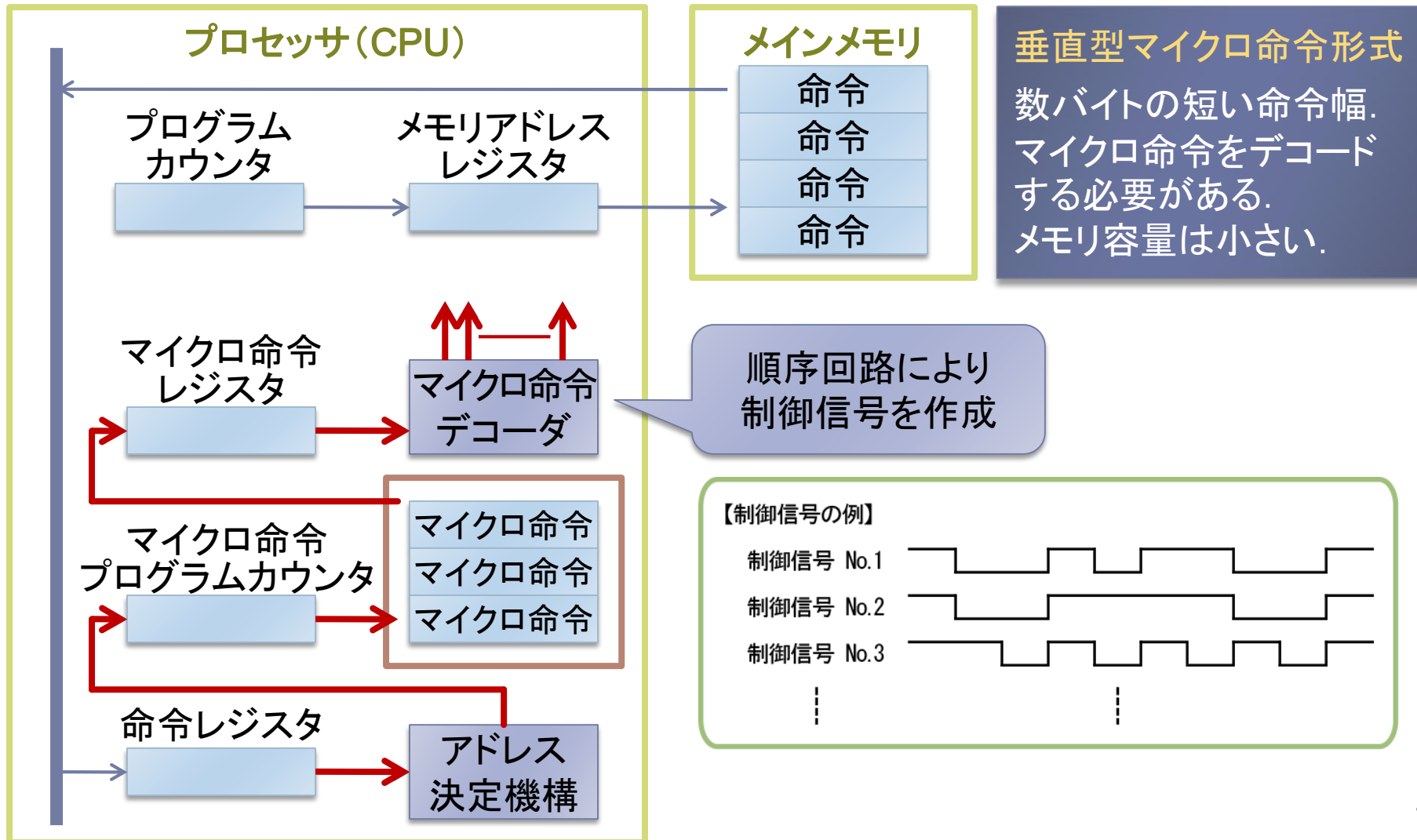


# マイクロプログラム制御方式

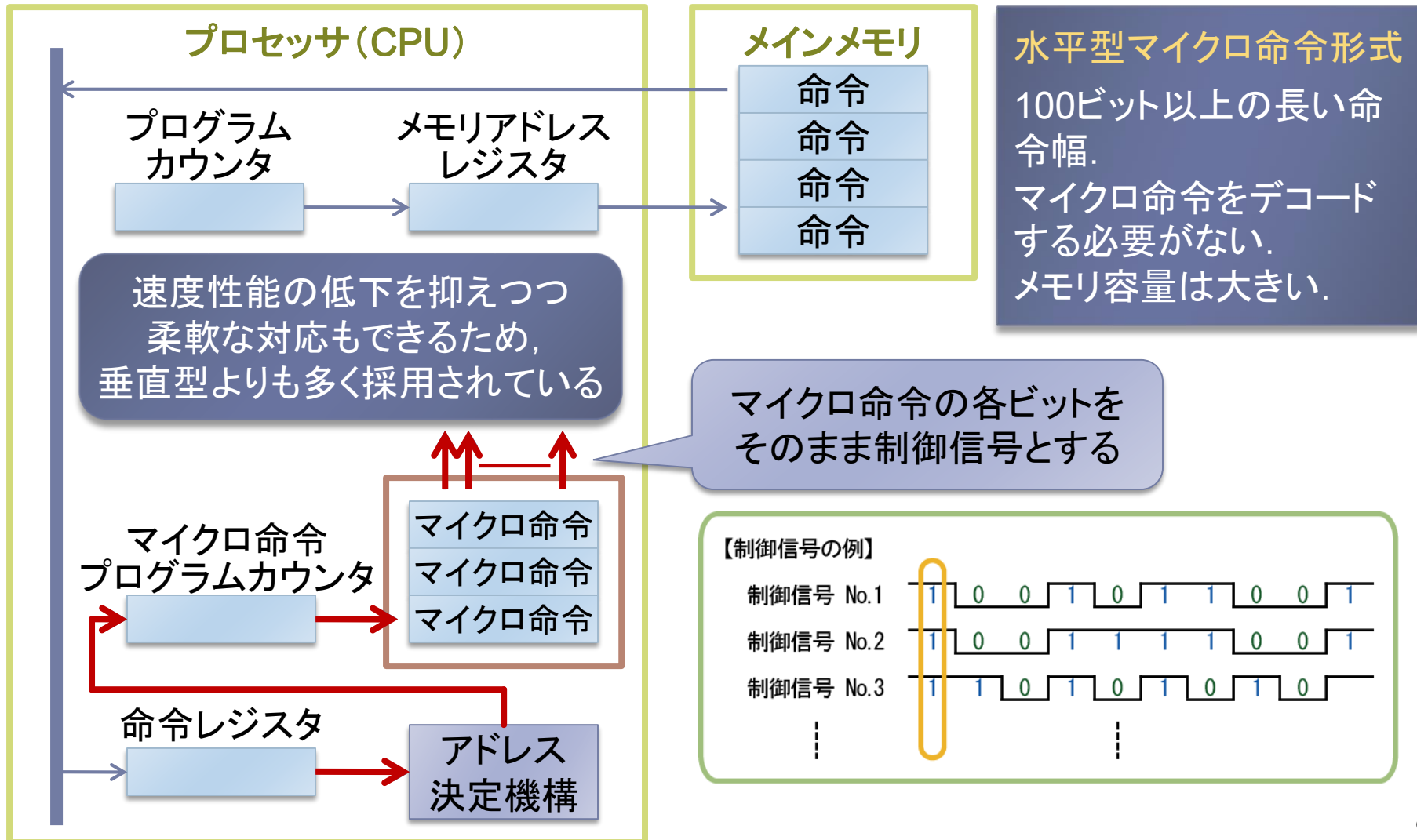
---

- ▶ マイクロプログラム制御方式
  - ▶ マイクロプログラム(ファームウェア : firmware )という一種のプログラムを介在させることによって, 制御信号を生成する方式.
  - ▶ 垂直型マイクロ命令形式と水平型マイクロ命令形式の2形式がある.
- ▶ 特徴
  - ▶ 長所
    - ▶ 制御機構の設計や修正が容易.
  - ▶ 短所
    - ▶ 制御メモリ(マイクロ命令格納用)とメインメモリ(マシン命令格納用)へのアクセスが必要となるため, ワイヤードロジック方式よりも一般に低速.

# マイクロプログラム制御方式の概念図 (垂直型マイクロ命令形式)



# マイクロプログラム制御方式の概念図 (水平型マイクロ命令形式)





## 制御方式の比較

---

- ▶ ワイヤードロジック制御方式
  - ▶ 制御機構の設計や修正は容易ではないが高速な制御が可能であるため、簡素な命令セットアーキテクチャの実現に適している。
  - ▶ RISCは、ワイヤードロジック制御方式を採用することが多い。
- ▶ マイクロプログラム制御方式
  - ▶ ファームウェアによって命令機能の拡張や改良などに柔軟に対応可能なため、複雑な命令セットアーキテクチャの実現に適している。
  - ▶ CISCは、マイクロプログラム制御方式を採用することが多い。

# 演習問題

---

## ▶ 問題1

- ▶ ワイヤードロジック制御方式とマイクロプログラム制御方式を比較説明せよ.

## ▶ 問題2

- ▶ マイクロプログラム制御方式における垂直型マイクロ命令形式と水平型マイクロ命令形式を比較説明せよ.

## 第9章 割り込みアーキテクチャ

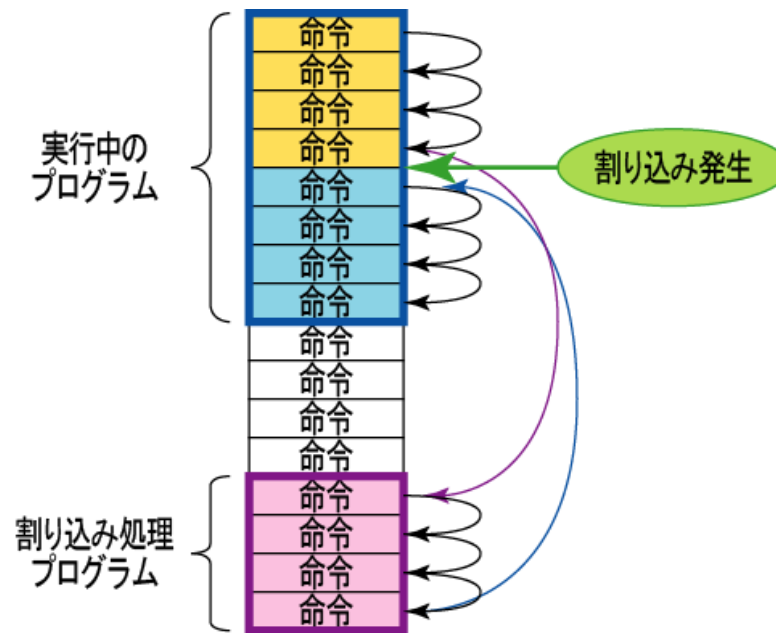
---

---

# 割り込みとは

## ▶ 割り込み

- ▶ 実行中の処理を一度停止して、他の処理を行った後に再開する機能のこと.
- ▶ プログラム実行中に割り込みが生じると、実行中のプログラムをいったん中断し、割り込み処理プログラムを実行する. そして、割り込み処理プログラムが終了した後、中断していたプログラムの実行を再開する.



## 割り込みの分類

---

- ▶ 内部割り込み
  - ▶ 内部装置(プロセッサとメインメモリ)が発生する割り込み.
  - ▶ ソフトウェア的要因によるのでソフトウェア割り込みともいう.
  - ▶ “明示的な割り込み”と“暗黙的な割り込み”がある。(後述)
- ▶ 外部割り込み
  - ▶ 外部装置(主として入出力装置)が発生する割り込み.
  - ▶ ハードウェア的要因によるのでハードウェア割り込みともいう.
- ▶ リセット
  - ▶ リセットスイッチによる割り込み.
  - ▶ ユーザが強制的にプログラムを中止するのに使用する.

# 内部割り込み

- ▶ 明示的(意図的)な割り込み (“トラップ(trap)”ともいう)
  - ▶ スーパーバイザコール(supervisor call)
    - ▶ ユーザプログラムからOSに制御を移す命令.
  - ▶ ブレークポイント(break point)
    - ▶ プログラムの中断をOSに知らせる命令.
- ▶ 暗黙的(非意図的)な割り込み (“例外(exception)”ともいう)
  - ▶ 演算オーバーフロー
    - ▶ ゼロを除数とする除算, 演算結果の格納装置からのあふれなど.
  - ▶ 未定義命令
    - ▶ 未定義のオペコードの使用など.
  - ▶ アクセス保護違反
    - ▶ ユーザモードでの特権命令の実行.
  - ▶ ページフォールト
    - ▶ (詳細は仮想メモリにて)

内部割り込みは,  
命令に同期して  
発生する.

## 外部割り込み

---

- ▶ ハードウェア障害
  - ▶ 電源異常, メモリ読み出しエラー, 温度異常など.
- ▶ 入出力装置
  - ▶ 入出力装置からの“動作完了”や“異常”などの状態通知.
- ▶ タイマ(timer)
  - ▶ プロセッサ内に装備するインタバルタイマからの“一定時間経過”の通知. OSが, “プロセススケジューリング”や“各装置からの応答のタイムアウトの決定”などのために用いる.

外部割り込みは,  
命令とは非同期に  
発生する.

# リセット割り込み

---

- ▶ リセット
  - ▶ 実行中のプログラムを強制的に終了する.



# 演習問題

---

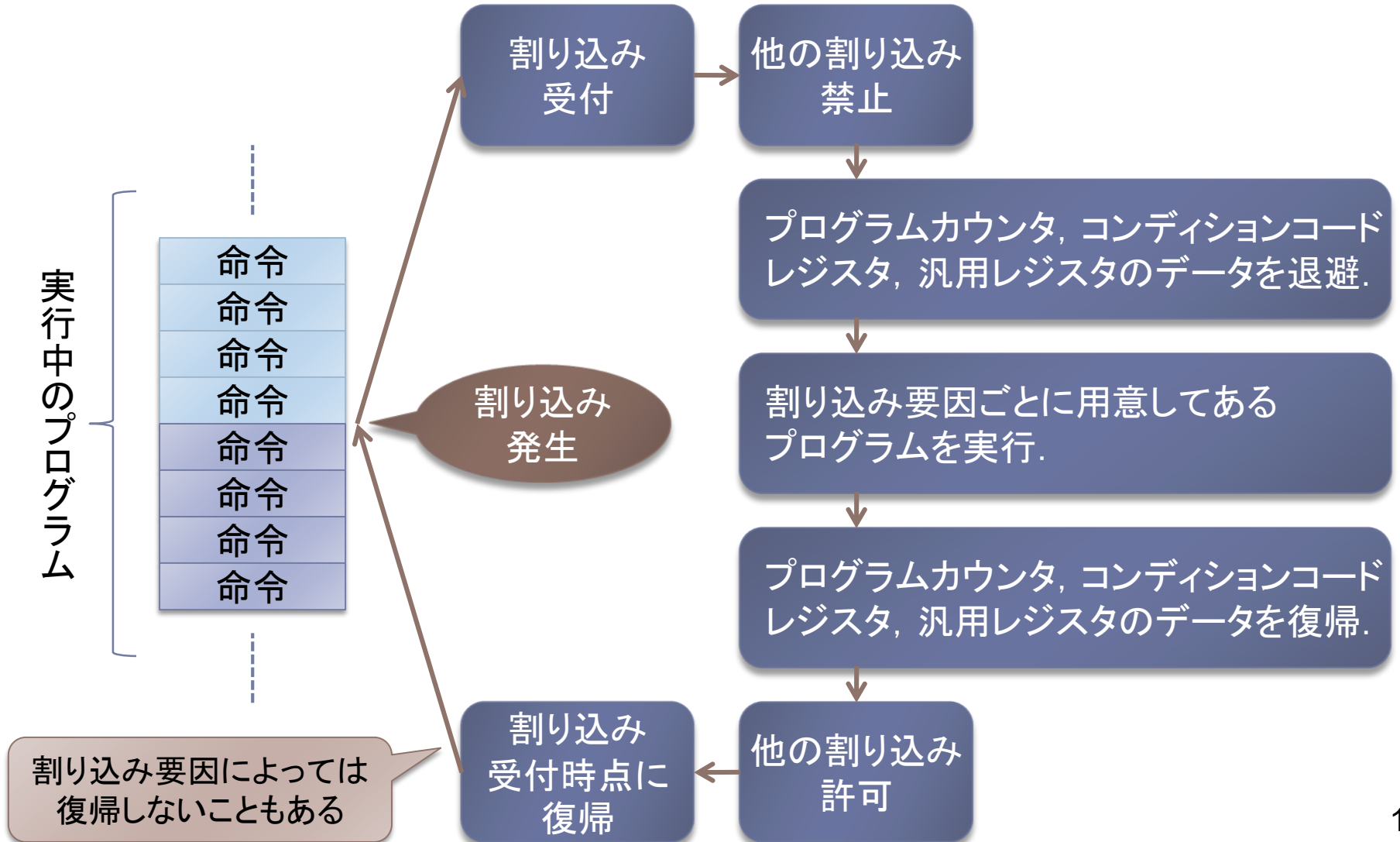
## ▶ 問題3

- ▶ 割り込みとは何か. 概略説明せよ.

## ▶ 問題4

- ▶ 内部割り込みとは, どのような割り込みか. 概略説明せよ.
- ▶ 外部割り込みとは, どのような割り込みか. 概略説明せよ.
- ▶ リセット割り込みとはどのような割り込みか. 概略説明せよ.

# 割り込み処理の流れ



## 割り込み受付のタイミングと処理後の措置

---

- ▶ 割り込み要因により、受付タイミングや割り込み処理後の措置が異なる。
  - ▶ 命令完了
    - ▶ 実行中の命令の完了まで受付を待たせる。
    - ▶ [例] 入出力割り込み, タイマ割り込み, 演算例外など。
  - ▶ 命令抑制
    - ▶ 実行中の命令をNOP (No Operation: 無操作) 命令にする。
    - ▶ [例] アクセス保護違反など。
  - ▶ 命令無効
    - ▶ 実行中の命令を中断する。命令は不実行扱いとなり, 再実行が可能。
    - ▶ [例] ページフォールトなど。
  - ▶ 命令中止
    - ▶ 実行中の命令を即時に中止し, 終了する。
    - ▶ [例] リセット, ハードウェア障害など。

# 多重レベル割り込み

---

- ▶ 多重レベル割り込み
  - ▶ 割り込み要因は多種多様であり、また、いずれも独立した事象である.
  - ▶ そのため、複数の割り込みが、同時あるいは他の割り込み処理中に発生する可能性がある.
  - ▶ 時間的に重複する割り込みが発生した場合には、多重レベル割り込み処理を行う.
  
- ▶ 多重レベル割り込み処理
  - ▶ 時間的に重複する割り込みに優先順位を付けて逐次処理する.
    - ▶ 優先度の高い割り込み処理時には、それよりも優先度が低い割り込みは禁止する.
    - ▶ ある割り込み処理時に、それよりも優先度が高い割り込みが発生した場合には、それを優先処理する.

# 割り込みの優先度

---

- ▶ 割り込みの優先度(例)
  - ▶ 割り込み処理の緊急性(必要性)によって順序付けする.
    1. リセット
    2. ハードウェア障害
    3. 命令実行例外  
(ページフォールト, アクセス保護違反, 演算例外の順番)
    4. 入出力割り込み  
(高速入出力装置, 低速入出力装置の順番)
    5. スーパバイザコールやブレークポイント命令

(実際には, 優先度はコンピュータにより異なる.)

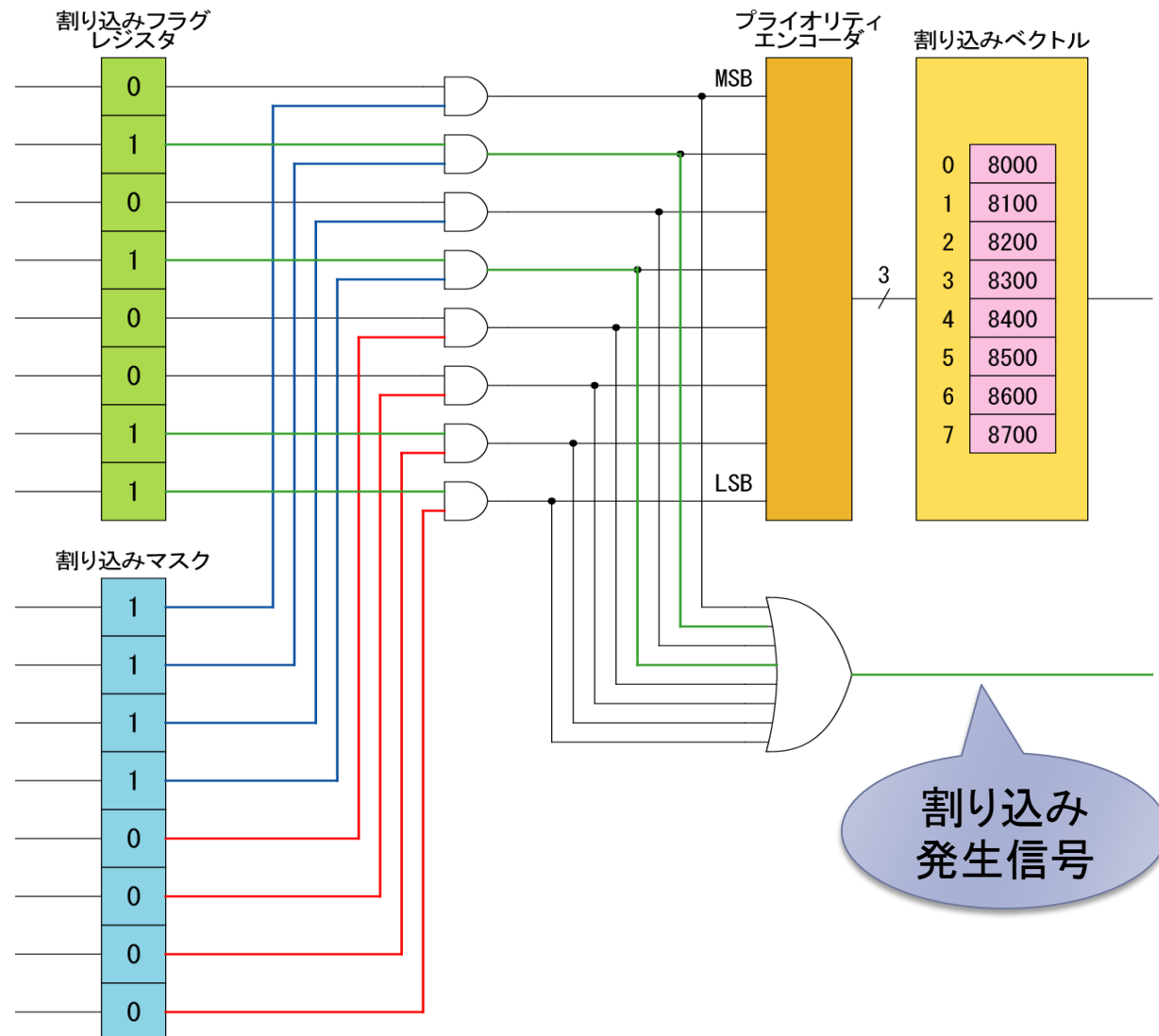
## 割り込み信号の検出

割り込み要因ごとの  
割り込みの有無を  
格納したレジスタ.  
(0:なし, 1:あり)

割り込みフラグレジスタ  
(割り込み信号レジスタ)という。

割り込み要因ごとの  
割り込みの許可・禁止  
状況を格納したレジス  
タ。(0:禁止, 1:許可)

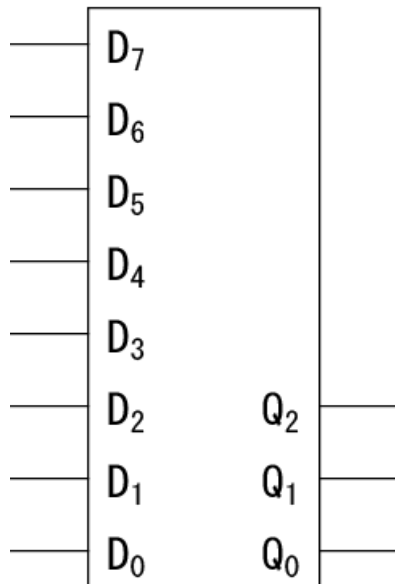
## 割り込みマスクレジスタ (割り込みイネーブルレ ジスタ)という.



# プライオリティエンコーダ

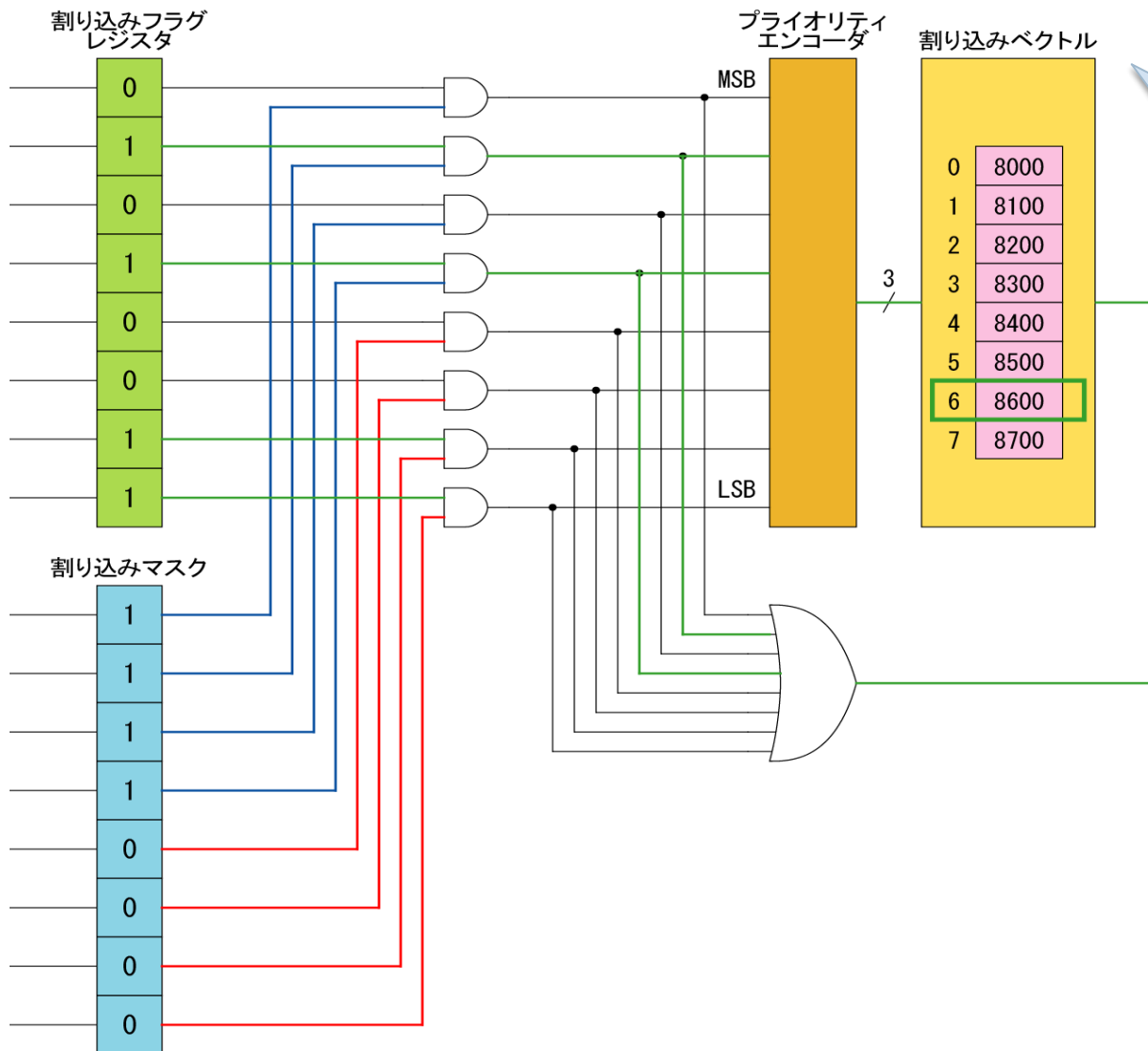
## ▶ プライオリティエンコーダ

- ▶ 8本の入力信号のうち、最上位の「1」を検出し、その位置を3ビットで出力する。



D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>	Q <sub>2</sub>	Q <sub>1</sub>	Q <sub>0</sub>
1	X	X	X	X	X	X	X	1	1	1
0	1	X	X	X	X	X	X	1	1	0
0	0	1	X	X	X	X	X	1	0	1
0	0	0	1	X	X	X	X	1	0	0
0	0	0	0	1	X	X	X	0	1	1
0	0	0	0	0	1	X	X	0	1	0
0	0	0	0	0	0	1	X	0	0	1
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0

# 割り込み要因の識別



割り込みに対応する  
割り込み処理の  
開始アドレスを  
割り込みベクトルという。



# 演習問題

## 問題5

- 割り込み識別機構の内部状態を以下に示す.
- 上記状態に対応する割り込み処理の開始アドレスを示せ.

