オペレーティングシステム(OS)

柴山　潔

8.　プロセス管理 (2)

- プロセス管理と割り込み(2)
- プロセス割り付けとプロセス領域(1)

（再掲）

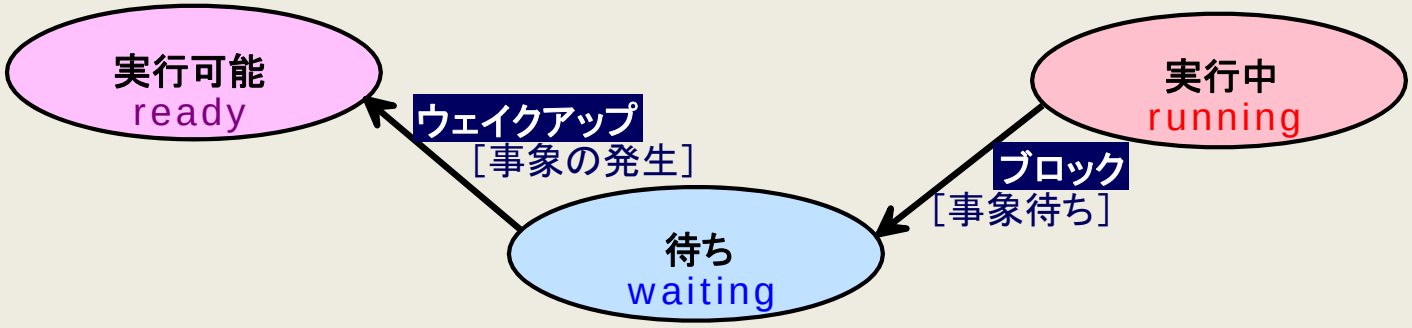
事象によるプロセス状態遷移 —ブロックとウェイクアップ—

［定義2.5］ **ブロック(block)** **事象待ち**

- プロセスの**実行中→待ち**状態遷移
  - 実行中状態のプロセスに関係する事象の発生を待つために行う

［定義2.6］ **ウェイクアップ(wake-up)**

- プロセスの**待ち→実行可能**状態遷移
  - 待ち状態のプロセスが待っている事象の発生による，そのプロセスの待ち→実行可能状態遷移



【まとめ】割り込みによるプロセス状態遷移 —割り込みと事象との関係による分類— (1)

- 「割り込みによって起動するOSのプロセス管理操作における(ユーザ)プロセスの状態遷移」について,「割り込みと事象との関係」を指標にして分類&時間経過順で整理

(A) ブロック割り込み

- 実行中(状態)のプロセスがブロック(＝実行中→待ち状態遷移)する ＝ 当該割り込み(例:SVCの実行)が「ブロック(事象待ち)する要因」(＝ブロック要因)そのものである割り込み
- 当該プロセス自身が割り込み要因である内部割り込みの大半はブロック割り込み

(1) 割り込み要因となる実行中状態プロセス自身はブロック(＝実行中→待ち状態遷移)

(2) プロセススケジューリング&プロセスディスパッチによって, 実行可能状態プロセスのうちから選定するプロセスをディスパッチ(＝実行可能→実行中状態遷移)

ブロックとブロック割り込み

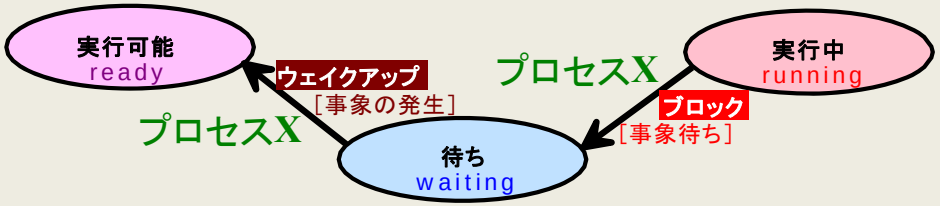
（厳密には）管理手順の1.において，割り込み要因が，

(1) 命令実行例外（の大半，代表例：ページフォールト）：OSへの「割り込み要因の解消処理」（＝OSのシステムサービス）の依頼
(2) SVC：OSの（無条件）呼び出し；
(3) ブレークポイント：OSへの「プログラムの中断点」の通知の場合

↓

- 割り込まれるプロセスXは，実行中状態を保持したままの“中断”ではなくて，ブロック（＝実行中→待ち状態遷移）
  - 実行中状態プロセスはなくなる

■ ブロック割り込み：ブロックを引き起こす割り込み
＝ (1)～(3)（内部割り込み）



【まとめ】割り込みによるプロセス状態遷移
—割り込みと事象との関係による分類— (2)

(B) ウェイクアップ割り込み

- 当該割り込みが「ウェイクアップする事象の発生」という要因（＝ウェイクアップ要因）そのもの ＝ その事象を待っているプロセスがウェイクアップ（＝待ち→実行可能状態遷移）する割 り込み

➢ ウェイクアップ割り込みは「待ち状態プロセスが待っている事象そのもの」

➢ 主要な外部割り込み（代表例：入出力割り込み）はウェイクアップ割り込み

(1) 割り込みそのものが事象→その事象を待っている待ち状態プロセスをウェイクアップ（＝待ち→実行可能状態遷移）

(2) (a)中断している実行中状態プロセスを再開；(b)プロセススケジューリング&プロセスディスパッチによって，中断している実行中状態プロセスを横取り（＝実行中→実行可能状態遷移），代わりに，実行可能状態プロセスのうちから選定する1プロセスをディスパッチ（ ＝実行可能→実行中状態遷移）；のどちらか

【まとめ】割り込みによるプロセス状態遷移
—割り込みと事象との関係による分類— (3)

(C) ブロック割り込み/ウェイクアップ割り込みのどちらでもない割り込み

（例）(a) タイマ割り込みの大半;(b) 一部の入出力割り込み;(c) ブロック要因やウェイクアップ要因でないプロセス間通信用SVC

- (B)の(2)((a)再開 / (b)横取り&ディスパッチのどちらか)だけが発生

（再掲）

(A) ブロック割り込み

(1) 実行中状態プロセスはブロック(＝実行中→待ち)

(2) プロセススケジューリング&プロセスディスパッチによって，ディスパッチ(＝実行可能→実行中)

(B) ウェイクアップ割り込み

(1) 待ち状態プロセスをウェイクアップ(＝待ち→実行可能)

(2) (a) 中断プロセスを再開;(b) プロセススケジューリング&プロセスディスパッチによって，中断プロセスを横取り(＝実行中→実行可能)，実行可能状態プロセスをディスパッチ(＝実行可能→実行中);のどちらか

（上級）

【まとめ】割り込みによるプロセス状態遷移
—割り込みと事象との関係による分類— (4)（注意）

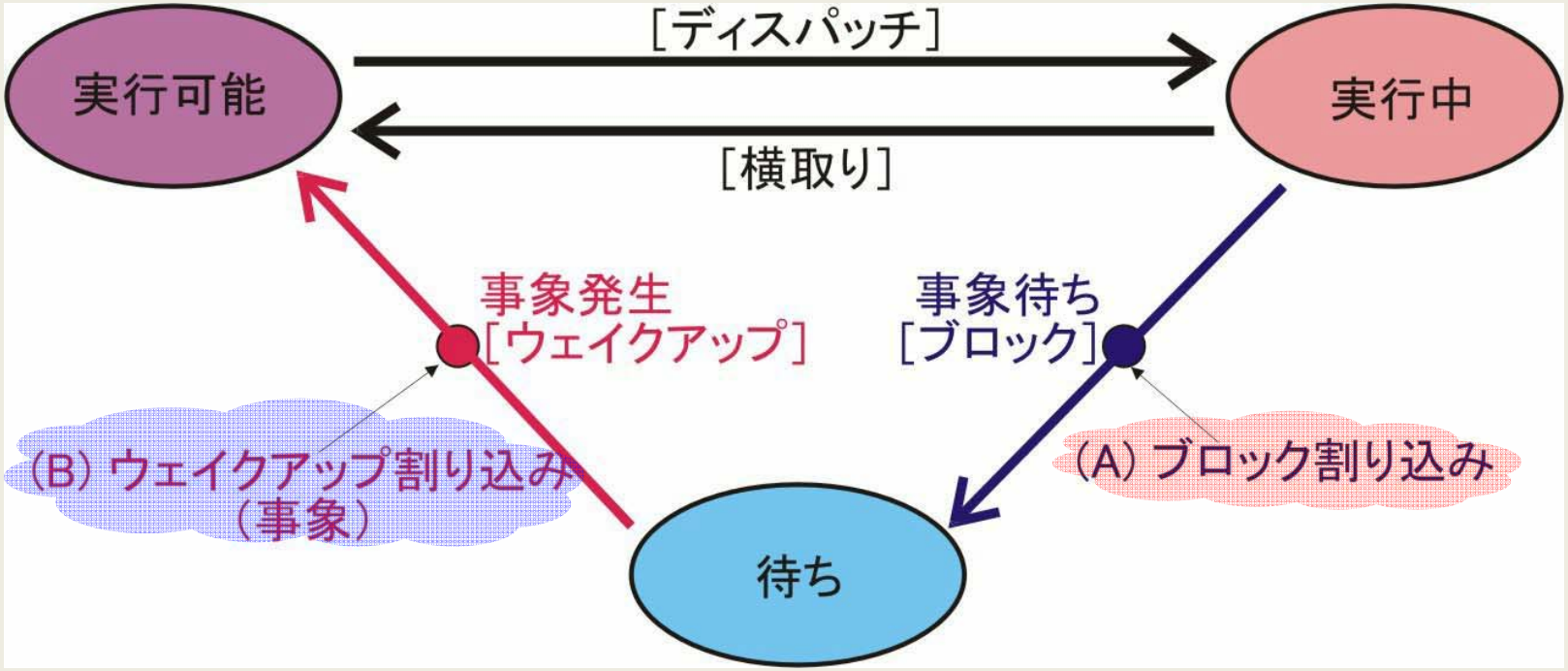
（注意）

- ブロック割り込みではなくウェイクアップ割り込みとなる内部割り込みも例外としてある → ウェイクアップ要因の例（(c)）として挙げる「プロセス通信やプロセス同期用SVC（の実行）」

（注意）

- (B) の(2)(a) は「OSによる割り込み処理」の間中断しているプロセスの再開 → 当該プロセスは「OSによる割り込み処理」の前後で状態遷移はせずに実行中状態を保持したまま
  - → 実際には，(B)(2)(a)（再開）では，プロセススケジューリングやプロセスディスパッチは不要
  - → しかし，(B)(2)(a)（再開）も「プロセススケジューリングによる当該中断プロセスの選定」および「プロセスディスパッチによる当該中断プロセスのディスパッチ（＝実行中→実行中状態遷移）」と見なして，プロセススケジューリングやプロセスディスパッチ手順に含める
  - → (A)(2)と(B)(2)とが，プロセス管理手順で実現するプロセススケジューリング（手順2）&プロセスディスパッチ（手順3）機能

事象によるプロセス状態遷移（図説）



事象によるプロセス状態遷移
—ブロック要因とウェイクアップ要因との関係—

- 「事象待ちであるブロック要因」や「事象そのものであるウェイクアップ要因」は割り込み

- ブロック割り込み（主として，内部割り込み）によるブロックでは，当該プロセス自身が実行中→待ち状態遷移を起こし，「事象の発生を待つ（＝事象待ち）」状態に

➢ ブロック要因 ＝ 種々のブロック割り込みの割り込み要因そのもの

- ウェイクアップ割り込み（主として，外部割り込み）によるウェイクアップでは，「事象の発生」によって，「その事象の発生」を待っているプロセスが待ち→実行可能状態遷移

➢ ウェイクアップ要因 ＝ 種々のウェイクアップ割り込みの割り込み要因（事象）そのもの

（上級）ブロック/ウェイクアップ要因としての割り込みの例 (1)

- 「ブロック要因としての割り込み（＝ブロック割り込み）要因」と対にして，

(a) 「入出力処理を依頼する」SVC（→ブロック割り込み）によってブロックするプロセス

- ウェイクアップするために，「『依頼した入出力操作の完了通知』としての入出力割り込み（→ウェイクアップ割り込み）」という事象を待つ

(b) 「当該ページがメインメモリ上にない」を通知するページフォールト（→ブロック割り込み）によってブロックするプロセス

- ウェイクアップするために，「『当該ページをファイル装置から（メインメモリに）読み出す入出力操作の完了通知』としての入出力割り込み（→ウェイクアップ割り込み）」という事象を待つ

(c) 「プロセス間通信（送信と受信）やプロセス同期用」SVC（→ブロック割り込み）によってブロックするプロセス

- ウェイクアップするために，「対となるSVC（→ウェイクアップ割り込み）」という事象を待つ
- ◆ ブロック要因もウェイクアップ要因もSVC（→内部割り込み）による例

（上級）

ブロック/ウェイクアップ要因としての割り込みの例 (2)

[(c)の実例]

◆プロセス間通信用の「送信（センド(send)）用SVC−受信（レシーブ(receive)）用SVC対」

- 「メッセージを受信（レシーブ）する」ためのSVC（＝受信用SVC）を実行するプロセス（＝**X**）は，「当該メッセージが未着である」場合には，ブロック（＝実行中→待ち状態遷移）

↓

➢ この場合に，当該受信用SVCはブロック割り込み/ブロック要因

- 「メッセージを送信（センド）する」ためのSVC（＝送信用SVC）を実行するプロセスは，ブロック（事象待ち）しているプロセス**X**を，「当該メッセージの到着」という事象によってウェイクアップ（＝待ち→実行可能状態遷移）

↓

➢ 当該送信用SVCはウェイクアップ割り込み/ウェイクアップ要因

事象としての割り込み —例：入出力割り込み— (1)

（例）
- OSを介して入出力装置に動作（＝入出力処理）を依頼するユーザプロセス
  - 事象 ＝入出力割り込み

- ユーザプロセス（ ＝プロセスX ）は「入出力処理の実行」を，内部割り込みであるSVCによって，OSに依頼
- このSVCを発したプロセスXは，「自身が依頼した『入出力処理（＝入出力装置の動作）』の完了」という事象待ちのために，ブロック（＝実行中→待ち状態遷移）

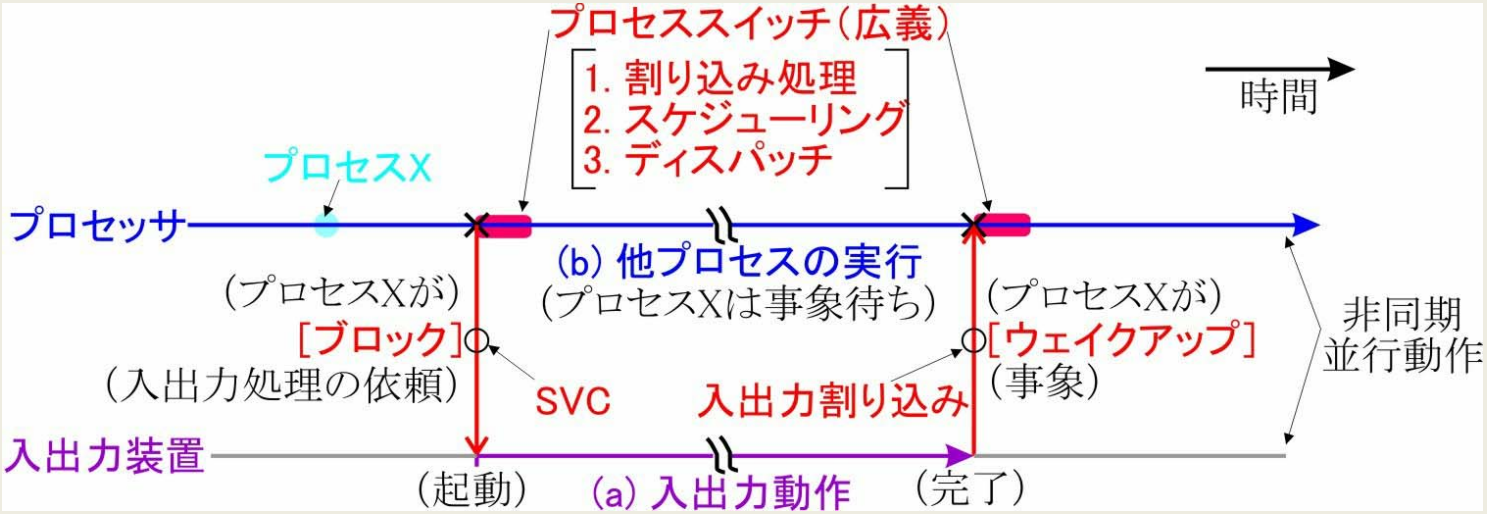
■ 入出力割り込み ＝ 代表的な外部割り込み
- 「プロセッサとは非同期に動作している入出力装置（外部装置，ハードウェア）が自分の状態（例：入出力動作の完了や異常など）を事象としてプロセッサ（内部装置）に通知する」手段
  - ウェイクアップ割り込みの代表例

事象としての割り込み —例：入出力割り込み— (2)

- 入出力処理を依頼したユーザプロセスXが入出力割り込みという事象の発生を待つ事象待ち状態の間に，
  - (a) 入出力装置はプロセスXが要求した入出力動作；
  - (b) プロセッサはプロセスX以外のプロセス；

  をそれぞれ独立して/非同期に実行

- プロセッサ（内部装置）と入出力装置（外部装置）とが相互に独立かつ並行して動作可能

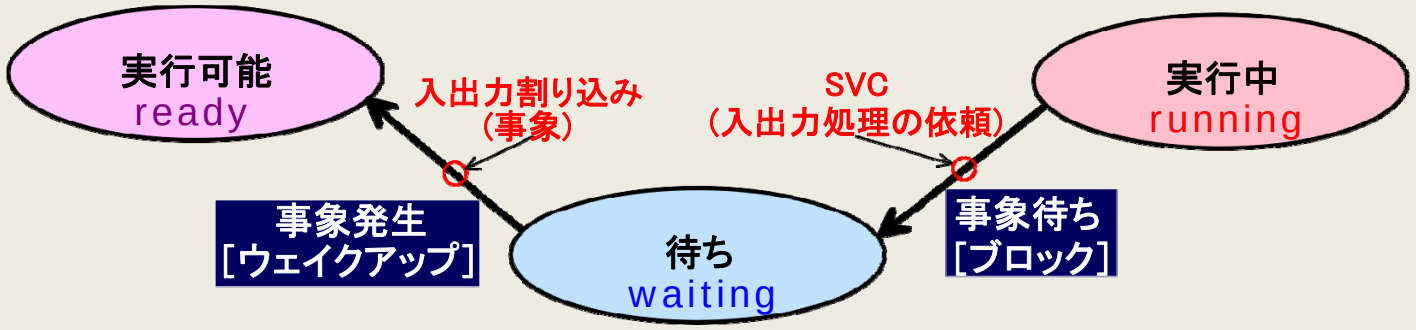


事象としての割り込み —例：入出力割り込み— (3)

（事象としては）

■ 入出力割り込み ＝ ウェイクアップ割り込み

- 「それぞれ非同期動作している入出力装置（外部装置）からプロセッサ（内部装置）へのウェイクアップという同期動作の要求」という事象の代表例



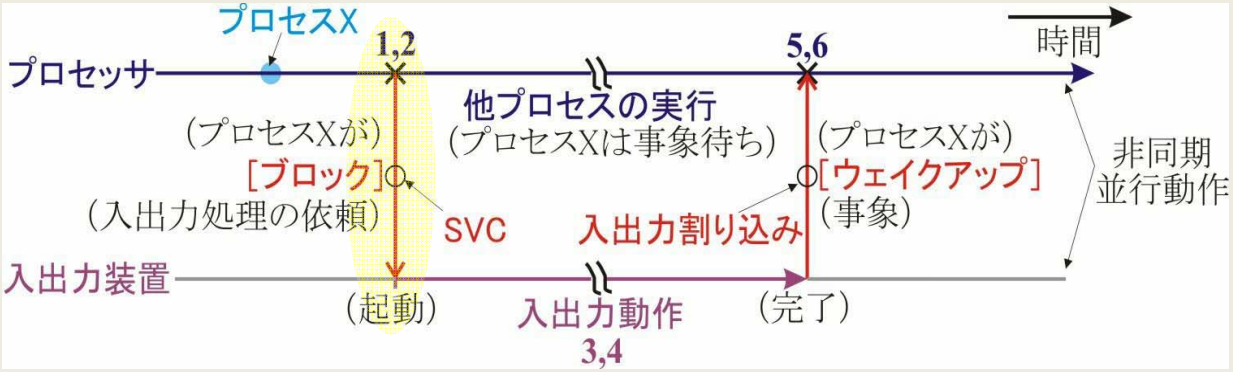
プロセスと割り込みとの関係 —例：入出力— (1)

■ 入出力管理サービス：入出力処理を管理・制御するOSのシステムサービス（→ そのOS機能は入出力管理サーバ）

■ 入出力管理サービス（入出力管理サーバ）と入出力装置の動作（入出力処理）との関係

[プロセッサ(OS)]

1. 「入出力処理をOSに依頼する」SVC（＝ブロック割り込み）を実行したプロセス（＝X）を[実行中→待ち]状態遷移（ブロック）
2. 1の割り込みをきっかけとするプロセススイッチ（広義）によって別のプロセスを[実行可能→実行中]状態遷移（＝切り替え，スイッチ）

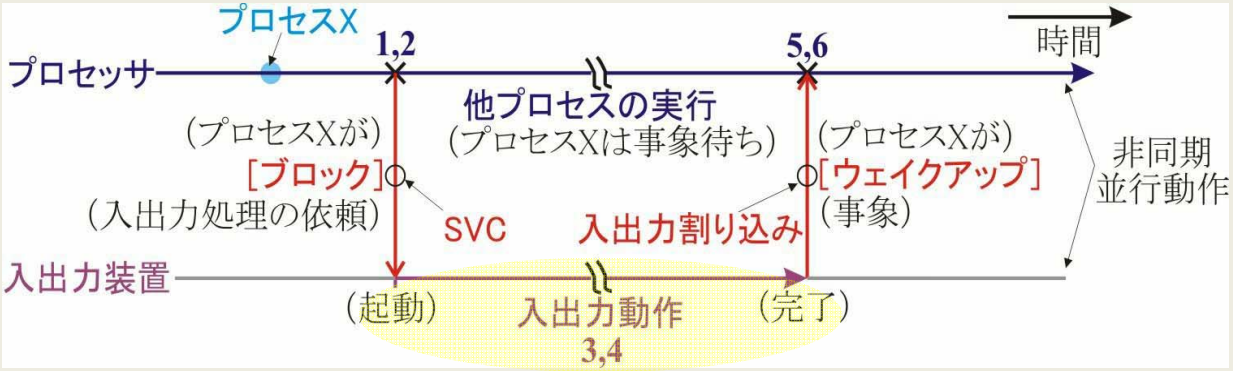


プロセスと割り込みとの関係 —例：入出力— (2)

［入出力装置］

3. プロセッサ（実際には，入出力管理サーバ）からの入出力指令によって入出力動作を開始
4. 入出力動作を完了するとプロセッサ（実際には，**OS**）に対して入出力割り込み（事象）によって，「入出力動作の完了」を通知

- 入出力動作中（3～4）→ プロセッサは入出力装置とは非同期&独立に（X以外の）別のプロセスを実行可

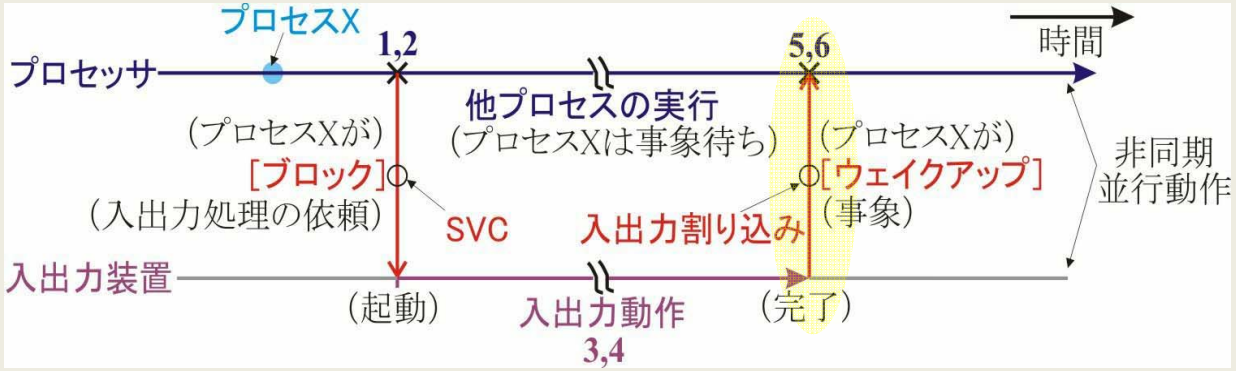


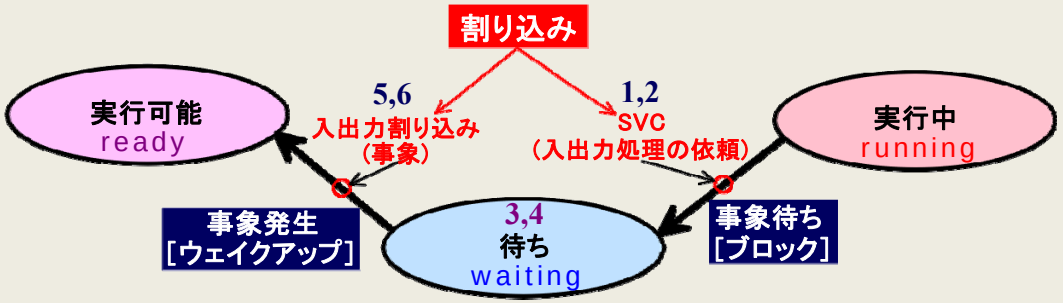
プロセスと割り込みとの関係 —例：入出力— (3)

[プロセッサ(**OS**)]

5. 入出力割り込みを受け付け，割り込み処理を開始
6. 入出力割り込み（＝ウェイクアップ**割り込み**）という事象を待つプロセス**X**を［待ち→実行可能］状態遷移（**ウェイクアップ**）

- 以降，ウェイクアップしたプロセス**X**はいつでも再開（＝実行可能→実行中状態遷移）可





【まとめ】OSから見るプロセスと割り込みとの関係
—例：入出力—

- 「入出力処理を依頼する」SVC：当該ユーザプロセス自身が「『入出力動作の完了』という事象待ちのために，プロセッサ（時間）を他のユーザプロセスに譲り渡す」依頼機能の実現

■ （入出力処理依頼）SVC：ユーザプロセス自身の［実行中→待ち］状態遷移（＝ブロック）通知
→ ブロック割り込み

↓ ↑

- 入出力割り込み：「入出力装置が非同期に並行動作するプロセッサに同期をとってもらう」通知機能の実現

■ 入出力割り込み：入出力処理を依頼したユーザプロセスの［待ち→実行可能］状態遷移（＝ウェイクアップ）を引き起こす事象（の発生）
→ ウェイクアップ割り込み

プロセス領域（プロセス本体）(1)

- **プロセス割り付け**：**OS**が，実行前（静的）に，メインメモリ(MM)上に領域（空間）を確保（割り付け）し，そこに実行するプログラム（＝プロセス）を置く　**（再掲）**

■ **プロセス領域**（プロセス本体）：**OS**が，**プロセス割り付け**によって，確保/生成するメインメモリ(MM)空間
＝「実行するプログラム」としてのプロセスの実体（本体）

プロセス領域（プロセス本体）(2)

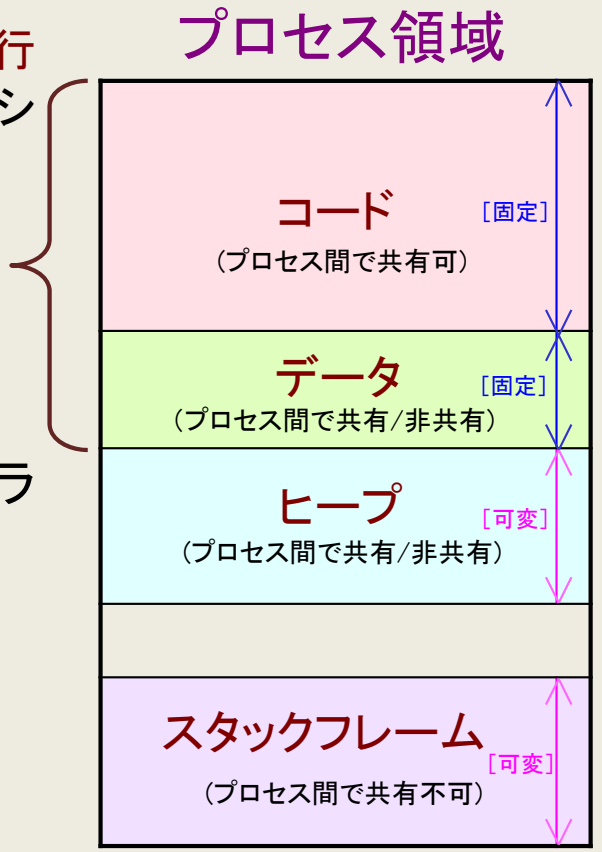
■ **OS**が静的（＝ 実行前）に確保するプロセス領域

→ サイズは実行前に定義・固定

- **OS**が，実行（直）前に，ファイルとしてファイル装置に格納してある実行可能プログラム（そのまま直ちに実行可能なプログラム，実際にはマシン命令列）をMMのプロセス領域にロード

(A) **コード**(code)：実行可能プログラムのマシン命令列部分

(B) **データ**：実行可能プログラムのデータ部分で，ソースプログラムでのグローバル（global; 大域）変数（静的データ）に対応



プロセス領域（プロセス本体）(3)

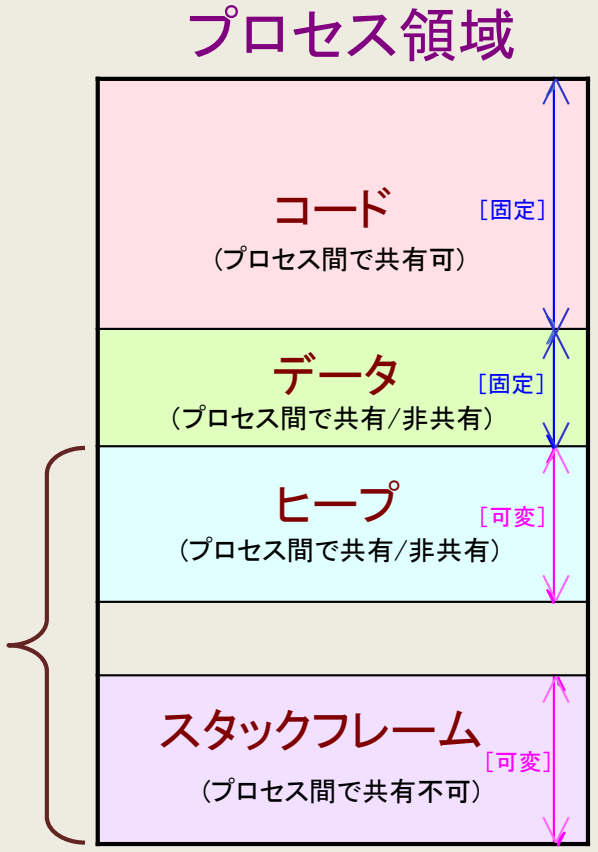
■**OS**が動的（＝ 実行時）に生成するプロセス領域

→ サイズは実行中に伸縮・可変

(C) **ヒープ**(heap)：主としてローカル（local;局所）変数（静的データ）に対応するデータ 用の一時作業領域

(D) **スタックフレーム**(stack frame)：スタック構造を備える，主として命令系用の一時作業領域

◆ MMを流用 & ソフトウェア機能で実現



（参考2.4）

スタック(stack)

- 唯一備えるアクセスポート（access port; アクセス口）を書き込みと読み出しで共用する1次元メモリ（構造）
  - LIFO (**L**ast **I**n **F**irst **O**ut):“一番最後に格納したものを一番最初に読み出す”順でアクセス
  - “書き込み順序を保持して，その書き込み順とは逆順で読み出す（ポップアップ(pop-up)）機能”を装備
  - ◆ スタックへの書き込み ＝ プッシュダウン(push-down)　→“プッシュダウンメモリ”