



東邦大学

いのち
生命の科学で未来をつなぐ

割込みとは？

聞いたことのない用語？

割込みとは



割り込みとは

マーケットの
列へ割り込む

本来の順番を
飛び越して
先に処理する

飛び越す
理由や緊急性
がある



ここでの対象は命令実行



東邦大学

ここでの対象は命令実行

本来の命令実行は、命令の並びどおり
(+ジャンプ命令でジャンプする)

ここでの対象は命令実行

本来の命令実行は、命令の並びどおり
(+ジャンプ命令でジャンプする)

何回実行しても、周囲がどう変わっても
(自分でチェックして変えない限り)
命令実行の順序は変わらない

ここでの対象は命令実行

本来の命令実行は、命令の並びどおり
(+ジャンプ命令でジャンプする)

何回実行しても、周囲がどう変わっても
(自分でチェックして変えない限り)
命令実行の順序は変わらない

周囲の変化を知らせたいことがあるのに...



周囲の変化を知らせたい

例： プログラムが止まらない
(周囲をチェックしないループに落ちた)

例： ハードエラーが起きたらすぐ止めたい

例： 一生懸命処理をしているときに
キーが押されたことを、繰り返し
チェックするのはやりたくない
(チェックの手間で性能も落ちるし)



(CPUへの)割込みとは

命令実行に対して割込んで
周囲の変化をプログラムに伝える仕組み

プログラムは、周囲の変化をチェック
し続けなくてもよい
(し続けるように書かなくてもよい)



割込みの原因 (周囲の変化)



東邦大学

割込みの原因 (周囲の変化)

プログラムの外的な原因

プログラムの内部的な原因



東邦大学

割込みの原因 (周囲の変化)

プログラムの外的な原因

例： ハードウェア故障

プログラムの内部的な原因



割込みの原因 (周囲の変化)

プログラムの外的な原因

例： ハードウェア故障
入出力機器からの要求

プログラムの内部的な原因



割込みの原因 (周囲の変化)

プログラムの外的な原因

例： ハードウェア故障
入出力機器からの要求
リセットボタンが押された

プログラムの内部的な原因



割込みの原因 (周囲の変化)

プログラムの外的な原因

例： ハードウェア故障
入出力機器からの要求
リセットボタンが押された

プログラムの内部的な原因

例： 命令の結果、計算あふれが起きた



割込みの原因 (周囲の変化)

プログラムの外的な原因

例： ハードウェア故障
入出力機器からの要求
リセットボタンが押された

プログラムの内部的な原因

例： 命令の結果、計算あふれが起きた
命令の結果、実行できない命令だった



割込みの原因 (周囲の変化)

プログラムの外的な原因

例： ハードウェア故障
入出力機器からの要求
リセットボタンが押された

外部割込み

プログラムの内部的な原因

例： 命令の結果、計算あふれが起きた
命令の結果、実行できない命令だった

内部割込み



割込みの仕組



割込みの仕組み

割込みはCPUに電気信号として伝えられる



割込みの仕組み

割込みはCPUに電気信号として伝えられる

CPUチップの外から： 割込み信号線



割込みの仕組み

割込みはCPUに電気信号として伝えられる
割込みが 1 になると、CPUは



割込みの仕組み

割込みはCPUに電気信号として伝えられる

割込みが 1 になると、CPUは
実行中の命令が終わった後、



割込みの仕組み

割込みはCPUに電気信号として伝えられる

割込みが1になると、CPUは
実行中の命令が終わった後、
予め決めたアドレスにジャンプする



割込みの仕組み

割込みはCPUに電気信号として伝えられる

割込みが1になると、CPUは
実行中の命令が終わった後、

予め決めたアドレスにジャンプする

この場所から始まるプログラム（ルーチン）を
割込み処理プログラム（ルーチン）と呼ぶ



割込みの仕組み

割込みはCPUに電気信号として伝えられる

割込みが1になると、CPUは

実行中の命令が終わった後、

予め決めたアドレスにジャンプする

割込み処理ルーチンでは、外からの信号に対応する（要求される）処理をする



割込みの仕組み

割込みはCPUに電気信号として伝えられる

割込みが1になると、CPUは

実行中の命令が終わった後、

予め決めたアドレスにジャンプする

割込み処理ルーチンでは、外からの信号に
対応する（要求される）処理をする

たとえばリセットボタンが押されたのなら
システムをリセットするなど



割込みの仕組み

割込みはCPUに電気信号として伝えられる

割込みが1になると、CPUは

実行中の命令が終わった後、

予め決めたアドレスにジャンプする

割込み処理ルーチンでは、外からの信号に
対応する（要求される）処理をする

たとえばリセットボタンが押されたのなら
システムをリセットするなど

（当然、要求ごとに対応も異なる）



割込みの仕組み

割込みはCPUに電気信号として伝えられる

割込みが1になると、CPUは
実行中の命令が終わった後、
予め決めたアドレスにジャンプする

割込み処理ルーチンでは、外からの信号に
対応する（要求される）処理をする

割込み処理が済んだら、前の処理に戻る



割込みの仕組み

割込みはCPUに電気信号として伝えられる

割込みが1になると、CPUは

実行中の命令が終わった後、

予め決めたアドレスにジャンプする

割込み処理ルーチンでは、外からの信号に対応する（要求される）処理をする

割込み処理が済んだら、前の処理に戻る

（「あたかも何事もなかったかのように」復帰する）



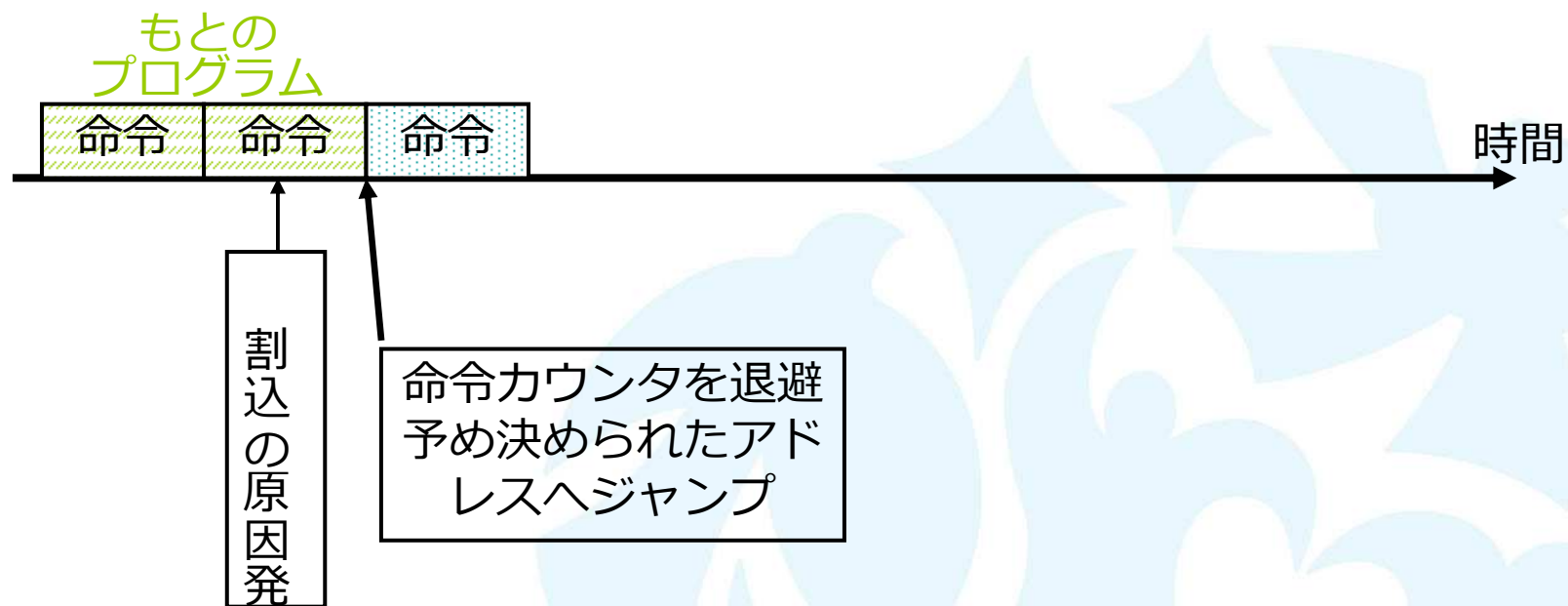
割込みの仕組み ~ 図解すると



- * 入出力機器からの信号
(動作完了など)
- * 電源断・回路異常など
- * 命令実行エラー
ゼロ割、オーバフロー
- * 割込発生命令による割込
(ソフトウェア割込み)

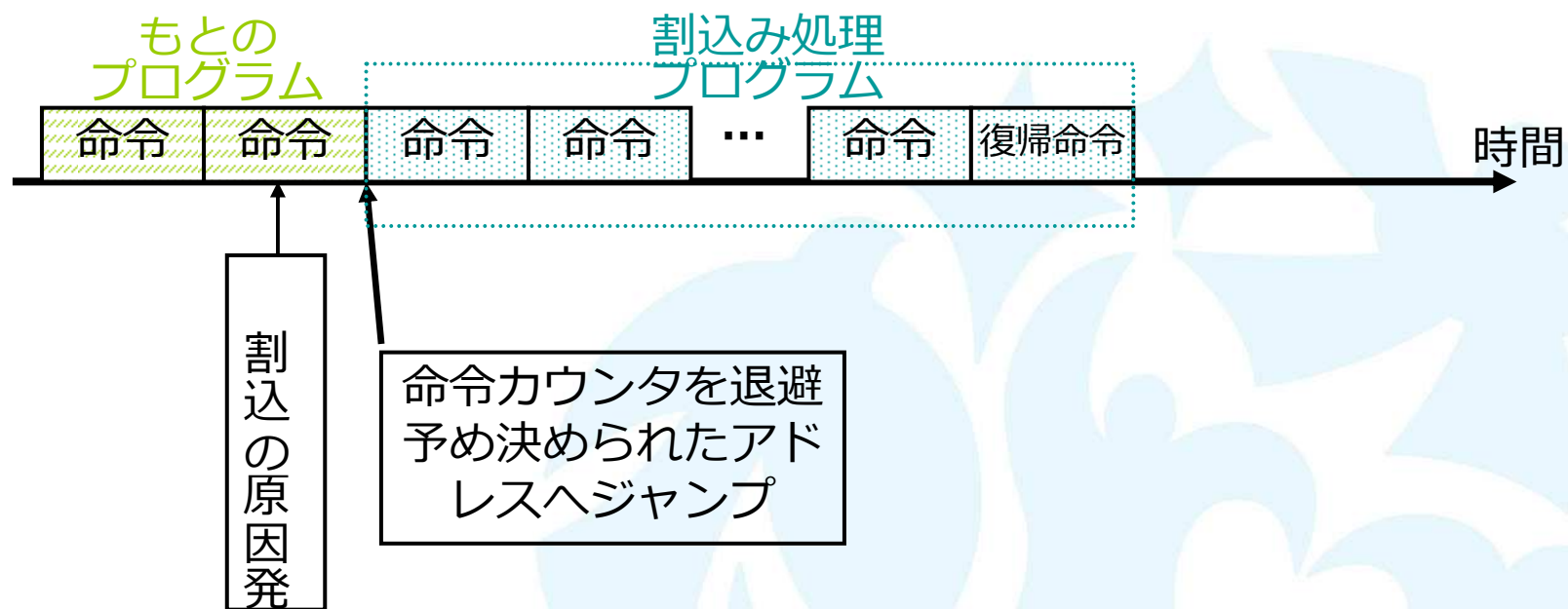


割込みの仕組み ~ 図解すると



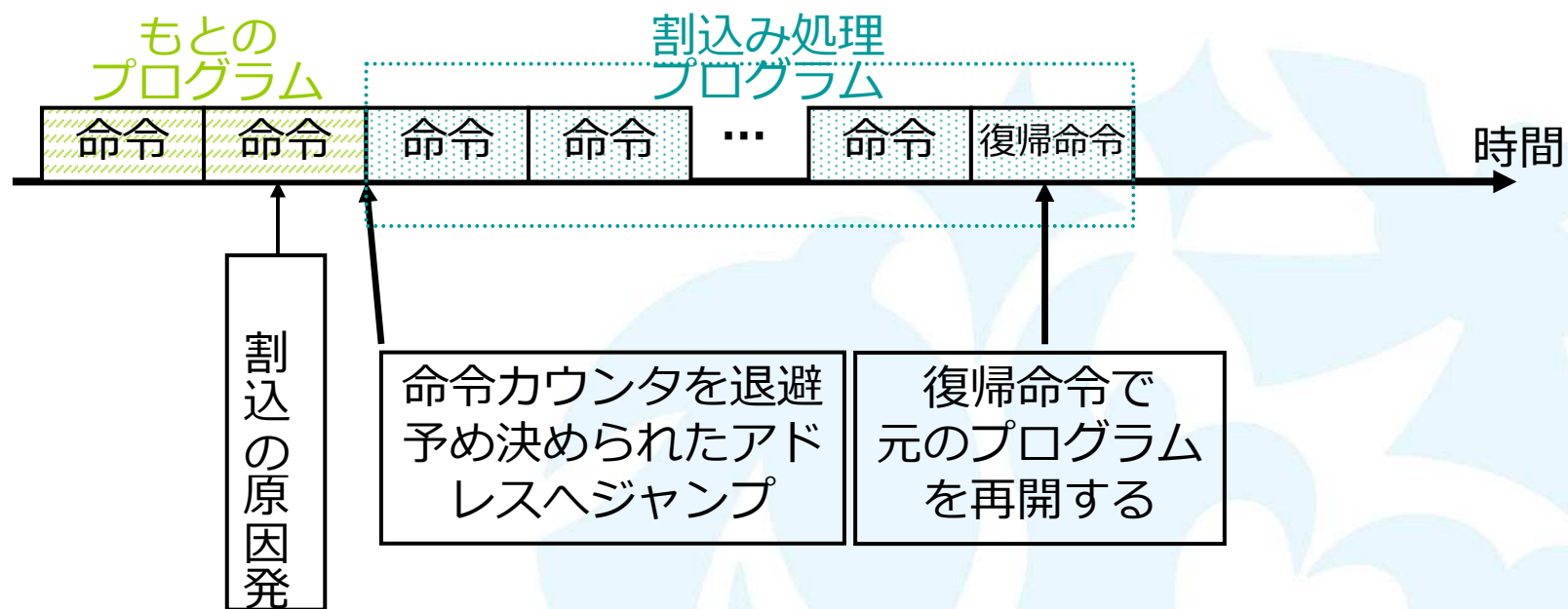
- * 入出力機器からの信号
(動作完了など)
- * 電源断・回路異常など
- * 命令実行エラー
ゼロ割、オーバフロー
- * 割込発生命令による割込
(ソフトウェア割込み)

割込みの仕組み ~ 図解すると



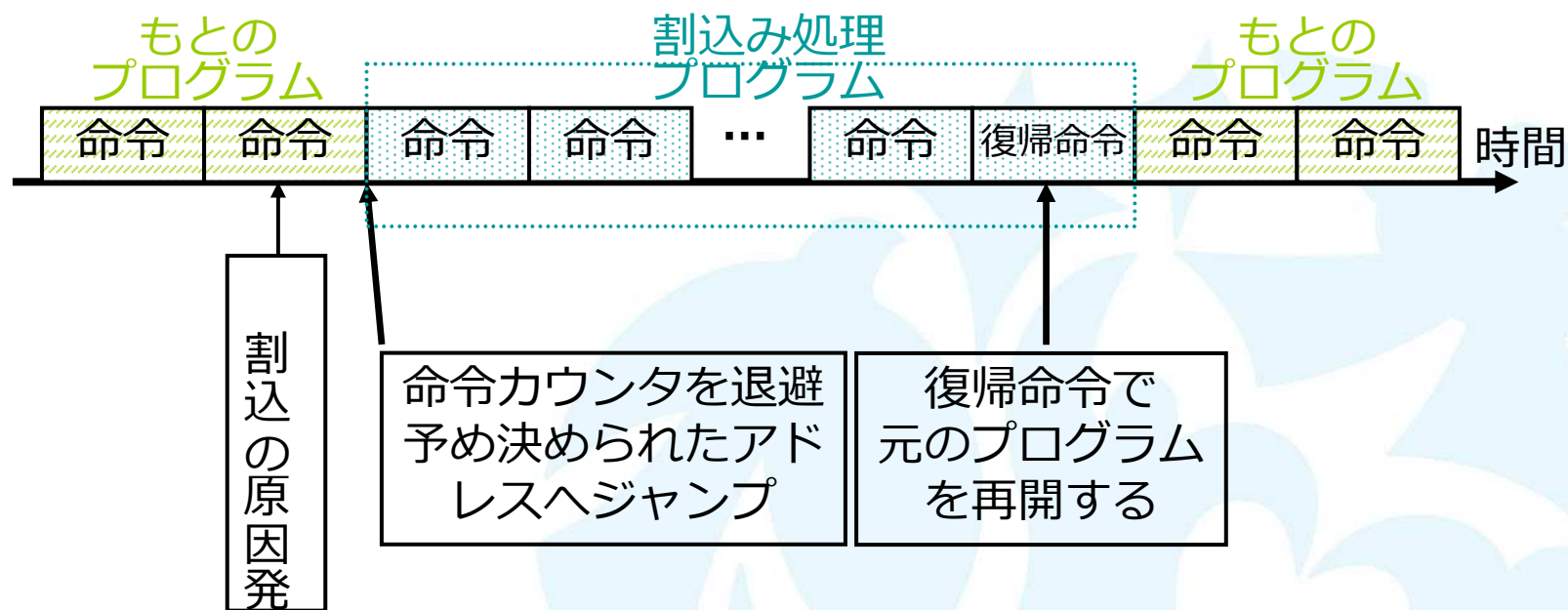
- * 入出力機器からの信号
(動作完了など)
- * 電源断・回路異常など
- * 命令実行エラー
ゼロ割、オーバフロー
- * 割込発生命令による割込
(ソフトウェア割込み)

割込みの仕組み ~ 図解すると



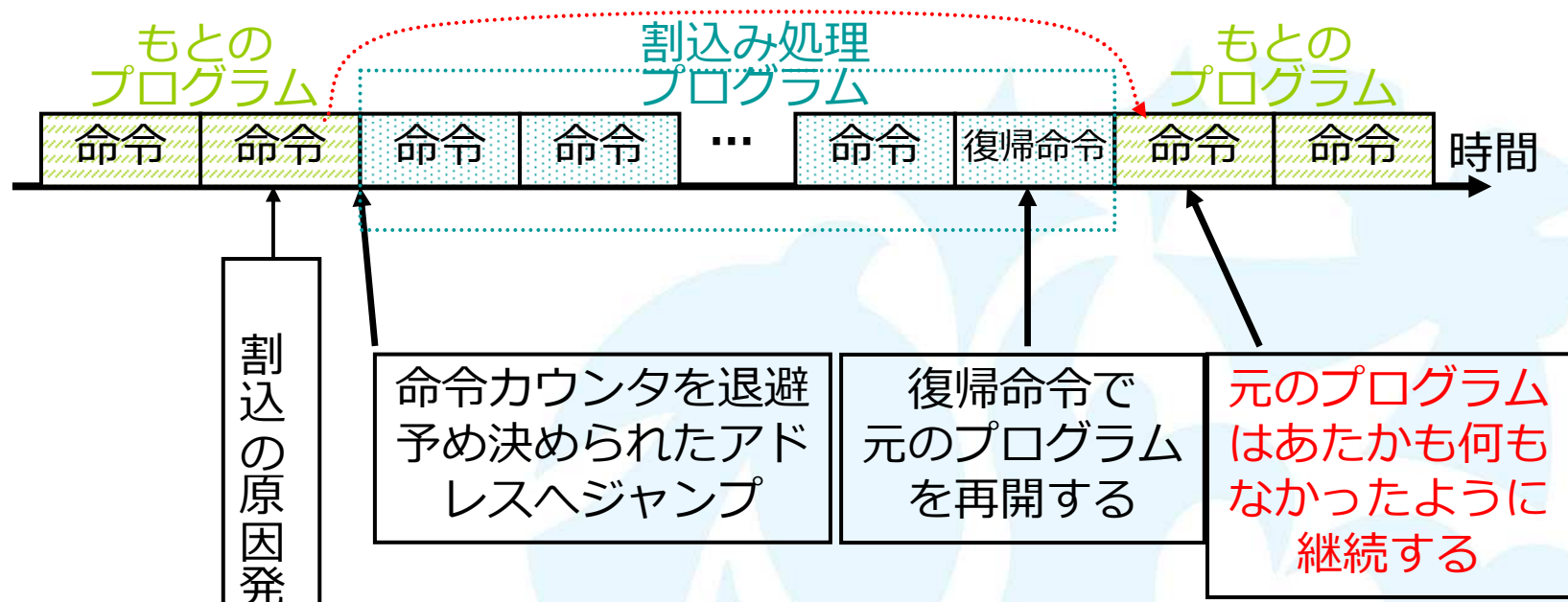
- * 入出力機器からの信号
(動作完了など)
- * 電源断・回路異常など
- * 命令実行エラー
ゼロ割、オーバフロー
- * 割込発生命令による割込
(ソフトウェア割込み)

割込みの仕組み ~ 図解すると



- * 入出力機器からの信号
(動作完了など)
- * 電源断・回路異常など
- * 命令実行エラー
ゼロ割、オーバフロー
- * 割り込発生命令による割り込
(ソフトウェア割り込み)

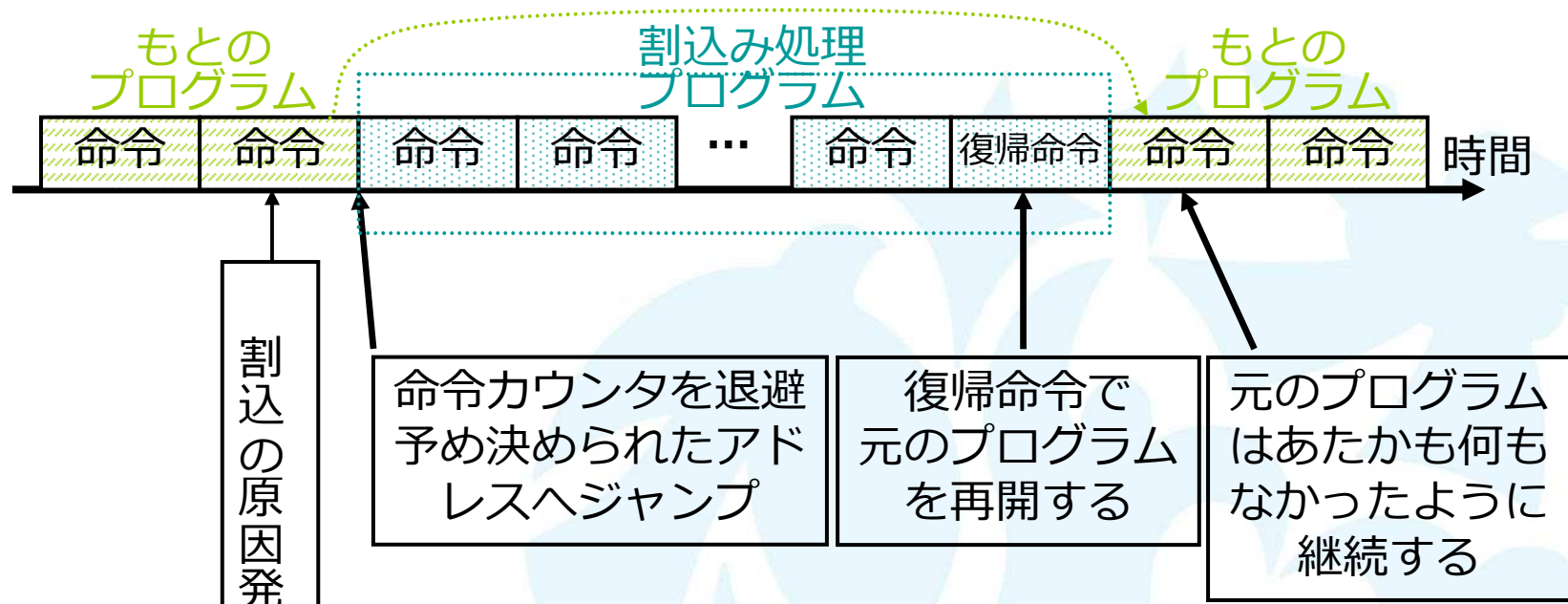
割込みの仕組み ~ 図解すると



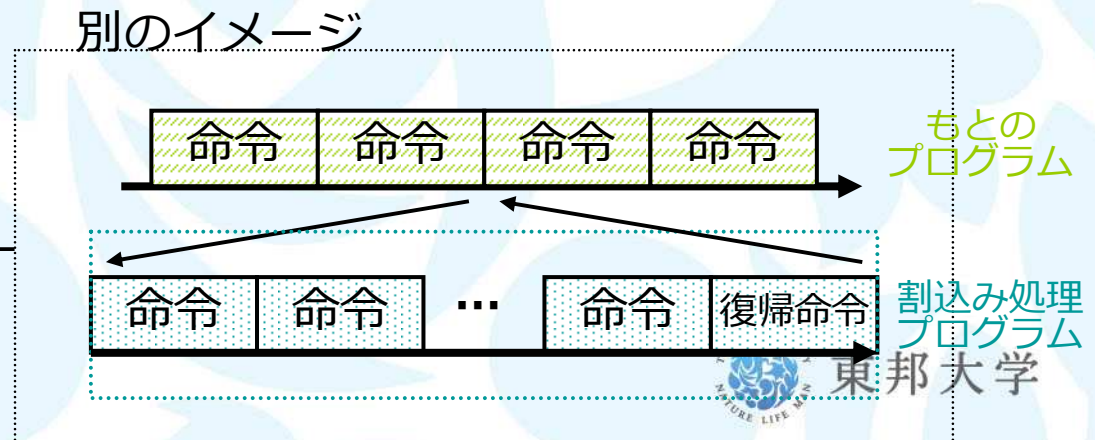
- * 入出力機器からの信号
(動作完了など)
- * 電源断・回路異常など
- * 命令実行エラー
ゼロ割、オーバフロー
- * 割込発生命令による割込
(ソフトウェア割込み)



割込みの仕組み ~ 図解すると



- * 入出力機器からの信号 (動作完了など)
- * 電源断・回路異常など
- * 命令実行エラー
ゼロ割、オーバフロー
- * 割り込発生命令による割り込 (ソフトウェア割り込み)

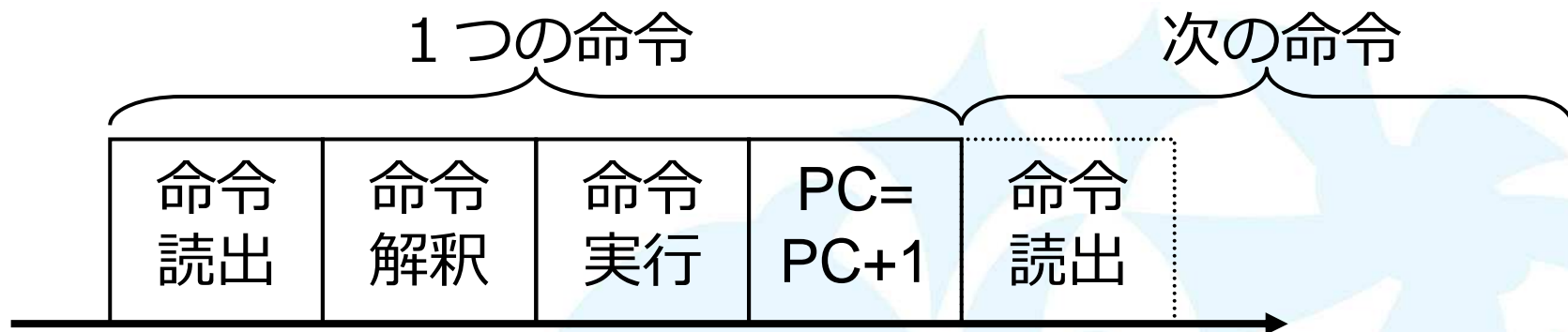


割込みの動作の説明ができますか？

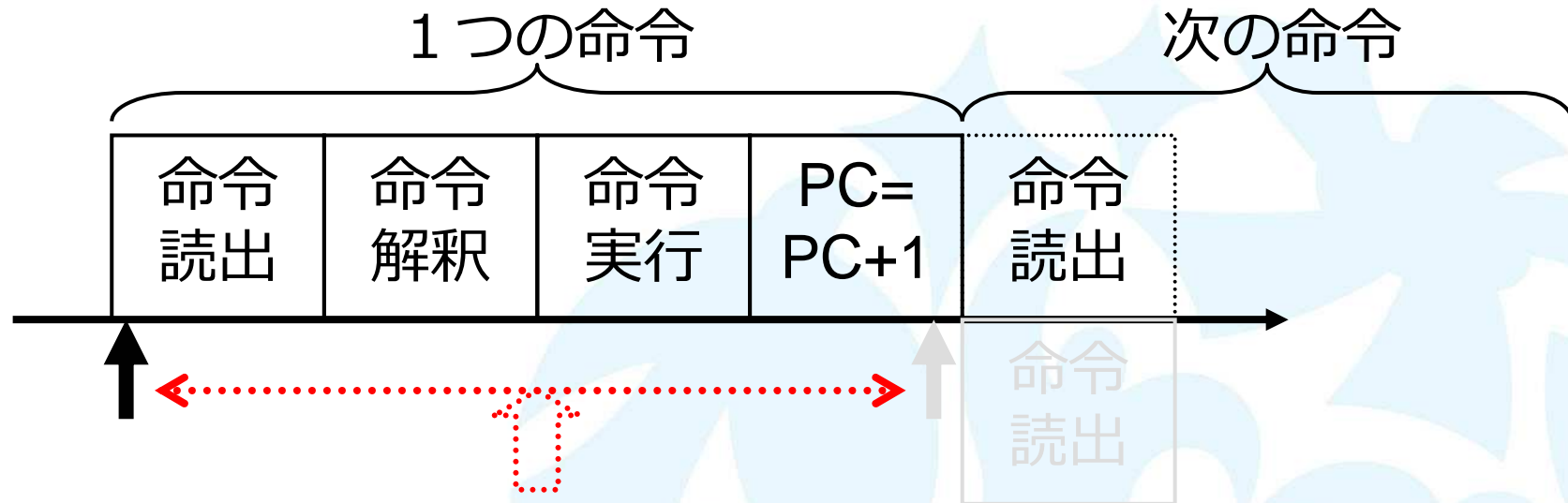


東邦大学

少し詳しく ～ 割り込みを起こす時点

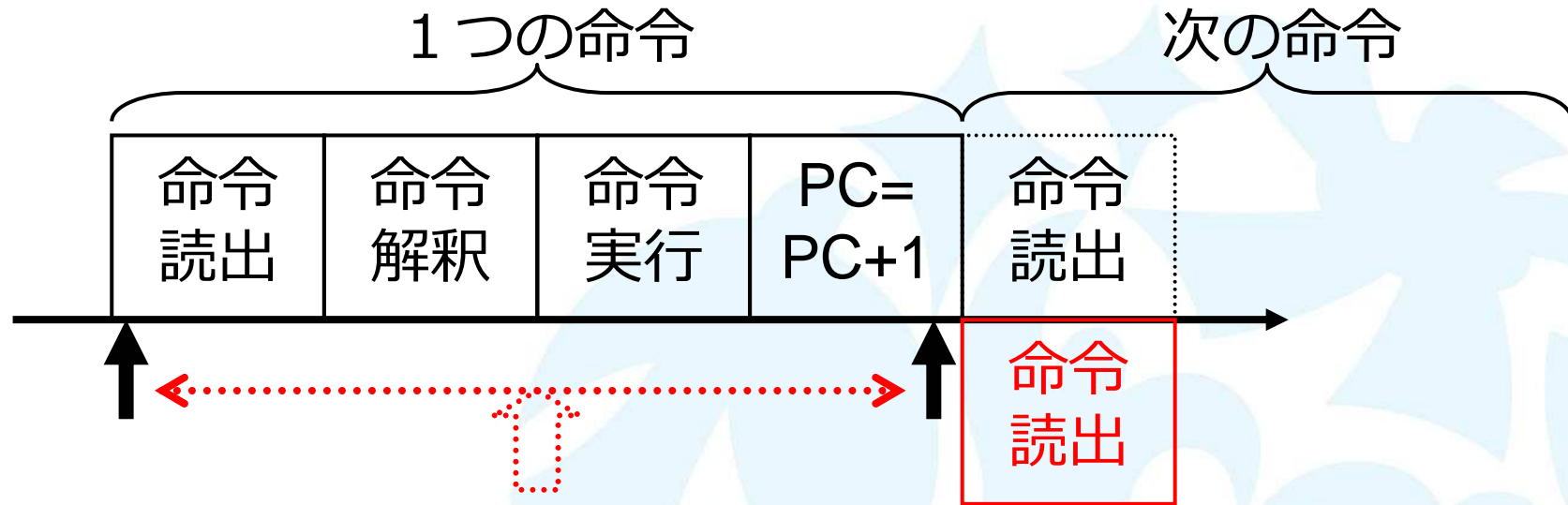


少し詳しく ~ 割り込みを起こす時点



この間に起こった
割り込み信号は

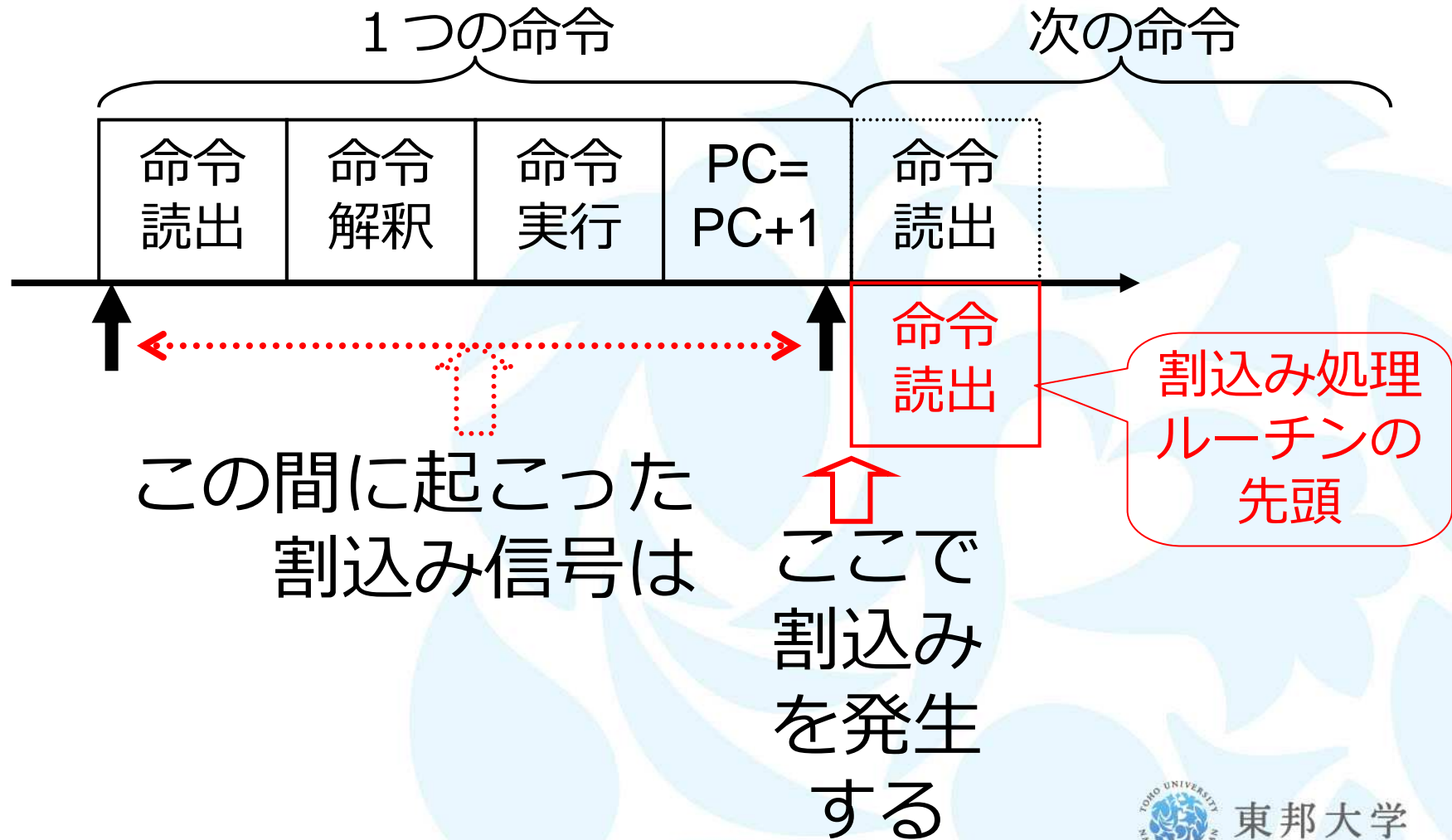
少し詳しく ~ 割り込みを起こす時点



この間に起こった
割り込み信号は

↑
ここで
割り込み
を発生
する

少し詳しく ~ 割込みを起こす時点



もう少し詳しく ～ 割込みからの復帰



東邦大学

もう少し詳しく ～ 割込みからの復帰

復帰 = あたかも何もなかったかのように！



東邦大学

もう少し詳しく ～ 割込みからの復帰

復帰 = あたかも何もなかったかのように！



割込み前に実行していた環境が
そのまま残っていないといけない



もう少し詳しく ～ 割込みからの復帰

復帰 = あたかも何もなかったかのように！



割込み前に実行していた環境が
そのまま残っていない

この環境のことを
コンテキスト（文脈）
と呼ぶ



もう少し詳しく ～ 割込みからの復帰

復帰 = あたかも何もなかったかのように！



割込み前に実行していた環境が
そのまま残っていない

環境（コンテキスト）とは？



もう少し詳しく ～ 割込みからの復帰

復帰 = あたかも何もなかったかのように！



割込み前に実行していた環境が
そのまま残っていない

環境（コンテキスト）とは？

汎用レジスタの内容、フラグレジスタの内容
プログラムカウンタの内容



もう少し詳しく ～ 割込みからの復帰

復帰 = あたかも何もなかったかのように！



割込み前に実行していた環境が
そのまま残っていないなければならない

環境（コンテキスト）とは？

汎用レジスタの内容、フラグレジスタの内容
プログラムカウンタの内容

（メモリ = 主記憶の内容は含めない）



もう少し詳しく ～ 割込みからの復帰

復帰 = あたかも何もなかったかのように！



割込み前に実行していた環境が
そのまま残っていないといけない



もし割込み処理ルーチンの中で上書きする
のなら、どこかへ値を退避しておく必要



もう少し詳しく ～ 割込みからの復帰

復帰 = あたかも何もなかったかのように！



割込み前に実行していた環境が
そのまま残っていないといけない



もし割込み処理ルーチンの中で上書きする
のなら、どこかへ値を退避しておく必要
汎用レジスタ、フラグレジスタ、
プログラムカウンタは
上書きされるので退避する必要



もう少し詳しく ～ 割込みからの復帰

復帰 = あたかも何もなかったかのように！



割込み前に実行していた環境が
そのまま残っていないといけない



汎用レジスタ・フラグレジスタは
割込み処理ルーチンの先頭で退避し、
終了復帰時に戻してやる必要がある



もう少し詳しく ～ 割り込みからの復帰

復帰 = あたかも何もなかったかのように！



割り込み前に実行していた環境が
そのまま残っていないといけない



汎用レジスタ・フラグレジスタは
割り込み処理ルーチンの先頭で退避し、
終了復帰時に戻してやる必要がある
プログラムカウンタはハードで退避し、
復帰命令で復帰する



もう少し詳しく ～ 割り込みからの復帰

復帰 = あたかも何もなかったかのように！



割り込み前に実行していた環境が
そのまま残っていない



コンテキスト（汎用レジスタ・フラグ
レジスタ・プログラムカウンタ）は
退避・復帰する必要がある



まとめると



割込みは

CPU(プログラム)の外で起こったことを
CPU(プログラム)に伝えるための仕組み

割り込みは

CPU(プログラム)の外で起こったことを
CPU(プログラム)に伝えるための仕組み

外で起こったこととは、
入出力装置の動作が終わったとか
ハードにエラーが起きたとか
計算でエラーが起きたとか
命令コードがおかしなコードだとか
リセットボタンが押されたとか



割り込みは

CPU(プログラム)の外で起こったことを
CPU(プログラム)に伝えるための仕組み

実行中の処理を中断して、割り込み処理
プログラムを実行し、終わると戻る

割込みは

CPU(プログラム)の外で起こったことを
CPU(プログラム)に伝えるための仕組み

実行中の処理を中断して、割込み処理
プログラムを実行し、終わると戻る

中断されたプログラムは、あたかも
何も起きなかったかのように続行する

割込みは

CPU(プログラム)の外で起こったことを
CPU(プログラム)に伝えるための仕組み

実行中の処理を中断して、割込み処理
プログラムを実行し、終わると戻る

中断されたプログラムは、あたかも
何も起きなかったかのように続行する

そのために、コンテキスト(文脈)を
退避・回復する



わかりましたね



東邦大学

追加

かなり細くてどうしてもよいことだけれど

同時に2つの割込みが発生？



東邦大学

同時に2つの割込みが発生？

「同時」ではなくて、
先の割込みの処理が終わらないうちに
次の割込みが発生してしまったとき
どうする？

同時に2つの割込みが発生？

「同時」ではなくて、
先の割込みの処理が終わらないうちに
次の割込みが発生してしまったとき

たとえば

プリンタ動作完了の処理をする中で
ハードエラー・計算エラーが起きたら？

同時に2つの割込みが発生？

「同時」ではなくて、
先の割込みの処理が終わらないうちに
次の割込みが発生してしまったとき

たとえば

プリンタ動作完了の処理をする中で
ハードエラー・計算エラーが起きたら？

キーボード入力の処理をする途中で
プリンタ動作完了割込みが起きたら？



どうするか？

プリンタ動作完了の処理をする中で
ハードエラー・計算エラーが起きたら？

どうするか？

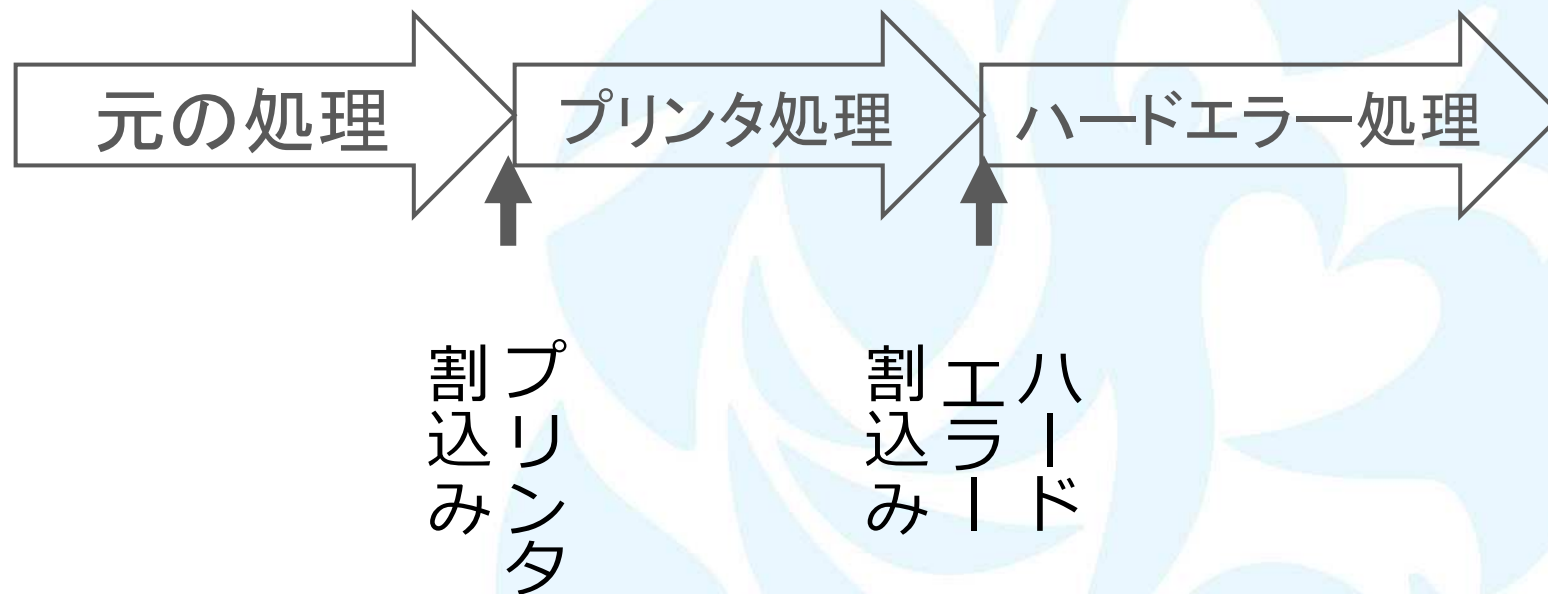
プリンタ動作完了の処理をする中で
ハードエラー・計算エラーが起きたら？

ハードエラーの方が**緊急度**が高い

どうするか？

プリンタ動作完了の処理をする中で
ハードエラー・計算エラーが起きたら？

ハードエラーの方が**緊急度**が高い



どうするか？

プリンタ動作完了の処理をする中で
ハードエラー・計算エラーが起きたら？

ハードエラーの方が**緊急度**が高い



割
込
み

割
込
み

ハードエラーでプリンタの処理を**中断**して
ハードエラー処理を行う必要がある

どうするか？

キーボード入力の処理をする途中で
プリンタ動作完了割込みが起きたら

どうするか？

キーボード入力の処理をする途中で
プリンタ動作完了割込みが起きたら

キーボードとプリンタは同じ入出力⇒同レベル

どうするか？

キーボード入力の処理をする途中で
プリンタ動作完了割り込みが起きたら

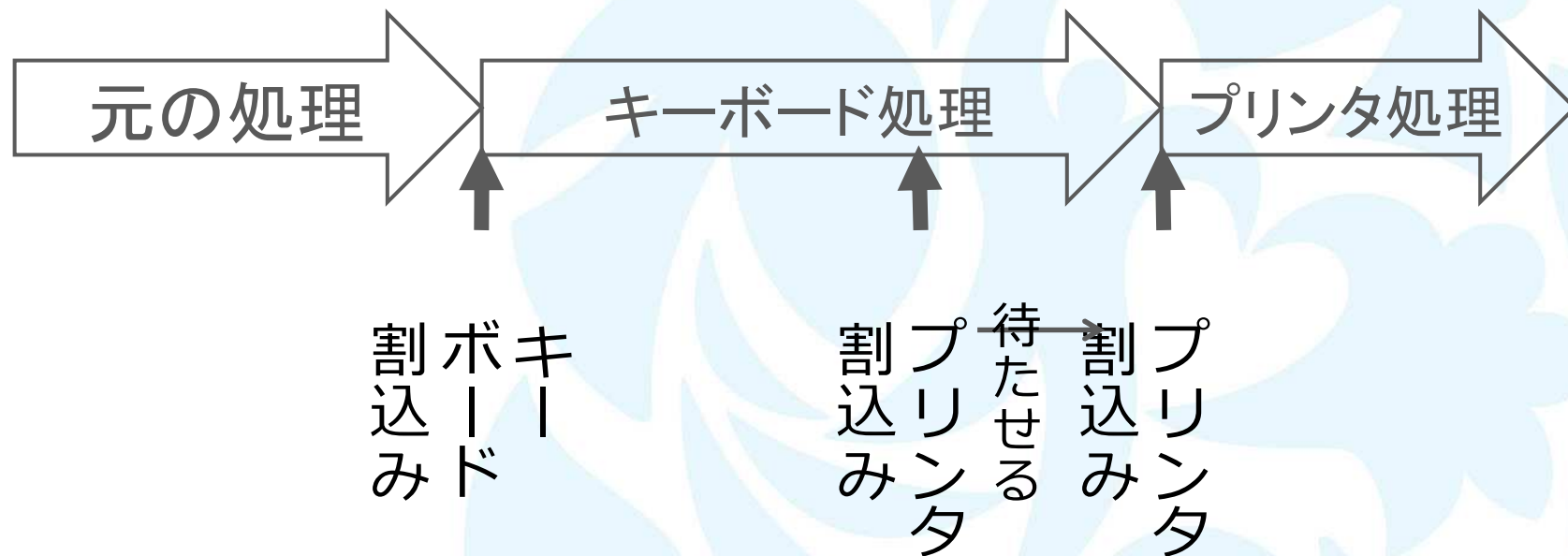
キーボードとプリンタは同じ入出力⇒同レベル



どうするか？

キーボード入力の処理をする途中で
プリンタ動作完了割り込みが起きたら

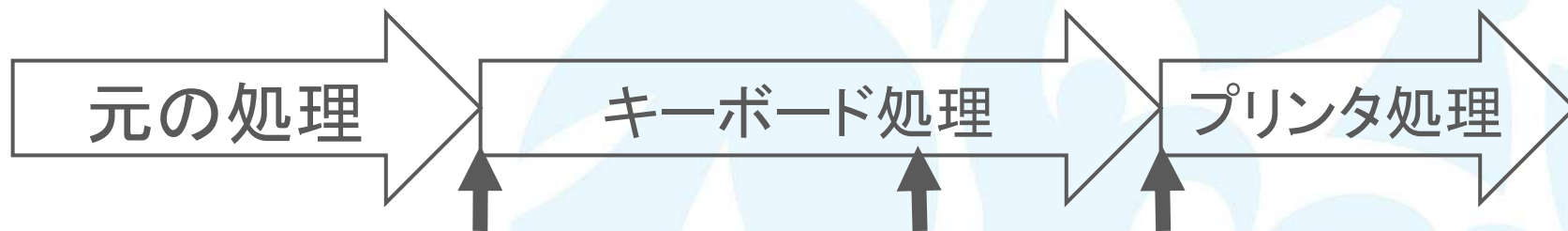
キーボードとプリンタは同じ入出力⇒同レベル



どうするか？

キーボード入力の処理をする途中で
プリンタ動作完了割り込みが起きたら

キーボードとプリンタは同じ入出力⇒同レベル



キーボード割り込み処理を続行させて
プリンタ割り込み処理を行えばよい

つまり

1つの割込み処理中に
後から別の割込みが発生した時

つまり

1つの割込み処理中に
後から別の割込みが発生した時

もし後の方が緊急度が高いなら
前の割込み処理を中断して割り込ませ

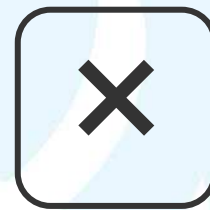
つまり

1つの割込み処理中に
後から別の割込みが発生した時

もし後の方が緊急度が高いなら
前の割込み処理を中断して割り込ませ

もし同じレベルか後の方が低いなら
前の割込み処理が終わるまで続けて
終わったら後の割込みを処理する

割込みの仕組みが
分かりましたか？



↓
次へ



東邦大学