

ソフトウェア実験I

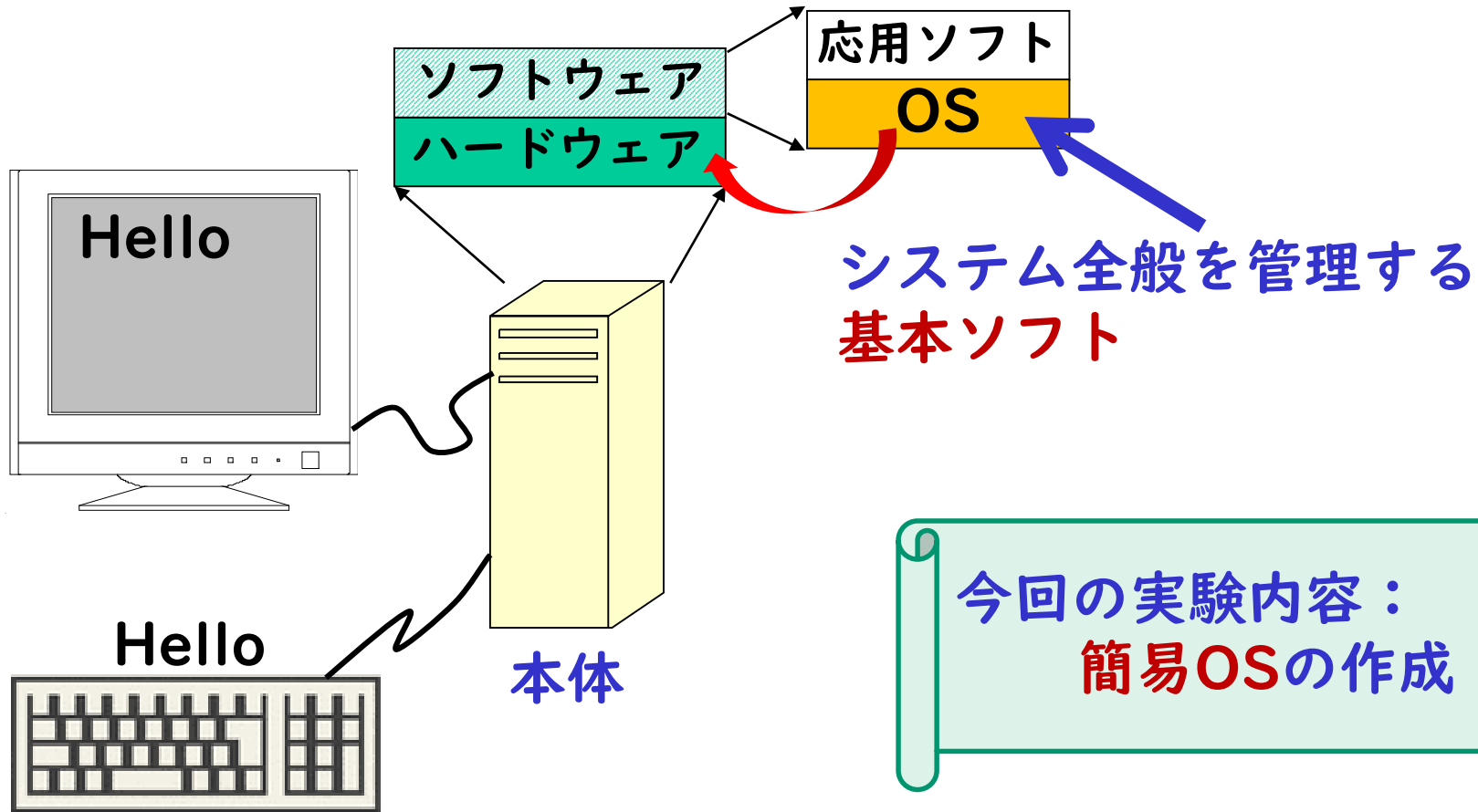
簡易OSの開発

担当教員： 峯、牟田、廣瀬、馮(フォン)、佐藤

令和元年10月25日(金)

Part I 背景と目的

OSの位置づけ



当たり前？

今回の実験内容：
簡易OSの作成

本実験で作成するもの

①簡易OS (班ごとに)

そして、作成したOSを利用して、

②エコーバックプログラム (班ごとに) } 応用ソフト
③選択課題 (個人で、Page 48) }



本実験の目的

- ・ OS全体および各部分への理解
- ・ グループワークの体験
- ・ アセンブリ言語によるプログラミング能力の向上

Part2 システムの説明

簡易OSとは？

ユーザーモードで動作

ユーザープログラム

TRAP#0

システムコール

RTE

スーパーバイザモードで動作

システムコールライブラリ

キュー入出力ルーチン

カーネル

ハードウェア割り込み処理ルーチン

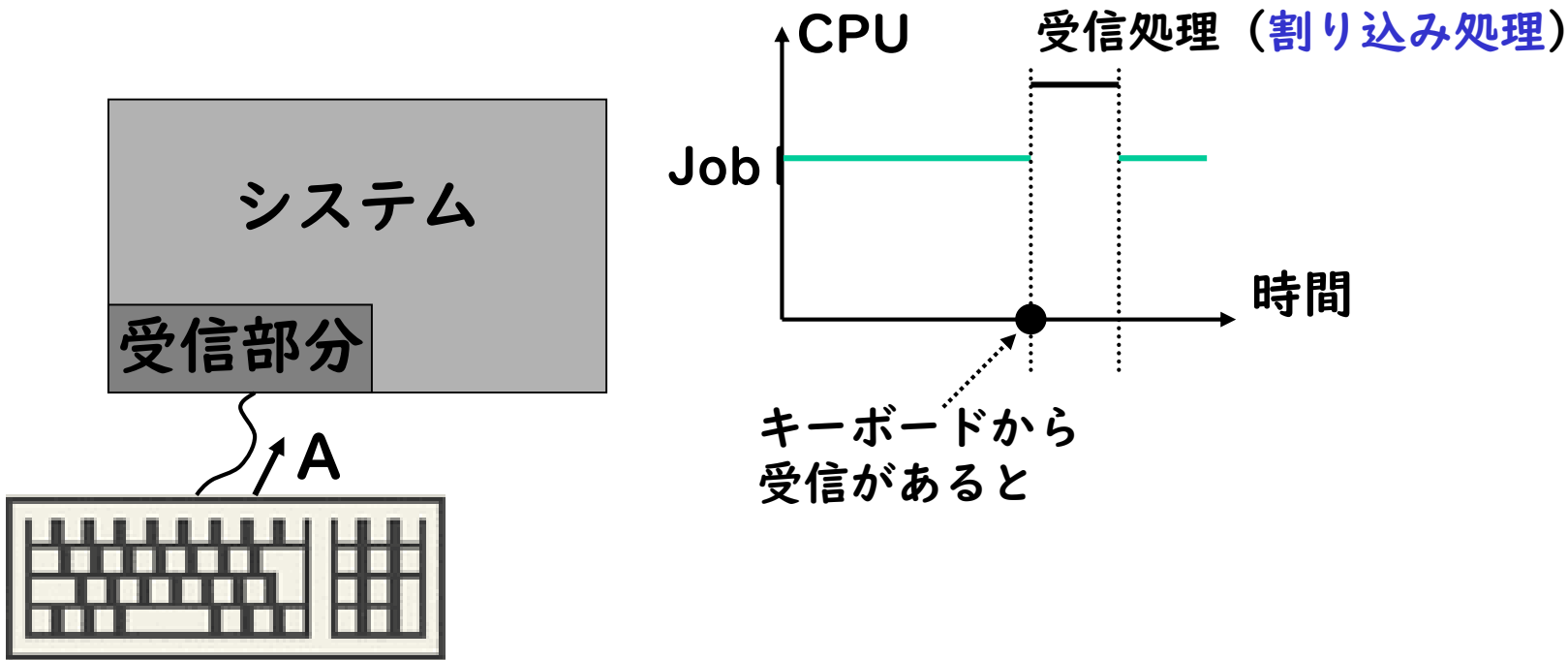
ハードウェア割り込み要求

ハードウェア

主に

- ・ ハードウェア割り込み処理
- ・ システムコール処理

ハードウェア割り込み（例）



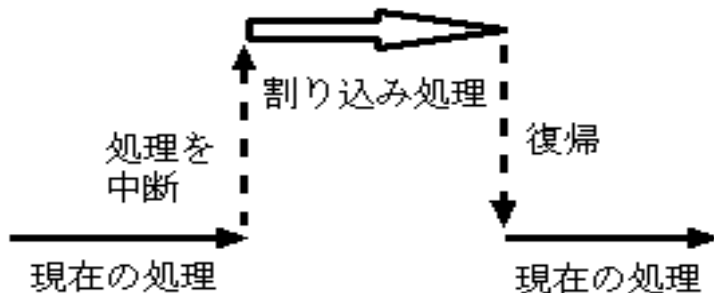
いま実行されている処理を、一時的に**中断**して、より**優先される**別の処理(**割り込み処理**)を行う

本システムの割り込み処理の流れ

68000CPUの走行モード

- **ユーザモード**：
ユーザプログラムが実行されている時
- **スーパーバイザモード**：
OS提供する処理ルーチンや
割り込み処理ルーチンが実行されている時

走行レベル



時間

現在の処理

割り込み信号を受信

スーパーバイザモードに移行
レジスタをスタックに保存

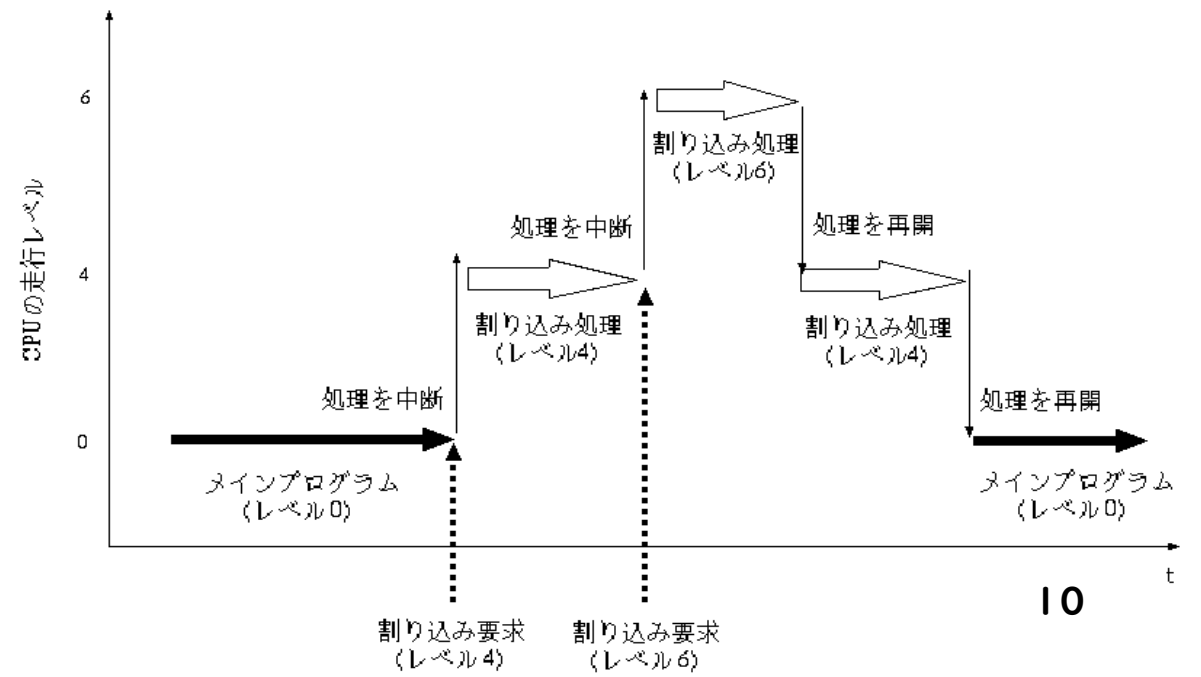
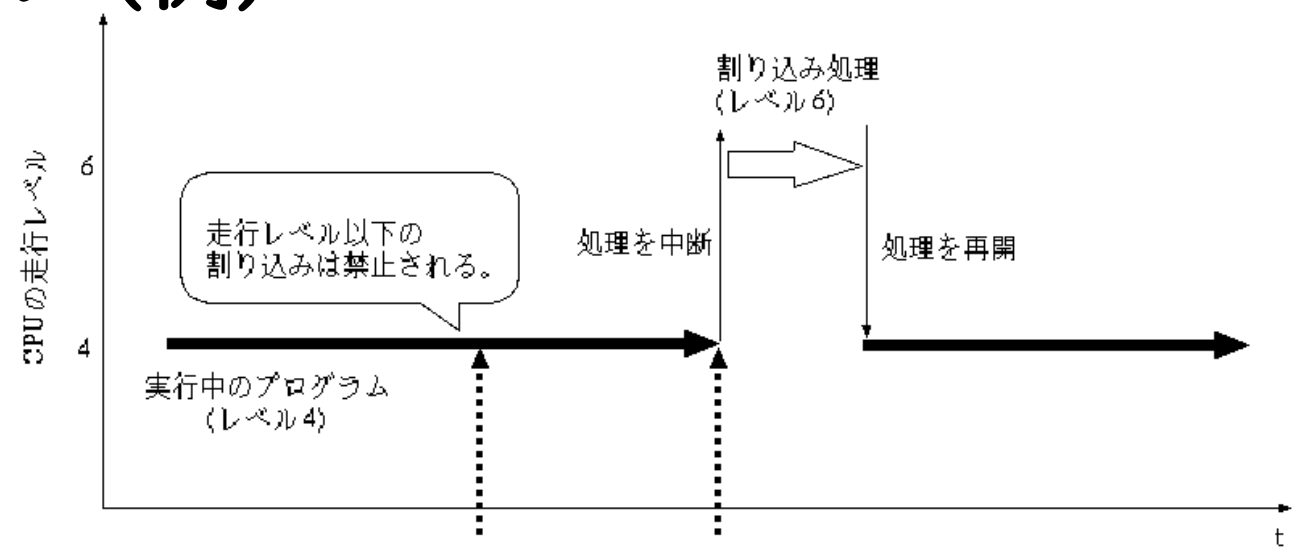
割り込み処理を実行

元のレジスタを復帰させる
CPU：以前の**走行レベル**に、
ユーザモードに戻る

中断された処理を再開

CPU走行レベル (例)

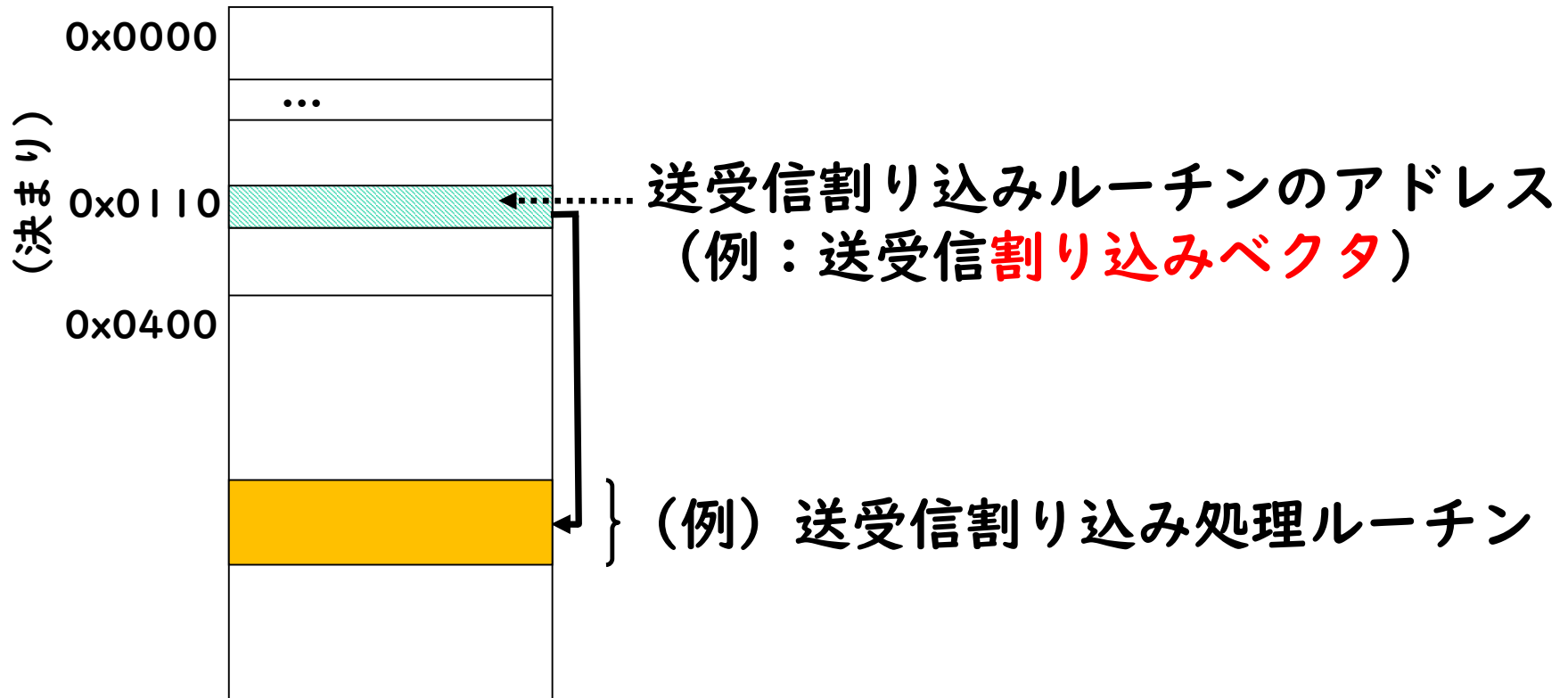
(優先順位)



割り込みベクタ

どのように割り込み処理ルーチンを探すのか？

割り込みベクタを利用する



CPUボードのメモリマップ

OSの構成

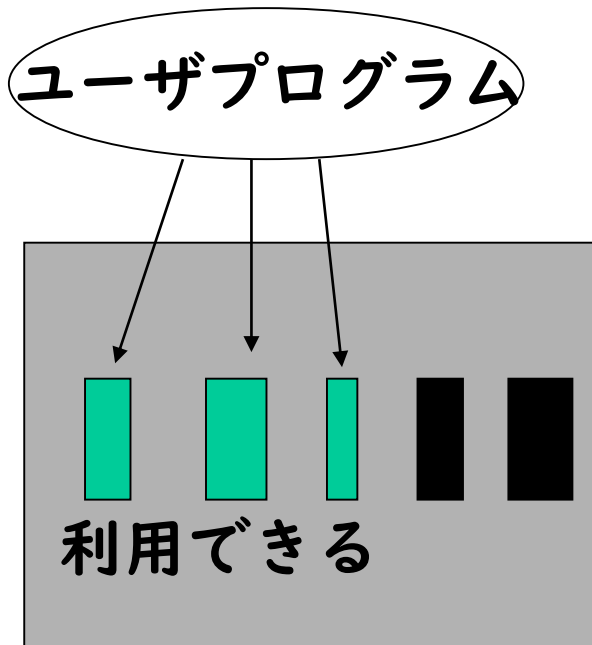
OSカーネル：

OSの基本的機能を提供するソフトウェア
(ハードウェアの監視、メモリ管理、割り込み処理
など)

システムコール：

OSカーネルの機能のうち、ユーザプログラムから
利用可能なものを呼び出すこと

システムコールの実現



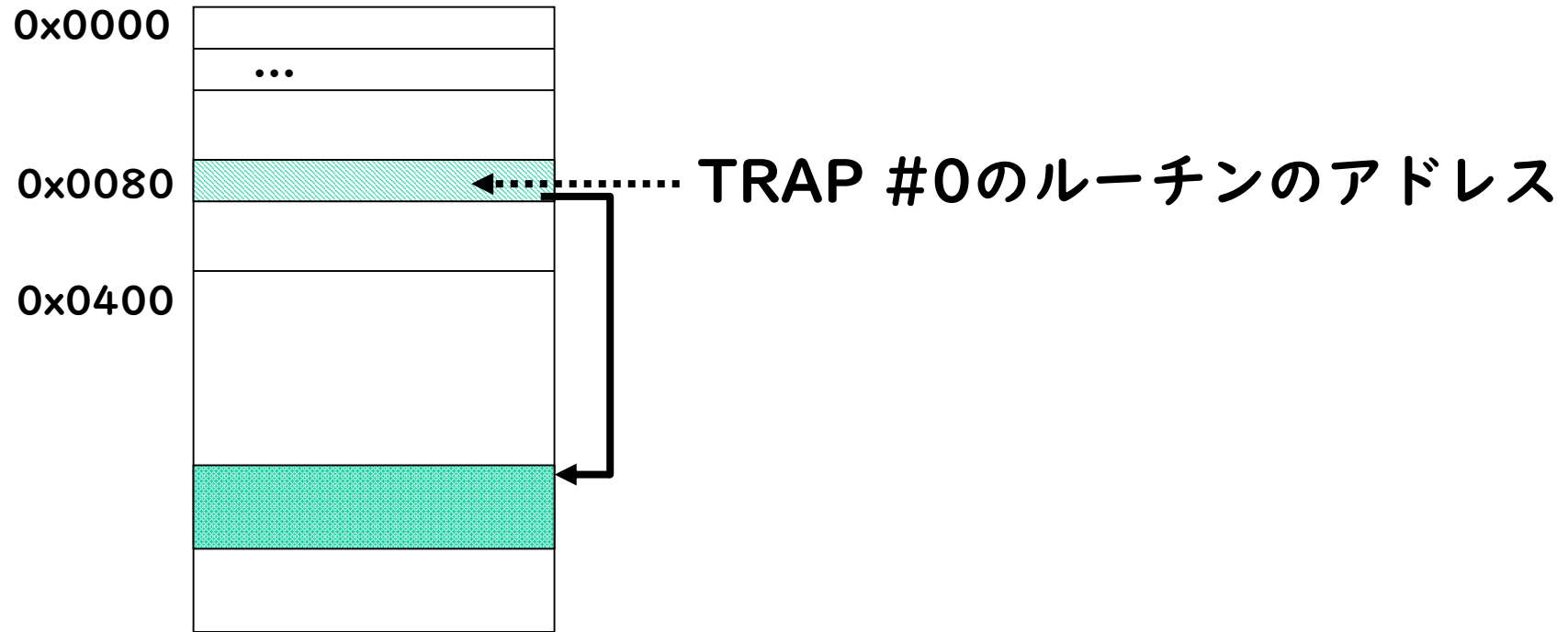
ユーザプログラムから
カーネルに仕事を依頼

TRAP命令

本実験では **TRAP # 0** 命令で

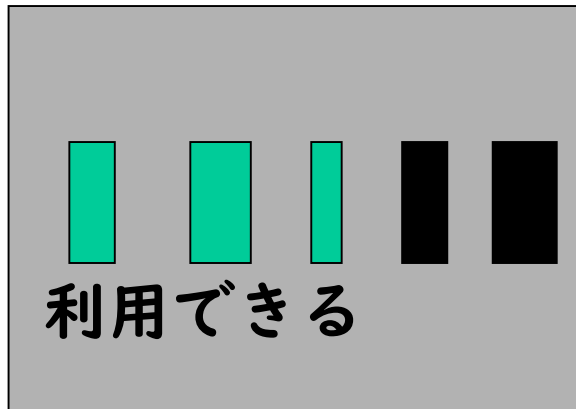
- ・ユーザプログラムの中のコマンドで行う
- ・タイミングはユーザが決める

TRAP #0の処理



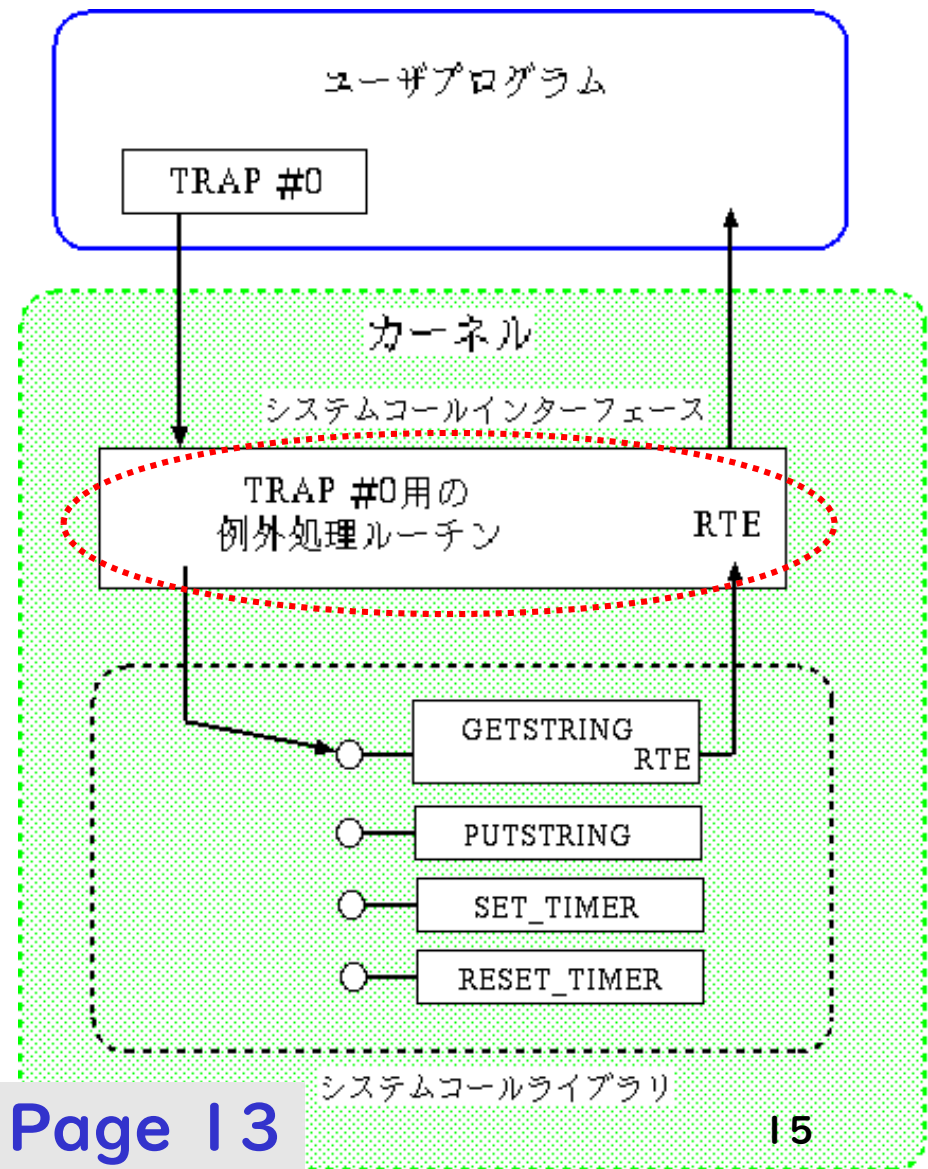
CPUボードのメモリマップ

複数の機能は **TRAP #0** の呼び出して、 使い分ける



TRAP命令

スーパーバイザモードへ
切り替える



簡易OSとは？（再掲）

ユーザーモードで動作

ユーザープログラム

TRAP#0

システムコール

RTE

スーパーバイザモードで動作

システムコールライブラリ

キュー入出力ルーチン

カーネル

ハードウェア割り込み処理ルーチン

ハードウェア割り込み要求

ハードウェア

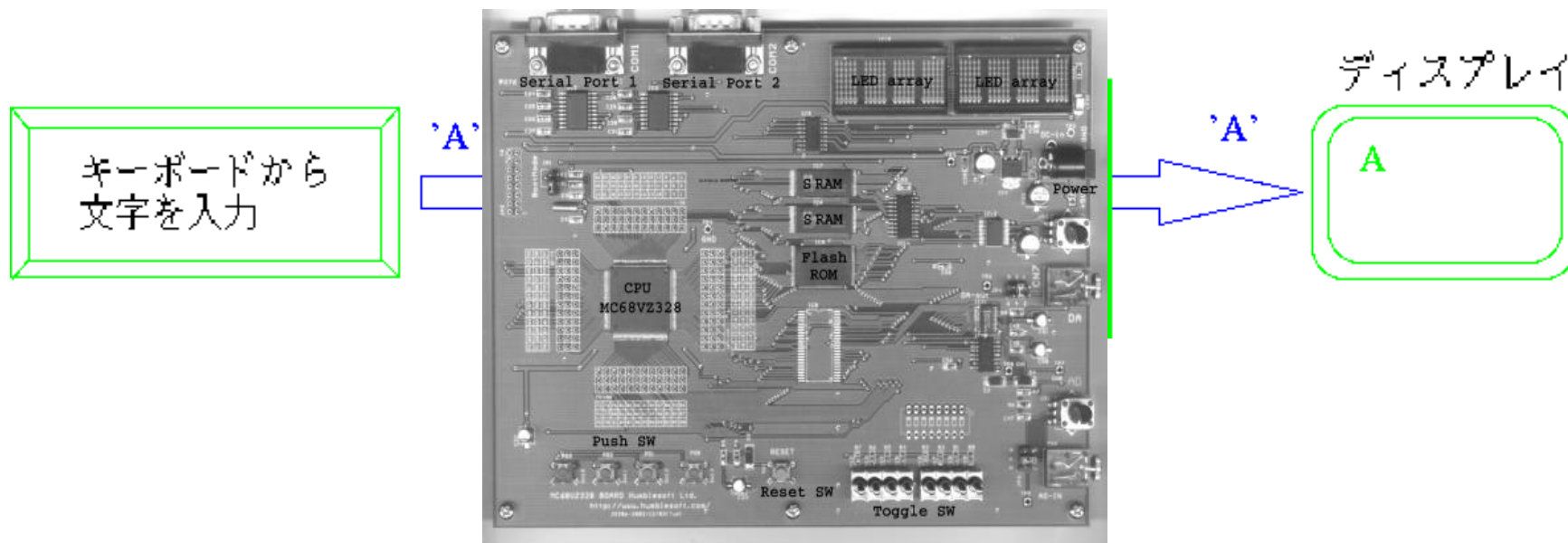
主に

- ・ ハードウェア割り込み処理
- ・ システムコール処理

Part3 本実験で作成する簡易 OS

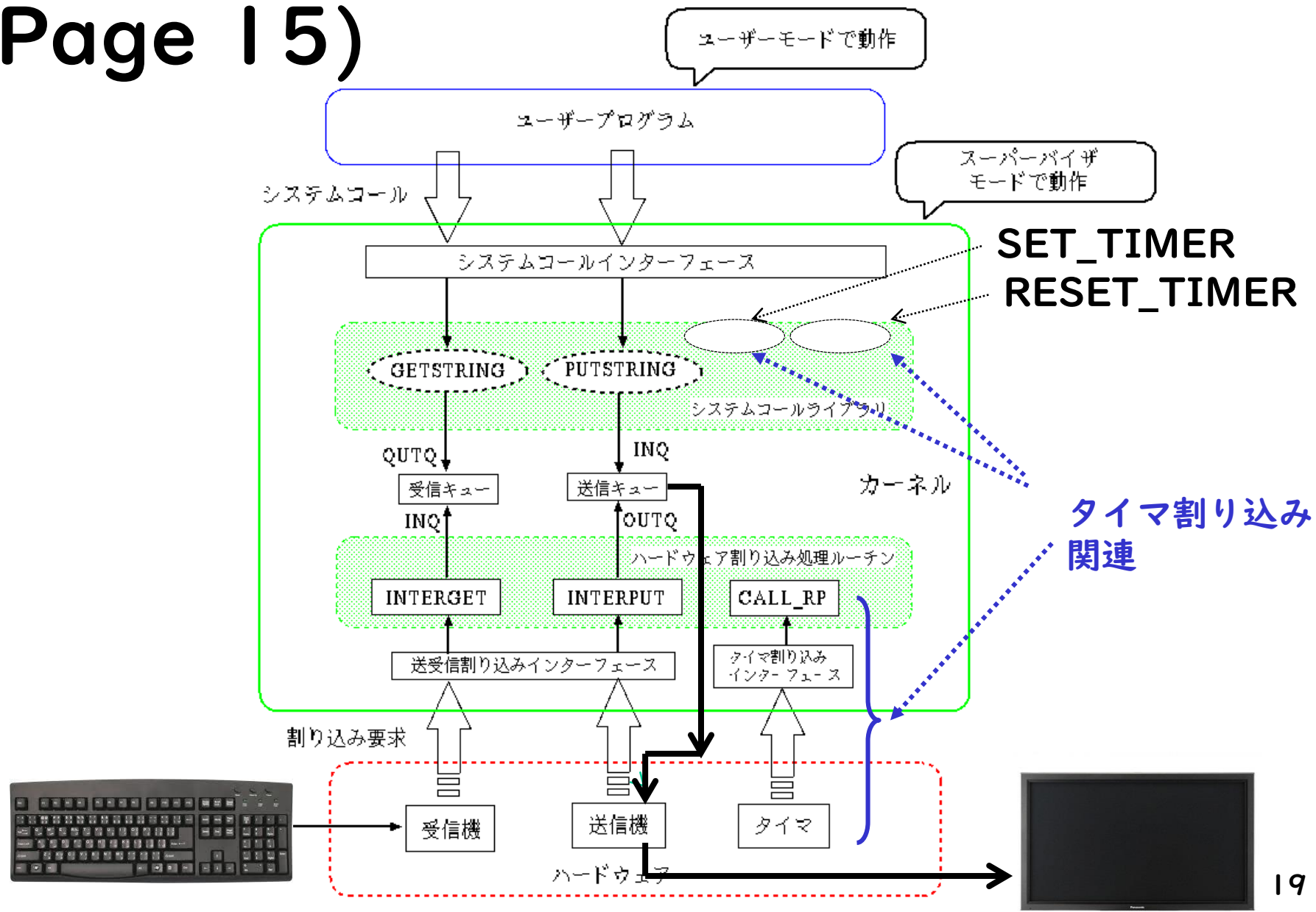
作成する簡易OS

- 1) キーボードで入力したものを
そのままモニターへ出力するために、**必要なOS**
- 2) タイマー



エコーバックシステムの流れ

(Page 15)



ハードウェア割り込み関連

1. 送受信割り込み：

- ・INTERGET: 送受信割り込みの受信処理ルーチン
- ・INTERPUT: 送受信割り込みの送信処理ルーチン
- ・インターフェース

2. タイマ割り込み

タイマ内のカウンタが予め設定した値に達すると発生
例：一定時刻毎に文字をモニターに表示させるなど

- ・CALL_RP: タイマ割り込み処理ルーチン
- ・インターフェース

システムコール関連

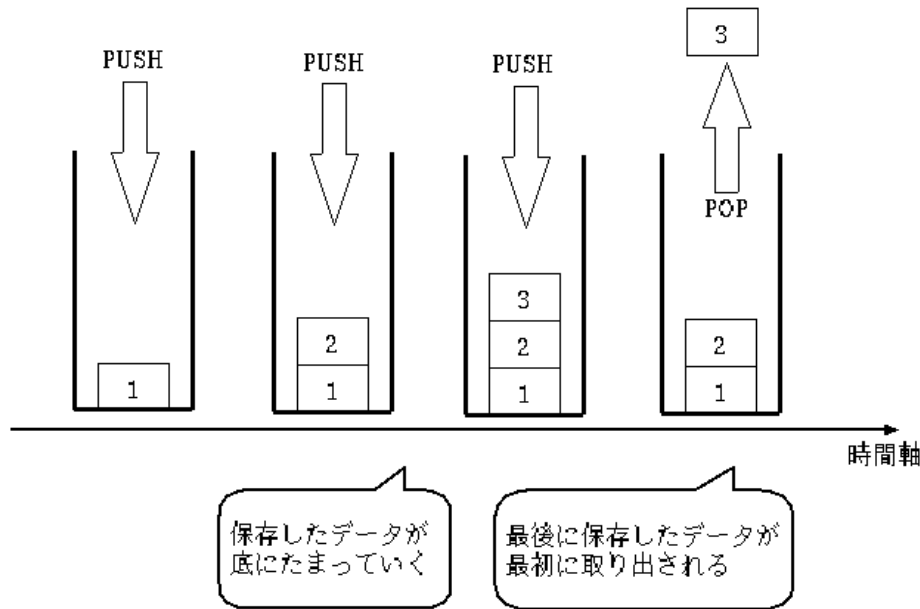
- ・ *PUTSTRING* 送信キューにデータを渡す
- ・ *GETSTRING* 受信キューからデータを取り出す
- ・ *RESET_TIMER* タイマのリセット
- ・ *SET_TIMER* タイマを起動
- ・ インターフェース

カーネル内部で用いるサブルーチン

- ・ *INQ, OUTQ* キューへのデータの入出力

スタック、キュー

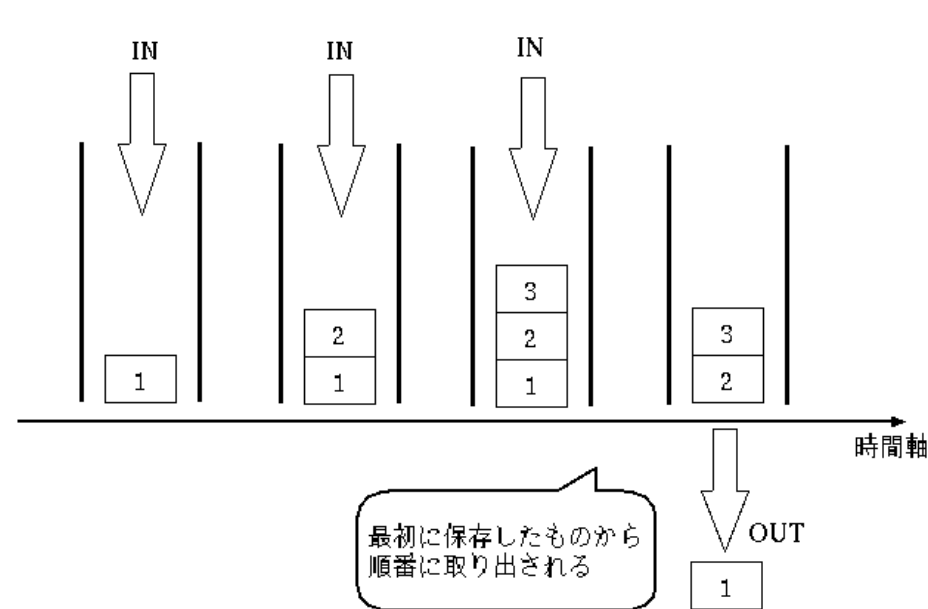
スタック (底のある一時記憶領域)



スタック

First In Last Out
後入れ 先出し

キュー (底のない一時記憶領域)

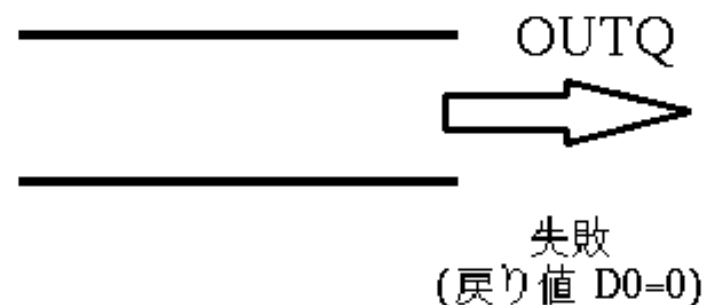
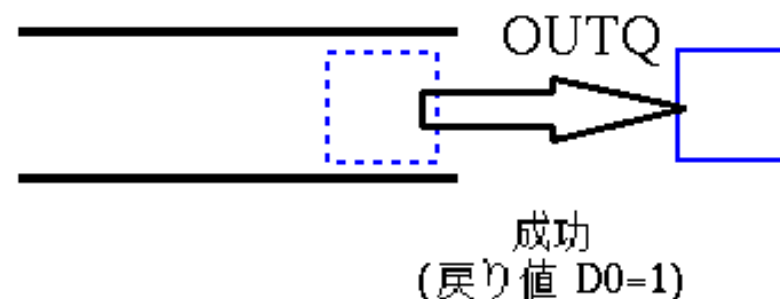
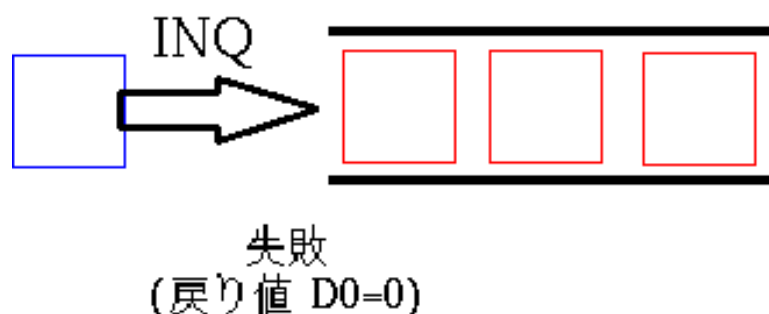
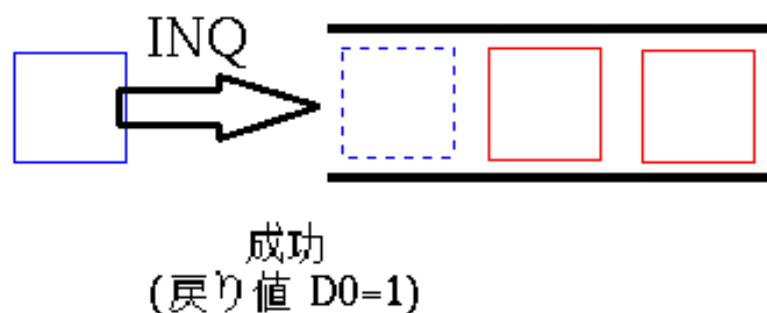


キュー

First In First Out
先入れ先出し

キュー (操作)

INQ と OUTQ



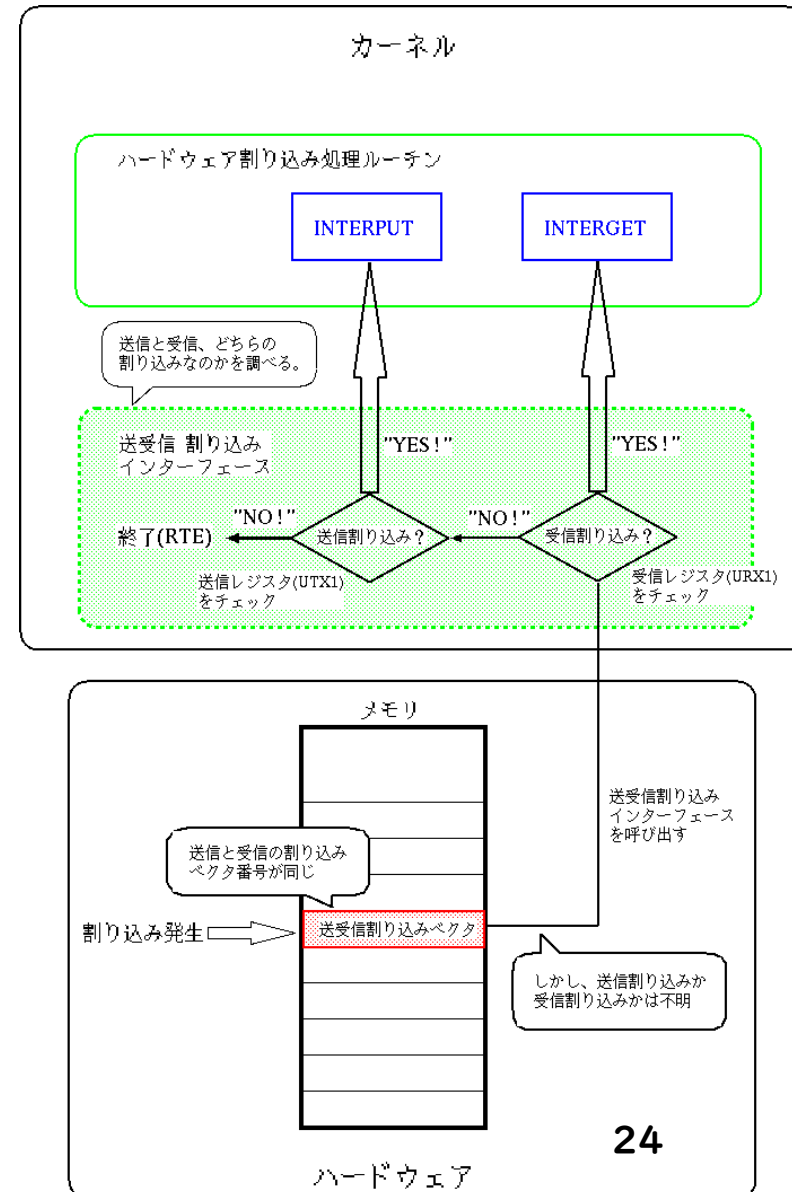
送信割り込みと受信割り込みの判定

(Page 17)

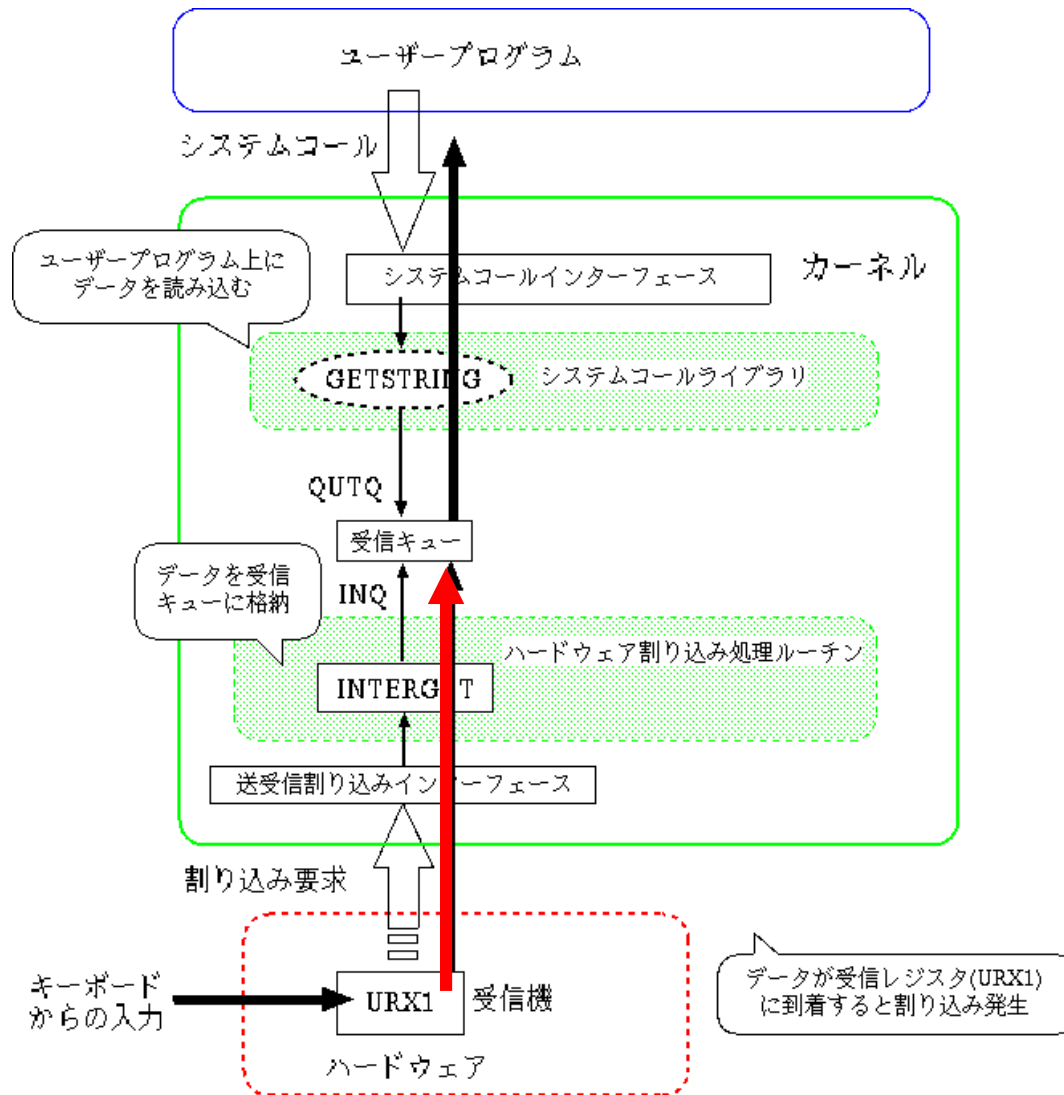
同じ**割り込みベクタ**を使用

判定方法：

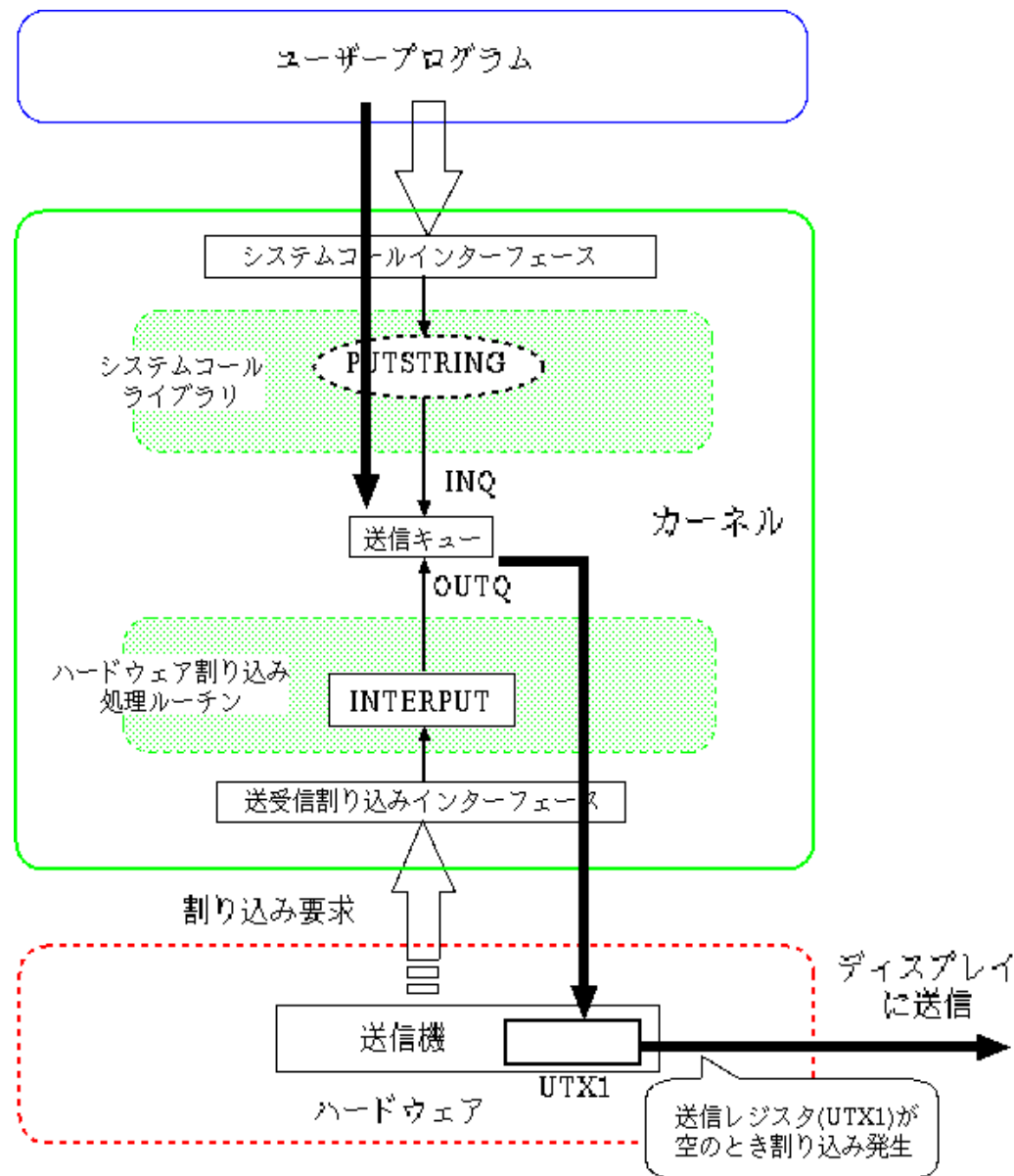
- 送受信割り込みインターフェースで
- 送信機と受信機の状態を示すレジスタをチェックする



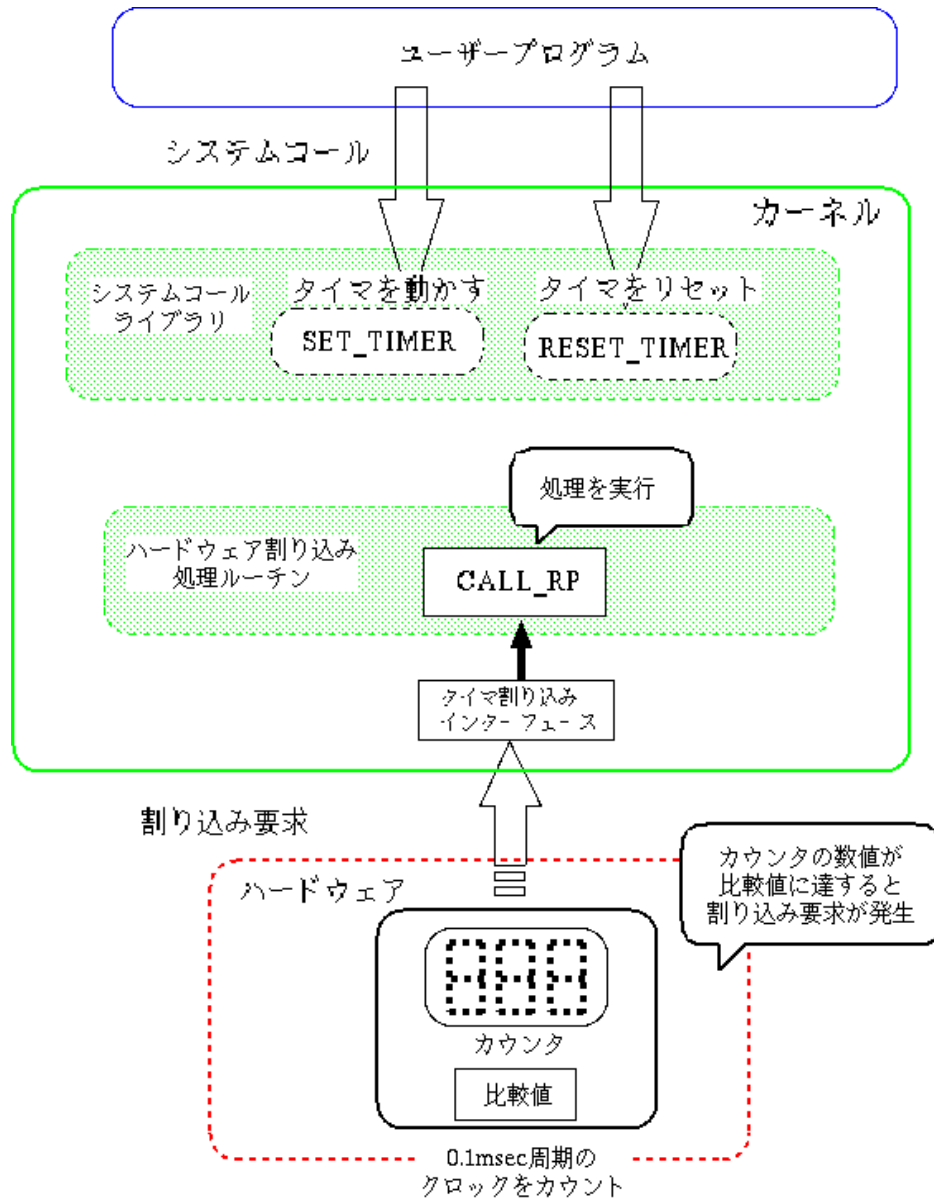
受信部における処理 (Page 18)



送信部における処理 (Page 18)



タイマ割り込み (Page 20)



カウンタは**基準信号**をカウント

カウンタと**比較値**が一致すると
割り込み信号が発生

Part4 その他

- 単位取得の条件
 - 役割分担
 - スケジュール
- など

単位取得の条件 (Page 5)

1. 班ごとに

- ①「簡易OS」、
 - ②「エコーバックプログラム」
- Step 9 までの部分に相当

2. 各自で

- ③ 1 4 個の選択課題から 1 つ選んで、完成させる
 - 班内で同じ課題は不可
 - 課題の難易度、工夫点等により加点

3. システム全体への理解と把握 個別試問あり

役割分担 (Page 22)

詳細はp.22 表3.1

- 1) 初期化担当 (1～2人)
 - 2) タイマー制御部担当 (1人)
 - 3) 受信制御部担当 (1人)
 - 4) 送信制御部担当 (1人)
- } キューの作成も担当

レビューについて

チームワーク：

役割分担、討議、レビュー (チェック、コメント)

初期化担当者	⇔	タイマ担当者
送信担当者	⇔	受信担当者

最終試問 (Page 50)

① 動作確認

- 1) Step 9のユーザプログラム (エコーバックプログラム)
- 2) 個人の選択課題のプログラム (Step 10)

② 班ごとの提出物

簡易OSの本体, エコーバックプログラム
(仕様説明書など, 十分にコメントを記載しておくこと)

③ 個人の提出物

選択課題の

- 1) コード
- 2) レポート (その内容/工夫点)

④ 個別試問

システム全体および各部分に対する理解度の確認

日報

- ・ 2種類： 個人用、班用
- ・ Moodle から毎回提出
- ・ 記入方法の詳細は付録D (Page 92)

実験日程

10月25日(金)	1
29日(火)	2
30日(水)	3
31日(木)	4
11月1日(金)	5
5日(火)	6
6日(水)	7
7日(木)	8
8日(金)	9
12日(火)	10
13日(水)	11
14日(木)	12
15日(金)	最終試問
19日(火)	最終試問

テキストの構成

- 第1章： 始めに（実験の目的、単位取得条件など）
- 第2章： 実験の概要（基本の概念、仕組み）
- 第3章： 実験の手順（具体的な進み方）

- 付録A： プログラムの作成と実行
- 付録B： CPUボードの機能説明、各設定方法など
- 付録C： 送受信チャンネルの2チャンネル化
- 付録D： 作業日報の様子とその書き方

指導体制

教員（５名）：

峯、牟田、廣瀬、馮（フォン）、佐藤

TA（１０名）：

小野未琴、チョヒョンス、山口翼、実崎直人、
弓場邦哲、佐藤祐大、細谷桃子、加藤博之、
浅倉健太、松井健

以上

頑張りましょう！