

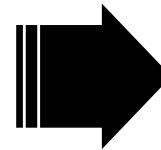
再配置可能プログラム

メモリ上のどこに配置しても動作するプログラムを再配置可能プログラムという。

プログラムを実行するマシン	配置位置 (アドレス)を 決める主体	配置位置 (アドレス)を 決める方法	メモリ上へ プログラムを 配置する主体	メモリ上へ プログラムを 配置する方法	再配置を 可能にする主体	再配置を 可能にする方法
一般の コンピュータ	OS				ソフトウェア開発ツール ／OS	・オブジェクトファイルのもつ 再配置情報を使用 ・位置独立コードを使用 (Relativeアドレッシング モードを使用)
09brd	あなた	擬似命令 ORGで指定	あなた	sコマンド +TeraTermの ファイル送信機能	プログラマ(あなた)	Indexed PC Relative アドレッシングモードを 使用
sim09brd				Sコマンド		

現在の一般のコンピュータでは、

- ・複数のプログラムを動かす(メモリ上に配置する)。
- ・ライブラリを利用してプログラムを作成する。
- ・配置は、メモリの状況を鑑みて、OSが行う。



プログラムの
再配置可能性は重要。

✓ Indexed PC Relativeアドレッシングモード

[CAE-3.pdf p.9](#)

[CAE-4.pdf p.7](#)

[CAE_AddressingModes_V2.pdf p.29](#)

✓ LEA(Load Effective Address: 実効アドレスロード)

[man6809.pdf p.A-45](#)

課題:再配置可能プログラムの作成

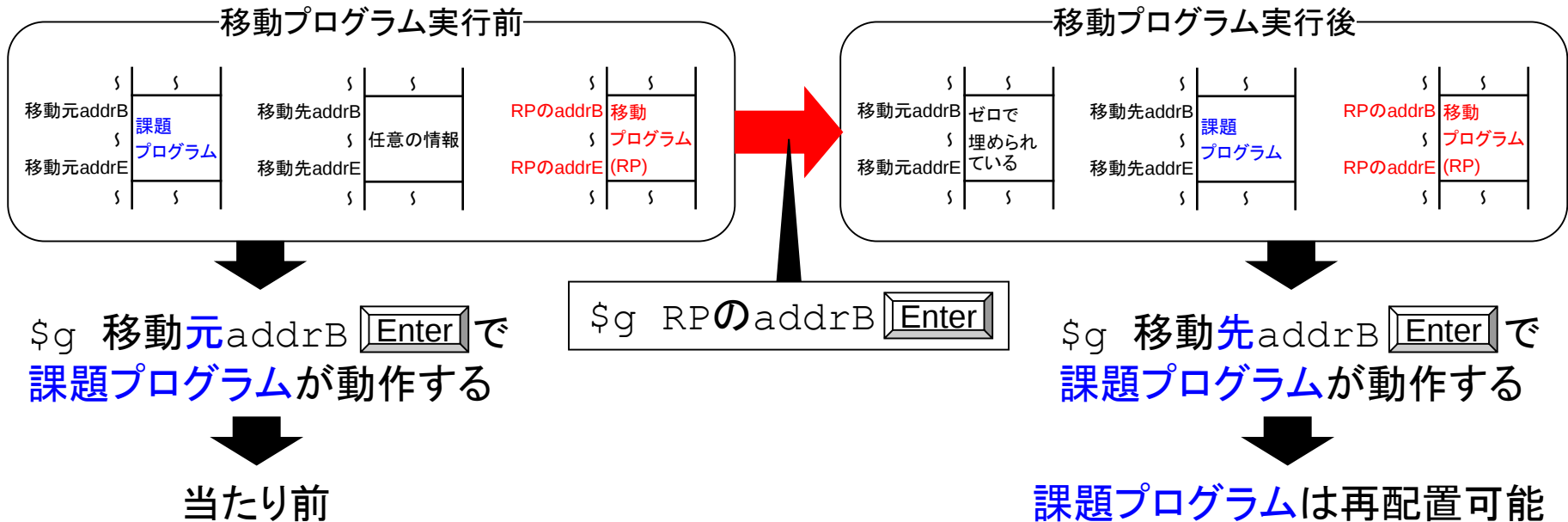
4つの1バイトデータ

\$07, \$35, \$81, \$C6

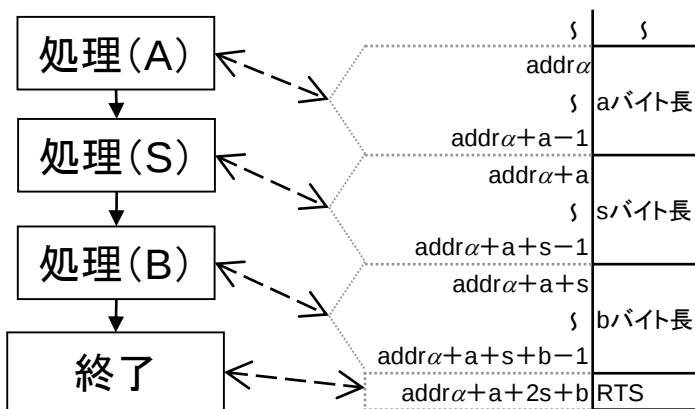
を、擬似命令FCBを用いてメモリに格納して参照するとき、
これらの総和を求め、4つのデータの格納位置に隙間無く続くアドレスに格納したい。
以上の動作をするプログラムを再配置可能なプログラムとして作成せよ。

再配置可能プログラムであるか否かの確認方法

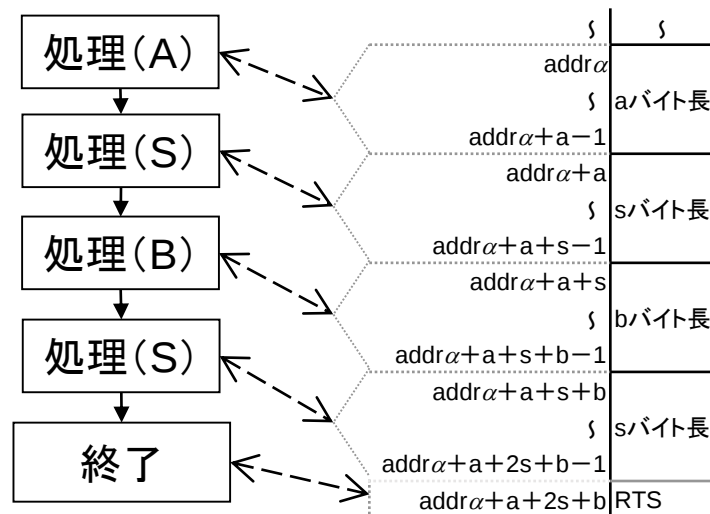
- 【注】・移動元addrBは「課題プログラム」中のORG擬似命令で指定した値。
・3つのメモリ領域が重なり合ってはいけない。(当たり前)
・移動元addrB、移動先addrB、RPのaddrB、の大小関係は問わない。



サブルーチン(1) サブルーチン化の有効性と方法(クラス2-GS&rts)



処理(B)の
後にも
処理(S)を
実行したい。



クラス2-GS
JSR BSR

クラス▲

メモリの無駄、処理(S)の修正の手間

処理(S)をサブルーチン化

一般に処理(S)をm回実行したいとき、($m \geq 2$)

◆処理(S)に要するメモリサイズは、
サブルーチン化前

ms

サブルーチン化後

$s+1+m$ (クラス2-GSのバイト長)

$\leq s+1+4m$ (∵クラス2-GSの命令の性質)

∴ $ms > s+1+4m$ を満たすならば、
サブルーチン化によって、メモリの無駄を解消可能。

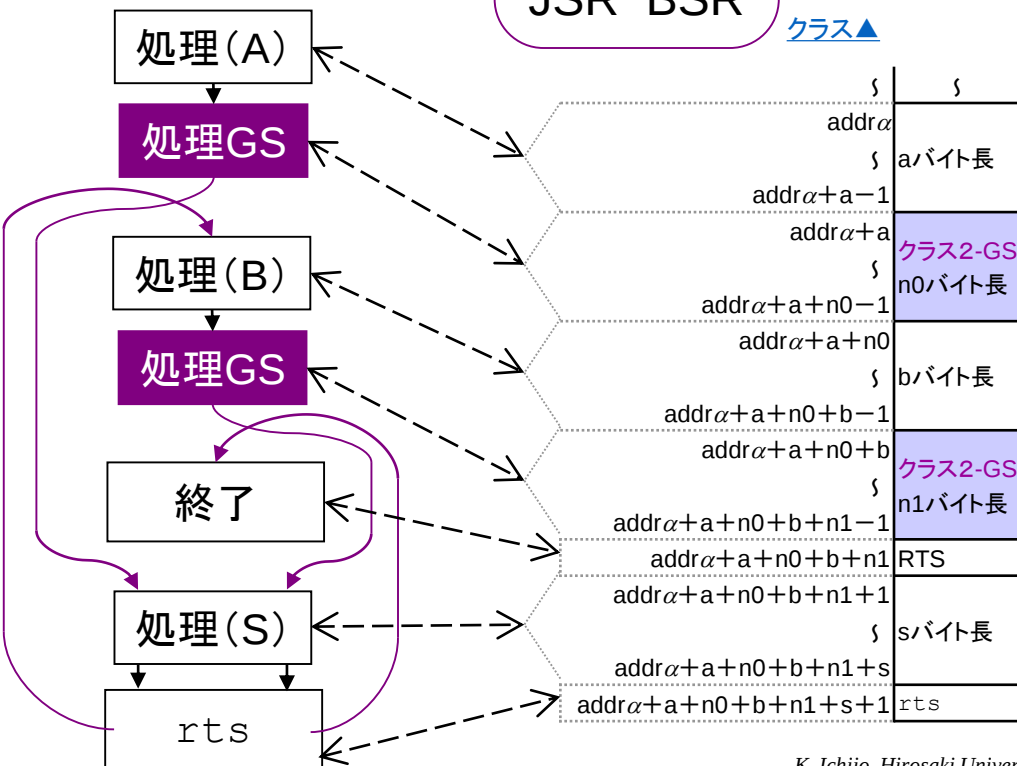
◆処理(S)の修正がk文字のとき全修正文字数は、
サブルーチン化前

mk

サブルーチン化後

k

∴ サブルーチン化によって、処理(S)の修正が容易に。



サブルーチン(2) クラス2-GS&rtsとクラス2-PCとの違い

クラス2-GS クラス▲

JSR BSR による処理(S)の呼出し

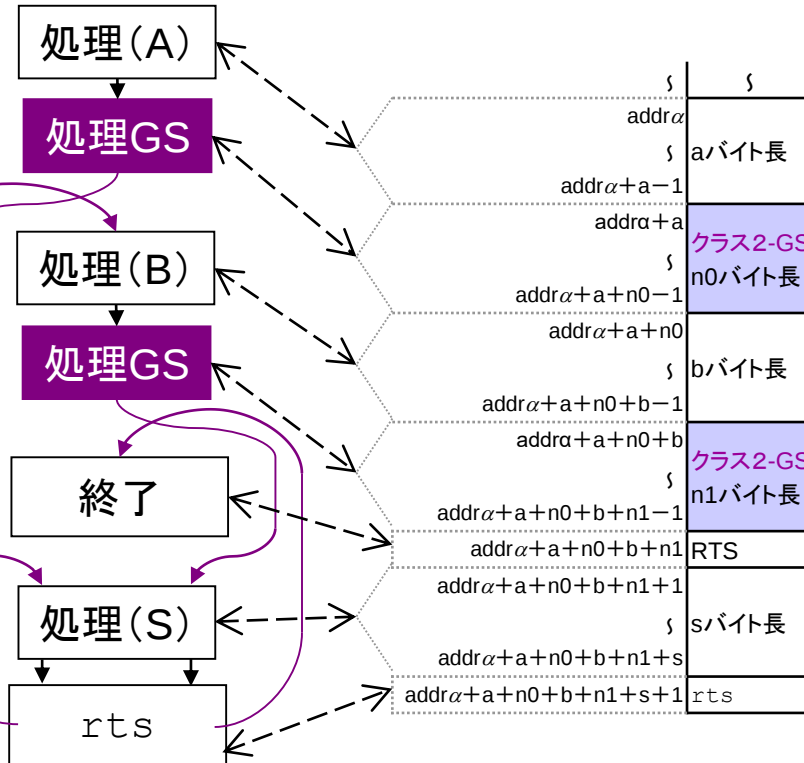
RTS による処理(S)からの復帰

クラス▲

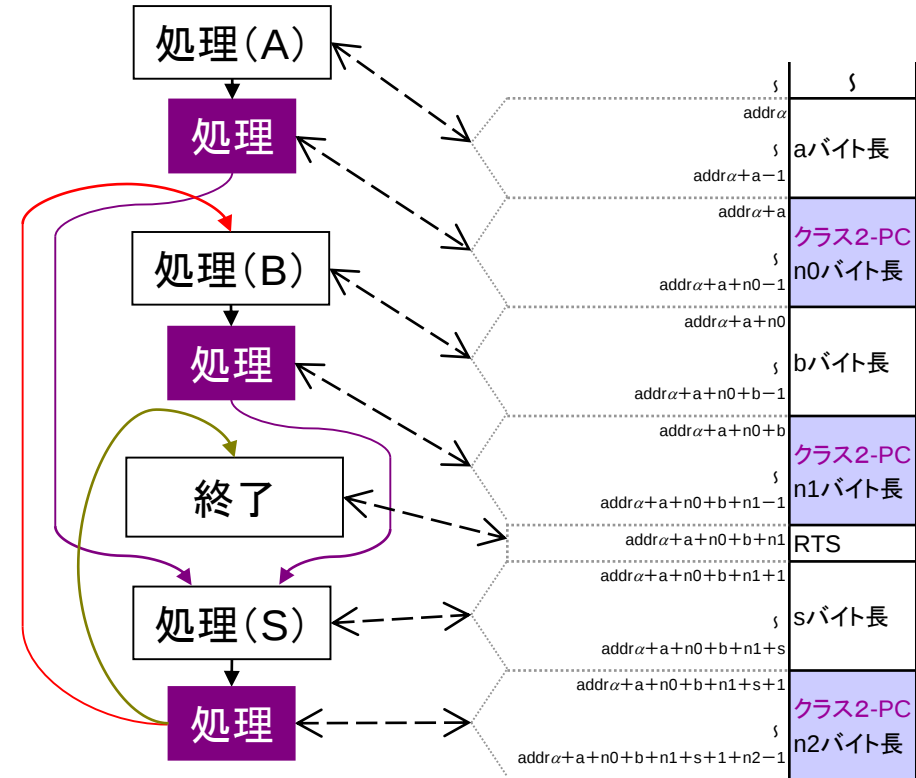
クラス2-PC

JMP BRA

による処理(S)の呼出し
による処理(S)からの復帰



問題なく実行される



⇒

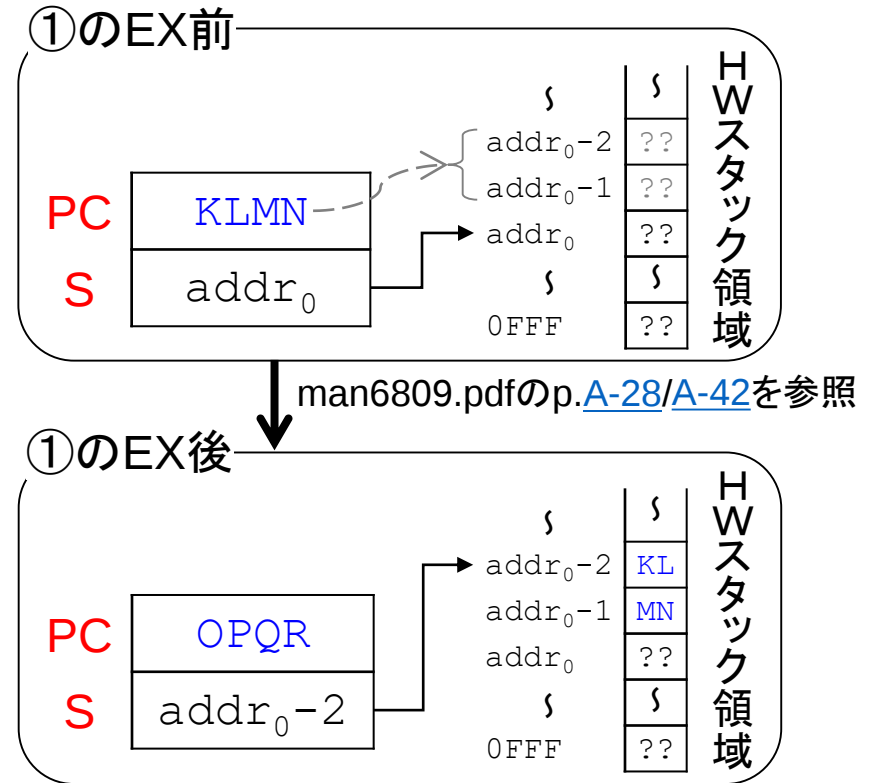
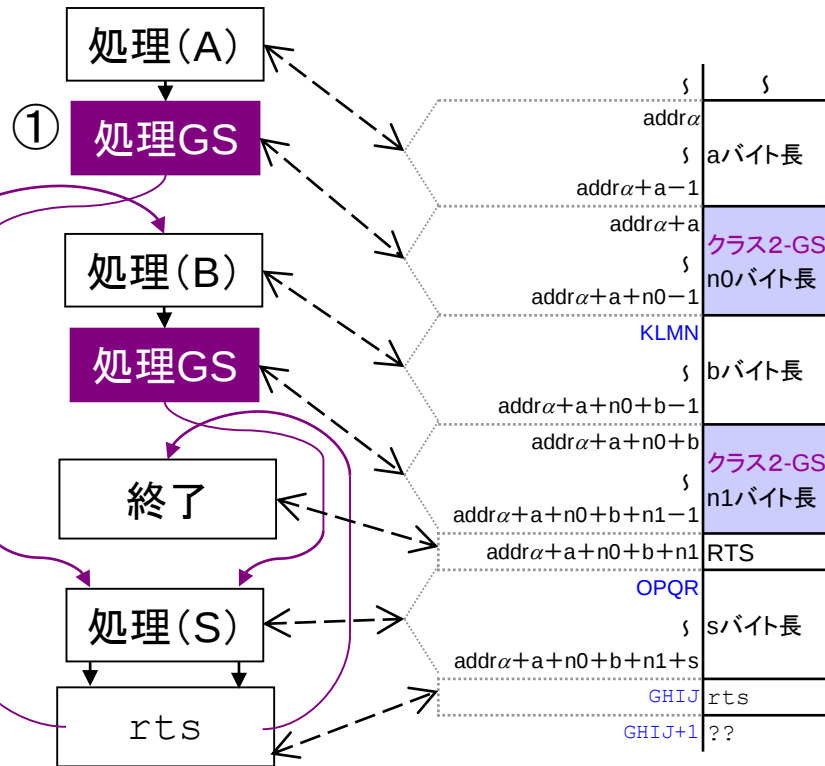
無限ループに陥る



⇒

処理(B)を実行できない

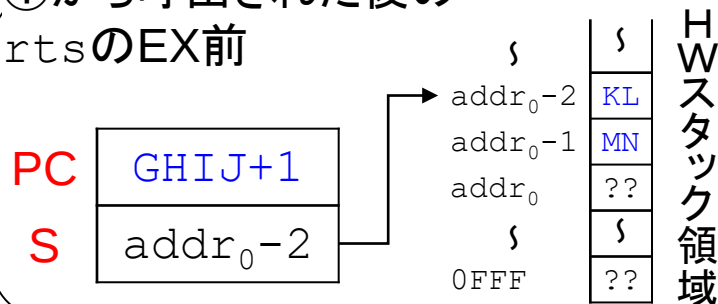
サブルーチン(3) クラス2-GS & rtsの動作とハードウェアスタック



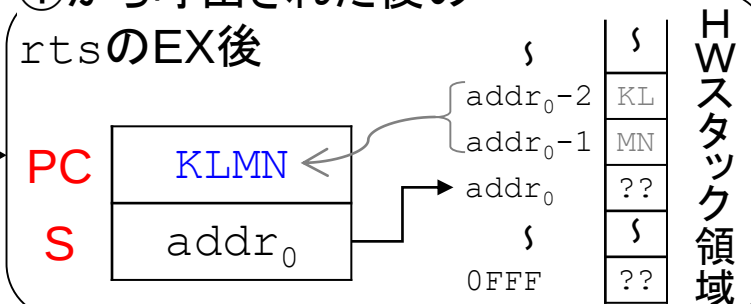
S: [Hardware Stack Register](#)

[HWスタック領域](#)

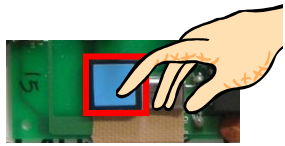
①から呼出された後の
rtsのEX前



①から呼出された後の
rtsのEX後



サブルーチン(4) モニタプログラムとサブルーチン



FFFFE0
FFFF09

初期化 FE09 1A
S
FE39 CD

コマンド
入力受付 FE3A 86
S
FE6F DF

コマンド
検索 FE70 8E
S
FE7A 27 beq FE85
FE7B 09
S
FE83 20 bra FE3A
FE84 B5

コマンド
呼出 FE85 AE
S
FE87 AD jsr ,X
FE88 84
FE89 20 bra FE3A
FE8A AF

mコマンド
Subroutine FF20 BD
S
FF40 39

dコマンド
Subroutine FF41 8E
S
FF91 39
S
FF97 39

gコマンド
Subroutine FF98 BD
S
FF9B 6E jmp ,X
FF9C 84

sコマンド
Subroutine FF9D BD
S
FFCE 39
S
FFDB EF

Other Subroutines

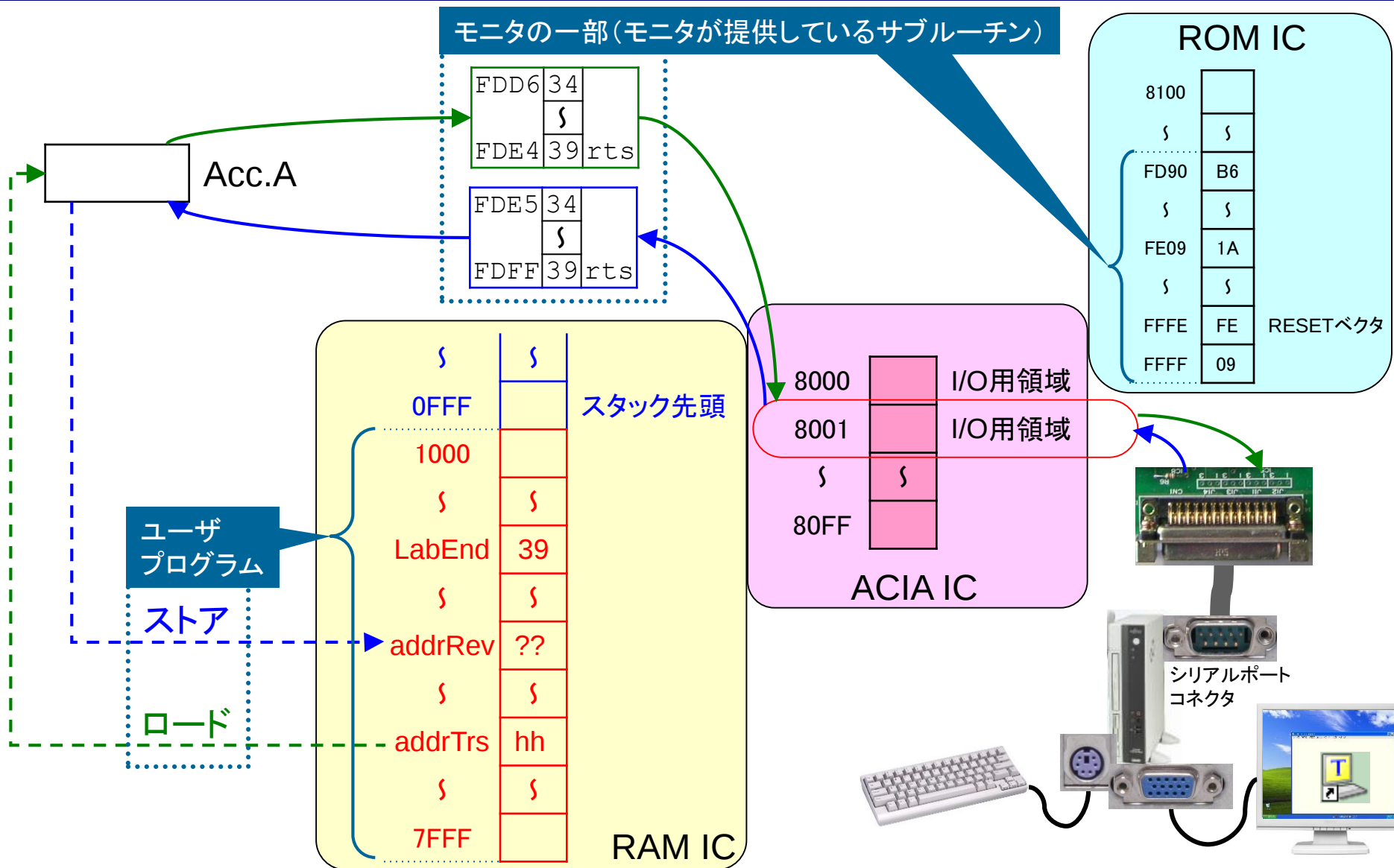
addrβ ??
S
39 rts
addy ??
S
39 rts

FDD6 34
S
FDE4 39 rts

UserProgram

addrα ??
S
39 rts

サブルーチン(5) 09brdの入出力とサブルーチン



1バイトデータ受信: FDE5のサブルーチン呼出『Acc.Aに受信データ格納』(→Acc.Aの値をスタア)

1バイトデータ送信: (Acc.Aに値をロード→) FDD6のサブルーチン呼出『Acc.Aの中身を送信』

課題: 1バイトデータ送信、文字列処理

(1)

自分の氏名のイニシャルの最初の文字データ(ASCIIコード)をFCBでメモリに配置し、**FDD6のサブルーチン**を呼び出して、その文字を表示するプログラムを作成せよ。

(2)

自分の氏名をローマ字表記した文字列データ(ASCIIコード)をFCBでメモリに配置し、**FDD6のサブルーチン**を繰り返し呼び出して、氏名を表示するプログラムを作成せよ。ただし、繰り返しの終了判定には、文字数ではなく、NUL文字を利用すること。

(3)

自分の氏名をローマ字表記した文字列データ(ASCIIコード)をFCBでメモリに配置し、**FDD6のサブルーチン**を繰り返し呼び出して、氏名を表示し、
さらに、氏名の各ローマ字の大文字⇄小文字変換をし、
FDD6のサブルーチンを繰り返し呼び出して、それを表示するプログラムを作成せよ。ただし、繰り返しの終了判定には、文字数ではなく、NUL文字を利用すること。